Data Analysis and Representation on a General Domain using Eigenfunctions of Laplacian

Naoki Saito

Department of Mathematics

University of California

Davis, CA 95616 USA

Abstract

We propose a new method to analyze and represent data recorded on a domain of general shape in $\mathbb{R}^d$ by computing the eigenfunctions of Laplacian defined over there and expanding the data into these eigenfunctions. Instead of directly solving the eigenvalue problem on such a domain via the Helmholtz equation (which can be quite complicated and costly), we find the integral operator commuting with the Laplacian and diagonalize that operator. Although our eigenfunctions satisfy neither the Dirichlet nor the Neumann boundary condition, computing our eigenfunctions via the integral operator is simple and has a potential to utilize modern fast algorithms to accelerate the computation. We also show that our method is better suited for small sample data than the Karhunen-Loève Transform/Principal Component Analysis. In fact, our eigenfunctions depend only on the shape of the domain, not the statistics of the data. As a further application, we demonstrate the use of our Laplacian eigenfunctions for solving the heat equation on a complicated domain.

Key words: Laplacian eigenfunctions, boundary conditions, Green's function, spectral decomposition, Karhunen-Loève transform, principal component analysis, heat equation

1 Introduction

Most of the currently available signal and image processing tools were designed and developed for signals and images that are sampled on regular/uniform grids and supported on a rectangular or cubic domain. For example, the conventional Fourier analysis using complex exponentials, sines and/or cosines, have been the crown jewels for such data. On the other hand, there is an increasing desire to analyze data sampled on irregular grids (e.g., meteorological data sampled at weather stations) or objects defined on a domain of general shape (e.g., cells in histological images). Unfortunately, the conventional tools cannot efficiently handle such data and objects. In this paper, we propose a new technique that can analyze spatial frequency information of such data and objects, filter the frequency contents if one wishes, and synthesize the data and objects at one's disposal. This is a direct generalization of the conventional Fourier analysis and synthesis. Our new tool explicitly incorporates geometric configuration of the domain or spatial location of the sensors. This is quite a contrast to the popular Karhunen-Loève Transform (KLT) or Principal Component Analysis (PCA), which only implicitly incorporate such geometric information via covariance. One of the goals of this paper is to demonstrate our tool's superiority over the KLT/PCA for such datasets.

Let us consider a bounded domain of general shape $\Omega \subset \mathbb{R}^d$, where typically $d = 2$ or 3. Let us also assume that the boundary $\Gamma = \partial\Omega$ consists of piecewise C^2

Email address: saito@math.ucdavis.edu (Naoki Saito).

surfaces (although one may be able to weaken this assumption by more subtle arguments). We want to analyze the spatial frequency information *inside* of the object (i.e., the data measured in Ω) without the annoying interference by the Gibbs phenomenon due to the boundary Γ. We also want to represent the object compactly for analysis, interpretation, discrimination, and so on, by expanding it into a basis that generates fast decaying expansion coefficients.

There are at least two approaches to this problem. One is to extend (or extrapolate) a general shape object smoothly to its outside, cut it by a circumscribed rectangle, and use the conventional tools to analyze the extended object on this rectangle. Using the idea from potential theory and elliptic partial differential equations, we developed the so-called generalized polyharmonic local trigonometric transform to do this extension and subsequent analysis [26], [35, Chap. 4]. Although this approach can analyze the spatial frequency contents of the object without being bothered by the Gibbs phenomenon, the resulting analysis (e.g., the Fourier cosine coefficients of the extended object) is still affected by the extended part which is smooth (in fact, harmonic) regardless of the intrinsic smoothness of the object inside the original domain Ω.

Instead, this paper proposes a second approach: find a genuine orthonormal basis tailored to the domain of general shape. To do so, we use the eigenfunctions of the Laplacian defined on the domain. After all, complex exponentials, sines, and cosines are the eigenfunctions of the Laplacian on a rectangular domain with specific boundary conditions, i.e., the periodic, the Dirichlet, and the Neumann boundary conditions, respectively. Also, our favorite special functions, e.g., spherical harmonics, Bessel functions, and prolate spheroidal wave functions, are again the part of the eigenfunctions of the Laplacian (via separation of variables) for spherical, cylindrical, and spheroidal domains, respectively.

The organization of this paper is as follows. In Section 2 we describe our proposed strategy for computing Laplacian eigenfunctions via eigenanalysis of an integral operator commuting with the Laplacian. In Section 3 we analyze a few simple examples that allow us to explicitly and analytically compute our Laplacian eigenfunctions using our proposed approach, which will be contrasted with the usual ones satisfying the Dirichlet or the Neumann boundary conditions. In Section 4, we examine our Laplacian eigenfunctions as a tool for approximating data on a complicated domain, compare its performance with that of the standard wavelet-based methods, and demonstrate the efficiency of our approach. In Section 5, we use our Laplacian eigenfunctions as a statistical data analysis tool, compare its performance with the standard KLT/PCA, and demonstrate that our tool can separate the statistics of data from the geometry of the domain where the data are supported, which is impossible for KLT/PCA. As a further application, we solve the heat equation on a complicated domain in Section 6 using our Laplacian eigenfunctions. In Section 7, we discuss a couple of strategies to speed up the Laplacian eigenfunction computations. Finally, we conclude this paper with our future research plan in Section 8.

2   Properties of the Laplacian Eigenfunctions and Their Computation

In this section, after briefly outlining the properties of the eigenfunctions of the Laplacian on a general domain, we describe our main idea on how to compute them.

Consider an operator $\mathcal{L} = -\Delta = -\dfrac{\partial^2}{\partial x_1{}^2} - \cdots - \dfrac{\partial^2}{\partial x_d{}^2}$ in $L^2(\Omega)$ with an *appropriate* boundary condition (we will be more specific about it later). The direct analysis of $\mathcal{L}$ is difficult due to its unboundedness that is well known and often covered

4

in any elementary functional analysis course (see e.g., [20]). A much better approach is to analyze its inverse $\mathcal{L}^{-1}$, which is referred to as the Green's operator because it is a *compact* and *self-adjoint* operator and consequently we can have a good grip of its spectral properties. In fact, $\mathcal{L}^{-1}$ for a reasonably regular boundary Γ has discrete spectrum (i.e., a countable number of eigenvalues with finite multiplicity) except 0 spectrum [8, Chap. 6, 7]. Moreover, thanks to this spectral property, $\mathcal{L}$ has a complete orthonormal basis of $L^2(\Omega)$, and this allows us to do *eigenfunction expansion* in $L^2(\Omega)$ [8,21].

The key difficulty is to compute such eigenfunctions. Directly solving the Helmholtz equation on a general domain, i.e., finding non-trivial solutions of $-\Delta\phi = \lambda\phi$ that satisfy $\mathcal{B}\phi = 0$ (where $\mathcal{B}$ is an operator specifying the boundary condition) is quite tough. Unfortunately, computing the Green's function for a general Ω satisfying the usual boundary condition such as the Dirichlet or the Neumann condition is also very difficult.

2.1 Integral operator commuting with the Laplacian

Our idea to avoid those difficulties is to find an integral operator *commuting* with the Laplacian without imposing the strict boundary condition a priori. Then, from the following well-known theorem (see e.g., [13, pp.63–67]), we know that the eigenfunctions of the Laplacian is the same as those of the integral operator that is much easier to deal with.

Theorem 1 *Let $\mathcal{K}$ and $\mathcal{L}$ be operators acting on $L^2(\Omega)$. Suppose $\mathcal{K}$ and $\mathcal{L}$ commute and one of them has an eigenvalue with finite multiplicity. Then, $\mathcal{K}$ and $\mathcal{L}$ share the same eigenfunction corresponding to that eigenvalue, i.e., there exists a*

function $\phi \in L^2(\Omega)$ such that $\mathcal{K}\phi = \mu\phi$ and $\mathcal{L}\phi = \lambda\phi$.

Here is the key step in our development. Let us replace the Green's function $G(\boldsymbol{x}, \boldsymbol{y})$ (the kernel of the Green's operator) by the *fundamental solution of the Laplacian* or the *harmonic kernel*:

$$K(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\Delta}{=} \begin{cases} -\frac{1}{2}|x - y| & \text{if } d = 1, \\ -\frac{1}{2\pi}\ln|\boldsymbol{x} - \boldsymbol{y}| & \text{if } d = 2, \\ \frac{|\boldsymbol{x}-\boldsymbol{y}|^{2-d}}{(d-2)\omega_d} & \text{if } d > 2, \end{cases} \tag{1}$$

where $\omega_d \stackrel{\Delta}{=} \frac{2\pi^{d/2}}{\Gamma(d/2)}$ is the surface area of the unit ball in $\mathbb{R}^d$, and $|\cdot|$ is the standard Euclidean norm. The price we pay for this replacement is to have rather implicit and non-local boundary condition (which we will discuss shortly) although we do not have to deal with this boundary condition directly. Let $\mathcal{K}$ be the integral operator with its kernel $K(\boldsymbol{x}, \boldsymbol{y})$:

$$\mathcal{K}f(\boldsymbol{x}) \stackrel{\Delta}{=} \int_\Omega K(\boldsymbol{x}, \boldsymbol{y})f(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}, \quad f \in L^2(\Omega). \tag{2}$$

We now have the following theorem.

Theorem 2 *The integral operator $\mathcal{K}$ commutes with the Laplacian $\mathcal{L} = -\Delta$ with the following* non-local *boundary condition:*

$$\int_\Gamma K(\boldsymbol{x}, \boldsymbol{y})\frac{\partial\phi}{\partial\nu_{\boldsymbol{y}}}(\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y}) = -\frac{1}{2}\phi(\boldsymbol{x}) + \mathrm{pv}\int_\Gamma \frac{\partial K(\boldsymbol{x}, \boldsymbol{y})}{\partial\nu_{\boldsymbol{y}}}\phi(\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y}), \tag{3}$$

for all $\boldsymbol{x} \in \Gamma$, where $\partial/\partial\nu_{\boldsymbol{y}}$ is the normal derivative operator at the point $\boldsymbol{y} \in \Gamma$ and $\mathrm{d}s(\boldsymbol{y})$ is the surface measure on Γ.

The proof of this theorem is given in Appendix A.

Consequently, we also have the following theorem (see e.g., [21, Sec. 4.5]).

Theorem 3 *The integral operator* $\mathcal{K}$ *is compact and self-adjoint on* $L^2(\Omega)$. *Thus, the kernel* $K(\boldsymbol{x}, \boldsymbol{y})$ *has the following eigenfunction expansion (in the sense of mean convergence):*

$$K(\boldsymbol{x}, \boldsymbol{y}) \sim \sum_{j=1}^{\infty} \mu_j \phi_j(\boldsymbol{x}) \overline{\phi_j(\boldsymbol{y})},$$

and $\{\phi_j\}_{j \in \mathbb{N}}$ *forms an orthonormal basis of* $L^2(\Omega)$.

We will use the basis $\{\phi_j\}_{j \in \mathbb{N}}$ to expand and represent the data supported on Ω.

Remark 4 *These eigenfunctions of the Laplacian are closely related to the so-called* Geometric Harmonics *proposed by Coifman and Lafon [6]. After all, our eigenfunctions are a specific example of the geometric harmonics with a specific kernel* (1). *Nevertheless, there are some important differences between their objectives and methods with those of ours. First of all, their motivation is to obtain the* extrinsic *geometric information of a given dataset or a manifold, i.e., how to extend a given function to the outside of the domain for various machine learning and statistical regression purposes. On the contrary, our objective is to use the Laplacian eigenfunctions for the* intrinsic *analysis of the data defined on the domain.*

Secondly, the geometric harmonics defined by Coifman and Lafon include not only harmonic kernel (1) *but also the kernels that do not commute with the usual Laplacian. Two such examples are: 1) the so-called* bandlimited *kernel:*

$$K_B(\boldsymbol{x}, \boldsymbol{y}) \triangleq \left(\frac{B}{2} \right)^{d/2} \frac{J_{d/2}(\pi B |\boldsymbol{x} - \boldsymbol{y}|)}{|\boldsymbol{x} - \boldsymbol{y}|^{d/2}},$$

where $J_{d/2}(\cdot)$ *is the Bessel function of the first kind of order* $d/2$ *and* $B > 0$ *is the bandwidth. The eigenfunctions of the integral operator with this kernel are the direct generalization of the prolate spheroidal wave functions [27]* [1] *; and 2) the*

---

[1] Although such bandlimited kernels do not commute with the usual Laplacian, they may commute with more general elliptic operators. For example, for $d = 1$, $K_B(x, y)$ be-

popular Gaussian kernel:

$$G_B(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\Delta}{=} \mathrm{e}^{-B^2|\boldsymbol{x}-\boldsymbol{y}|^2}.$$

There are some intrinsic differences between these kernels and the harmonic kernel (1). Without going into too much details, let us say that the integral operator with the bandlimited kernel is more difficult to deal with numerically due to its oscillatory nature. The eigenfunctions associated with the Gaussian kernel have intrinsic "scale" related to the parameter B while the Laplacian eigenfunctions are "global" in nature. We rather prefer to use the harmonic kernel $K(\boldsymbol{x}, \boldsymbol{y})$ (1) for our intrinsic data analysis purposes because it is: 1) easier to deal with mathematically; 2) possible to make it multiscale by explicitly splitting the domain; and 3) more amenable to fast algorithms such as the wavelets and Fast Multipole Method; see Section 7 for more about these aspects.

2.2 Discretization of the eigenvalue problem

Let us recall our original aim once again. We would like to analyze an object of general shape defined on a digitized image or 3D dataset using the Laplacian eigenfunctions. Therefore, we must discretize our eigenvalue problem in order to compute the eigenfunctions and analyze such an object. In this subsection, we describe our discretization strategy and assumptions on the dataset.

Let us first assume that the whole dataset consists of a collection of data sampled on a regular grid, and that each sampling cell is a box of size $\prod_{i=1}^{d} \Delta x_i$. Let us also assume that the shape of an object of our interest Ω consists of a subset of those cells or boxes (i.e., pixels in 2D and voxels in 3D), and the object itself is defined

---

comes the sinc kernel $\frac{\sin \pi B(x-y)}{\pi(x-y)}$, and the corresponding integral operator commutes with the differential operator $\mathcal{L} = \frac{\mathrm{d}}{\mathrm{d}x}(1-x^2)\frac{\mathrm{d}}{\mathrm{d}x} - \pi^2 B^2 x^2$.

as a collection of the data samples taken on those cells. Let $\{\boldsymbol{x}_i\}_{i=1}^N \subset \Omega$ be the centers of those cells in Ω. Under these assumptions, we can approximate the integral eigenvalue problem $\mathcal{K}\phi = \mu\phi$, where $\mathcal{K}$ is defined in (2), with a simple quadrature rule with node-weight pairs $(\boldsymbol{x}_j, w_j)$ as follows.

$$\sum_{j=1}^N w_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)\phi(\boldsymbol{x}_j) = \mu\phi(\boldsymbol{x}_i), \quad i = 1, \dots, N.$$

Let us use the simplest weights $w_j = \prod_{i=1}^d \Delta x_i$, with which the above quadrature rule becomes the midpoint rule. Let $K_{i,j} \stackrel{\Delta}{=} w_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $\phi_i \stackrel{\Delta}{=} \phi(\boldsymbol{x}_i)$, and $\boldsymbol{\phi} \stackrel{\Delta}{=} (\phi_1, \dots, \phi_N)^T \in \mathbb{R}^N$. Then, the above equation can be written in a matrix-vector form as: $K\boldsymbol{\phi} = \mu\boldsymbol{\phi}$, where $K = (K_{ij}) \in \mathbb{R}^{N \times N}$. Under our assumptions, the weight w_j does not depend on j, which makes K *symmetric*.

Once the discretized eigenvalue problem, $K\boldsymbol{\phi} = \mu\boldsymbol{\phi}$, is formed, we can compute its eigenvectors (and the corresponding eigenvalues). In this paper, we use the conventional technique to compute the eigenvalues and eigenvectors of such a matrix, i.e., a slow algorithm of $O(N^3)$, where N is the number of samples in the discretization process. We note, however, that we can considerably speed up the eigenvector computation, i.e., up to $O(N \log N)$ or $O(N^2)$ using the wavelets or the Fast Multipole Method, which we will briefly discuss in Section 7.

Remark 5 *For analyzing data generated by a distributed sensor network, the location of the sensors are often very irregular and we cannot assume the weight w_j above is independent of the sensor location $\{\boldsymbol{x}_j\}_{j=1}^N$. In order to set up a correct discretized eigenvalue problem under such circumstances, we may need to either: 1) compute w_j using the Voronoi diagram (or some other computational geometric tools); or 2) interpolate on regular grids from the given sample points $\{\boldsymbol{x}_j\}_{j=1}^N$. We are currently investigating these cases and will report our findings at a later date.*

3 Examples

In this section, we will show a few analytic examples to contrast our eigenfunctions with the conventional basis functions to deepen our understanding of those eigenfunction-based representation.

3.1 1D Example

Consider the unit interval $\Omega = (0,1)$. Then, the kernel of the integral operator becomes $K(x,y) = -|x - y|/2$, and we can obtain the Laplacian eigenfunctions explicitly as the following corollary of Theorem 2 shows.

Corollary 6 *The eigenfunctions of the integral operator $\mathcal{K}$ for the unit interval $\Omega = (0,1)$ satisfy the following Laplacian eigenvalue problem:*

$$-\phi'' = \lambda\phi, \quad x \in (0,1);$$

$$\phi(0) + \phi(1) = -\phi'(0) = \phi'(1), \tag{4}$$

which can be solved explicitly as follows.

- $\lambda_0 \approx -5.756915$ *is the smallest (and the only negative) eigenvalue and is the solution of the following secular equation:*

$$\tanh\frac{\sqrt{-\lambda_0}}{2} = \frac{2}{\sqrt{-\lambda_0}}. \tag{5}$$

The corresponding eigenfunction is:

$$\phi_0(x) = A_0 \cosh\sqrt{-\lambda_0}\left(x - \tfrac{1}{2}\right), \tag{6}$$

where $A_0 = \sqrt{2}\left(1 + \frac{\sinh\sqrt{-\lambda_0}}{\sqrt{-\lambda_0}}\right)^{-1/2} \approx 0.7812598$ is a normalization constant to have $\|\phi_0\|_{L^2(\Omega)} = 1$.

