Graph Laplacian Tomography From Unknown Random Projections

Ronald R. Coifman, Yoel Shkolnisky, Fred J. Sigworth, and Amit Singer

***Abstract*—We introduce a graph Laplacian-based algorithm for the tomographic reconstruction of a planar object from its projections taken at random unknown directions. A Laplace-type operator is constructed on the data set of projections, and the eigenvectors of this operator reveal the projection orientations. The algorithm is shown to successfully reconstruct the Shepp–Logan phantom from its noisy projections. Such a reconstruction algorithm is desirable for the structuring of certain biological proteins using cryo-electron microscopy.**

***Index Terms*—Dimensionality reduction, graph laplacian, tomography.**

I. Introduction

A standard problem in computerized tomography (CT) is reconstructing an object from samples of its projections. Focusing our attention to a planar object characterized by its density function $\rho(x,y)$, its Radon transform $P_\theta(t)$ is the line integral of ρ along parallel lines L inclined at an angle θ with distances t from the origin (see, e.g, [1]–[3])

$$P_\theta(t) = \int_L \rho(x,y)ds = \int\int_{-\infty}^{\infty} \rho(x,y)\delta(x\cos\theta + y\sin\theta - t)dx\,dy. \quad (1)$$

The function ρ represents a property of the examined object which depends on the imaging modality. For example, ρ represents the X-ray attenuation coefficient in the case of X-ray tomography (CT scanning), the concentration of some radioactive isotope in PET scanning, or the refractive index of the object in ultrasound tomography.

Tomographic reconstruction algorithms estimate the function ρ from a finite set of samples of $P_\theta(t)$, assuming that the sampling points (θ,t) are known. See [4] for a survey of tomographic reconstruction methods. However, there are cases in which the projection angles are unknown, for example, when reconstructing certain biological proteins or moving objects. In such cases, one is given samples of the projection function $P_{\theta_i}(t)$ for a finite but unknown set of angles $\{\theta_i\}$, and the problem at hand is to estimate the underlying function ρ without knowing the angle values. The sampling set for the parameter t is usually known and dictated by the physical setting of the acquisition process; for example, the values of t correspond to the location of the detectors along the detectors line, while the origin may be set at the center of mass.

In this paper, we consider the reconstruction problem for the 2-D parallel-beam model with unknown acquisition angles. Formally, we consider the following problem: Given N projection vectors $(P_{\theta_i}(t_1), P_{\theta_i}(t_2), \ldots, P_{\theta_i}(t_n))$ taken at unknown angles $\{\theta_i\}_{i=1}^N$ that were randomly drawn from the uniform distribution of $[0,2\pi]$ and $t_1, t_2, \ldots, t_n$ are fixed n equally spaced points in t, find the underlying density function $\rho(x,y)$ of the object.

Various aspects of this problem were previously considered in [5] and [6]. In particular, [5] derives conditions for the existence of unique reconstruction from unknown angles and shifts. The angle recovery problem is formulated as a nonlinear system using the Helgason–Ludwig consistency conditions, that is used to derive uniqueness conditions. Stability conditions for the angle recovery problem under deterministic and stochastic perturbation models are derived in [6], where Cramér–Rao lower bounds on the variance of angle estimators for noisy projections are also given. An algorithm for estimating the angles is introduced in [6], and it consists of three steps: 1) initial angle estimation; 2) angle ordering; 3) joint maximum likelihood refinement of the angles and shifts. Step 2 uses a simple symmetric nearest neighbor algorithm for projection ordering. Once the ordering is determined, the projection angles are estimated to be equally spaced on the unit circle, as follows from the properties of the order statistics of uniformly distributed angles. Thus, the problem boils down to sorting the projections with respect to their angles.

Our proposed algorithm sorts the projections by using the graph Laplacian [7], [8]. Graph Laplacians are widely used in machine learning for dimensionality reduction, semi-supervised learning and spectral clustering. However, their application to this image reconstruction problem seems to be new. Briefly speaking, we construct an $N \times N$ weight matrix related to the pairwise projection distances, followed by a computation of its first few eigenvectors. The eigenvectors reveal the correct ordering of the projections in a manner to be later explained. This algorithm may also be viewed as a generalization of the nearest-neighbor insertion algorithm [6] as it uses several weighted nearest neighbors at once. More importantly, the graph Laplacian incorporates all local pieces

Manuscript received May 10, 2007; revised April 30, 2008. Current version published September 10, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Peter C. Doerschuk.

R. R. Coifman, Y. Shkolnisky, and A. Singer are with the Department of Mathematics, Program in Applied Mathematics, Yale University, New Haven, CT 06520-8283 USA (e-mail: coifman-ronald@yale.edu; yoel.shkolnisky@yale.edu; amit.singer@yale.edu).

F. J. Sigworth is with the Department of Cellular and Molecular Physiology, Yale University School of Medicine, New Haven, CT 06520 USA (e-mail: fred.sigworth@yale.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2008.2002305

of information into a coherent global picture, eliminating the dependence of the outcome on any single local datum. Small local perturbations of the data points have almost no effect on the outcome. This global information is encoded in the first few smooth and slowly-varying eigenvectors, which depend on the entire dataset. Our numerical examples show that increasing the number of projections improves the performance of the sorting algorithm. We examine the influence of corrupting the projections by white Gaussian additive noise on the performance of the graph Laplacian sorting algorithm and its ability to reconstruct the underlying object. We find that applying classical filtering methods to de-noise the projection vectors, such as the translation invariant spin-cycle wavelet de-noising [9], allows to reconstruct the underlying object at even higher levels of noise.

This work was motivated by a similar problem in three dimensions, where a 3-D object is to be reconstructed from its 2-D line integral projections (X-ray transform) taken at unknown directions, as is the case in cryo-electron microscopy [10]–[12]. Though there is no sense of order anymore, the graph Laplacian may be used to reveal the projection directions also in this higher dimensional case. See Section V for a brief discussion of extensions to higher dimensions.

The organization of the paper is as follows. In Section II, we survey graph Laplacians, which are being used in Section III for solving the tomography problem. The performance of the algorithm is demonstrated in Section IV. Finally, Section V contains some concluding remarks and a discussion of future extensions.

II. Graph Laplacians Sort Projections

Though graph Laplacians are widely used in machine learning for dimensionality reduction of high-dimensional data, semi-supervised learning and spectral clustering, their usage in tomography is uncommon. For that reason, a self-contained albeit limited description of graph Laplacians is included here. The presentation alternates between a general presentation of graph Laplacians and a specific consideration of their role in the tomography problem at hand. For a more detailed description of graph Laplacians and their applications, the reader is referred to [7], [8], and [13]–[15] (and references therein).

