Recent Advances in Image Analysis: On the Use of Laplacian Eigenfunctions for Images and Datasets

Naoki Saito

Department of Mathematics
University of California, Davis

February 15, 2008

Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

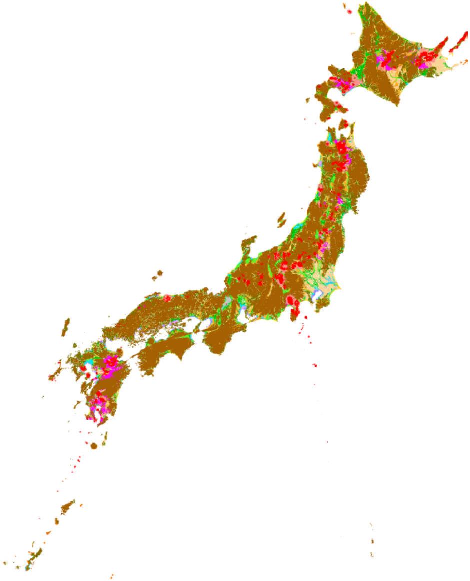
Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

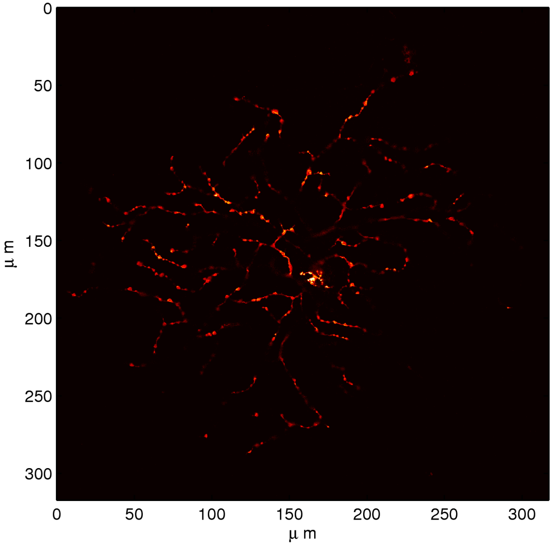
Motivations

- Consider a bounded domain of general (may be quite complicated) shape $\Omega \subset \mathbb{R}^d$.
- Want to analyze the spatial frequency information inside of the object defined in $\Omega \Longrightarrow$ need to avoid the Gibbs phenomenon due to $\Gamma = \partial\Omega$.
- Want to represent the object information efficiently for analysis, interpretation, discrimination, etc. $\Longrightarrow$ fast decaying expansion coefficients relative to a meaningful basis.
- Want to extract geometric information about the domain $\Omega \Longrightarrow$ shape clustering/classification.

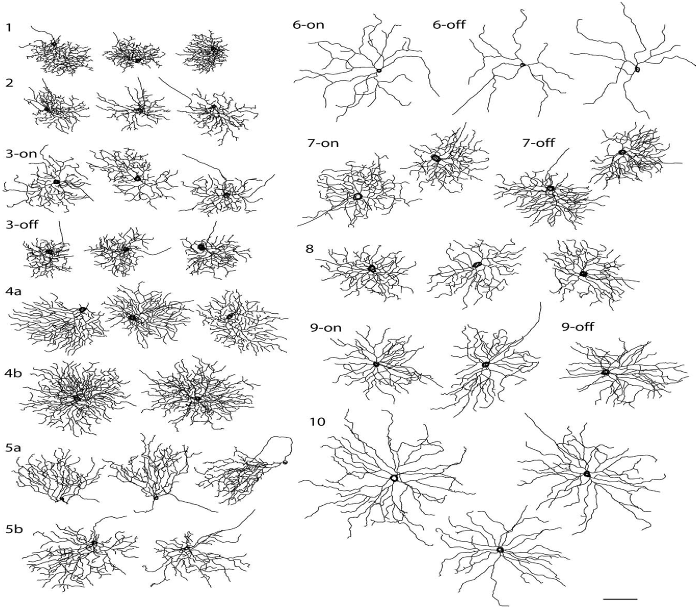
Motivations . . . Data Analysis on a Complicated Domain



Motivations . . . Clustering Complicated Objects



Motivations . . . Clustering Complicated Objects . . .



Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Eigenfunctions of Laplacian

- Our previous attempt was to extend the object to the outside smoothly and then bound it nicely with a rectangular box followed by the ordinary Fourier analysis.
- Why not analyze (and synthesize) the object using genuine basis functions tailored to the domain?
- After all, *sines* (and *cosines*) are the eigenfunctions of the Laplacian on the *rectangular* domain with Dirichlet (and Neumann) boundary condition.
- *Spherical harmonics, Bessel functions, and Prolate Spheroidal Wave Functions*, are part of the eigenfunctions of the Laplacian (via separation of variables) for the *spherical, cylindrical, and spheroidal* domains, respectively.