10

- $\lambda_{2m-1} = (2m-1)^2\pi^2$, $m = 1, 2, \ldots$, and the corresponding eigenfunction is:

$$\phi_{2m-1}(x) = \sqrt{2}\cos(2m-1)\pi x. \tag{7}$$

These are canonical cosines with odd modes.

- λ_{2m}, $m = 1, 2, \ldots$, is the solution of the secular equation:

$$\tan\frac{\sqrt{\lambda_{2m}}}{2} = -\frac{2}{\sqrt{\lambda_{2m}}}, \tag{8}$$

and the corresponding eigenfunction is:

$$\phi_{2m}(x) = A_{2m}\cos\sqrt{\lambda_{2m}}\left(x - \tfrac{1}{2}\right), \tag{9}$$

where $A_{2m} = \sqrt{2}\left\{1 + \frac{\sin\sqrt{\lambda_{2m}}}{\sqrt{\lambda_{2m}}}\right\}^{-1/2}$ is a normalization constant.

The proof of this corollary is given in Appendix B.

Figure 1 shows these Laplacian eigenfunctions of the lowest five frequencies.

Remark 7 *The kernel $K(x,y)$ is of* Toeplitz *form in this case, and consequently, the eigenvectors must have even and odd symmetry [4], which is indeed the case.*

Remark 8 *The Laplacian eigenfunctions for the Dirichlet boundary condition on the unit interval satisfy $-\phi'' = \lambda\phi$, $\phi(0) = \phi(1) = 0$, and they are sines. The Green's function in this case is:*

$$G_D(x,y) = \min(x,y) - xy.$$

Those satisfying the Neumann boundary condition, i.e., $\phi'(0) = \phi'(1) = 0$, are cosines, and its Green's function is:

$$G_N(x,y) = -\max(x,y) + \frac{1}{2}(x^2 + y^2) + \frac{1}{3}.$$

One can easily imagine that it is a rather difficult task to find these Green's functions for a general domain in higher dimensions. Incidentally, when we discretize
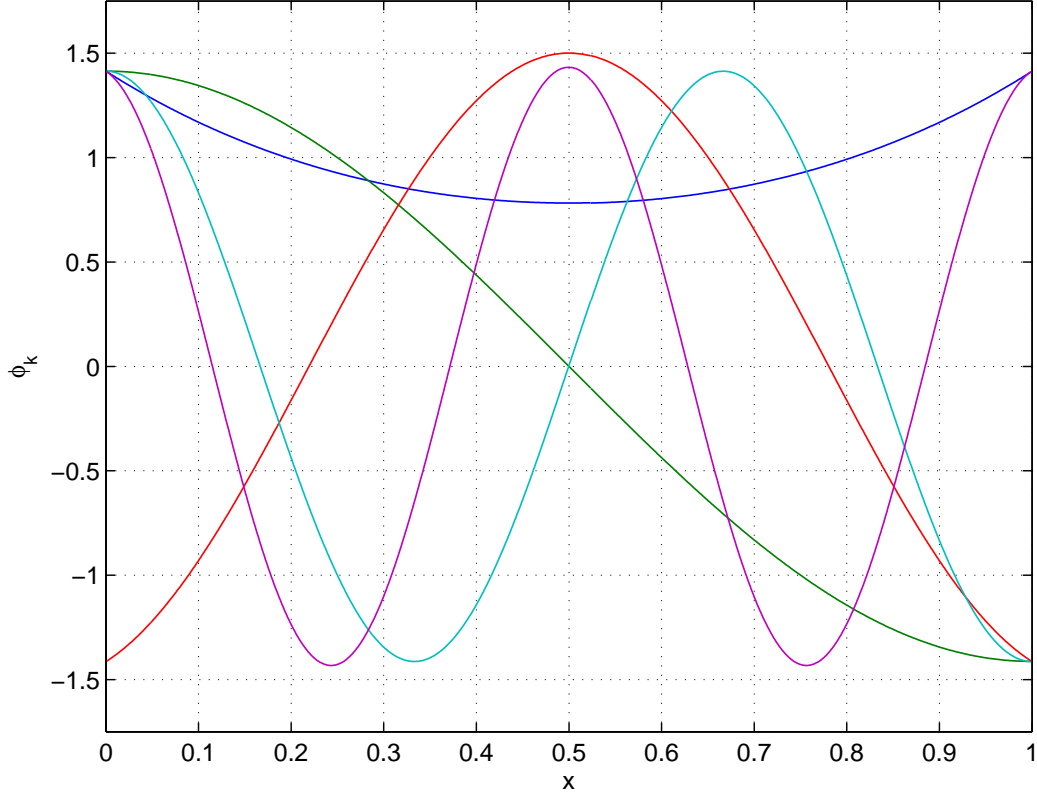
Fig. 1. The first five eigenfunctions of the Laplacian on the unit interval with the non-local boundary condition (4). The eigenfunctions with odd symmetry are in fact usual cosine functions. Those with even symmetry are cosh function or cosines with non-integer periodicity (also known as nonharmonic or almost-periodic cosines).

and approximate the Green's operator for the Dirichlet boundary condition by the gridpoint sampling (i.e., sampling at $\left\{x_j = j/N\right\}_{j=0}^{N}$ over the interval $[0,1]$) and the trapezoidal rule, then the eigenvectors are the so-called Discrete Sine Transform Type I (DST-I for short). The same discretization scheme for the Neumann boundary condition leads to the Discrete Cosine Transform Type I (DCT-I for short). In other words, these matrices commutes with the differentiation matrices with appropriate boundary conditions listed in [28, p. 140]. Here, one can also see that the asymmetry of the discretized kernel matrix for the Neumann boundary condition due to the trapezoidal rule corresponds to the special weighting at the two end points used for the DCT-I basis vectors to be orthonormal. On the other hand,

when we discretize these Green's operators with midpoint sampling (i.e., sampling at $\left\{ x_j = (j+0.5)/N \right\}_{j=0}^{N-1}$ over the interval $[0,1]$) and use the midpoint rule for the integration, then we obtain DST-II/DCT-II basis vectors as the eigenvectors, which do not require any special weighting of the end points.

Remark 9 *It is interesting to note that our boundary condition* (4) *is somewhat similar to the so-called the Robin boundary condition, i.e.,*

$$u(0) - a_0\, u'(0) = 0 = u(1) + a_1\, u'(1),$$

where a_0 and a_1 are some constants. These boundary conditions happen to generate similar eigenfunctions as ours and the eigenvalues are also determined by solving similar secular equations. See [29, Sec. 4.3] for the details. However, these are still different from ours, i.e., these are local boundary condition whereas ours are nonlocal.

3.2   2D Example

Let us now consider the unit disk Ω in $\mathbb{R}^2$, where the kernel of our integral operator $\mathcal{K}$ becomes $K(\boldsymbol{x}, \boldsymbol{y}) = -\frac{1}{2\pi} \ln |\boldsymbol{x} - \boldsymbol{y}|$ for $\boldsymbol{x}, \boldsymbol{y} \in \overline{\Omega}$. Tailoring Theorem 2 for the unit disk, we have the following corollary:

Corollary 10 *The eigenfunctions of the integral operator $\mathcal{K}$ for the unit disk in $\mathbb{R}^2$ satisfies the following Laplacian eigenvalue problem:*

$$-\Delta\phi = \lambda\phi, \quad in \;\; \Omega;$$

$$\left.\frac{\partial\phi}{\partial\nu}\right|_\Gamma = \left.\frac{\partial\phi}{\partial r}\right|_\Gamma = -\left.\frac{\partial\mathcal{H}\phi}{\partial\theta}\right|_\Gamma, \tag{10}$$

where $\mathcal{H}$ is the Hilbert transform on the unit circle in $\mathbb{R}^2$, i.e.,

$$\mathcal{H}f(\theta) \triangleq \frac{1}{2\pi} \, pv \int_{-\pi}^{\pi} f(\eta) \cot\left(\frac{\theta - \eta}{2}\right) d\eta \quad \theta \in [-\pi, \pi].$$

Moreover, these eigenfunctions are of the form:

$$\phi_{m,n}(r,\theta) = \begin{cases} J_m(\beta_{m-1,n}r)\binom{\cos}{\sin}(m\theta) & \text{if } m = 1,2,\ldots, \; n = 1,2,\ldots, \\ J_0(\beta_{0,n}r) & \text{if } m = 0, \; n = 1,2,\ldots, \end{cases}$$

where $\beta_{k,\ell}$ is the ℓ th zero of the Bessel function of the first kind of order k, i.e., $J_k(\beta_{k,\ell}) = 0$. The corresponding eigenvalues are

$$\lambda_{m,n} = \begin{cases} \beta_{m-1,n}^2, & \text{if } m = 1,\ldots, \; n = 1,2,\ldots, \\ \beta_{0,n}^2 & \text{if } m = 0, \; n = 1,2,\ldots. \end{cases} \tag{11}$$

See Appendix C for the proof. Note that $\beta_{k,\ell}$ are tabulated in many places, e.g., [1, Table 9.5].

Remark 11 *This corollary suggests that out of the Laplacian eigenfunctions computed with our formulation, those corresponding to J_0, i.e., the radially symmetric eigenfunctions satisfy the Dirichlet boundary condition, but the other eigenfunctions do not. Also, we note that there are three eigenfunctions corresponding to each $\beta_{0,n}$, namely, $J_0(\beta_{0,n}r)$, $J_1(\beta_{0,n}r)\cos\theta$, and $J_1(\beta_{0,n}r)\sin\theta$, which can also be numerically confirmed.*

It is also interesting to compare (11) with the eigenvalues $\lambda_{m,n}^D$ of the Dirichlet-Laplacian and $\lambda_{m,n}^N$ of the Neumann-Laplacian:

$$\lambda_{m,n}^D = \beta_{m,n}^2, \quad \lambda_{m,n}^N = \alpha_{m,n}^2, \quad \text{for } m = 0,1,\ldots, \; n = 1,2,\ldots,$$
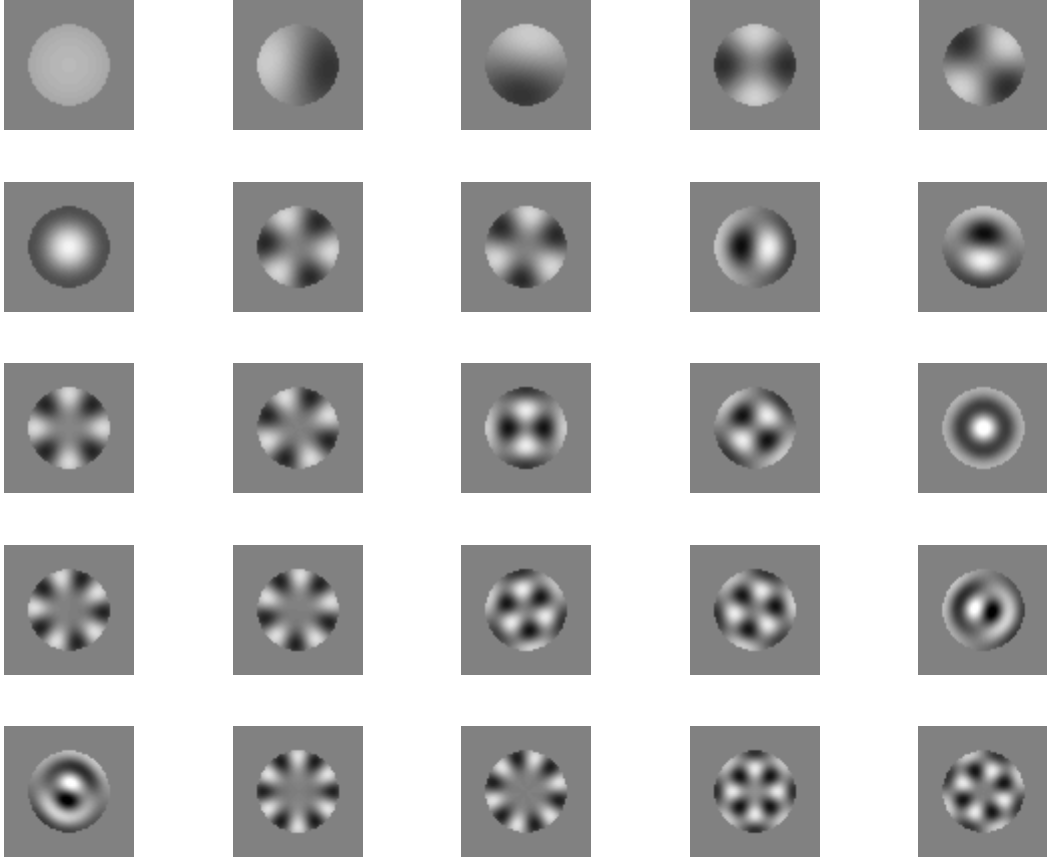
where $J_m'(\alpha_{m,n}) = 0$.

Fig. 2. The first 25 Laplacian eigenfunctions on the unit disk.

Figure 2 shows the Laplacian eigenfunctions corresponding to the lowest 25 frequencies that were computed numerically using our formulation. In other words, they are the eigenfunctions of the integral operator commuting with the Laplacian with the boundary condition (10). Note that the lowest frequencies (i.e., the smallest eigenvalues) in the Laplacian side correspond to the largest eigenvalues in the integral operator side. These eigenfunctions can be viewed as "modes" of the vibration of the domain if the domain is interpreted as a "drum" although not all of them satisfy the Dirichlet boundary condition.

(a) χ_Ω (b) $\chi_\Omega \cdot$ Barbara
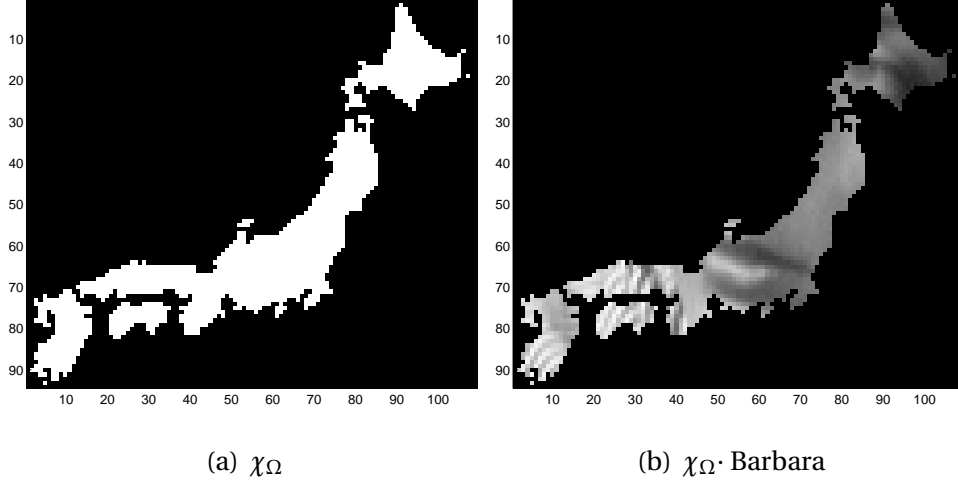
Fig. 3. The characteristic function of the Japanese Islands (a) and the Barbara image overlaid over the islands (b).

4 Application to Image Approximation

In this section, we examine the approximation capability of our Laplacian eigenfunctions for given image data on a rather irregular domain, and compare the performance with that of the standard wavelet-based methods.

As an example of irregularly-shaped domains, we decided to use a coarsely digitized image of the islands of Japan, as shown in Figure 3(a), which was obtained by the google image search. We then defined the characteristic function $\chi_\Omega(\boldsymbol{x})$ to indicate the shape of the islands. As for the data living on this domain Ω, we multiplied the standard Barbara image with χ_Ω, which is shown in Figure 3(b). The number of samples (i.e., pixels) forming the data on the islands is 1625. We computed the Laplacian eigenfunctions defined on Ω from the kernel matrix of 1625×1625. We display the first 25 eigenfunctions in Figure 4.

We then computed the 1625 expansion coefficients relative to this Laplacian eigenbasis, sorted in terms of their magnitudes, and approximated the data using the top 100 coordinates. In other words, we performed 100-term nonlinear
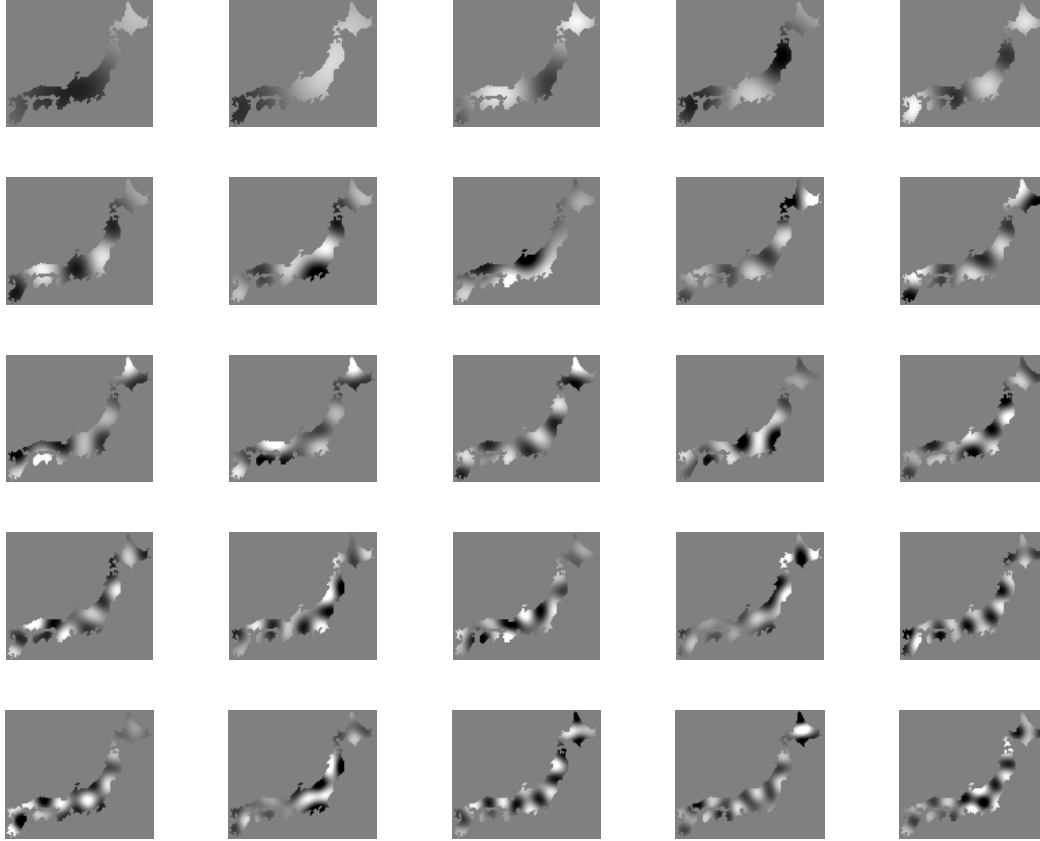
Fig. 4. The first 25 Laplacian eigenfunctions over the islands of Japan.

approximation using the Laplacian eigenfunctions. The result is shown in Figure 5(a) and the reconstruction error (or the residual) is shown in Figure 5(b). Note that these two figures are displayed with different dynamic ranges in order to the details. We note that the scarf region of the Barbara image was not captured well by these 100 terms of the Laplacian eigenfunctions. To capture the high frequency features, we need more terms.