A. Spectral Embedding

In the context of dimensionality reduction, high-dimensional data points are described by a large number of coordinates n, and a reduced representation that uses only a few effective coordinates is wanted. Such a low-dimensional representation is sought to preserve properties of the high-dimensional dataset, such as, local neighborhoods [13], [16], geodesic distances [17], and diffusion distances [7]. It is often assumed that the data points approximately lie on a low-dimensional manifold, typically a nonlinear one. In such a setting, the N data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ are viewed as points in the ambient Euclidean space $\mathbb{R}^n$, while it is assumed that they are restricted to an intrinsic low-dimensional manifold $\mathcal{M}$. In other words, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathcal{M} \subset \mathbb{R}^n$ and $d = \dim \mathcal{M} \ll n$. For example, consider a curve Γ in $\mathbb{R}^3$. In this case, $n = 3$ and $d = 1$. One choice, out of many, of a dimensionality reducing mapping is given by the arc-length $s : \boldsymbol{\gamma}(s) \mapsto s$ for $\boldsymbol{\gamma}(s) \in \Gamma$. In the general setting, we assume that the structure of the manifold $\mathcal{M}$ is unknown, and one can only access the data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ as points in $\mathbb{R}^n$. Fitting the data points using linear methods such as linear regression, least squares approximation or principal component analysis, to name a few, usually performs poorly when the manifold is nonlinear. The graph Laplacian, however, is a nonlinear method that overcomes the shortcomings of the linear methods. The construction of the graph Laplacian is given below, as well as its relation to the well-known Laplace operator. In fact, we show that this construction gives rise to a family of Laplace-type operators, known in differential geometry and stochastic processes as the Laplace–Beltrami and the Fokker–Plank operators.

In our tomography problem, each data point corresponds to a projection at some fixed angle θ_i, sampled at n equally spaced points in the t direction

$$\boldsymbol{x}_i = \left(P_{\theta_i}(t_1), P_{\theta_i}(t_2), \ldots, P_{\theta_i}(t_n)\right), \quad i = 1, 2, \ldots, N. \tag{2}$$

The vector that corresponds to each projection is viewed as a point $\boldsymbol{x}_i \in \mathbb{R}^n$; however, all points $\boldsymbol{x}_i$ lie on a closed curve $\Gamma \subset \mathbb{R}^n$ parameterized by θ

$$\Gamma = \left\{\boldsymbol{\gamma}(\theta) = \left(P_\theta(t_1), \ldots, P_\theta(t_n)\right) \mid \theta \in [0, 2\pi)\right\}. \tag{3}$$

The closed curve Γ is a 1-D manifold of $\mathbb{R}^n$ $(d = 1)$ parameterized by the projection angle θ. The exact shape of the curve depends on the underlying imaged object $\rho(x, y)$, so different objects give rise to different curves. The particular curve Γ is unknown to us, because the object $\rho(x, y)$ is unknown. However, recovering the curve, or, in general, the manifold, from a sufficiently large number of data points sounds plausible.

In practice, the manifold is recovered by constructing the graph Laplacian and computing its first few eigenvectors. The starting point is constructing an $N \times N$ weight matrix $\boldsymbol{W}$ using a suitable semi-positive kernel k as follows:

$$W_{ij} = k\left(\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\varepsilon}\right), \quad i, j = 1, \ldots, N \tag{4}$$

where $\|\cdot\|$ is the Euclidean norm of the ambient space $\mathbb{R}^n$ and $\varepsilon > 0$ is a parameter known as the bandwidth of the kernel. A popular choice for the kernel function is $k(x) = \exp(-x)$, though other choices are also possible [7], [8]. The weight matrix $\boldsymbol{W}$ is then normalized to be row stochastic, by dividing it by a diagonal matrix $\boldsymbol{D}$ whose elements are the row sums of $\boldsymbol{W}$

$$D_{ii} = \sum_{j=1}^{N} W_{ij}. \tag{5}$$

The (negatively defined) normalized graph Laplacian $\boldsymbol{L}$ is then given by

$$\boldsymbol{L} = \boldsymbol{D}^{-1}\boldsymbol{W} - \boldsymbol{I} \tag{6}$$

where $\boldsymbol{I}$ is the $N \times N$ identity matrix. There exist normalizations other than the row stochastic one; the choice of normalization and the differences between them are addressed below.

The row stochastic matrix $\boldsymbol{D}^{-1}\boldsymbol{W}$ has a complete set of eigenvectors $\boldsymbol{\phi}_i$ and eigenvalues $1 = \lambda_0 > \lambda_1 \geq \cdots \geq \lambda_{N-1} \geq 0$, and the first eigenvector is constant, that is, $\boldsymbol{\phi}_0 = (1, 1, \ldots, 1)^T$. The remaining eigenvectors $\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_k$, for some $k \ll N$, define a k-dimensional nonlinear spectral embedding of the data

$$\boldsymbol{x}_i \mapsto (\phi_1(i), \phi_2(i), \ldots, \phi_k(i)), \quad i = 1, \ldots, N. \tag{7}$$

Examples demonstrating the rationale, properties and advantages of this embedding are given below. It is sometimes advantageous to incorporate the eigenvalues into the embedding by defining for some $t \geq 0$ [7]

$$\boldsymbol{x}_i \mapsto \left(\lambda_1^t \phi_1(i), \lambda_2^t \phi_2(i), \ldots, \lambda_k^t \phi_k(i)\right), \quad i = 1, \ldots, N.$$

B. Uniform Datasets and the Laplace–Beltrami Operator

The embedding (7) has many nice properties and we choose to emphasize only one of them, namely the intimate connection between the graph Laplacian matrix $\boldsymbol{L}$ and the continuous Laplace–Beltrami operator $\Delta_{\mathcal{M}}$ of the manifold $\mathcal{M}$.[1]

The connection between the graph Laplacian and the Laplace–Beltrami operator is manifested in the following theorem [18]: if the data points $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$ are independently uniformly distributed over the manifold $\mathcal{M}$ then with high probability

$$\frac{1}{\varepsilon}\sum_{j=1}^{N} L_{ij} f(\boldsymbol{x}_j) = \frac{1}{2}\Delta_{\mathcal{M}} f(\boldsymbol{x}_i) + O\left(\frac{1}{N^{1/2}\varepsilon^{1/2+d/4}}, \varepsilon\right) \tag{8}$$

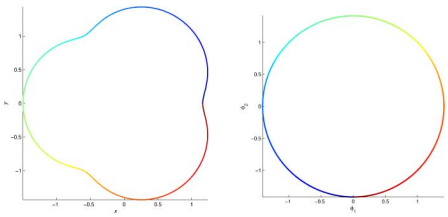
where $f : \mathcal{M} \mapsto \mathbb{R}$ is any smooth function. The approximation in (8) incorporates two error terms: a bias term of $O(\varepsilon)$ which is independent of N and a variance term of $O\left(N^{-1/2}\varepsilon^{-(1/2+d/4)}\right)$. The theorem implies that the discrete operator $\boldsymbol{L}$ converges pointwise to the continuous Laplace–Beltrami operator in the limit $\varepsilon \to 0$ and $N \to \infty$ as long as $N\varepsilon^{1+d/2} \to \infty$. This theorem justifies the name "graph Laplacian" given to the weighted adjacency matrix $\boldsymbol{L}$ in (6). In the sense given by (8), the graph Laplacian is a numerical machinery for approximating a specific operator on the underlying manifold, by using only a finite subset of its points. Note that the order of the data points in this theorem is irrelevant; the theorem holds for any arbitrary ordering of the points.