Eigenfunctions of Laplacian ...

- Consider an operator $\mathcal{L} = -\Delta$ in $L^2(\Omega)$ with *appropriate* boundary condition.
- Analysis of $\mathcal{L}$ is difficult due to unboundedness, etc.
- Much better to analyze its inverse, i.e., the Green's operator because it is compact and self-adjoint.
- Thus $\mathcal{L}^{-1}$ has discrete spectra (i.e., a countable number of eigenvalues with finite multiplicity) except 0 spectrum.
- $\mathcal{L}$ has a complete orthonormal basis of $L^2(\Omega)$, and this allows us to do eigenfunction expansion in $L^2(\Omega)$.

Eigenfunctions of Laplacian . . . Difficulties

- The key difficulty is to compute such eigenfunctions; directly solving the Helmholtz equation (or eigenvalue problem) on a general domain is tough.
- Unfortunately, computing the Green's function for a general Ω satisfying the usual boundary condition (i.e., Dirichlet, Neumann) is also very difficult.

Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Integral Operators Commuting with Laplacian

- The key idea is to find an integral operator commuting with the Laplacian without imposing the strict boundary condition a priori.
- Then, we know that the eigenfunctions of the Laplacian is the same as those of the integral operator, which is easier to deal with, due to the following

Theorem (G. Frobenius 1878?; B. Friedman 1956)

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. That is, $\mathcal{L}\varphi = \lambda\varphi$ and $\mathcal{K}\varphi = \mu\varphi$.

Integral Operators Commuting with Laplacian . . .

- Let's replace the Green's function $G(\mathbf{x}, \mathbf{y})$ by the fundamental solution of the Laplacian:

$$K(\mathbf{x}, \mathbf{y}) = \begin{cases} -\frac{1}{2}|x - y| & \text{if } d = 1, \\ -\frac{1}{2\pi} \log |\mathbf{x} - \mathbf{y}| & \text{if } d = 2, \\ \frac{|\mathbf{x}-\mathbf{y}|^{2-d}}{(d-2)\omega_d} & \text{if } d > 2. \end{cases}$$

- The price we pay is to have rather implicit, non-local boundary condition although we do not have to deal with this condition directly.

Integral Operators Commuting with Laplacian . . .

- Let $\mathcal{K}$ be the integral operator with its kernel $K(\mathbf{x}, \mathbf{y})$:

$$\mathcal{K}f(\mathbf{x}) \triangleq \int_\Omega K(\mathbf{x}, \mathbf{y}) f(\mathbf{y})\, \mathrm{d}\mathbf{y}, \quad f \in L^2(\Omega).$$

Theorem (NS 2005)

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

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

for all $\mathbf{x} \in \Gamma$, where φ is an eigenfunction common for both operators.

Corollary (NS 2005)

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

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

and $\{\varphi_j\}_j$ *forms an orthonormal basis of* $L^2(\Omega)$*.*

Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

1D Example

- Consider the unit interval $\Omega = (0, 1)$.
- Then, our integral operator $\mathcal{K}$ with the kernel $K(x, y) = -|x - y|/2$ gives rise to the following eigenvalue problem:

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

$$\varphi(0) + \varphi(1) = -\varphi'(0) = \varphi'(1).$$

- The kernel $K(\mathbf{x}, \mathbf{y})$ is of Toeplitz form $\Longrightarrow$ Eigenvectors must have even and odd symmetry (Cantoni-Butler '76).
- In this case, we have the following explicit solution.

1D Example . . .

- $\lambda_0 \approx -5.756915$, which is a solution of $\tanh \frac{\sqrt{-\lambda_0}}{2} = \frac{2}{\sqrt{-\lambda_0}}$,