We also approximated the same image using the top 100 coefficients computed by the standard 2D wavelet basis called "Symmlet 8" [7, pp. 198–199]. Note that this comparison is not really fair in the sense that the input image to the 2D wavelet transform is the whole rectangular image shown in Figure 3(b), i.e., including not only the islands, but also the outer ocean part. The approximation

17

(a) 100-term Approx.
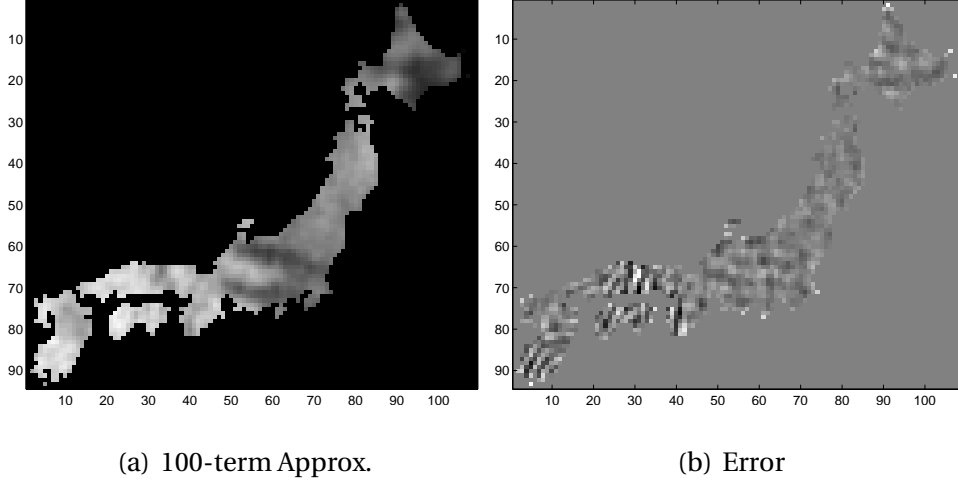
(b) Error

Fig. 5. The 100-term approximation and the residual error using the Laplacian eigen-functions.

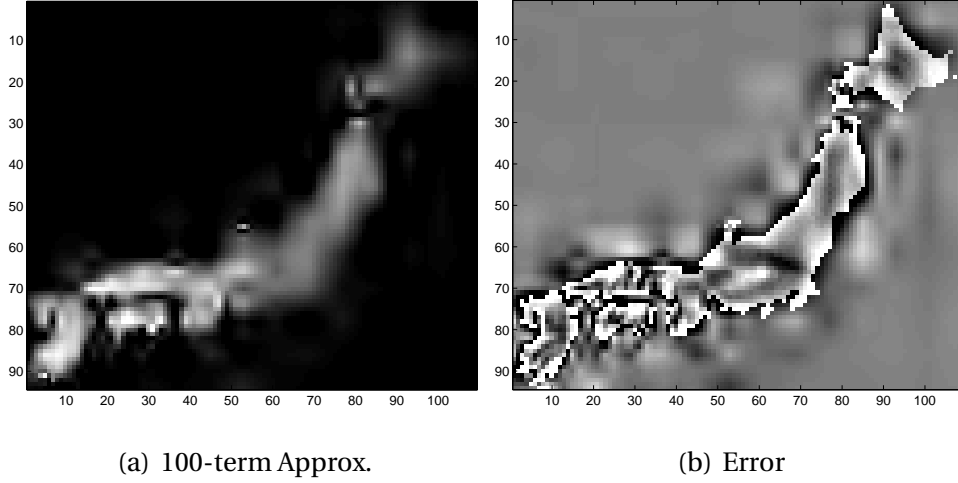

(a) 100-term Approx.

(b) Error

Fig. 6. The 100-term approximation and the residual error using the 2D Wavelets (Symmlet 8).

and its residual error are shown in Figure 6. In this case, most of the top 100 wavelet coordinates were used to capture the boundary of the islands and could not afford to capture the internal structure within the domain. It is clear that simply keeping the top 100 wavelet coefficients was not enough to capture even the boundary of the domain precisely for this image.

To be fairer, we organized these 1625 data points into a one-dimensional array by scanning each column of Figure 3(b), and applied the 1D wavelet transform

(a) 100-term Approx.　　　　　　　　(b) Error
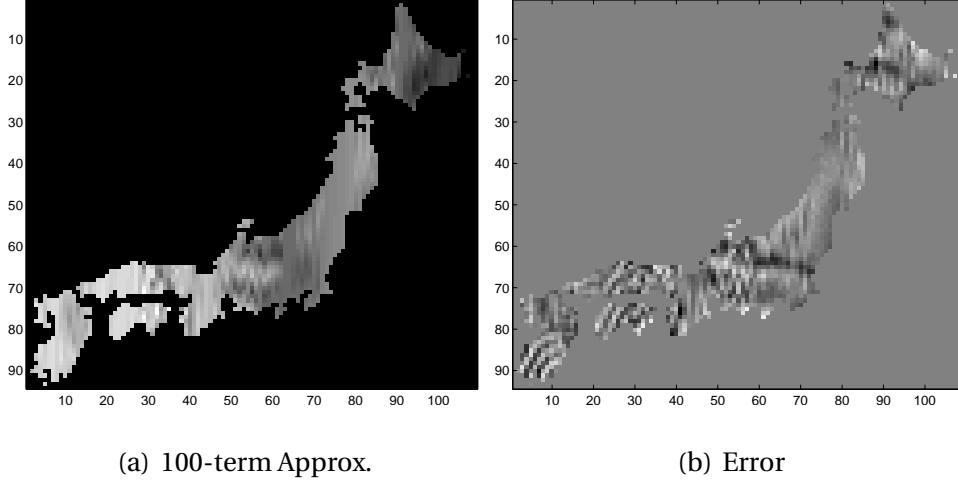
Fig. 7. The 100-term approximation and the residual error using the 1D Wavelets (Symmlet 8).

using the Symmlet 8 filter to the resulting 1D array. Then, the 100-term approximation and its residual error were computed. The results are shown in Figure 7. In this case, the boundary shape was not blurred because we explicitly used the geometric information. However, observe the stripe-shape artifacts in the approximation and in the residual. This is because we deconstructed the 2D spatial coherency of the original data by putting them into the 1D array.

Figure 8 compares the relative ℓ^2 errors of the m-term approximations of various methods. From these plots, we can observe that the nonlinear approximation using the Laplacian eigenfunctions was the best in terms of both the ℓ^2 error and the visual fidelity. We also would like to note that the difference in ℓ^2 error between the approximation using the Laplacian eigenfunctions and that by the 1D wavelet transform is not great while the visual fidelity of the former is far better than the latter as can be seen in Figures 5(a) and 7(a). It is important to realize that the visual fidelity may not be measured well by the ℓ^2 error; see e.g., [32] for the details about this intricate problem.

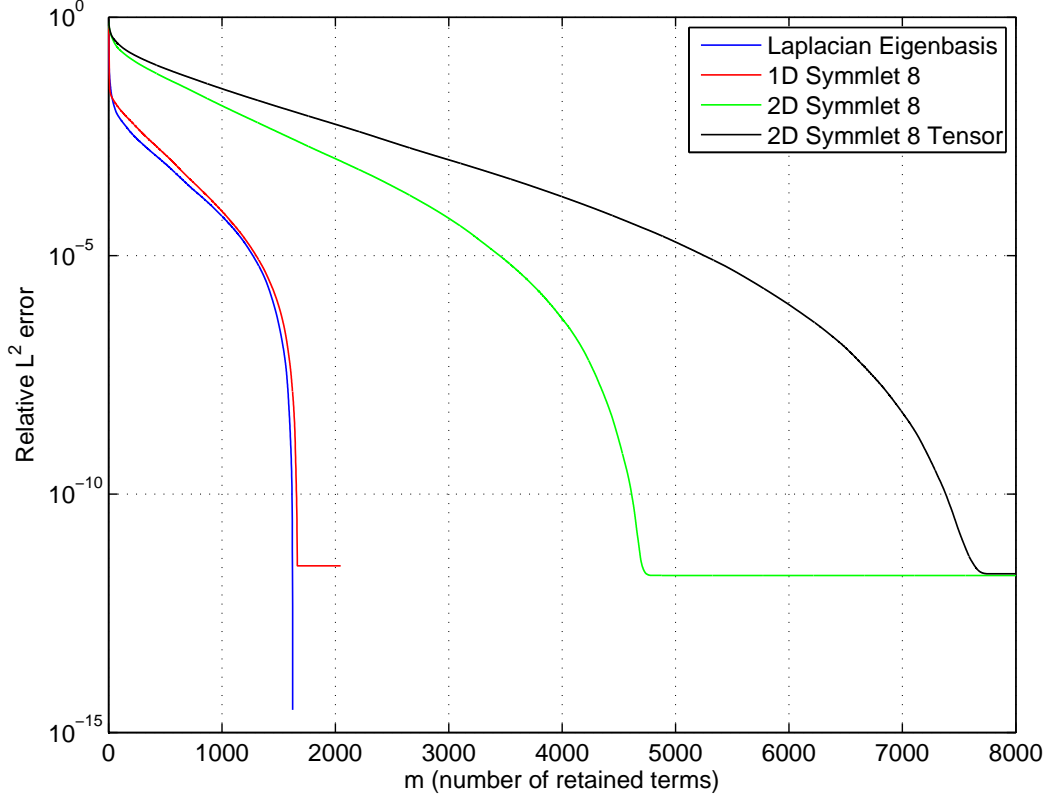For a variety of applications, we wish to prove the following conjecture:

19

Fig. 8. Comparison of the relative ℓ^2 errors of the m-term nonlinear approximations of various transforms. '2D Symmlet 8 Tensor' means the tensor product of 1D Symmlet 8 wavelet transform along rows and columns whereas '2D Symmlet 8' means the 2D wavelet transform based on the genuine 2D multiresolution analysis.

Conjecture 12 *Let Ω be a C^2-domain of general shape and let $f \in C\left(\overline{\Omega}\right)$ with $\dfrac{\partial f}{\partial x_j} \in BV\left(\overline{\Omega}\right)$ for $j = 1, \ldots, d$. Let $\{c_k = \langle f, \phi_k \rangle\}_{k \in \mathbb{N}}$ be the expansion coefficients of f with respect to our Laplacian eigenbasis on this domain. Then, $|c_k|$ decays with rate $O(k^{-\alpha})$ with $1 < \alpha < 2$ as $k \to \infty$. Thus, the approximation error using the first m terms measured in the L^2-norm, i.e., $\|f - \sum_{k=1}^{m} c_k \phi_k\|_{L^2(\Omega)}$ should have a decay rate of $O(m^{-\alpha+0.5})$ as $m \to \infty$.*

Essentially, this conjecture implies that what our Laplacian eigenfunctions do for data on a general domain is similar to what the Fourier cosine/DCT basis functions do for data fully supported on a rectangular domain. This should be

contrasted with the Fourier sine or complex (i.e., periodic) Fourier bases for data on a rectangular domain, which provide only $\alpha = 1$ in the above approximation statement in general. See [25,34] for the details about the boundary effect of the conventional Fourier/trigonometric bases.

This conjecture was derived from our numerous numerical experiments and we shall show some of them here to support the above conjecture. In the following experiments, we prepared four different domains, all of which were derived from the Japanese Islands discussed above. More presicely, we set

Ω_1: the Japanese Islands (doubly enlarged from the original one in Figure 3 to match the number of sampling points approximately to those of the other domains below);

Ω_2: the smoothed and connected version of Ω_1;

Ω_3: the same as Ω_2 but with a "jaggy" boundary curve (more precisely, two boundary pixels are added to each odd row and column of Ω_2 and two are removed from each even row and column of Ω_2 to form Ω_3); and

Ω_4: the two-component version of Ω_2.

The shapes of these domains are displayed in the top row of Figure 9. As for the data on these domains, we adopted three functions with different smoothness: 1) a discontinuous function (in fact, a simple step function whose discontinuity is a straight line along the "spine" or the main axis of the domain); 2) a pyramid-shaped function, which is continuous and its first order partial derivatives are of bounded variation; and finally 3) the standard Gaussian function.

The middle and the bottom rows of Figure 9 show the magnitudes of the expansion coefficients of these three functions of different smoothness with respect to our Laplacian eigenbases in the linear order (i.e., the order of increasing fre-
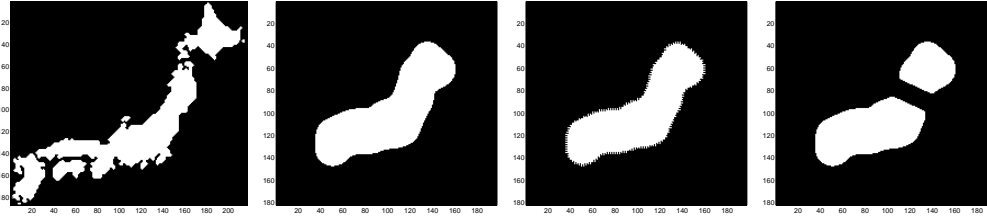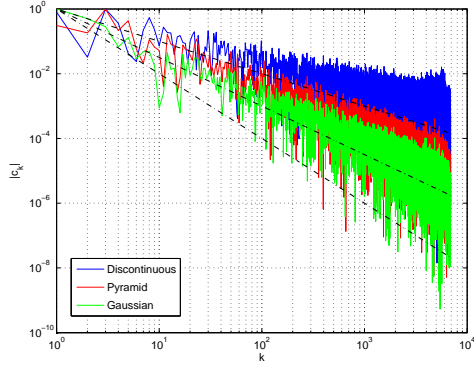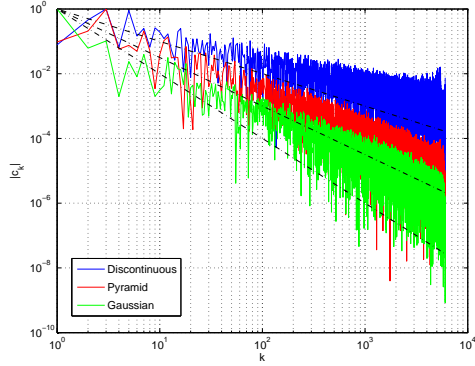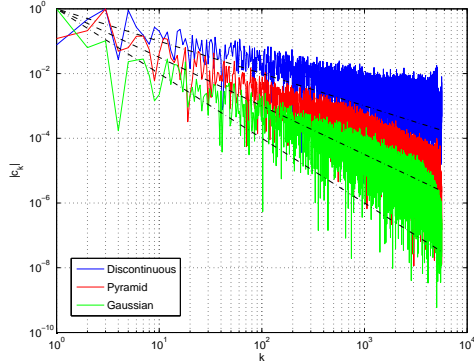
(a) χ_{Ω_1} (b) χ_{Ω_2} (c) χ_{Ω_3} (d) χ_{Ω_4}

(e) Decay rates on Ω_1

(f) Decay rates on Ω_2
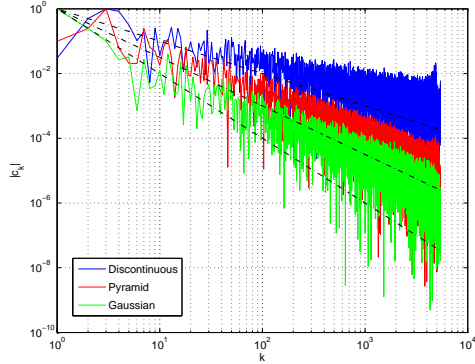
(g) Decay rates on Ω_3

(h) Decay rates on Ω_4

Fig. 9. Comparison of the decay rates of the expansion coefficients (with respect to the Laplacian eigenbases) of functions of various smoothness over four different domains. These are plotted in the log-log scale. The blue, red, and green curves correspond to the discontinuous, pyramid-shape, and Gaussian functions, respectively. The three straight lines plotted with the 'dashdot' pattern are for the reference: they indicate decay rates of k^{-1}, $k^{-1.5}$, k^{-2}, respectively. The top row shows the four different domains we examined.

quencies). There are several things we can observe from this figure.

- The decay rates reflect the intrinsic smoothness of the functions living in the domain, but are not affected by the existence of the boundary of the domains. If we were to extend these datasets to the outside of Ω_i's by zeros up to the circumscribed rectangles and apply the conventional bases appropriate for data on a rectangular domain (e.g., 2D wavelets or 2D Fourier cosine bases) to these extended datasets, then the results would be much worse than those by our Laplacian eigenfunctions, i.e., at most $O(k^{-1})$ or slower. In other words, these conventional basis functions view such extended datasets as discontinuous functions across the boundaries where as our Laplacian eigenfunctions do not sense such artificial discontinuities.

- The decay rates are rather insensitive to the smoothness of the boundary curves. In particular, the plots for Ω_2, Ω_3, and Ω_4 are virtually the same whereas those for Ω_1—the most complicated domain among these four—seem slightly worse than the others. Yet all behave better than $O(k^{-1})$. It will be interesting to compare the performance of our Laplacian eigenfunctions with the conventional ones satisfying the Dirichlet or the Neumann boundary conditions because it is well known that the latter eigenfunctions reveal complicated behavior near the rough boundary. We will compare the behavior of these Laplacian eigenfunctions satisfying different boundary conditions with ours and report our findings at a later date.