In other words, (8) states that applying the discrete operator $\boldsymbol{L}$ to a smooth function sampled at the data points approximates the Laplace–Beltrami of that function evaluated at those data points. Moreover, the eigenvectors of the graph Laplacian $\boldsymbol{L}$ approximate the eigenfunctions of $\Delta_{\mathcal{M}}$ that correspond to homogenous Neumann boundary condition (vanishing normal derivative) in the case that the manifold has a boundary [19].

[1]The Laplace–Beltrami operator on a manifold $\mathcal{M}$ is given by

$$\Delta f = \operatorname{divgrad} f = \sum_{ij} \frac{1}{\sqrt{|g|}} \partial_i \left(\sqrt{|g|} g^{ij} \partial_j f\right)$$

where g^{ij} are the components of the inverse metric tensor of $\mathcal{M}$, and $|g|$ is its determinant. For example, for the n-dimensional Euclidean space, the Laplace–Beltrami operator coincides with the ordinary Laplacian and has the form $\Delta f = \sum_{i=1}^{n} (\partial^2 f / \partial x_i^2)$, because $g^{ij} = \delta_{ij}$.



(a) (b)

Fig. 1. Equally spaced sampled epitrochoid and its spectral embedding. (a) Epitrochoid that corresponds to $R = 1$, $r = 1/3$, $d = 1/6$. Points are equally spaced in arc-length. (b) Embedding the epitrochoid into the eigenvectors (ϕ_1, ϕ_2) of the graph Laplacian.

This connection between the graph Laplacian and the Laplace–Beltrami operator sheds light on the spectral embedding (7). For example, consider a closed curve $\Gamma \subset \mathbb{R}^n$ of length l parameterized by the arc-length s. The Laplace–Beltrami operator Δ_Γ of Γ is simply the second order derivative with respect to the arc-length, $\Delta_\Gamma f(s) = f''(s)$. The eigenfunctions of Δ_Γ satisfy

$$f''(s) = -\lambda f(s), \quad s \in (0, l) \tag{9}$$

with the periodic boundary conditions $f(0) = f(l)$, $f'(0) = f'(l)$. The first eigenfunction is the constant function $\phi_0(s) = 1$ with eigenvalue $\lambda_0 = 0$. The remaining eigenfunctions are $\{\cos(2\pi m s/l), \sin(2\pi m s/l)\}_{m=1}^{\infty}$ with corresponding degenerated eigenvalues $\lambda_m = 4\pi^2 m^2/l^2$ of multiplicity 2. It follows that embedding $\boldsymbol{\gamma}(s) \in \Gamma$ using the first two nontrivial eigenfunctions results in the unit circle in the plane

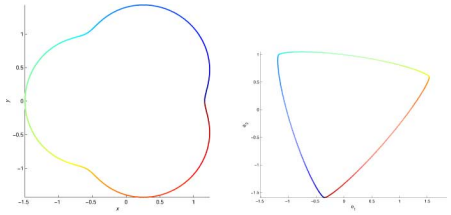
$$\boldsymbol{\gamma}(s) \mapsto \left(\cos\left(\frac{2\pi s}{l}\right), \sin\left(\frac{2\pi s}{l}\right)\right), \quad s \in [0, l]. \tag{10}$$

For data points $\boldsymbol{x}_i = \boldsymbol{\gamma}(s_i)$ that are uniformly distributed over the curve Γ, the first two nontrivial eigenvectors of the graph Laplacian are approximately $\cos(2\pi s/l)$ and $\sin(2\pi s/l)$ and the embedding (7) reads

$$\boldsymbol{x}_i \mapsto (\boldsymbol{\phi}_1(i), \boldsymbol{\phi}_2(i)) \approx \left(\cos\left(\frac{2\pi s_i}{l}\right), \sin\left(\frac{2\pi s_i}{l}\right)\right). \tag{11}$$

Due to the multiplicity of the eigenvalues, the computed eigenvectors may be any orthogonal 2×2 linear transformation of $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$. The specific orthogonal combination depends on the numerical procedure used to compute the eigenvectors. Thus, the embedding is unique up to an arbitrary rotation and possibly a reflection (orientation of the curve). Fig. 1(b) shows that the graph Laplacian embedding of data points equally spaced with respect to arc-length along the epitrochoid in Fig. 1(a) is, indeed, a circle.

The 2-D embedding (11) reveals the ordering of the data points $\boldsymbol{x}_i$ along the curve Γ. The graph Laplacian provides an approximate solution to the traveling salesman problem in $\mathbb{R}^n$. Going back to the tomography problem, the graph Laplacian embedding of the projection vectors (2) reveals their true ordering. The last statement is indeed correct, but we should exercise more carefulness in its justification.



(a) (b)

Fig. 2. Density dependent embedding. (a) Epitrochoid that corresponds to $R = 1, r = 1/3, d = 1/6$. Points are equally spaced in $\theta \in [0, 2\pi)$. (b) Embedding the epitrochoid into the eigenvectors (ϕ_1, ϕ_2) of the graph Laplacian.

The graph Laplacian approximates the Laplace–Beltrami operator if the data points are uniformly distributed over the manifold. However, this is not the case in our tomography problem. Even though the projection angles are uniformly distributed on the unit circle, the projection vectors are not uniformly distributed over the curve in $\mathbb{R}^n$ on which they lie with respect to its arc-length. To see this, we examine the relationship between the probability density function $p_\Theta(\theta)$ of the projection angle θ and the probability density function $p_S(s)$ of the projection vectors over the curve Γ. This relationship is given by $p_\Theta(\theta)\, d\theta = p_S(s)\, ds$, for infinitesimals arc-length ds and angle $d\theta$, because the mapping of the unit circle S^1 to Γ conserves the number of mapped points. The uniform distribution of the angle θ means $p_\Theta(\theta) = (1/2\pi)$. Hence

$$p_S(s) = p_\Theta(\theta) \left(\frac{ds}{d\theta}\right)^{-1} = \frac{1}{2\pi} \left\| \frac{d\boldsymbol{\gamma}(\theta)}{d\theta} \right\|^{-1} \tag{12}$$

where $\boldsymbol{\gamma}(\theta) = (P_\theta(t_1), \ldots, P_\theta(t_n))$, and $(d\boldsymbol{\gamma}/d\theta) = ((\partial/\partial\theta)P_\theta(t_1), \ldots, (\partial/\partial\theta)P_\theta(t_n))$. The density $p_S(s)$ depends on the specific object $\rho(x, y)$ through Γ and is usually not uniform, because $\|d\boldsymbol{\gamma}/d\theta\|$ is not constant.