$$\varphi_0(x) = A_0 \cosh \sqrt{-\lambda_0}\left(x - \frac{1}{2}\right);$$

- $\lambda_{2m-1} = (2m-1)^2\pi^2$, $m = 1, 2, \ldots$,

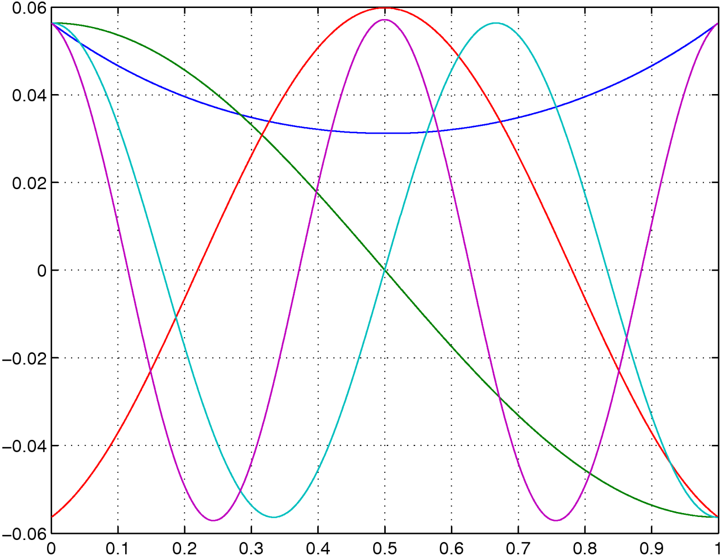
$$\varphi_{2m-1}(x) = \sqrt{2}\cos(2m-1)\pi x;$$

- λ_{2m}, $m = 1, 2, \ldots$, which are solutions of $\tan \frac{\sqrt{\lambda_{2m}}}{2} = -\frac{2}{\sqrt{\lambda_{2m}}}$,

$$\varphi_{2m}(x) = A_{2m} \cos \sqrt{\lambda_{2m}}\left(x - \frac{1}{2}\right),$$

where A_k, $k = 0, 1, \ldots$ are normalization constants.

First 5 Basis Functions



1D Example: Comparison

- The Laplacian eigenfunctions with the Dirichlet boundary condition: $-\varphi'' = \lambda\varphi$, $\varphi(0) = \varphi(1) = 0$, are *sines*. The Green's function in this case is:
$$G_D(x, y) = \min(x, y) - xy.$$
- Those with the Neumann boundary condition, i.e., $\varphi'(0) = \varphi'(1) = 0$, are *cosines*. The Green's function is:
$$G_N(x, y) = -\max(x, y) + \frac{1}{2}(x^2 + y^2) + \frac{1}{3}.$$
- Remark: Gridpoint $\Leftrightarrow$ DST-I/DCT-I;
Midpoint$\Leftrightarrow$ DST-II/DCT-II.

Outline

1 Motivations
2 Laplacian Eigenfunctions
3 Integral Operators Commuting with Laplacian
4 Examples
- 1D Example
- 2D Example
- 3D Example

5 Discretization of the Problem
6 Applications
- Image Approximation
- Statistical Image Analysis; Comparison with PCA
- Clustering Mouse Retinal Ganglion Cells

7 Fast Algorithms for Computing Eigenfunctions
8 Conclusions
9 References/Acknowledgment

2D Example

- Consider the unit disk Ω. Then, our integral operator $\mathcal{K}$ with the kernel $K(\mathbf{x},\mathbf{y}) = -\frac{1}{2\pi}\log|\mathbf{x}-\mathbf{y}|$ gives rise to:

$$-\Delta\varphi = \lambda\varphi, \quad \text{in } \Omega;$$

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

where $\mathcal{H}$ is the Hilbert transform for the circle, i.e.,

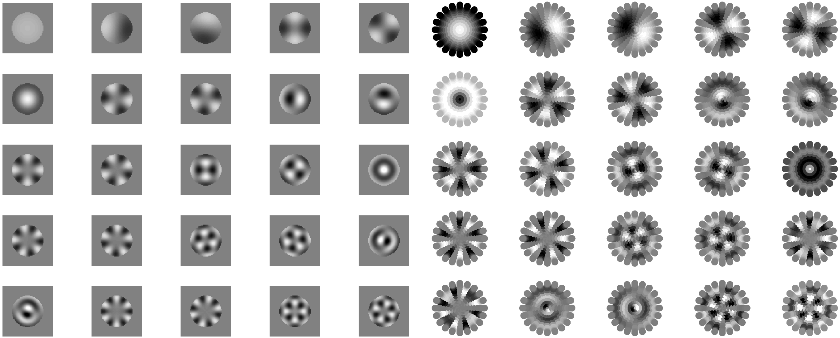
$$\mathcal{H}f(\theta) \triangleq \frac{1}{2\pi}\,\mathrm{pv}\int_{-\pi}^{\pi} f(\eta)\cot\left(\frac{\theta-\eta}{2}\right)\,\mathrm{d}\eta \quad \theta\in[-\pi,\pi].$$

- Let $\beta_{k,\ell}$ is the ℓth zero of the Bessel function of order k, $J_k(\beta_{k,\ell}) = 0$. Then,