- The decay rates are rather insensitive to the number of the separated subdomains. Again, it will be also of interest to investigate the behavior the conventional Laplacian eigenfunctions in this respect.

- Although the coefficient plots oscillate around the linear lines (in the log-log scale), the decay rates $O(k^{-\alpha})$, regardless of the domain shapes, behave as fol-

lows. For the discontinous functions, $\alpha < 1$. For the pyramid-shape function, $1 < \alpha < 1.5$. For the Gaussian function, $\alpha \geq 1.5$. In Figure 10, we plot the decreasing rearragement of these coefficients. The oscillations around the linear lines are of course completely suppressed and the decay rates get faster than those in the linear canonical order. This is one of the reasons why nonlinear approximation [10] provides better approximation than linear ones. Although it is interesting and important to investigate what is a good approximation space for nonlinear approximation using our Laplacian eigenfunctions, it would be most likely quite tough to nail down the *best* approximation space considering the nature of our general shape domains and their boundaries.

Based on the above numerical experiments as well as other numerous experiments we conducted, we have reached to Conjecture 12. Although the assumptions of the conjecture such as the C^2 boundary smoothness and the first partial derivatives being in BV might be weakened, we would like to keep the conjecture "as is" to be on a safe side.

Remark 13 *The example of the Japanese Islands discussed in this section was of small size. We recently obtained a digital map called "Japan Engineering Geomorphologic Classification Map" (JEGM) [31]. The number of sampling points in this map is $387,924$ over the Japanese Islands. Each point is associated with a vector of length 11 representing a type of geological layer, an elevation, a slope, etc. In other words, this is a vector-valued dataset. The coordinate of each point is specified by four values because each point here approximately represents a square region of $1km \times 1km$. These four values are the longitude and the latitude of South West corner and North East corner of each square. The corresponding kernel matrix would be $387,924 \times 387,924$, which is just too huge to handle with usual eigenvalue solvers. A fast algorithm that will be discussed in Section 7 and that was*

(a) Decay rates on Ω_1

(b) Decay rates on Ω_2

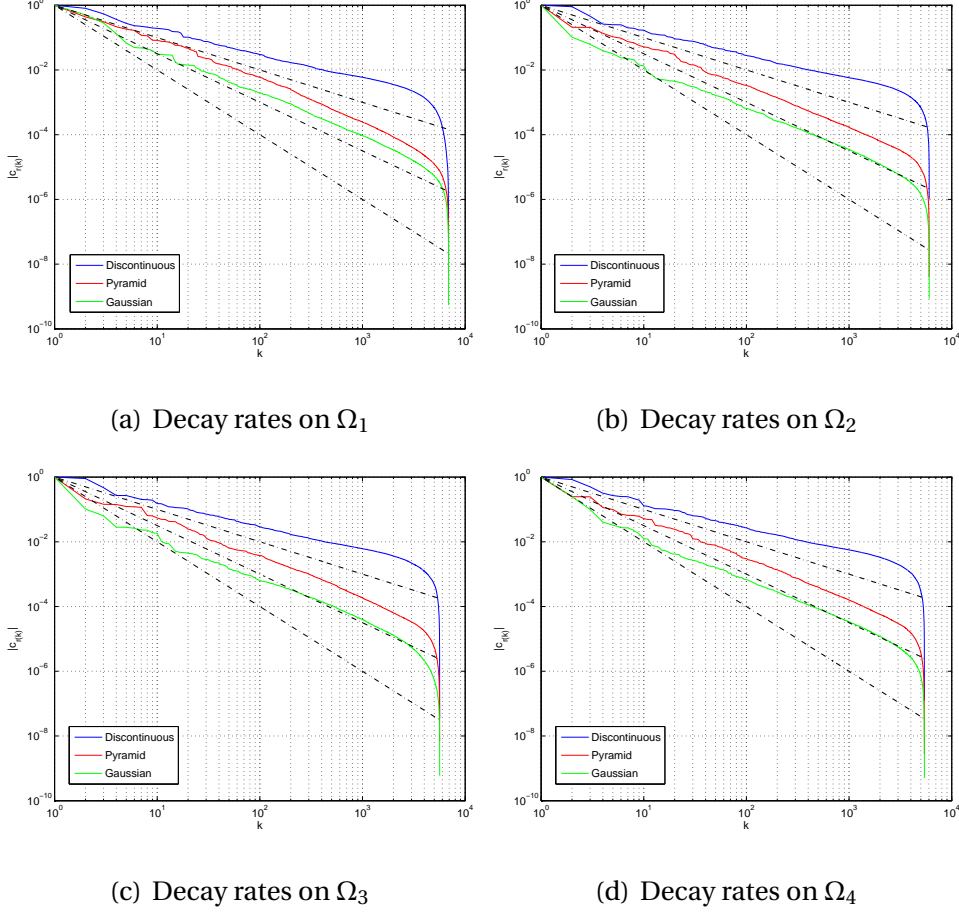(c) Decay rates on Ω_3

(d) Decay rates on Ω_4

Fig. 10. Comparison of the decay rates of the decreasing rearrangement of the expansion coefficients (with respect to the Laplacian eigenbases) of functions of various smoothness over four different domains. These are plotted in the log-log scale. The blue, red, and green curves correspond to the discontinuous, pyramid-shape, and Gaussian functions, respectively. It is obvious that these curves show no oscillations and their decay rates are faster than those in Figure 9. Moreover, the decay rates can be read off easily from the plots. The three straight lines plotted with the 'dashdot' pattern are for the reference: they indicate decay rates of k^{-1}, $k^{-1.5}$, k^{-2}, respectively.

implemented and tested in the Ph.D. thesis of Xue [33] is indispensable to compute the Laplacian eigenfunctions for such a large scale problem.

25

5   Comparison with KLT/PCA

In this section, we shall discuss the use of the Laplacian eigenfunctions for analysis of a stochastic process that lives on a general domain and generates its realizations over there, and compare them with KLT/PCA. As we mentioned in Introduction, KLT/PCA implicitly incorporate geometric information of the measurement (or sampling) location through the autocorrelation or covariance matrices whereas our Laplacian eigenfunctions use explicit geometric information through the integral operator (2) defined on the domain Ω. Moreover, it is important to point out that our Laplacian eigenfunctions are computed once and for all when the geometry of the domain is fixed, and they are *independent* of the statistics of the stochastic process and do not require any autocorrelation or covariance information of the process. This means that we can compute these eigenfunctions even if we have only one realization of the stochastic process. Furthermore, the statistics of the data are completely reflected in the expansion coefficients, not in the basis functions themselves. On the other hand, KLT/PCA requires a good number of realizations of the underlying stochastic process for stably estimating the autocorrelation or covariance matrices. Hence the KLT/PCA basis functions, being the eigenvectors of these matrices, heavily depend on the statistics of the data. In other words, our Laplacian eigenfunctions allow us to *separate geometry and statistics of the underlying data for data analysis purposes*, which can never be realized by KLT/PCA.

The dataset we use for demonstration is the so-called *"Rogue's Gallery"* dataset that we obtained through the courtesy of Prof. Larry Sirovich at Mount Sinai School of Medicine. See [18,23] for more about this dataset. Out of 143 face images in the dataset, 72 are used as a training dataset from which we compute the
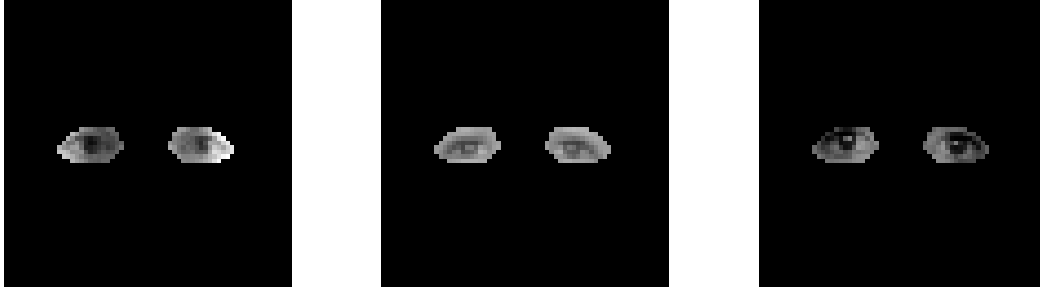
Fig. 11. Three samples (or realizations) of the eye data.

autocorrelation matrix for KLT/PCA. The remaining 71 faces are used as a test dataset to check the validity of KLT/PCA. We cut out the left and right eye regions from the face images and set them as our domain Ω as shown in Figure 11. Therefore in this case, Ω consists of two separate components. The total number of pixels in Ω is 190, i.e., this is a relatively small scale problem. Figure 12 shows the first 25 KL basis vectors. Note that all the KL basis vectors are simply linear combinations of the eyes in the training dataset. This is the reason why they all look like some variations of the actual eyes. Figure 13 shows the Laplacian eigenvectors that have the lowest 25 frequencies. These basis vectors are completely independent from the statistics of the eye training dataset; they only depend on the shape of the domain. Note also that they reveal the even and odd symmetry similar to cosines and sines in the conventional Fourier analysis. This property turns out to be decisive for certain applications such as the detection of "asymmetric eyes" that will be discussed later in this section. Figure 14 shows the log magnitude distribution of the data over the first 25 KLT/PCA coordinates and that over the Laplacian eigenvector coordinates corresponding to the lowest 25 spatial frequencies. As we can observe from these figures, KLT/PCA push more energy of the data into the top few coordinates. In terms of interpretability of the coordinates, however, the Laplacian eigenvectors are far more intuitive. For example, we can see that there are several Laplacian eigencoordinates with
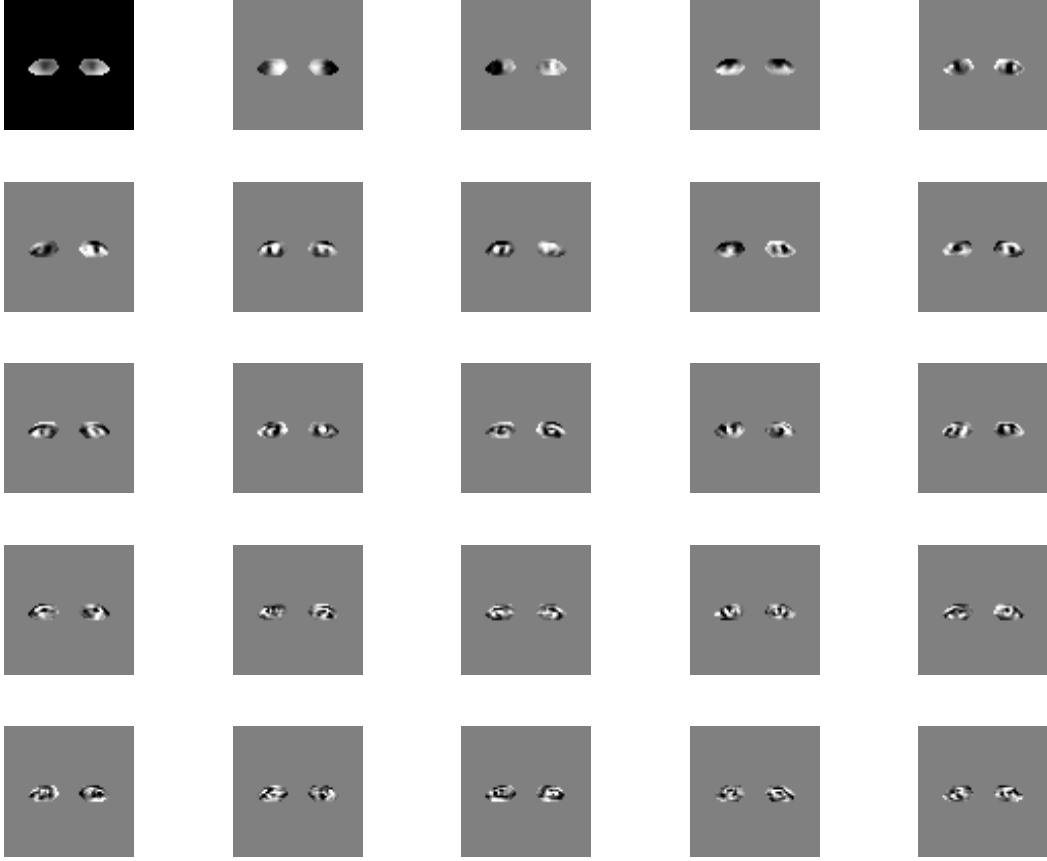
Fig. 12. Top 25 KL basis vectors for the eye region.

high energy, e.g., the coordinates #7 and #13. If we check what these coordinates are in Figure 13, the Laplacian eigenvector #7 correlates well with the iris in the eyes as shown in Figure 15, which also includes the actual eyes with high and low correlations with this particular eigenvector. Similarly, the Laplacian eigenvector #13 indicates how wide the eyes are open as shown in Figure 16. On the other hand, it is very difficult to do this type of interpretation with the KLT/PCA coordinates, as one can easily see from Figures 12 and 14(a).

Figures 17 and 18 well demonstrate the high dependence of the KLT/PCA on the training dataset mentioned earlier. Figure 17 compares the mean energy of the training data distributed over the KLT/PCA coordinates with that over the Laplacian eigencoordinates. From this figure, one can observe a few things. First, the
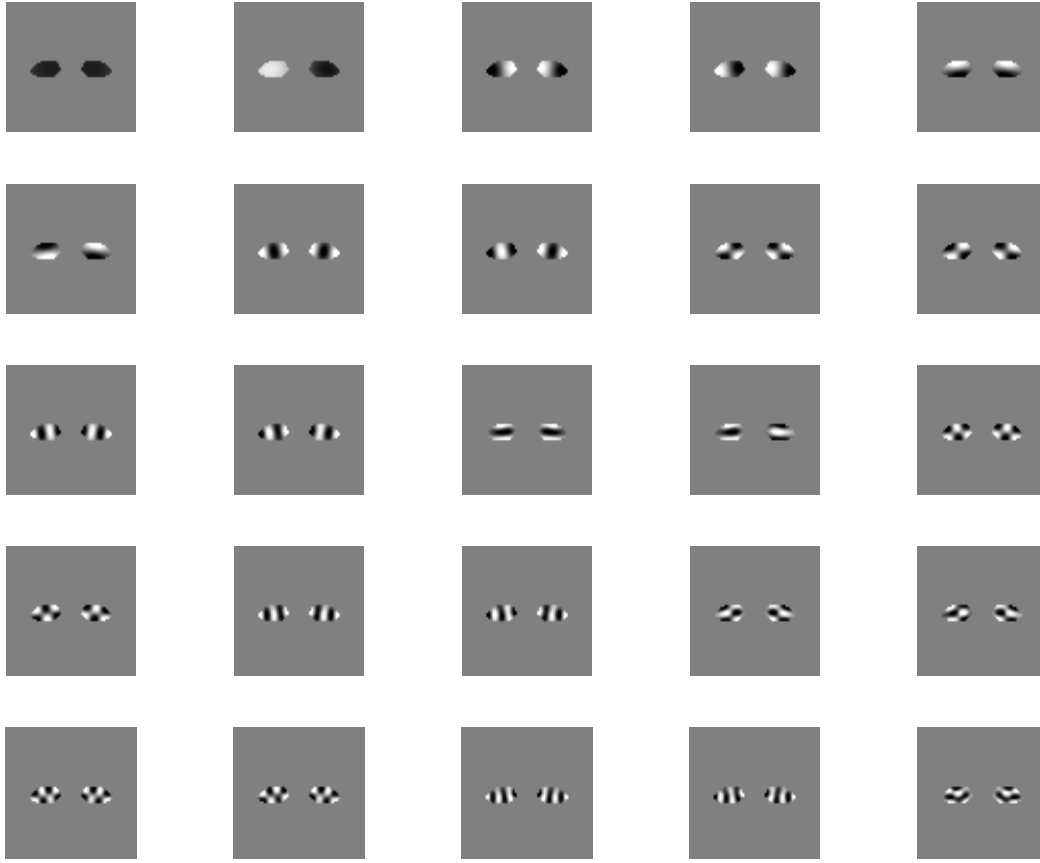
28

Fig. 13. The Laplacian eigenvectors with the lowest 25 spatial frequencies for the eye region.
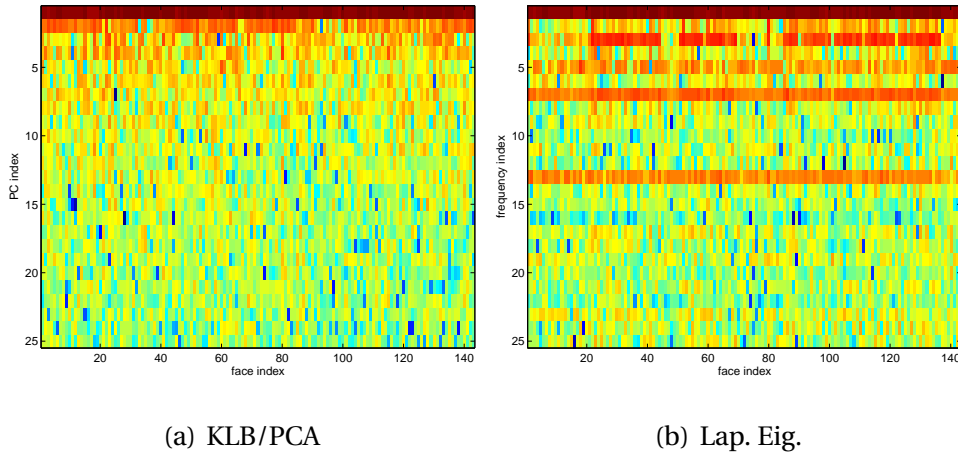

(a) KLB/PCA

(b) Lap. Eig.