C. Nonuniform Densities and the Fokker–Planck Operator

When data points are distributed over a manifold $\mathcal{M}$ according to a nonuniform density $p(\boldsymbol{x})$, their graph Laplacian does not approximate the Laplace–Beltrami operator, but rather a different differential operator, known as the backward Fokker–Planck operator $\mathcal{L}$, [8], [20]

$$\mathcal{L}f = \Delta_{\mathcal{M}} f - \nabla U \cdot \nabla f \tag{13}$$

where $U(\boldsymbol{x}) = -2 \log p(\boldsymbol{x})$ is the potential function. Thus, the more general form of (8) is

$$\frac{1}{\varepsilon} \sum_{j=1}^{N} L_{ij} f(\boldsymbol{x}_j) \approx \frac{1}{2} \mathcal{L} f(\boldsymbol{x}_i). \tag{14}$$

Note that the Fokker–Planck operator $\mathcal{L}$ coincides with the Laplace–Beltrami operator in the case of a uniform distribution for which the potential function U is constant, so its gradient vanishes.

The Fokker–Planck operator has a complete set of eigenfunctions and eigenvalues. In particular, when the manifold is a closed curve Γ of length l, the eigenfunctions satisfy

$$f'' - U' f' = -\lambda f \tag{15}$$

with the periodic boundary conditions

$$f(0) = f(l), \quad f'(0) = f'(l). \tag{16}$$

We rewrite the eigenfunction problem (15) as a Sturm–Liouville problem $\left(e^{-U} f'\right)' + \lambda e^{-U} f = 0$. Although the eigenfunctions are no longer the sine and cosine functions, it follows from the classical Sturm–Liouville theory of periodic boundary conditions and positive coefficients $\left(e^{-U(s)} = p_S^2(s) > 0\right)$ [21] that the embedding consisting of the first two nontrivial eigenfunctions $\boldsymbol{\phi}_1$, $\boldsymbol{\phi}_2$ of (15)–(16) also circles the origin exactly once in a manner that the angle is monotonic. In other words, upon writing the embedding in polar coordinates

$$\boldsymbol{\gamma}(s) \mapsto (\phi_1(s), \phi_2(s)) = r(s) e^{i\varphi(s)}, \quad s \in [0, l]$$

the argument $\varphi(s)$ is a monotonic function of s, with $\varphi(0) = 0$, $\varphi(l) = 2\pi$. Despite the fact that the explicit form of the eigenfunctions is no longer available, the graph Laplacian embedding reveals the ordering of the projections through the angle φ_i attached to $\boldsymbol{x}_i$. Fig. 2(a) and (b) shows a particular embedding of a curve into the eigenfunctions of the Fokker–Planck operator. The embedding is no longer a circle as it depends on the density along the curve, but still, ordering the points according to the angle $\varphi(s)$ of the embedding produces the correct ordering of the points along the original curve.

D. Density Invariant Graph Laplacian

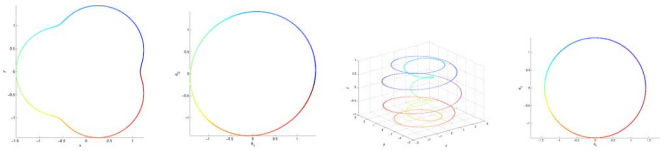
What if data points are distributed over $\mathcal{M}$ with some nonuniform density, and we still want to approximate the Laplace–Beltrami operator on $\mathcal{M}$ instead of the Fokker–Planck operator? This can be achieved by replacing the row stochastic normalization in (5) and (6) by the so-called density invariant normalization. Such a normalization is described in [8] and leads to the density invariant graph Laplacian. This normalization is obtained as follows. First, normalize both the rows and columns of $\boldsymbol{W}$ to form a new weight matrix $\tilde{\boldsymbol{W}}$

$$\tilde{\boldsymbol{W}} = \boldsymbol{D}^{-1} \boldsymbol{W} \boldsymbol{D}^{-1} \tag{17}$$

where $\boldsymbol{D}$ is the diagonal matrix (5) whose elements are the row sums of $\boldsymbol{W}$. Next, normalize the new weight matrix $\tilde{\boldsymbol{W}}$ to be row stochastic by dividing it by a diagonal matrix $\tilde{\boldsymbol{D}}$ whose elements are the row sums of $\tilde{\boldsymbol{W}}$ $\left(\tilde{D}_{ii} = \sum_{j=1}^{N} \tilde{W}_{ij}\right)$. Finally, the (negatively defined) density invariant graph Laplacian $\tilde{L}$ is given by

$$\tilde{\boldsymbol{L}} = \tilde{\boldsymbol{D}}^{-1} \tilde{\boldsymbol{W}} - \boldsymbol{I}. \tag{18}$$

The density invariant graph Laplacian $\tilde{\boldsymbol{L}}$ approximates the Laplace–Beltrami operator on the underlying manifold $\mathcal{M}$, with $\tilde{\boldsymbol{L}}$ replacing $\boldsymbol{L}$ in (8) [8], even when the data points are nonuniformly distributed over $\mathcal{M}$. Therefore, embedding data points which are nonuniformly distributed over a closed curve Γ using the density-invariant graph Laplacian results in a circle given by (10) and (11). As mentioned before, although θ is uniformly distributed in $[0, 2\pi)$, the arc-length s is not uniformly distributed in $[0, l]$, but rather has some nonconstant density $p_S(s)$. It follows that the embedded points that are given by (11) are nonuniformly distributed on the circle. Nonetheless, the embedding reveals the ordering of the projection vectors.



(a) (b) (c) (d)

Fig. 3. Density invariant embedding of the epitrochoid and a closed helix. (a) Epitrochoid that corresponds to $R = 1$, $r = 1/3$, $d = 1/6$. Points are equally spaced in $\theta \in [0, 2\pi)$. (b) Embedding the epitrochoid into the eigenvectors (ϕ_1, ϕ_2) of the density invariant graph Laplacian. (c) A closed helix in $\mathbb{R}^3$. Points are nonequally spaced in arc-length. (d) Embedding the closed helix into the eigenvectors (ϕ_1, ϕ_2) of the density invariant graph Laplacian.

Fig. 3(b) and (d) shows the embedding of the epitrochoid [Fig. 3(a)] and a closed helix in $\mathbb{R}^3$ [Fig. 3(c)] into the first two eigenfunctions of the Laplace–Beltrami operator, obtained by applying the density-invariant normalization.

The graph Laplacian integrates local pieces of information into a global consistent picture. Each data point interacts only with a few of its neighbors, or a local cloud of points, because the kernel is rapidly decaying outside a neighborhood of size $\sqrt{\varepsilon}$. However, the eigenvector computation involves the entire matrix and glues those local pieces of information together.