Fig. 14. Log magnitudes of the eye data over the first 25 PCA coordinates (a); those over the Laplacian eigencoordinates corresponding to the 25 lowest frequencies (b).
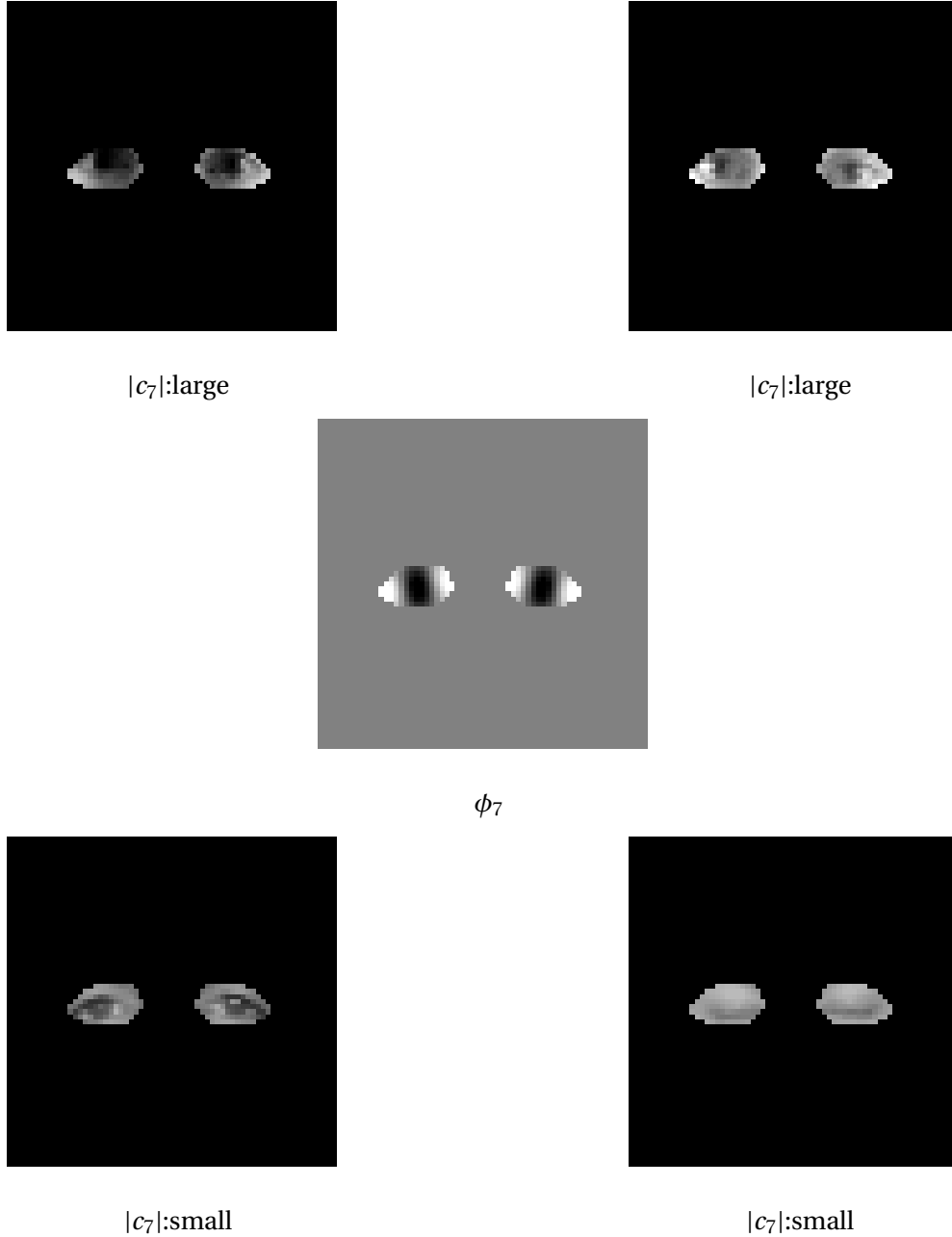
$|c_7|$:large

$|c_7|$:large

ϕ_7

$|c_7|$:small

$|c_7|$:small

Fig. 15. ϕ_7, the 7th Laplacian eigenfunction and the actual eyes with high and low correlations with ϕ_7.

energy of the KLT/PCA coordinates drops suddenly at the coordinate #73. This is because the training dataset consists of 72 realizations (eyes), and consequently the rank of the autocorrelation matrix is only 72. Thus, the KLT/PCA coordinates beyond #72 are useless. Second, the mean energy of the training data distributed over the KLT/PCA coordinates monotonically and nicely decreases as expected.
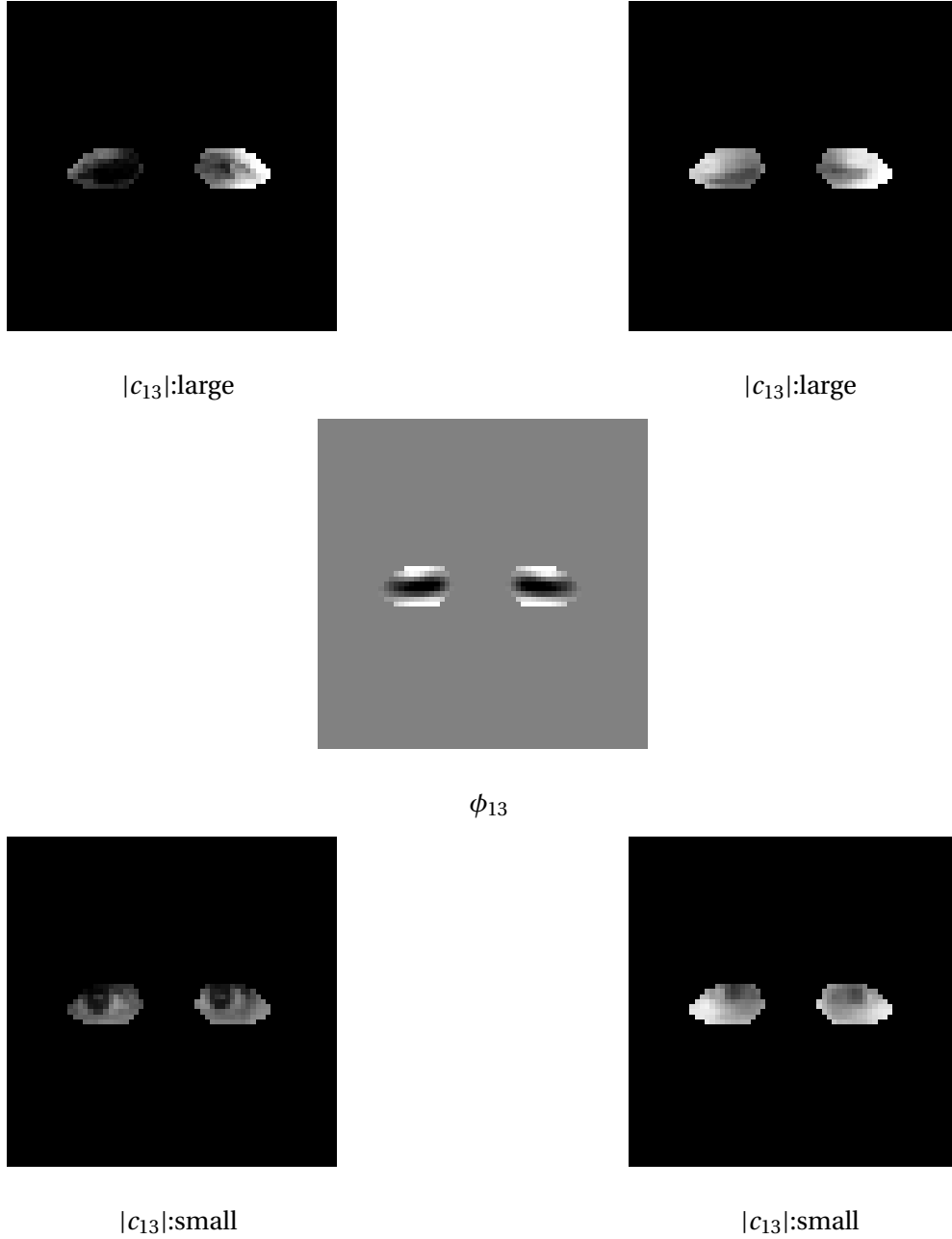
$|c_{13}|$:large $|c_{13}|$:large

ϕ_{13}

$|c_{13}|$:small $|c_{13}|$:small

Fig. 16. ϕ_{13}, the 13th Laplacian eigenfunction and the actual eyes with high and low correlations with ϕ_{13}.

As shown in Figure 18, however, the mean energy of the test data distributed over those KLT/PCA coordinates does not monotonically decrease anymore and the considerable energy are still distributed over the coordinates beyond the coordinate #73. In other words, the behavior of the mean energy distribution of the test data over the KLT/PCA coordinates and that of the training data are quite differ-
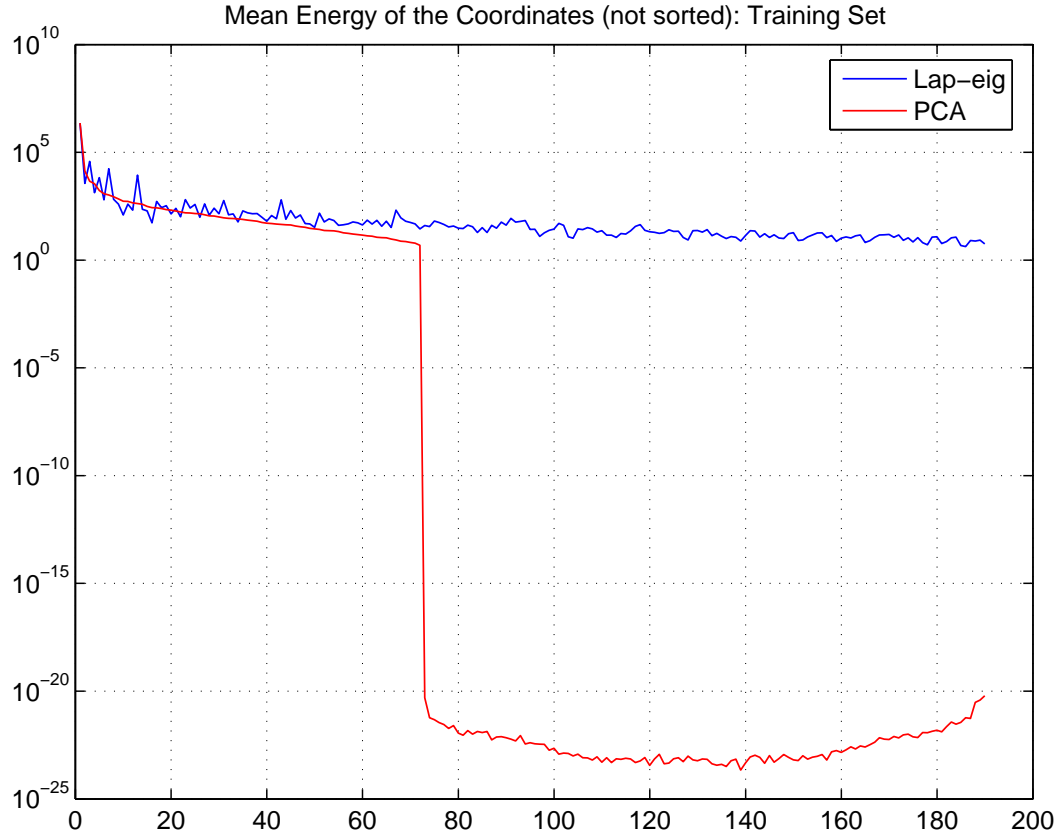
Fig. 17. Comparison of the mean energy of the training data over the KLT/PCA coordinates and the Laplacian eigencoordinates.

ent. On the other hand, because the Laplacian eigenfunctions do not depend on the statistics of the data at all, the behavior of the mean energy distribution of the test data over the Laplacian eigencoordinates is essentially the same as that of the training data.

Finally, we shall discuss another pattern analysis/recoginition application that KLT/PCA cannot handle: the detection of asymmetry of the eyes. Our eye domain has two mirror symmetric subdomains as shown in Figure 11, i.e., the left and right eye regions. It turns out that our Laplacian eigenfunctions naturally consist of those having mirror symmetry (i.e., even functions with respect to the center line between the left and right eyes) and those having mirror antisymmetry (i.e., odd functions) as shown in Figure 13. Note also that the eigenfunc-
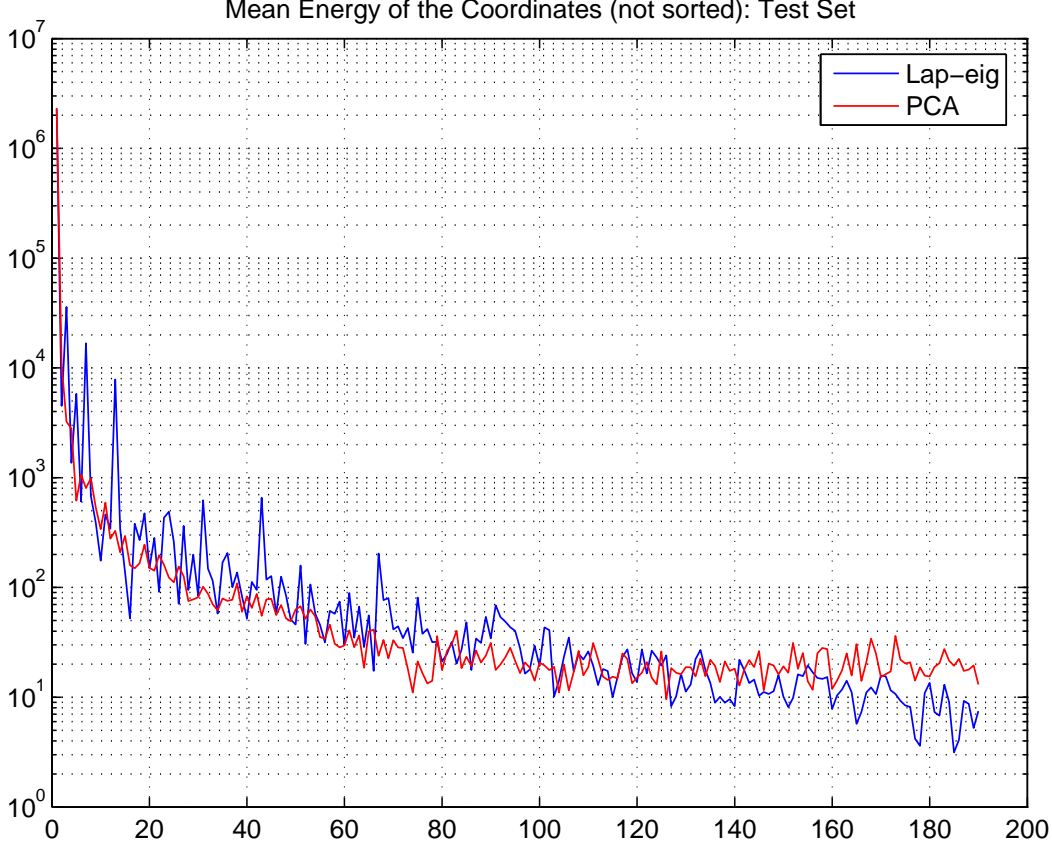
32

Fig. 18. Comparison of the mean energy of the test data over the KLT/PCA coordinates and the Laplacian eigencoordinates.

tions ϕ_7 and ϕ_{13} shown in Figures 15 and 16 are mirror symmetric. Observe that the eigenfunctions with odd indices correspond to the mirror symmetric ones and vice versa. Therefore, by computing the energy of the eye data distributed only along the antisymmetric eigenfunctions ϕ_{2k}, $k = 1,\ldots,95$, we can form a "asymmetry detector". Figure 19 shows the asymmetry detector for the eyes of these 143 people at work. From this figure, we can easily see that the eyes with high asymmetry (the top row) have inhomogeneous illumination conditions or asymmetric eye features (e.g., eye locations) whereas those with low asymmetry (the bottom row) have rather uniform illumination conditions *and* symmetric eye features. It is clear from both its construction and Figure 12 that KLT/PCA cannot handle this type of asymmetry detection at all because KLT/PCA basis
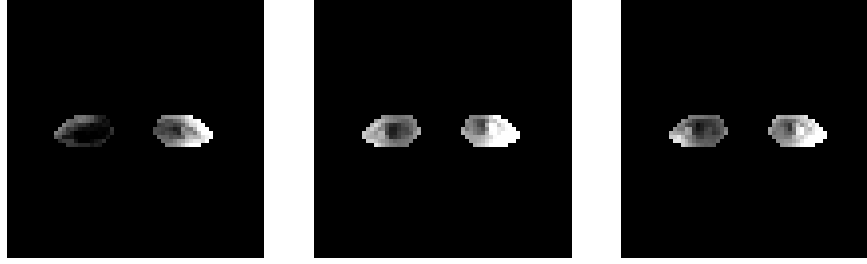
Eyes #80   Eyes #22   Eyes #52
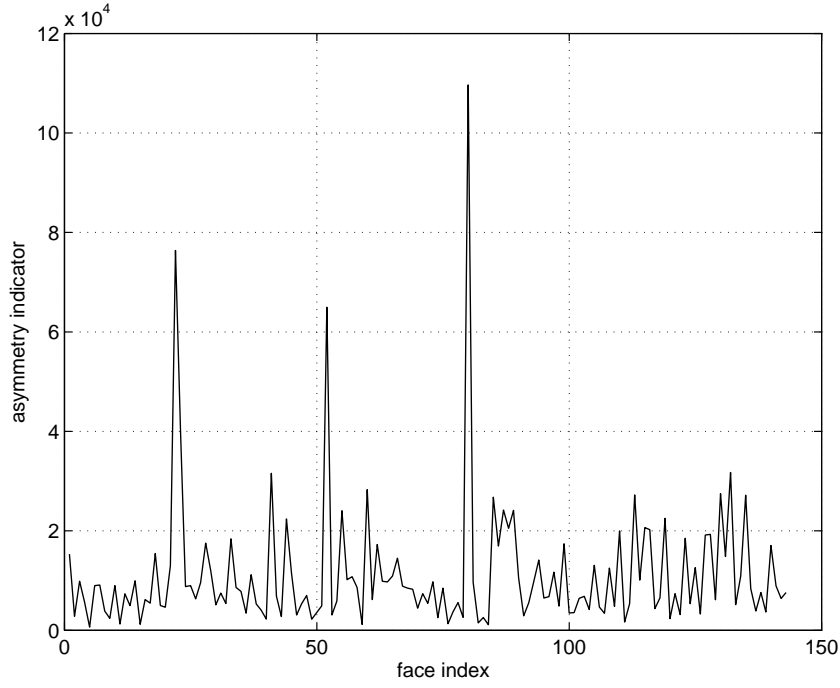
Asymmetry detector
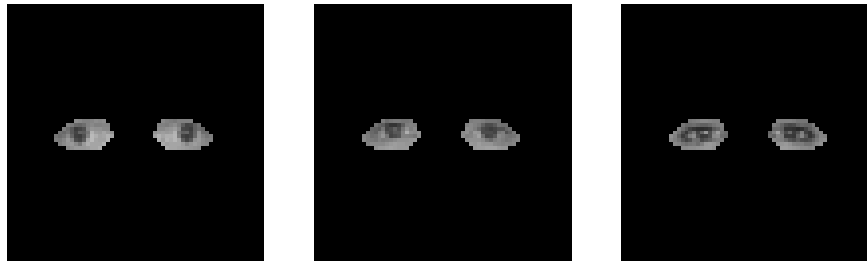
Eyes #5   Eyes #84   Eyes #59

Fig. 19. The "asymmetry detector" at work. The three peaks in the asymmetry detector plot shown in the middle row indicate that there are three sets of eyes that are highly asymmetric compared to the others, which are displayed in the top row. The bottom row shows the most symmetric eyes, i.e., those having the smallest energy in the anti-symmetric Laplacian eigencoordinates.