III. Recovering the Projection Angles

So far, we have shown that by constructing the graph Laplacian from the given projections and embedding each projection into the first two eigenvectors, it is possible to recover the correct ordering of the projections along the underlying curve in $\mathbb{R}^n$. Once the projection vectors are sorted, the values of the projection angles $\theta_1, \ldots, \theta_N$ need to be estimated. Since the projection angles are assumed to be uniformly distributed over the circle, we estimate the sorted projection angles

$$\theta_{(1)} < \theta_{(2)} < \cdots < \theta_{(N)}$$

by equally spaced angles $\bar{\theta}_{(k)}$ (the bar indicates that these are estimated angle values rather than true values)

$$\bar{\theta}_{(k)} = \frac{2\pi k}{N}, \quad k = 1, \ldots, N. \tag{19}$$

Due to rotation invariance, we fix $\theta_{(N)} = 2\pi$. The remaining $N - 1$ random variables $\theta_{(k)}$ $(k = 1, \ldots, N-1)$ are known as the kth order statistics [22] and their (marginal) probability density functions $p_{\theta_{(k)}}(\theta)$ are given by

$$\frac{(N-1)!}{2\pi(k-1)!(N-1-k)!}\left(\frac{\theta}{2\pi}\right)^{k-1}\left(1-\frac{\theta}{2\pi}\right)^{N-1-k}.$$

The mean value and variance of $\theta_{(k)}$ are

$$\mathbb{E}\theta_{(k)} = \frac{2\pi k}{N}, \quad \mathrm{Var}\theta_{(k)} = \frac{4\pi^2 k(N-k)}{(N+1)N^2}.$$

Thus, the equally spaced estimation (19) of the kth order statistics is in fact the mean value estimation, and the mean square error (MSE) given by $\mathrm{Var}\,\theta_{(k)}$ is maximal for $k = \lfloor N/2 \rfloor$

$$\mathrm{Var}\,\theta_{(\lfloor N/2 \rfloor)} \sim \frac{\pi^2}{N} + O\left(\frac{1}{N^2}\right).$$

The MSE vanishes as the number of data points $N \to \infty$, and the typical estimation error is $O(1/\sqrt{N})$.

Now that the projection angles have been estimated, any classical tomography algorithm may be applied to reconstruct the image. The image can be reconstructed either from the entire set of N projection vectors, or it can be reconstructed from a smaller subset of them. Given a set of mN projection vectors, where m is an over-sampling factor, we first sort all mN angles $\theta_{(1)} < \theta_{(2)} < \cdots < \theta_{(mN)}$ using the density-invariant graph Laplacian, but use only N of them (every mth projection) $\theta_{(m)} < \theta_{(2m)} < \cdots < \theta_{(mN)}$ for the image reconstruction. The effect of sub-sampling is similarly understood in terms of the order statistics.

Note that the symmetry of the projection function (1) $P_\theta(t) = P_{\theta+\pi}(-t)$ practically doubles the number of projections. For every given projection vector $\boldsymbol{x}_i = (P_{\theta_i}(t_1), \ldots, P_{\theta_i}(t_n))$ that corresponds to an unknown angle θ_i, we create the projection vector

$$\boldsymbol{x}_{i+mN} = (P_{\theta_i}(-t_1), \ldots, P_{\theta_i}(-t_n)) \tag{20}$$

that corresponds to the unknown angle $\theta_i + \pi$.

The reconstruction algorithm is summarized in Algorithm 1. Note that in steps 2 and 3 of Algorithm 1 the graph Laplacian $\boldsymbol{L}$ can be used instead of $\tilde{\boldsymbol{L}}$.

Algorithm 1 Reconstruction from random orientations

Require: Projection vectors $\boldsymbol{x}_i = (P_{\theta_i}(t_1), \ldots, P_{\theta_i}(t_n))$, for $i = 1, 2, \ldots, mN$

1: Double the number of projections to $2mN$ using (20).

2: Construct $\tilde{\boldsymbol{L}}$ following (4), (5), and (17)–(18).

3: Compute $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$, the first nontrivial eigenvectors of $\tilde{\boldsymbol{L}}$.

4: Sort $\boldsymbol{x}_i$ according to $\varphi_i = \tan^{-1}(\phi_1(i)/\phi_2(i))$.

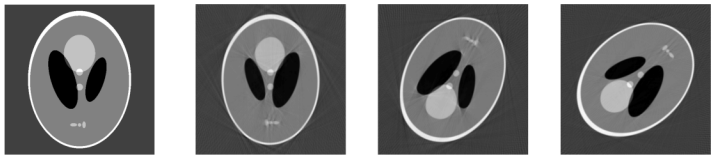
5: Reconstruct the image using the sorted projections corresponding to the estimated angles $\bar{\theta}_{(2mi)} = 2\pi i/N$.

If the distribution of the projection angles is not uniform, then the estimation (19) should be replaced by the mean value of the order statistics of the corresponding distribution.

IV. Numerical Examples and Noise Tolerance

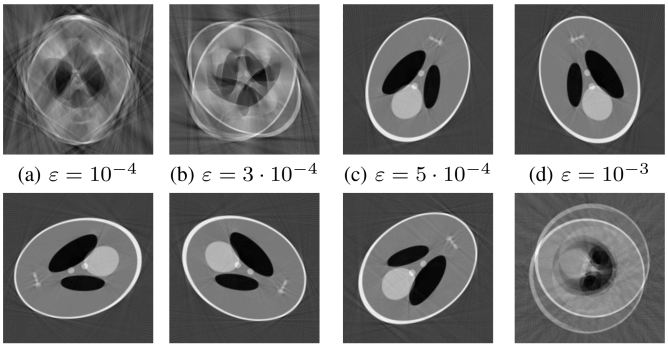
We applied to the above algorithm to the reconstruction of the 2-D Shepp–Logan phantom, shown in Fig. 4(a), from its projections at random angles. The results are illustrated in Fig. 4(a)–(d).

The figures were generated as follows. We set $N = 256$, and for each over-sampling factor $m = 4$, 8, 16, we generated mN uniformly distributed angles in $[0, 2\pi]$, denoted $\theta_1, \ldots, \theta_{mN}$. Then, for each θ_i, we evaluated the analytic expression of the Radon transform of the Shepp–Logan phantom [2] at $n = 500$ equally spaced points between -1.5 and 1.5. That is, each projection vector $\boldsymbol{x}_i$ is a vector in $\mathbb{R}^{500}$. We then applied Algorithm 1 and reconstructed the Shepp–Logan phantom using $N = 256$ projections. The results are presented in Fig. 4(b)–(d) for $m = 4$,



(a) Original Shepp-Logan phantom (b) $mN = 1024$ (c) $mN = 2048$ (d) $mN = 4096$

Fig. 4. Reconstructing the Shepp–Logan phantom from its random projections while using the symmetry of the Radon transform. $N = 256$.



(a) $\varepsilon = 10^{-4}$ (b) $\varepsilon = 3 \cdot 10^{-4}$ (c) $\varepsilon = 5 \cdot 10^{-4}$ (d) $\varepsilon = 10^{-3}$

(e) $\varepsilon = 10^{-2}$ (f) $\varepsilon = 5 \cdot 10^{-2}$ (g) $\varepsilon = 7.5 \cdot 10^{-2}$ (h) $\varepsilon = 10^{-1}$

Fig. 5. Reconstructing the Shepp–Logan phantom from its random projections for different values of ε [increasing from (a) to (h)]. All reconstructions use $N = 256$, $mN = 4096$, and $n = 500$ pixels per projection. High-quality reconstructions are obtained for a wide range of ε values.

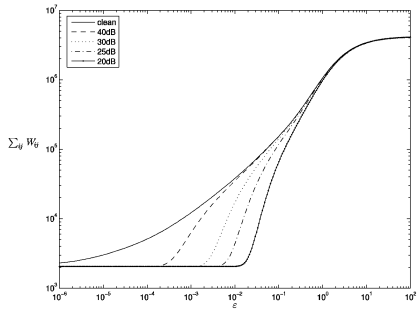


Fig. 6. Logarithmic scale plot of $\sum_{i,j=1}^{N} W_{ij}(\varepsilon)$ against ε for various levels of noise. The top (blue) curve corresponds to noiseless projections.