34

functions do not have any well-defined symmetry or antisymmetry.

6 Solving the Heat Equation on a Complicated Domain

In this section, we shall discuss yet another application of our Laplacian eigen-functions: solving the heat equation on a general shape domain in $\mathbb{R}^d$. It is well known that the semigroup $e^{t\Delta}$ can be diagonalized using the Laplacian eigenbasis, i.e., for any initial heat distribution $u_0(\boldsymbol{x}) \in L^2(\overline{\Omega})$, we have the heat distribution at time t formally as

$$u(\boldsymbol{x}, t) = e^{t\Delta} u_0(\boldsymbol{x}) = \sum_{j=1}^{\infty} e^{-t\lambda_j} \langle u_0, \phi_j \rangle \phi_j(\boldsymbol{x}),$$

which is based on the expansion of the Green's function for the heat equation $p_t(\boldsymbol{x}, \boldsymbol{y})$ via the Laplacian eigenfunctions as follows (see e.g., [11]).

$$p_t(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j=1}^{\infty} e^{-\lambda_j t} \phi_j(\boldsymbol{x}) \overline{\phi_j(\boldsymbol{y})} \quad (t, \boldsymbol{x}, \boldsymbol{y}) \in (0, \infty) \times \overline{\Omega} \times \overline{\Omega}.$$

In practice, the domain Ω must be discretized by a finite number (i.e., $N \in \mathbb{N}$) of sample points (or pixels) as discussed in Section 2.2. Consequently, the Laplacian eigenfunctions become the Laplacian eigenvectors of length N as in the previous sections. Therefore, we can write $e^{t\Delta}$ in the matrix-vector form as

$$\Phi \, e^{-t\Lambda} \, \Phi^T = \Phi \, \mathrm{diag}\left(e^{-t\lambda_1}, \dots, e^{-t\lambda_N}\right) \Phi^T = \sum_{j=1}^{N} e^{-\lambda_j t} \boldsymbol{\phi}_j \boldsymbol{\phi}_j^T,$$

where $\Phi = \left(\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_N\right)$ is the Laplacian eigenbasis matrix of size $N \times N$, and Λ is the diagonal matrix consisting of the eigenvalues of the Laplacian, which are the inverse of the eigenvalues of the discretized kernel matrix, i.e., $\Lambda_{k,k} = \lambda_k = 1/\mu_k$.

Given an initial heat distribution over the domain, $\boldsymbol{u}_0 \in \mathbb{R}^N$, we can compute the
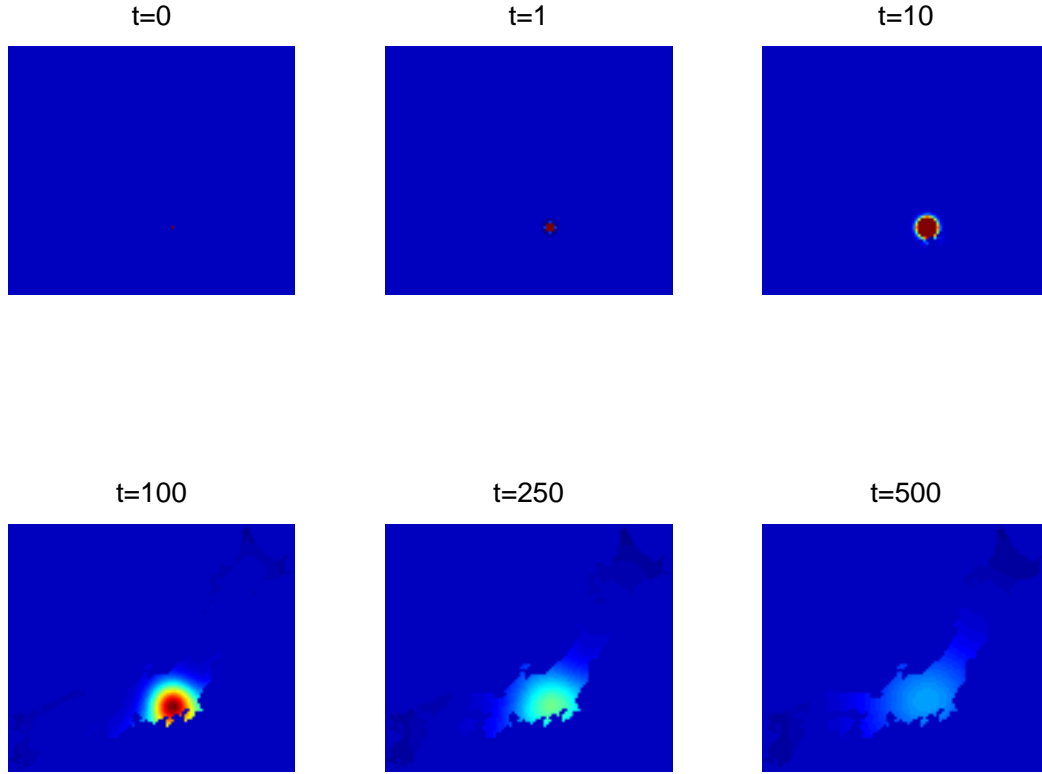
Fig. 20. The propagation of the heat distribution over the Japanese Islands. The initial point heat source was put at the location of Mt. Fuji.

heat distribution at time t as

$$\boldsymbol{u}(t) = \Phi\, e^{-t\Lambda}\, \Phi^T \boldsymbol{u}_0. \tag{12}$$

Figure 20 shows our simple numerical experiment using the Japanese Islands of Section 4 as our Ω. The physical meaning of our boundary condition (3) is not completely obvious: it is different from the Dirichlet, the Neumann, and the Robin boundary conditions as we discussed in Sections 2 and 3. Therefore, it is important to investigate how to synthesize those standard physically-meaningful Laplacian eigenfunctions using our own Laplacian eigenfunctions. All we can say at this point is that the heat distribution over the Japanese Islands eventually ceases to zero regardless of the initial heat distribution as can be observed in Figure 20 because the Laplacian eigenvalues are all positive in this example.

Remark 14 *It is interesting to examine the positivity of the eigenvalues of the Laplacian in our setting. Although the 1D example in Section 3.1 has one negative eigenvalue, many examples of higher dimensional domains* without symmetry *(e.g., the Japanese Islands) that we examined have only positive eigenvalues. It is well known that the Laplacian associated with the Dirichlet (or the Neumann) boundary condition has only positive (or nonnegative) eigenvalues. The basis of the proof is the following argument (see also [29, Sec. 10.1]). Let ϕ be an eigenfunction of the Laplacian over Ω satisfying either the Dirichlet or the Neumann boundary condition, and let λ be the corresponding eigenvalue. Then, using the Green's first identity (see e.g., [17, p.94]), we have*

$$
\begin{aligned}
\|\nabla\phi\|_{L^2(\Omega)}^2 &= \int_\Omega \nabla\phi \cdot \nabla\phi \, \mathrm{d}\boldsymbol{x} \\
&= \int_\Omega \phi(\boldsymbol{x})(-\Delta\phi(\boldsymbol{x})) \, \mathrm{d}\boldsymbol{x} + \int_\Gamma \phi(\boldsymbol{x})\frac{\partial\phi}{\partial\nu}(\boldsymbol{x}) \, \mathrm{d}s(\boldsymbol{x}) \\
&= \lambda\|\phi\|_{L^2(\Omega)}^2 + \int_\Gamma \phi(\boldsymbol{x})\frac{\partial\phi}{\partial\nu}(\boldsymbol{x}) \, \mathrm{d}s(\boldsymbol{x}).
\end{aligned}
$$

Therefore, if ϕ satisfies either the Dirichlet or the Neumann condition, then the second term in the righthand side vanishes. Thus we have: $\lambda = \|\nabla\phi\|_{L^2(\Omega)}^2 / \|\phi\|_{L^2(\Omega)}^2 > 0$ for the Dirichlet case and ≥ 0 for the Neumann case where $\phi = $ const. is the first eigenfunction. In the case of the Robin boundary condition, $\frac{\partial\phi}{\partial\nu} + a\phi = 0$, it is known that $\lambda \geq 0$ if $a \geq 0$. This is because the above equation leads to $\lambda = \left(\|\nabla\phi\|_{L^2(\Omega)}^2 + a\|\phi|_\Gamma\|_{L^2(\Gamma)}^2\right) / \|\phi\|_{L^2(\Omega)}^2$. The above proof critically depends on the Dirichlet/Neumann/Robin boundary conditions. As for our Laplacian eigenvalue problem with our special boundary condition (3), *it would be rather difficult to derive exact conditions on the shape of the domain where all the eigenvalues become positive.*

Remark 15 *Our approach to solving heat equations and to computing Laplacian*

eigenfunctions via the integral operator commuting with the Laplacian in general may become a useful tool for machine learning problems, in particular, clustering of high dimensional data. Popular procedures for such tasks include Laplacian eigenmaps [3] and diffusion maps [5]. Both start with constructing a graph from available data. Then, the former forms a graph Laplacian or a diffusion kernel while the latter form a normalized diffusion kernel. Finally, both compute the eigenvectors of these kernel matrices and embed the original data into a smaller number of those eigencoordinates to seek their clustering structure. Computing the eigenvectors of the graph Laplacian on a complicated graph is often difficult due to the slow convergence of iterative eigenvalue solvers for such matrices even if the graph Laplacian matrix is usually quite sparse. On the other hand, our approach based on the integral operator commuting with the Laplacian is computationally quite stable although it uses dense matrices. Using an approach using the celebrated Fast Multipole Method (FMM) [15] that we shall discuss in the next section and that was implemented and tested in [33], we should be able to speed up the eigenvector computations despite of the denseness of the kernel matrices. As for computing eigenvectors of diffusion kernels, one may want to use the improve fast Gauss transform [22], which is an improved version of the fast Gauss transform of Greengard and Strain [16], which in turn is based on FMM again. But if one wants to compute the evolution of the diffusion process by varying the time parameter t, then this method seems less effective since it requires computing the eigenvectors of the diffusion kernel for each fixed t. On the other hand, our method once and for all computes the eigenvectors of the underlying Laplacian and can compute the diffusion process at any time t quite easily as shown in Eq. (12).
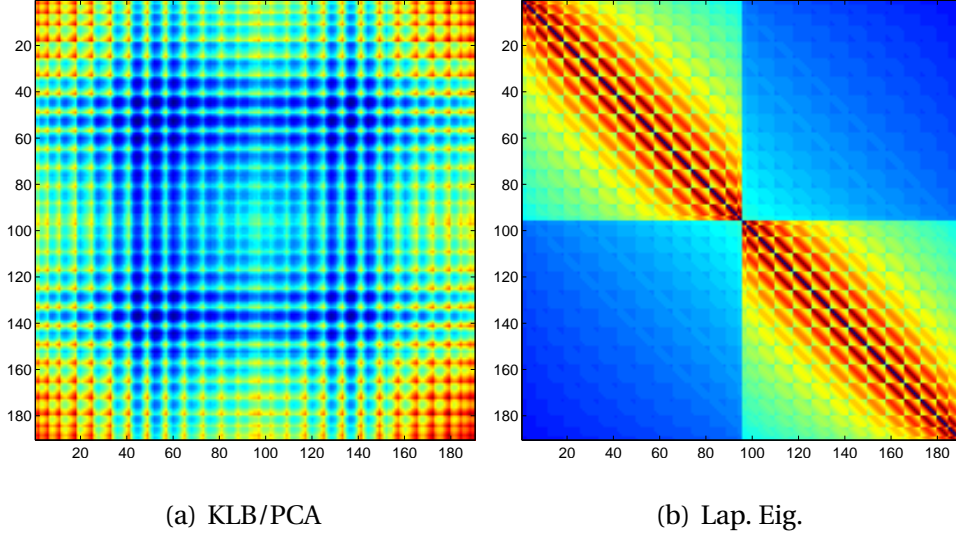
(a) KLB/PCA   (b) Lap. Eig.

Fig. 21. The autocorrelation matrix of the eye data (a) vs. the harmonic kernel (1) for the eye region as a matrix (b).

7   Strategies for Fast Computation

For computing our Laplacian eigenfunctions for a domain sampled by a large number of cells, the use of fast algorithms becomes indispensable. There are at least two possibilities, both of which we are currently actively investigating and hope to report the result of our investigation at a later date. Xue's Ph.D. thesis [33] contains our effort along this direction to date. Both of them use the special properties of the harmonic kernel (1). Unlike the autocorrelation matrix of the eye data we examined in Section 5, which is not really structured except that it is symmetric as we can see from Figure 21(a), the kernel matrix displayed in Figure 21(b) is similar to a block Toeplitz form and the entries in each block decays logarithmically away from the diagonal. Potentially, we may get a better (i.e., smoother) kernel matrix by rearranging its entries. Therefore, one possibility is to use the "Alpert wavelets" [2] to *sparsify* this matrix (possibly with some rearrangement), and then use the sparse eigenvalue solver for the resulting matrix. Another more promising possibility is to use the Krylov subspace method

(such as the Lanczos iteration) [30] combined with the fast matrix-vector multiplication derived by the celebrated Fast Multipole Method (FMM) [15]. This is possible because our integral operator (2) with the harmonic kernel (1) is the one for computing the electrostatic potential field caused by the point charges distributed on the domain Ω. See [33] for the details on and the further development of the use of FMM for our Laplacian eigenvalue problem.

If a domain consists of disconnected components, then one can reduce the original problem into a set of smaller problems. For example, let us consider the example of the Japanese Islands shown in Figure 3, where there are four clearly separated major islands (or components). Figure 22(a) shows the kernel matrix for the whole islands from which we computed the Laplacian eigenfunctions show in Figure 4. We can clearly see four main diagonal blocks corresponding to the four islands. Note also that the size of each block is proportional to the volume of (i.e., the number of samples on) the corresponding component. But we can also see in this figure that there are nonzero entries in the off-diagonal blocks. These resulted from the "communication" between the geographically-separated islands when we constructed this kernel matrix. In other words, we computed $K_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) \neq 0$ even if $\boldsymbol{x}_i$ belongs to one island and $\boldsymbol{x}_j$ belongs to the other. Now, it is at our disposal to decide whether we keep the communication among the separated components alive or cut that communication down when we construct the kernel matrix. If we decide to disconnect the communication among the separated components, the only thing we need to do is to set $K_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = 0$ if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ belong to the different components. Such disconnection operations set the entries of the off-diagonal blocks of the matrix to completely zero and decouple the original matrix into a set of smaller matrices as shown in Figure 22(b). This means that we can decompose a large eigenvalue
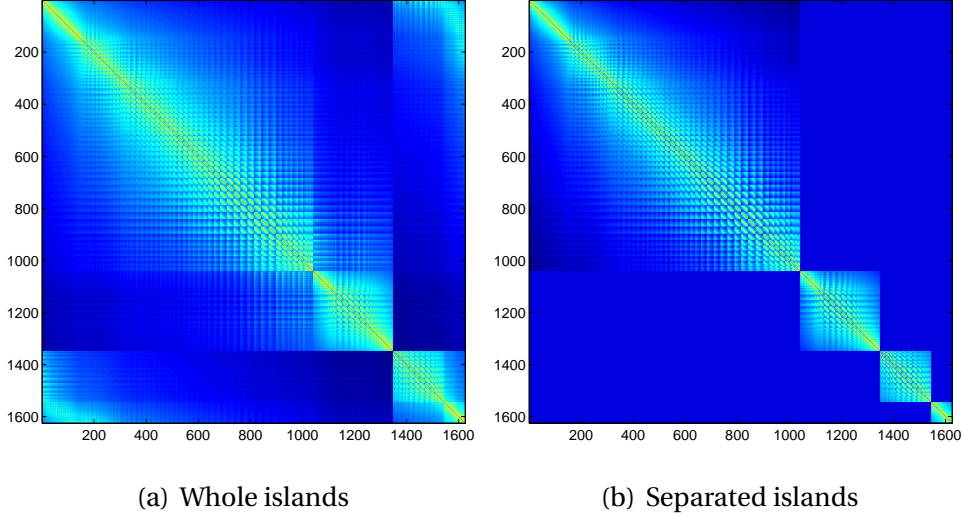
| (a) Whole islands | (b) Separated islands |

Fig. 22. The kernel matrix where the communications between and within the islands are kept (a) and the one without the communication between the islands (b).

problem into a set of smaller subproblems, each of which can be solved definitely faster than the original large problem. Moreover, because of the supports of the eigenfunctions are disjoint, their orthogonality is maintained. Figure 23 shows the five lowest frequency eigenfunctions for each major island in Japan. Of course the user has the ultimate responsibility for cutting such communication among the components. There may be some cases where keeping the communication between separated components makes pattern analysis tasks easier. In fact, we have already seen one such example: the asymmetry detector of eyes in Section 5.

8 Conclusion

In this paper, we proposed a new method to compute the Laplacian eigenfunctions for a domain of general shape $\Omega \subset \mathbb{R}^d$ via the eigenanalysis of the integral operator commuting with the Laplacian, and demonstrated their usefulness for a variety of applications. In particular, we demonstrated: 1) the expansion coeffi-
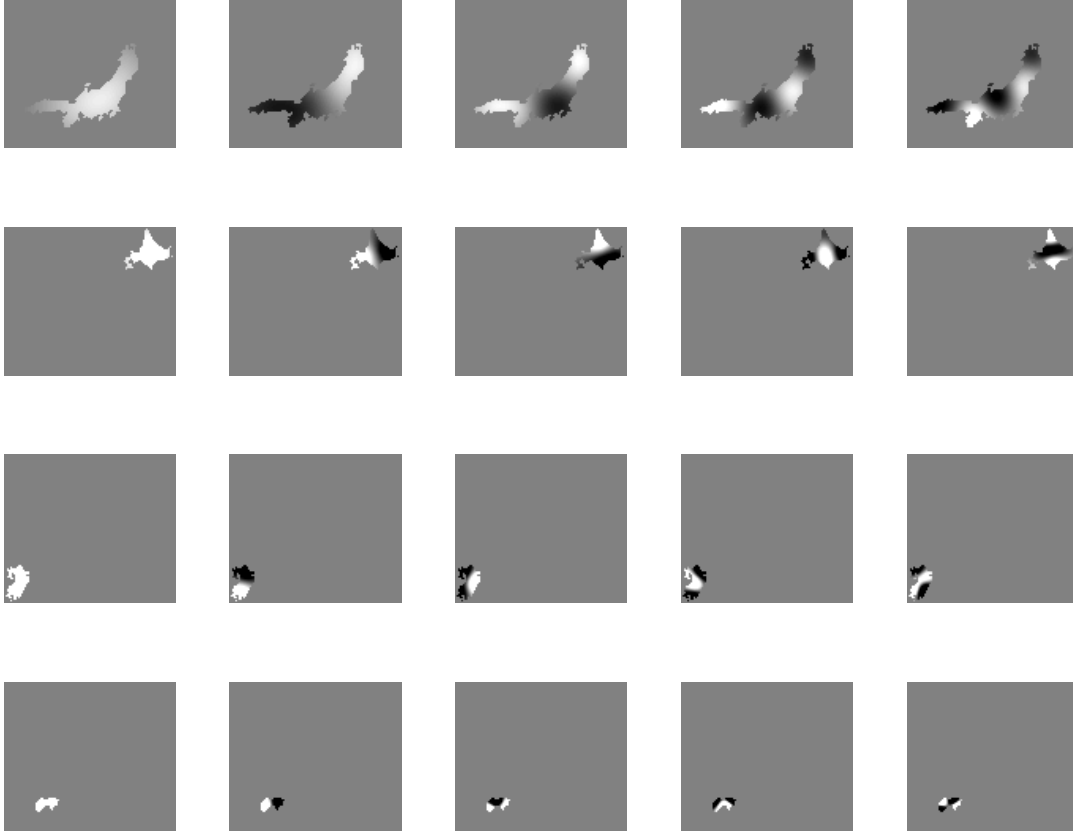
Fig. 23. Top five Laplacian eigenfunctions for separated islands.

cients of a function on Ω with respect to our Laplacian eigenfunctions reflect the intrinsic smoothness of that function on Ω and are not affected by the boundary Γ, which form the basis for efficient approximation; 2) our Laplacian eigenfunctions achieve the *separation of geometry and statistics*, i.e., they are determined once and for all when the domain Ω and the data sampling strategy on Ω are determined, and are independent from the statistics of the measured data over there, which should be contrasted with KLT/PCA; 3) implementing a stable numerical algorithm to compute our Laplacian eigenfunctions is quite straightforward and our formulation is amenable to modern fast algorithms.

There are a number of potential areas to which our Laplacian eigenfunctions may contribute. On the fundamental side, we can list interpolation, extrapola-

tion, and local feature computation. On the applied side, medical and geological image analysis and analysis of distributed sensor networks immediately come to mind. We plan to investigate those important areas in the near future.

Now, there are several challenges in front of us that we also plan to investigate. First, computing the Laplacian eigenfunctions on 3D domains. This is rather straightforward as long as the number of samples N on the domain is not too huge; one just needs to switch the kernel to $K(\boldsymbol{x}, \boldsymbol{y}) = \left(4\pi|\boldsymbol{x} - \boldsymbol{y}|\right)^{-1}$. As a matter of fact, we computed the Laplacian eigenfunctions for simple 3D shapes such as the unit ball in $\mathbb{R}^3$ without any difficulty. If N is huge, however, computing 3D Laplacian eigenfunctions becomes a major challange; we need the 3D FMM, which is not trivial to implement. Mathematical analysis of the boundary condition for a 3D domain also becomes a challenge. For example, it seems quite cumbersome to derive a neat boundary condition (like Eq. 10 of the 2D unit disk) for the 3D unit ball. To do so, we may need to employ a higher dimensional analog of the Hilbert transform involving "Clifford Analysis" (see e.g., [9] and the references therein).

Second, understanding the physical meaning of our unconventional non-local boundary condition (3). This is important for physical applications such as the heat equation that we discussed in Section 6. Since the conventional Laplacian eigenfunctions satisfying either the Dirichlet or the Neumann boundary condition are used in numerous applications and yet difficult to compute on a complicated domain, it is also important to investigate whether we can synthesize the conventional eigenfunctions using our own Laplacian eigenfunctions and to compare the mathematical and numerical properties of our eigenfunctions with those of the conventional eigenfunctions.

Last but not least, computing the eigenfunctions of the polyharmonic operators Δ^m with $m > 1$. Although these may be computed easily by simply replacing the harmonic kernel by the polyharmonic kernel and we may not need to worry about the boundary condition as in the Laplacian case, analyzing mathematical properties of such eigenfunctions may become a real challege.

Finally, we would like to conclude this article by noting that our method has a connection to many interesting mathematics such as spectral geometry, Toeplitz operators, PDEs, potential theory, radial basis functions, almost-periodic functions, etc., and we expect further interplay among these fields!

Acknowledgment

A   Proof of Theorem 2

PROOF. Let $\mathcal{L} = -\Delta$ and $\mathcal{K}$ be defined as (2). Then, for $f \in C^2(\Omega) \cup C^1(\overline{\Omega})$, we have

$$\mathcal{L}\mathcal{K}f(\boldsymbol{x}) = -\Delta_{\boldsymbol{x}}\mathcal{K}f(\boldsymbol{x}) = f(\boldsymbol{x}) \quad \boldsymbol{x} \in \Omega,$$

which is referred to as "Poisson's formula" [17, p.99]. On the other hand, using the Green's second identity (see e.g., [17, p.94]), we have

$$\mathcal{K}\mathcal{L}f(\boldsymbol{x}) = -\int_{\Omega} K(\boldsymbol{x}-\boldsymbol{y})\Delta_{\boldsymbol{y}}f(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}$$
$$= f(\boldsymbol{x}) - \int_{\Gamma} K(\boldsymbol{x}-\boldsymbol{y})\frac{\partial f}{\partial v_{\boldsymbol{y}}}(\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y}) + \int_{\Gamma} \frac{\partial K}{\partial v_{\boldsymbol{y}}}(\boldsymbol{x}-\boldsymbol{y})f(\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y}).$$

Thus, $\mathcal{K}$ and $\mathcal{L}$ commute if and only if

$$\int_{\Gamma} K(\boldsymbol{x}-\boldsymbol{y})\frac{\partial f}{\partial v_{\boldsymbol{y}}}(\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y}) = \int_{\Gamma} \frac{\partial K}{\partial v_{\boldsymbol{y}}}(\boldsymbol{x}-\boldsymbol{y})f(\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y}) \quad \boldsymbol{x} \in \Omega. \tag{A.1}$$

We now would like to move $\boldsymbol{x} \in \Omega$ to the boundary Γ in Eq. (A.1). While we do not have any problem in the lefthand side (the single layer potential), we must treat the righthand side (the double layer potential) carefully. Here, we will follow Folland [12, Chap. 2] (see also Kress [19, Sec. 6.3]). Let us consider the righthand side at $\boldsymbol{x} + tv_{\boldsymbol{x}} \in \Omega$ for $\boldsymbol{x} \in \Gamma$ with a sufficiently small $t < 0$ instead of $\boldsymbol{x} \in \Omega$. Thus, for $\boldsymbol{x} \in \Gamma$, we have

$$\int_{\Gamma} \frac{\partial K}{\partial v_{\boldsymbol{y}}}(\boldsymbol{x}+tv_{\boldsymbol{x}}-\boldsymbol{y})f(\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y}) = f(\boldsymbol{x})\int_{\Gamma} \frac{\partial K}{\partial v_{\boldsymbol{y}}}(\boldsymbol{x}+tv_{\boldsymbol{x}}-\boldsymbol{y})\,\mathrm{d}s(\boldsymbol{y})$$
$$+ \int_{\Gamma} \frac{\partial K}{\partial v_{\boldsymbol{y}}}(\boldsymbol{x}+tv_{\boldsymbol{x}}-\boldsymbol{y})(f(\boldsymbol{y})-f(\boldsymbol{x}))\,\mathrm{d}s(\boldsymbol{y}). \tag{A.2}$$

The first term in the righthand side is $-f(\boldsymbol{x})$ thanks to the following

Lemma 16 (a variant of Lemma (3.19) in [12])

$$\int_\Gamma \frac{\partial K}{\partial v_y}(\boldsymbol{x} - \boldsymbol{y}) \, \mathrm{d}s(\boldsymbol{y}) = \begin{cases} -1 & \text{if } \boldsymbol{x} \in \Omega; \\ -\frac{1}{2} & \text{if } \boldsymbol{x} \in \Gamma; \\ 0 & \text{if } \boldsymbol{x} \notin \overline{\Omega}. \end{cases}$$

As for the second integral in Eq. (A.2), because $\psi(\boldsymbol{y}) \overset{\Delta}{=} f(\boldsymbol{y}) - f(\boldsymbol{x})$ is continuous and $\psi(\boldsymbol{x}) = 0$ for $\boldsymbol{x} \in \Gamma$, we can use Lemma (3.21) of Folland [12, p. 127] to conclude that as $t \to 0$ the second integral in (A.2) approaches to

$$\int_\Gamma \frac{\partial K}{\partial v_y}(\boldsymbol{x} - \boldsymbol{y}) f(\boldsymbol{y}) \, \mathrm{d}s(\boldsymbol{y}) - f(\boldsymbol{x}) \int_\Gamma \frac{\partial K}{\partial v_y}(\boldsymbol{x} - \boldsymbol{y}) f(\boldsymbol{y}) \, \mathrm{d}s(\boldsymbol{y})$$
$$= \int_\Gamma \frac{\partial K}{\partial v_y}(\boldsymbol{x} - \boldsymbol{y}) f(\boldsymbol{y}) \, \mathrm{d}s(\boldsymbol{y}) + \tfrac{1}{2} f(\boldsymbol{x}), \quad \boldsymbol{x} \in \Gamma,$$

where we used Lemma 16 again in the last equality. Therefore, we can finally have

$$\int_\Gamma K(\boldsymbol{x} - \boldsymbol{y}) \frac{\partial f}{\partial v_y}(\boldsymbol{y}) \, \mathrm{d}s(\boldsymbol{y}) = -\frac{1}{2} f(\boldsymbol{x}) + \mathrm{pv} \int_\Gamma \frac{\partial K}{\partial v_y}(\boldsymbol{x} - \boldsymbol{y}) f(\boldsymbol{y}) \, \mathrm{d}s(\boldsymbol{y}) \quad \boldsymbol{x} \in \Gamma,$$

which is the same as (3). This completes the proof. □

B   Proof of Corollary 6

PROOF. Rather than dealing with (3), we will derive the Laplacian eigenvalue problem directly from the integral eigenvalue problem, $\mathcal{K}\phi = \mu\phi$.

$$\mathcal{K}\phi(x) = -\frac{1}{2}\int_0^1 |x-y|\phi(y)\,\mathrm{d}y$$
$$= -\frac{1}{2}\left(\int_0^x (x-y)\phi(y)\,\mathrm{d}y + \int_x^1 (y-x)\phi(y)\,\mathrm{d}y\right)$$
$$= -\frac{1}{2}\left(x\int_0^x \phi(y)\,\mathrm{d}y - \int_0^x y\phi(y)\,\mathrm{d}y + \int_x^1 y\phi(y)\,\mathrm{d}y - x\int_x^1 \phi(y)\,\mathrm{d}y\right)$$
$$= \mu\phi(x).$$

Differentiating both sides with respect to x, we have

$$\int_0^x \phi(y)\,\mathrm{d}y - \int_x^1 \phi(y)\,\mathrm{d}y = -2\mu\phi'(x). \tag{B.1}$$

By setting $x = 0$ and $x = 1$ in this equality, we have

$$\phi'(0) = -\phi'(1) = \frac{1}{2\mu}\int_0^1 \phi(y)\,\mathrm{d}y. \tag{B.2}$$

Now, evaluating $\mathcal{K}\phi(x) = \mu\phi(x)$ at $x = 0, 1$ yields

$$\phi(0) = -\frac{1}{2\mu}\int_0^1 y\phi(y)\,\mathrm{d}y,$$

$$\phi(1) = -\frac{1}{2\mu}\int_0^1 (1-y)\phi(y)\,\mathrm{d}y = -\frac{1}{2\mu}\left(\int_0^1 \phi(y)\,\mathrm{d}y - \int_0^1 y\phi(y)\,\mathrm{d}y\right).$$

Adding these two equations and using (B.2), we obtain

$$\phi(0) + \phi(1) = -\frac{1}{2\mu}\int_0^1 \phi(y)\,\mathrm{d}y = -\phi'(0) = \phi'(1),$$

which is (4), i.e., what we wanted to show. Differentiating (B.1) in x once again, we can easily obtain the Laplacian eigenvalue equation as follows.

$$2\phi(x) = -2\mu\phi''(x) \quad \Longleftrightarrow \quad -\phi''(x) = \lambda\phi(x),\ \lambda = \frac{1}{\mu}.$$

Let us now compute the solutions to this Laplacian eigenvalue problem with the boundary condition (4). The characteristic equation for $\phi'' + \lambda\phi = 0$ is $r^2 + \lambda = 0$. Therefore, we need to consider the following three cases:

Case I $\lambda < 0$: In this case, the eigenfunction is of the form $\phi(x) = A\cosh\sqrt{-\lambda}x + B\sinh\sqrt{-\lambda}x$ where A, B are some constants. Putting this form to the boundary condition (4), we have

$$
\begin{aligned}
\phi(0) + \phi(1) &= A(1 + \cosh\sqrt{-\lambda}) + B\sinh\sqrt{-\lambda} \\
&= -\phi'(0) = -\sqrt{-\lambda}B \\
&= \phi'(1) = \sqrt{-\lambda}\left(A\sinh\sqrt{-\lambda} + B\cosh\sqrt{-\lambda}\right)
\end{aligned}
$$

From these equalities, we have the following 2×2 linear system for A and B:

$$
\begin{bmatrix}
1 + \cosh\sqrt{-\lambda} & \sqrt{-\lambda} + \sinh\sqrt{-\lambda} \\
\sinh\sqrt{-\lambda} & 1 + \cosh\sqrt{-\lambda}
\end{bmatrix}
\begin{bmatrix} A \\ B \end{bmatrix}
=
\begin{bmatrix} 0 \\ 0 \end{bmatrix}.
$$

Thus, in order to have nontrivial eigenfunctions, the determinant of this equation must be zero. This leads to

$$
\begin{aligned}
0 &= \left(1 + \cosh\sqrt{-\lambda}\right)^2 - \sinh\sqrt{-\lambda}\left(\sqrt{-\lambda} + \sinh\sqrt{-\lambda}\right) \\
&= 2 + 2\cosh\sqrt{-\lambda} - \sqrt{-\lambda}\sinh\sqrt{-\lambda} \\
&= 4\cosh^2\frac{\sqrt{-\lambda}}{2} - 2\sqrt{-\lambda}\sinh\frac{\sqrt{-\lambda}}{2}\cosh\frac{\sqrt{-\lambda}}{2} \\
&= 2\cosh^2\frac{\sqrt{-\lambda}}{2}\left(2 - \sqrt{-\lambda}\tanh\frac{\sqrt{-\lambda}}{2}\right).
\end{aligned}
$$

The last equality is justified because $\cosh\left(\sqrt{-\lambda}/2\right) \neq 0$. Thefore the second factor must be zero, i.e., $\tanh\left(\sqrt{-\lambda}/2\right) = 2/\sqrt{-\lambda}$, which is exactly (5). Let λ_0 be the solution of this secular equation, which can be found numerically as $\lambda_0 \approx -5.756915$. For this λ_0, the relationship between the constants A and B above must have:

$$
A\sinh\sqrt{-\lambda_0} + B(1 + \cosh\sqrt{-\lambda_0}) = 0 \Longleftrightarrow B = -\frac{\sinh\sqrt{-\lambda_0}}{1 + \cosh\sqrt{-\lambda_0}}A.
$$

Thus, we have

$$\phi_0(x) = A\left(\cosh\sqrt{-\lambda_0}x - \frac{\sinh\sqrt{-\lambda_0}}{1 + \cosh\sqrt{-\lambda_0}}\sinh\sqrt{-\lambda_0}x\right)$$

$$= A'\left(\cosh\sqrt{-\lambda_0}x + \cosh\sqrt{-\lambda_0}\cosh\sqrt{-\lambda_0}x - \sinh\sqrt{-\lambda_0}\sinh\sqrt{-\lambda_0}x\right)$$

$$= A'\left(\cosh\sqrt{-\lambda_0}x + \cosh\sqrt{-\lambda_0}(1 - x)\right)$$

$$= A_0\cosh\sqrt{-\lambda_0}\left(x - \tfrac{1}{2}\right),$$

which is exactly (6). The constant A_0 is a normalization constant to have $\|\phi_0\|_{L^2(\Omega)} = 1$. Thus,

$$A_0 = \left(\int_0^1\left(\cosh\sqrt{-\lambda_0}\left(x - \tfrac{1}{2}\right)\right)^2 dx\right)^{-1/2}$$

$$= \left(\tfrac{1}{2} + \tfrac{1}{2}\int_0^1\cosh 2\sqrt{-\lambda_0}\left(x - \tfrac{1}{2}\right) dx\right)^{-1/2}$$

$$= \sqrt{2}\left(1 + \frac{\sinh\sqrt{-\lambda_0}}{\sqrt{-\lambda_0}}\right)^{-1/2}$$

$$\approx 0.7812598.$$

Case II $\lambda = 0$**:** In this case, we have $\phi''(x) = 0$. Thus $\phi(x) = Ax + B$. But the boundary condition (4) leads to $2A + B = -A = A$, i.e., $A = B = 0$. Therefore, $\lambda = 0$ is not an eigenvalue for this problem.