8, 16, respectively. The density invariant graph-Laplacian (Algorithm 1) was constructed using the kernel $k(x) = e^{-x}$ with $\varepsilon = 0.05$. The dependence of the algorithm on ε is demonstrated below. All tests were implemented in Matlab. The Radon transform was inverted using Matlab's `iradon` function with spline interpolation and a hamming filter.

Note that Fig. 4(b)–(d) exhibits an arbitrary rotation, and possibly a reflection as is the case in Fig. 4(c), due to the random orthogonal mixing of the eigenfunctions $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ that consists of merely rotations and reflections.

A. Choosing ε

For the reconstructions in Fig. 4(b)–(d), we used $\varepsilon = 0.05$. According to (8), in general, the value of ε should be chosen to balance the bias term that calls for small ε with the variance term that calls for large ε. In practice, however, the value of ε is set such that for each projection vector $\boldsymbol{x}_i$ there are several neighboring projection vectors $\boldsymbol{x}_j$ for which W_{ij} in (4) are non-negligible. Fig. 5(a)–(h) depicts the dependence of the quality of reconstruction on the value of ε. We conclude that the algorithm is stable with respect to ε in the sense that high-quality reconstructions are obtained when ε is changed by as much as two orders of magnitude, from $5 \cdot 10^{-4}$ to $7.5 \cdot 10^{-2}$.

The value of ε can also be chosen in an automated way without manually verifying the reconstruction quality and without computing the eigenvectors of the graph Laplacian matrix. Following [23], we use a logarithmic scale to plot the sum of the N^2 weight matrix elements

$$\sum_{i,j} W_{ij}(\varepsilon) = \sum_{ij} \exp\left\{\frac{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\varepsilon}\right\} \tag{21}$$

against ε (Fig. 6). As long as the statistical error in (8) is small, the sum (21) is approximated by its mean value integral

$$\sum_{ij} \exp\left\{\frac{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\varepsilon}\right\} \approx \frac{N^2}{\mathrm{vol}^2(\mathcal{M})} \int_{\mathcal{M}} \int_{\mathcal{M}} \exp\left\{\frac{-\|\boldsymbol{x} - \boldsymbol{y}\|^2}{2\varepsilon}\right\} d\boldsymbol{x} d\boldsymbol{y} \tag{22}$$

where $\mathrm{vol}(\mathcal{M})$ is the volume of the manifold $\mathcal{M}$ and assuming uniformly distributed data points. For small values of ε, we approximate the narrow Gaussian integral

$$\int_{\mathcal{M}} \exp\left\{\frac{-\|\boldsymbol{x} - \boldsymbol{y}\|^2}{2\varepsilon}\right\} d\boldsymbol{x} \approx \int_{\mathbb{R}^d} \exp\left\{\frac{-\|\boldsymbol{x} - \boldsymbol{y}\|^2}{2\varepsilon}\right\} d\boldsymbol{x} = (2\pi\varepsilon)^{d/2} \tag{23}$$

because the manifold looks locally like its tangent space $\mathbb{R}^d$. Substituting (23) in (21)–(22) gives

$$\sum_{i,j} W_{ij}(\varepsilon) \approx \frac{N^2}{\mathrm{vol}(\mathcal{M})} (2\pi\varepsilon)^{d/2}$$

or, equivalently, upon taking the logarithm

$$\log\left(\sum_{i,j} W_{ij}(\varepsilon)\right) \approx \frac{d}{2} \log\varepsilon + \log\left(\frac{N^2 (2\pi)^{d/2}}{\mathrm{vol}(\mathcal{M})}\right) \tag{24}$$

which means that the slope of the logarithmic scale plot is $d/2$. In the limit $\varepsilon \to \infty$, $W_{ij} \to 1$, so $\sum_{ij} W_{ij} \to N^2$. On the other hand, as $\varepsilon \to 0$, $W_{ij} \to \delta_{ij}$, so $\sum_{ij} W_{ij} \to N$. Those two limiting values assert that the logarithmic plot cannot be linear for all values of ε. In the linearity region, both the statistical and bias errors are small, and it is, therefore, desirable to choose ε from that region.

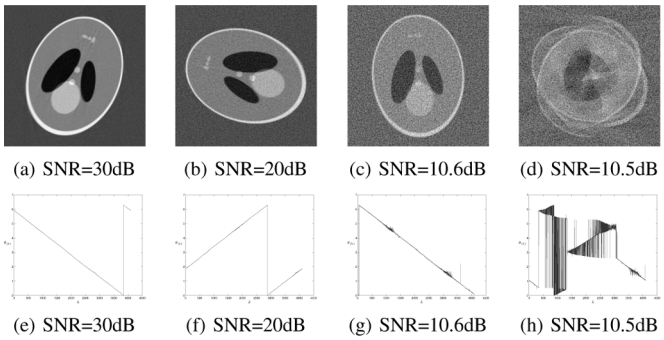


Fig. 7. Top: Reconstructing the Shepp–Logan phantom from its random projections that were corrupted by different levels of additive white noise. Bottom: Projection angles as estimated by the graph Laplacian sorting algorithm plotted against their true values. The jump discontinuity is due to the rotation invariance of the problem. Reflection flips the slope to -1. ($mN = 4096$, $N = 256$, $n = 500$).

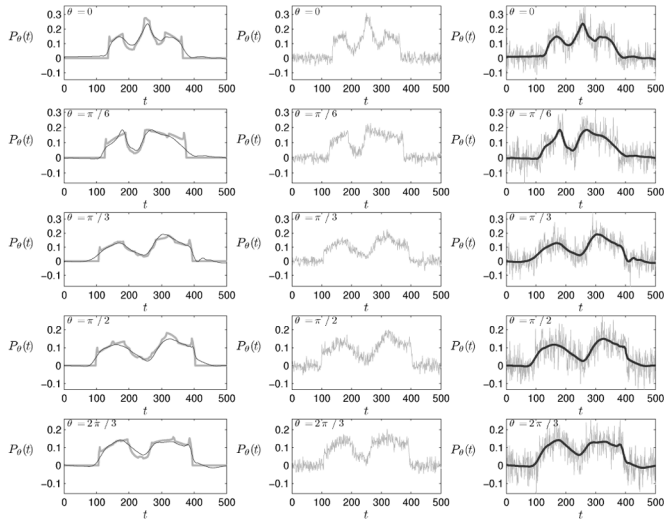


Fig. 8. Five different projections that differ by $\Delta\theta = \pi/6$ (left column, blue thick curve), their noisy version at 10.6 dB (center column) and their noisy version at 2.0 dB (right column). The solid red curves (right column and left column) correspond to applying the hard thresholding full spin-cycle de-noising algorithm with Daubechies "db2" wavelets to the 2.0 dB noisy projections of the right column.