Case III $\lambda > 0$**:** In this case, the eigenfunction is of the form $\phi(x) = A\cos\sqrt{\lambda}x + B\sin\sqrt{\lambda}x$. Similarly to Case I, using the boundary condition (4), we have

$$\phi(0) + \phi(1) = A(1 + \cos\sqrt{\lambda}) + B\sin\sqrt{\lambda}$$

$$= -\phi'(0) = -\sqrt{\lambda}B$$

$$= \phi'(1) = \sqrt{\lambda}\left(-A\sin\sqrt{\lambda} + B\cos\sqrt{\lambda}\right)$$

From these equalities, we have the following 2×2 linear system for A and B:

$$\begin{bmatrix} 1 + \cos\sqrt{\lambda} & \sqrt{\lambda} + \sin\sqrt{\lambda} \\ -\sin\sqrt{\lambda} & 1 + \cos\sqrt{\lambda} \end{bmatrix}\begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Again, the vanishing determinant of this equation leads to

$$0 = \left(1 + \cos\sqrt{\lambda}\right)^2 + \sin\sqrt{\lambda}\left(\sqrt{\lambda} + \sin\sqrt{\lambda}\right)$$
$$= 2 + 2\cos\sqrt{\lambda} + \sqrt{\lambda}\sin\sqrt{\lambda}$$
$$= 2\cos\frac{\sqrt{\lambda}}{2}\left(2\cos\frac{\sqrt{\lambda}}{2} + \sqrt{\lambda}\sin\frac{\sqrt{\lambda}}{2}\right).$$

Thus, we have $\cos\left(\sqrt{\lambda}/2\right) = 0$ or $2\cos\left(\sqrt{\lambda}/2\right) + \sqrt{\lambda}\sin\left(\sqrt{\lambda}/2\right) = 0$. From the former, we can easily get the eigenvalues $\lambda = \lambda_{2m-1} = (2m-1)^2\pi^2$ and the corresponding eigenfunctions $\phi_{2m-1}(x) = \sqrt{2}\cos(2m-1)\pi x$, $m = 1,2,\ldots$, which is (7). From the latter, we get the secular equation, $\tan\left(\sqrt{\lambda}/2\right) = -2/\sqrt{\lambda}$, which is exactly (8). Considering the graph of periodic asymptotes of $\tan\left(\sqrt{\lambda}/2\right)$ at $\lambda = (2m-1)^2\pi^2$, we observe that the eigenvalues satisfying (8) and λ_{2m-1} are interlacing. Thus, we naturally denote the eigenvalues satisfying (8) by λ_{2m}, $m = 1,2,\ldots$, which must be computed numerically. The corresponding eigenfunctions are

$$\begin{aligned}
\phi_{2m}(x) &= A\left(\cos\sqrt{\lambda_{2m}}x + \frac{\sin\sqrt{\lambda_{2m}}}{1 + \cos\sqrt{\lambda_{2m}}}\sin\sqrt{\lambda_{2m}}x\right)\\
&= A'\left(\cos\sqrt{\lambda_{2m}}x + \cos\sqrt{\lambda_{2m}}\cos\sqrt{\lambda_{2m}}x + \sin\sqrt{\lambda_{2m}}\sin\sqrt{\lambda_{2m}}x\right)\\
&= A'\left(\cos\sqrt{\lambda_{2m}}x + \cos\sqrt{\lambda_{2m}}(1-x)\right)\\
&= A_{2m}\cos\sqrt{\lambda_{2m}}\left(x - \tfrac{1}{2}\right),
\end{aligned}$$

which is exactly (9). The constant A_{2m} is a normalization constant to have $\|\phi_{2m}\|_{L^2(\Omega)} = 1$. Thus,

$$\begin{aligned}
A_{2m} &= \left(\int_0^1 \left(\cos\sqrt{\lambda_{2m}}\left(x - \tfrac{1}{2}\right)\right)^2 dx\right)^{-1/2}\\
&= \left(\tfrac{1}{2} + \tfrac{1}{2}\int_0^1 \cos 2\sqrt{\lambda_{2m}}\left(x - \tfrac{1}{2}\right) dx\right)^{-1/2}\\
&= \sqrt{2}\left(1 + \frac{\sin\sqrt{\lambda_{2m}}}{\sqrt{\lambda_{2m}}}\right)^{-1/2},
\end{aligned}$$

which completes the proof. $\quad\square$

C Proof of Corollary 10

PROOF. In $\mathbb{R}^2$, we have $K(\boldsymbol{x}, \boldsymbol{y}) = -\frac{1}{2\pi} \ln |\boldsymbol{x} - \boldsymbol{y}|$. Thus,

$$\nabla_{\boldsymbol{y}} K(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{2\pi} \frac{\boldsymbol{x} - \boldsymbol{y}}{|\boldsymbol{x} - \boldsymbol{y}|^2}.$$

Now, the normal derivative of $K(\boldsymbol{x}, \boldsymbol{y})$ at $\boldsymbol{y} \in \Gamma$ can be computed easily as:

$$\frac{\partial K}{\partial \boldsymbol{v}_{\boldsymbol{y}}} (\boldsymbol{x} - \boldsymbol{y}) = \boldsymbol{v}_{\boldsymbol{y}} \cdot \nabla_{\boldsymbol{y}} K(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{2\pi} \frac{(\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{v}_{\boldsymbol{y}}}{|\boldsymbol{x} - \boldsymbol{y}|^2}.$$

When Ω is the unit disk in $\mathbb{R}^2$, things simplify considerably. Let $\boldsymbol{x} = e^{i\theta}$, $\boldsymbol{y} = e^{i\eta}$ be any two boundary points. Then, it is easy to show that

$$|\boldsymbol{x} - \boldsymbol{y}|^2 = 4 \sin^2 \frac{\theta - \eta}{2}, \quad (\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{v}_{\boldsymbol{y}} = (\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{y} = -2 \sin^2 \frac{\theta - \eta}{2},$$

which lead to

$$\frac{\partial K}{\partial \boldsymbol{v}_{\boldsymbol{y}}} (\boldsymbol{x} - \boldsymbol{y}) = \frac{1}{2\pi} \frac{(\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{v}_{\boldsymbol{y}}}{|\boldsymbol{x} - \boldsymbol{y}|^2} = -\frac{1}{4\pi}. \tag{C.1}$$

Let $\phi \in C^2(\Omega) \cap C^1(\overline{\Omega})$ in (3) be represented in the polar coordinates as $\phi(r, \theta)$. Plugging (C.1) and the 2D kernel above into (3) and multiplying 2 on both sides, we get

$$-\frac{1}{\pi} \int_{-\pi}^{\pi} \ln \left| 2 \sin \frac{\theta - \eta}{2} \right| \frac{\partial \phi}{\partial r} (1, \eta) \, d\eta = -\phi(1, \theta) - \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi(1, \eta) \, d\eta. \tag{C.2}$$

Note that the second term in the righthand side is a constant. Differentiating both sides in θ leads to

$$\frac{\partial \phi}{\partial \theta} (1, \theta) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\partial \phi}{\partial r} (1, \eta) \cot \frac{\theta - \eta}{2} \, d\eta = \mathcal{H} \frac{\partial \phi}{\partial r} (1, \theta),$$

where $\mathcal{H}$ is the Hilbert transform on the unit circle in $\mathbb{R}^2$. We now note that $\mathcal{H}^2 = -Id$ where Id is the identity operator. Thus, we have

$$\frac{\partial \phi}{\partial r} (1, \theta) = -\mathcal{H} \frac{\partial \phi}{\partial \theta} (1, \theta) = -\frac{\partial \mathcal{H} \phi}{\partial \theta} (1, \theta),$$

which is exactly (10).

Finally, let us compute the eigenfunctions and the corresponding eigenvalues satisfying (10). Using the separation of variables (see e.g., [29, Sec. 10.2]), it is easy to derive that each eigenfunction is of the form:

$$\phi(r,\theta) = J_m\left(\sqrt{\lambda}r\right)\begin{pmatrix}\cos\\\sin\end{pmatrix}(m\theta), \quad m = 0,1,\dots. \tag{C.3}$$

If our boundary condition were the standard Dirichlet condition, then the eigenvalue λ could be obtained by the condition $J_m(\sqrt{\lambda}) = 0$. However, our boundary condition is quite different from the Dirichlet case: it must satisfy (10). Let us first consider the case $m > 0$. In this case, plugging (C.3) into (10), we have

$$\begin{aligned}\frac{\partial\phi}{\partial r}(1,\theta) &= \sqrt{\lambda}J'_m\left(\sqrt{\lambda}\right)\begin{pmatrix}\cos\\\sin\end{pmatrix}(m\theta)\\ &= -\frac{\partial\mathcal{H}\phi}{\partial\theta}(1,\theta)\\ &= -J_m\left(\sqrt{\lambda}\right)\frac{\partial}{\partial\theta}\begin{pmatrix}\sin\\-\cos\end{pmatrix}(m\theta)\\ &= -mJ_m\left(\sqrt{\lambda}\right)\begin{pmatrix}\cos\\\sin\end{pmatrix}(m\theta).\end{aligned}$$

From these, we have $\sqrt{\lambda}J'_m\left(\sqrt{\lambda}\right) = -mJ_m\left(\sqrt{\lambda}\right)$ for $m > 0$. Now, inserting $z = \sqrt{\lambda}$ in the following standard recursion formulas (see e.g., [1, Formula 9.1.27]):

$$zJ'_m(z) = -zJ_{m+1}(z) + mJ_m(z), \quad z(J_{m-1}(z) + J_{m+1}(z)) = 2mJ_m(z) \quad \text{for } z \in \mathbb{C},$$

we have

$$\sqrt{\lambda}J'_m\left(\sqrt{\lambda}\right) = -mJ_m\left(\sqrt{\lambda}\right) \iff \sqrt{\lambda}J_{m+1}\left(\sqrt{\lambda}\right) = 2mJ_m\left(\sqrt{\lambda}\right)$$
$$\iff \sqrt{\lambda}J_{m+1}\left(\sqrt{\lambda}\right) = \sqrt{\lambda}\left(J_{m-1}\left(\sqrt{\lambda}\right) + J_{m+1}\left(\sqrt{\lambda}\right)\right) \iff \sqrt{\lambda}J_{m-1}\left(\sqrt{\lambda}\right) = 0.$$

However, we cannot have $\lambda = 0$ because $J_m(0) = 0$ for all $m > 0$. Hence, we must have $J_{m-1}\left(\sqrt{\lambda}\right) = 0$ for $m = 1,2,\dots$, and consequently the eigenvalues are deter-

mined as the zeros of J_{m-1}. In other words, we can set $\lambda_{m,n} = \beta^2_{m-1,n}$, $m, n = 1, 2, \ldots$.

Now, let us consider the case of $m = 0$. The eigenfunction is of the form $\phi(r, \theta) = J_0\left(\sqrt{\lambda}r\right)$, i.e., a radial function. In this case, we need to go back to (C.2) because (10) simply says $0 = 0$. Now, plugging $\phi(r, \theta) = J_0\left(\sqrt{\lambda}r\right)$ into (C.2), we have

$$-\frac{1}{\pi}\int_{-\pi}^{\pi}\ln\left|2\sin\frac{\theta-\eta}{2}\right|\sqrt{\lambda}J_0\left(\sqrt{\lambda}\right)d\eta = -J_0\left(\sqrt{\lambda}\right) - \frac{1}{2\pi}\int_{-\pi}^{\pi}J_0\left(\sqrt{\lambda}\right)d\eta = -2J_0\left(\sqrt{\lambda}\right).$$

Since the integral

$$\int_{-\pi}^{\pi}\ln\left|2\sin\frac{\theta-\eta}{2}\right|d\eta = 0 \quad\text{for any } \theta \in [-\pi, \pi]^2,$$

we must have $J_0\left(\sqrt{\lambda}\right) = 0$. Thus, we have $\lambda_{0,n} = \beta^2_{0,n}$, $n = 1, 2, \ldots$. This completes the proof. $\square$

References

[1] M. Abramowitz, I. A. Stegun, Handbook of Mathematical Functions, Dover Publications, Inc., New York, 1972, 9th printing.

[2] B. K. Alpert, A class of bases in L^2 for the sparse representation of integral operators, SIAM J. Math. Anal. 24 (1) (1993) 246–262.

[3] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation 15 (2003) 1373–1396.

[4] A. Cantoni, P. Butler, Eigenvalues and eigenvectors of symmetric centrosymmetric matrices, Linear Algebra Appl. 13 (1976) 275–288.

---

[2] This can be derived by the formula $\int_0^{\pi/2}\ln\sin x\,dx = -\frac{\pi}{2}\ln 2$ via [14, Formula 4.224.3] and utilizing the periodicity of the integrand.

[5] R. R. Coifman, S. Lafon, Diffusion maps, Applied and Computational Harmonic Analysis 21 (1) (2006) 5–30.

[6] R. R. Coifman, S. Lafon, Geometric harmonics, Applied and Computational Harmonic Analysis 21 (1) (2006) 32–52.

[7] I. Daubechies, Ten Lectures on Wavelets, vol. 61 of CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, Philadelphia, PA, 1992.

[8] E. B. Davies, Spectral Theory and Differential Operators, vol. 42 of Cambridge Studies in Advanced Mathematics, Cambridge Univ. Press, 1995.

[9] R. Delanghe, On some properties of the Hilbert transform in Euclidean space, Bull. Belg. Math. Soc. 11 (2004) 163–180.

[10] R. A. DeVore, Nonlinear approximation, in: Acta Numerica, Cambridge Univ. Press, 1998, pp. 51–150.

[11] J. Dodziuk, Eigenvalues of the Laplacian and the heat equation, Amer. Math. Monthly 88 (9) (1981) 686–695.

[12] G. B. Folland, Introduction to Partial Differential Equations, 2nd ed., Princeton Univ. Press, 1995.

[13] B. Friedman, Principles and Techniques of Applied Mathematics, John Wiley & Sons, Inc., New York, 1956, republished by Dover Publications, Inc. in 1990.

[14] I. S. Gradshteyn, I. M. Ryzhik, Table of Integrals, Series, and Products, seventh ed., Academic Press, 2007.

[15] L. Greengard, V. Rokhlin, A fast algorithm for particle simulation, J. Comput. Phys. 73 (2) (1987) 325–348.

[16] L. Greengard, J. Strain, The fast Gauss transform, SIAM J. Sci. Stat. Comput. 12 (1) (1991) 79–94.

[17] F. John, Partial Differential Equations, vol. 1 of Applied Mathematical Sciences, 4th ed., Springer-Verlag, New York, 1982.

[18] M. Kirby, L. Sirovich, Application of the Karhunen-Loève procedure for the characterization of human faces, IEEE Trans. Pattern Anal. Machine Intell. 12 (1) (1990) 103–108.

[19] R. Kress, Linear Integral Equations, 2nd ed., Springer-Verlag, New York, 1999.

[20] E. Kreyszig, Introductory Functional Analysis with Applications, Wiley Classics Library, John Wiley & Sons, Inc., New York, 1989.

[21] D. Porter, D. S. G. Stirling, Integral Equations: A Practical Treatment from Spectral Theory to Applications, Cambridge Texts in Applied Mathematics, Cambridge Univ. Press, New York, 1990.

[22] V. C. Raykar, R. Duraiswami, The improve fast Gauss transform with applications to machine learning, in: L. Bottou, O. Chapelle, D. Decoste, , J. Weston (eds.), Large Scale Kernel Machines, Neural Information Processing, chap. 8, The MIT Press, Cambridge, MA, 2007, pp. 175–202.

[23] N. Saito, Image approximation and modeling via least statistically dependent bases, Pattern Recognition 34 (2001) 1765–1784.

[24] N. Saito, Geometric harmonics as a statistical image processing tool for images defined on irregularly-shaped domains, in: Proc. 13th IEEE Workshop on Statistical Signal Processing, IEEE, 2005.

[25] N. Saito, J.-F. Remy, The polyharmonic local sine transform: A new tool for local image analysis and synthesis without edge effect, Applied and Computational Harmonic Analysis 20 (1) (2006) 41–73.

[26] N. Saito, K. Yamatani, J. Zhao, Generalized polyharmonic trigonometric transform: A tool for object-oriented image analysis and synthesis, Tech. rep., Dept. Math., Univ. California, Davis, in preparation (2007).

[27] D. Slepian, Some comments on Fourier analysis, uncertainty and modeling, SIAM Review 25 (1983) 379–393.

[28] G. Strang, The discrete cosine transform, SIAM Review 41 (1) (1999) 135–147.

[29] W. A. Strauss, Partial Differential Equations: An Introduction, John Wiley & Sons, Inc., New York, 1992.

[30] L. N. Trefethen, D. Bau, III, Numerical Linear Algebra, SIAM, Philadelphia, 1997.

[31] K. Wakamatsu, S. Kubo, M. Matsuoka, K. Hasegawa, M. Sugiura, Japan Engineering Geomorphologic Classification Map, University of Tokyo Press, 2005, (product serial number: JEGM0001), with CD-ROM.

[32] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: From error measurement to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–613.

[33] X. Xue, On a fast algorithm for computing the Laplacian eigenpairs via commuting integral operators, Ph.D. thesis, Dept. Math., Univ. California, Davis (2007).

[34] K. Yamatani, N. Saito, Improvement of DCT-based compression algorithms using Poisson's equation, IEEE Trans. Image Process. 15 (12) (2006) 3672–3689, to appear.

[35] J. Zhao, Efficient approximations: Overcoming boundary effects, Ph.D. thesis, Dept. Math., Univ. California, Davis (2006).