In Fig. 6, the top (blue) curve corresponds to noiseless projections. The slope of that curve in the region of linearity, $10^{-3} \leq \varepsilon \leq 10^{-1}$, is approximately 0.5, as predicted by (24) for data points that lie on a curve ($d = 1$).

B. Noise Tolerance

The effect of additive noise on the reconstruction is depicted in Fig. 7(a)–(d). For each figure, we randomly drew 4096 angles from a uniform distribution, computed the projections of the Shepp–Logan phantom corresponding to those angles and added noise to the computed projections. For a given SNR, the noise was Gaussian with zero mean and a variance that satisfied SNR [dB] $= 10\log_{10}(\operatorname{Var} S/\operatorname{Var}\delta)$, where S is the array of the noiseless projections and δ is a realization of the noise. As before, once applying the algorithm, the images were reconstructed from $N = 256$ projections.

The reconstruction algorithm performs well above ≈ 10.6 dB and performs poorly below this SNR value. As was pointed out in [6], this phenomenon is related to the threshold effect in nonlinear parameter estimation [24], [25] that predicts a sharp transition in the success rate of detecting and estimating the signal from its noisy measurements as a function of the SNR. The manifestation of this phenomena in our case is that the distances in (4) become meaningless above a certain level of noise. Fig. 7(c) and (d) demonstrates the breakdown of the algorithm when the SNR decreases by just 0.1 dB. Fig. 7(e)–(h) demonstrates the same breakdown by comparing the estimated projection angles with their true value. Fig. 8 shows five different projections $P_{\theta_i}(t)$ separated by $\theta_{i+1} - \theta_i = \pi/6$ (left column, thick blue curve) and their noisy realizations at 10.6 dB (center column), gauging the level of noise that can be tolerated.

The threshold effect can also be understood by Fig. 6, where it is shown that higher levels of noise result in higher slope values, rendering larger empirical dimensions. In other words, adding noise thickens the curve $\Gamma \subset \mathbb{R}^n$ and effectively enlarges the dimensionality of the data. The graph Laplacian treats the data points as if they lie on a surface rather than a curve and stumbles upon the threshold effect.

The threshold point can be pushed down by initially de-noising the projections and constructing the graph Laplacian

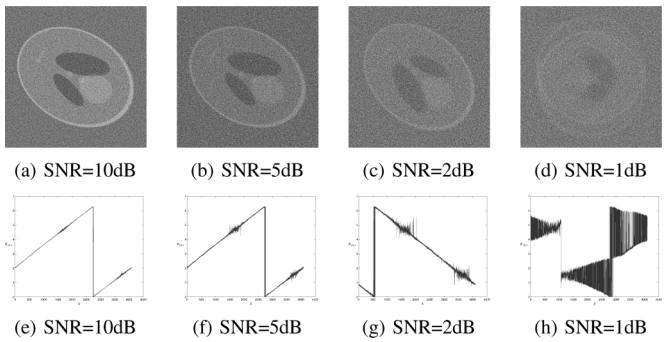


Fig. 9. Top: Reconstructing the Shepp–Logan phantom from its random projections that were corrupted by additive white noise by first spin-cycle filtering the projections. The main features of the phantom are reconstructed even at 2 dB. Bottom: Projection angles as estimated by the graph Laplacian sorting algorithm with spin-cycled denoised projections plotted against their true values for different levels of noise ($mN = 4096$, $N = 256$, $n = 500$).

using the de-noised projections rather than the original noisy ones. In practice, we used the fast $O(n\log n)$ implementation of the full translation invariant wavelet spin-cycle algorithm [9] with Daubechies wavelets "db2" of length 4 combined with hard thresholding the wavelet coefficients at $\sigma\sqrt{2\log n}$, where $\sigma = \sqrt{\operatorname{Var}\delta/\operatorname{Var} S}$ [9]. Using this classical nonlinear filtering method, we were able to push down the threshold point from 10.6 to 2.0 dB as illustrated in Fig. 9(a)–(h). Samples of spin-cycled de-noised projections (originally 2.0 dB) are shown in Fig. 8 in solid red.

For the sorting algorithm to succeed, we need to be able to identify local neighborhoods along the projections curve. The information that is required for such identification is carried by a few robust features. For example, the support of the projection or the number of peaks in it provide very strong cues for its neighboring projections. Without de-noising, the Euclidean distance between projections is dominated by noise. However, since these features are very robust, they survive even a very

aggressive de-noising procedure. This suggests that designing a suitable metric to construct the graph in (4) is of great practical importance. Spatial correlations take no advantage of robust features specific to the underlying problem, and the result is dominated by noise. Applying a specifically designed metric (de-noising) pulls out these features even in very high noise levels.

V. Summary and Discussion

In this paper, we introduced a graph Laplacian-based algorithm for imaging a planar object given its projections at random unknown directions. The graph Laplacian is widely used nowadays in machine learning and high-dimensional data analysis, however, its usage in tomography seems to be new. The graph Laplacian embeds the projection functions into a closed planar curve from which we estimate the projection angles. The graph Laplacian ensembles local projection similarities into a global embedding of the data. In that respect, our algorithm may be viewed as the natural generalization of the nearest neighbor algorithm of [6], and can be viewed as an approximate solution to the traveling salesman problem in high dimensions.

We tested the graph Laplacian reconstruction algorithm for the Shepp–Logan phantom and examined its tolerance to noise. We observed the threshold effect and were able to improve the tolerance to noise by constructing a graph Laplacian based on de-noised projections. Our success in pushing down the threshold limit using the wavelets spin-cycle algorithm suggests that more sophisticated filtering techniques may tolerate even higher levels of noise. In particular, we speculate that filtering the entire set of projection vectors all together, rather than one at a time, using neighborhood filters [26], nonlocal means [27] and functions adapted kernels [28], may push down the threshold even further. The original non-noisy projection vectors have similar features or building blocks (e.g., peaks, jumps, quiet regions, etc.) when the underlying imaged object is not too complex. We expect better recovery of those features when averaging similar slices across many different projections.

The application which motivated the current work is the reconstruction of 3-D objects from their line integrals taken at random unknown orientations, as is the case for cryo-electron microscopy of molecular structures [10]–[12]. The algorithm presented in this paper relies on the special structure of the underlying projections manifold. That is, the fact that the projections manifold is a curve in $\mathbb{R}^n$. When considering the 3-D reconstruction problem, the underlying manifold is no longer a curve but rather a 2-D surface. Therefore, the presented method is not directly applicable, as manifolds which are not curves have no notion of order. To extend our method to higher dimensions, we need to take a different approach. Instead of using the embedding to order the projections, it is possible to modify the graph Laplacian so that the embedding gives directly the orientation of each projection, by constructing the Laplacian over the orientations manifold instead over the projections manifold. Such an extension can be derived along the lines of the approach in [29]. However, in the case of 3-D random tomography, the special geometry induced by the Fourier slice theorem gives rise to a much simpler and numerically superior approach. Instead of constructing the graph Laplacian from the projection data, it is possible to design a different operator (an averaging operator), whose eigenvectors are exactly the projection orientations. The relation between the orientations and the eigenvectors of this operator is exact, and not asymptotic as for the graph Laplacian case [see (8)]. This will be described in a separate publication.

References

[1] S. R. Deans, *The Radon Transform and Some of its Applications*. New York: Krieger, 1993.

[2] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*, ser. Classics in Applied Mathematics. Philadelphia, PA: SIAM, 2001.

[3] F. Natterer, *The Mathematics of Computerized Tomography*, ser. Classics in Applied Mathematics. Philadelphia, PA: SIAM, 2001.

[4] F. Natterer and F. Wûbbeling, *Mathematical Methods in Image Reconstruction*, ser. Mographs on Mathematical Modeling and Computation, 1st ed. Philadelphia, PA: SIAM, 2001.

[5] S. Basu and Y. Bresler, "Uniqueness of tomography with unknown view angles," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1094–1106, Jun. 2000.

[6] S. Basu and Y. Bresler, "Feasibility of tomography with unknown view angles," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1107–1122, Jun. 2000.

[7] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 5–30, Jul. 2006.

[8] S. Lafon, "Diffusion maps and geometric harmonics" Ph.D. dissertation, Yale Univ., New Haven, CT, 2004.

[9] R. R. Coifman and D. Donoho, "Translation invariant de-noising," in *Wavelets in Statistics*, A. Antoniadis and G. Oppenheim, Eds. New York: Springer, 1995, pp. 125–150.

[10] J. Frank, *Three-Dimensional Electron Microscopy of Macromolecular Assemblies: Visualization of Biological Molecules in Their Native State*. Oxford, U.K.: Oxford Univ. Press, 2006.

[11] P. Doerschuk and J. E. Johnson, "Ab initio reconstruction and experimental design for cryo electron microscopy," *IEEE Trans. Inf. Theory*, vol. 46, no. 5, pp. 1714–1729, Aug. 2000.

[12] M. van Heel, B. Gowen, R. Matadeen, E. V. Orlova, R. Finn, T. Pape, D. Cohen, H. Stark, R. Schmidt, M. Schatz, and A. Patwardhan, "Single-particle electron cryo-microscopy: towards atomic resolution," *Q. Rev. Biophys.*, vol. 33, pp. 307–369, 2001.

[13] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, pp. 1373–1396, 2003.

[14] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proc. Nat. Acad. Sci.*, vol. 102, no. 21, pp. 7426–7431, 2005.

[15] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Multiscale methods," *Proc. Nat. Acad. Sci.*, vol. 102, no. 21, pp. 7432–7437, 2005.

[16] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[17] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[18] A. Singer, "From graph to manifold Laplacian: The convergence rate," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 128–134, Jul. 2006.

[19] U. von Luxburg, O. Bousquet, and M. Belkin, "Limits of spectral clustering," in *Advances in Neural Information Processing Systems (NIPS)*, L. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2005, vol. 17, pp. 857–864.

[20] B. Nadler, S. Lafon, R. R. Coifman, and I. Kevrekidis, "Diffusion maps, spectral clustering and eigenfunctions of Fokker–Planck operators," in *Advances in Neural Information Processing Systems*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA: MIT Press, 2006, vol. 18, pp. 955–962.

[21] E. A. Coddington and N. Levinson, *Theory of Ordinary Differential Equations*. New York: McGraw-Hill, 1984.

[22] H. A. David and H. N. Nagaraja, *Order Statistics*, 3rd ed. New York: Wiley, 2003.

[23] M. Hein and Y. Audibert, L. De Raedt and S. Wrobel, Eds., "Intrinsic dimensionality estimation of submanifolds in $\mathbb{R}^d$," in *Proc. 22nd Int. Conf. Machine Learning*, 2005, pp. 289–296.

[24] M. Zakai and J. Ziv, "On the threshold effect in radar range estimation," *IEEE Trans. Inf. Theory*, vol. 15, no. 1, pp. 167–170, Jan. 1969.
[25] J. Ziv and M. Zakai, "Some lower bounds on signal parameter estimation," *IEEE Trans. Inf. Theory*, vol. IT-15, no. 3, pp. 386–391, May 1969.
[26] L. P. Yaroslavsky, *Digital Picture Processing-An Introduction*. Berlin, Germany: Springer-Verlag, 1985.
[27] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, 2005.
[28] A. D. Szlam, M. Maggioni, and R. R. Coifman, "A general framework for adaptive regularization based on diffusion processes on graphs," *J. Mach. Learn. Res.*, to be published.
[29] A. Singer and R. R. Coifman, "Non linear independent component analysis with diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 25 (2), pp. 226–239, 2008.

Ronald R. Coifman received the Ph.D. degree from the University of Geneva, Geneva, Switzerland, in 1965.

He is a member of the National Academy of Sciences and the American Academy of Arts and Sciences. He is the Phillips Professor of Mathematics at Yale University, New Haven, CT. His research interests include: nonlinear Fourier analysis, wavelet theory, singular integrals, numerical analysis and scattering theory, and new mathematical tools for efficient computation and transcriptions of physical data, with applications to numerical analysis, feature extraction recognition, and denoising. He received the DARPA Sustained Excellence Award in 1996 and the 1999 Pioneer Award from the International Society for Industrial and Applied Mathematics. He is a recipient of National Medal of Science.

Yoel Shkolnisky received his B.Sc. degree in mathematics and computer science and the M.Sc. and Ph.D. degrees in computer science from Tel-Aviv University, Tel-Aviv, Israel, in 1996, 2001, and 2005, respectively.

From July 2005 to July 2008, he was a Gibbs Assistant Professor in applied mathematics at the Department of Mathematics, Yale University, New Haven, CT. His research interests include computational harmonic analysis, scientific computing, and data analysis.

Fred J. Sigworth was born in Berkeley, CA, in 1951. He received the B.S. degree in applied physics from the California Institute of Technology, Pasedena, in 1974, and the Ph.D. degree in physiology from Yale University, New Haven, CT, in 1979.

He was a Research Fellow at the Max Planck Institute for Biophysical Chemistry, Göttingen, Germany, from 1979 to 1984, where he worked with E. Neher on the development and applications of the patch-clamp technique for recording single ion-channel currents. Since 1984, he has been a Faculty Member with the Department of Cellular and Molecular Physiology, Yale University. His research interests include the structure and function of ion channel proteins and the development of techniques to study ion channels.

Dr. Sigworth is a Member of Biophysical Society, the Society of Neuroscience, and the American Scientific Affiliation.

Amit Singer received the B.Sc. degree in physics and mathematics and the Ph.D. degree in applied mathematics from Tel Aviv University, Tel Aviv, Israel, in 1997 and 2005, respectively.

From July 2005 to July 2008, he was a Gibbs Assistant Professor in applied mathematics at the Department of Mathematics, Yale University, New Haven, CT. Since July 2008, he has been an Assistant Professor in the Department of Mathematics and PACM (Program in Applied and Computational Mathematics), Princeton University, Princeton, NJ. He served in the Israeli Defense Forces from 1997–2003. His research in applied mathematics focuses on problems of statistical data analysis, structural biology, molecular biophysics, and statistical physics.