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

$$\lambda_{m,n} = \begin{cases} \beta^2_{m-1,n}, & \text{if } m=1,\ldots,\ n=1,2,\ldots, \\ \beta^2_{0,n} & \text{if } m=0,\ n=1,2,\ldots. \end{cases}$$

First 25 Basis Functions



(a) Our Basis (b) Dirichlet-Laplace

Outline

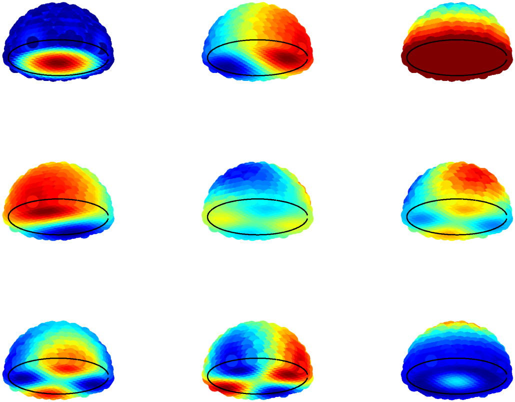
1 Motivations
2 Laplacian Eigenfunctions
3 Integral Operators Commuting with Laplacian
4 Examples
- 1D Example
- 2D Example
- 3D Example

5 Discretization of the Problem
6 Applications
- Image Approximation
- Statistical Image Analysis; Comparison with PCA
- Clustering Mouse Retinal Ganglion Cells

7 Fast Algorithms for Computing Eigenfunctions
8 Conclusions
9 References/Acknowledgment

3D Example

- Consider the unit ball Ω in $\mathbb{R}^3$. Then, our integral operator $\mathcal{K}$ with the kernel $K(\mathbf{x}, \mathbf{y}) = \frac{1}{4\pi|\mathbf{x}-\mathbf{y}|}$.
- Top 9 eigenfunctions cut at the equator viewed from the south:



Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Discretization of the Problem

- 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 $\prod_{i=1}^{d} \Delta x_i$.
- Assume that an object of our interest Ω consists of a subset of these boxes whose centers are$\{\mathbf{x}_i\}_{i=1}^{N}$.
- Under these assumptions, we can approximate the integral eigenvalue problem $\mathcal{K}\varphi = \mu\varphi$ with a simple quadrature rule with node-weight pairs $(\mathbf{x}_j, w_j)$ as follows.

$$\sum_{j=1}^{N} w_j K(\mathbf{x}_i, \mathbf{x}_j)\varphi(\mathbf{x}_j) = \mu\varphi(\mathbf{x}_i), \quad i = 1, \ldots, N, \quad w_j = \prod_{i=1}^{d} \Delta x_i.$$

- Let $K_{i,j} \triangleq w_j K(\mathbf{x}_i, \mathbf{x}_j)$, $\varphi_i \triangleq \varphi(\mathbf{x}_i)$, and $\boldsymbol{\varphi} \triangleq (\varphi_1, \ldots, \varphi_N)^T \in \mathbb{R}^N$. Then, the above equation can be written in a matrix-vector format as: $K\boldsymbol{\varphi} = \mu\boldsymbol{\varphi}$, 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.

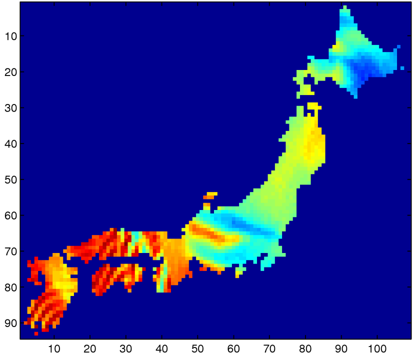
Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Outline

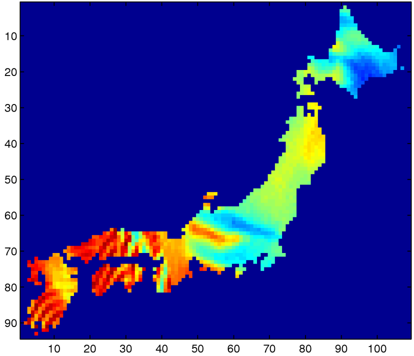
1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Image Approximation; Comparison with Wavelets

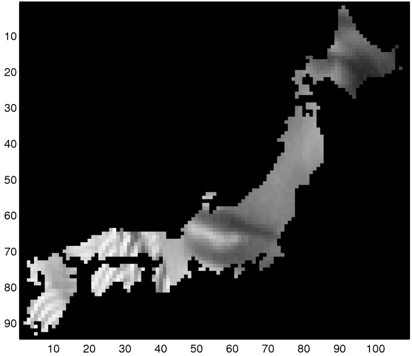


(a) What data?

Image Approximation; Comparison with Wavelets

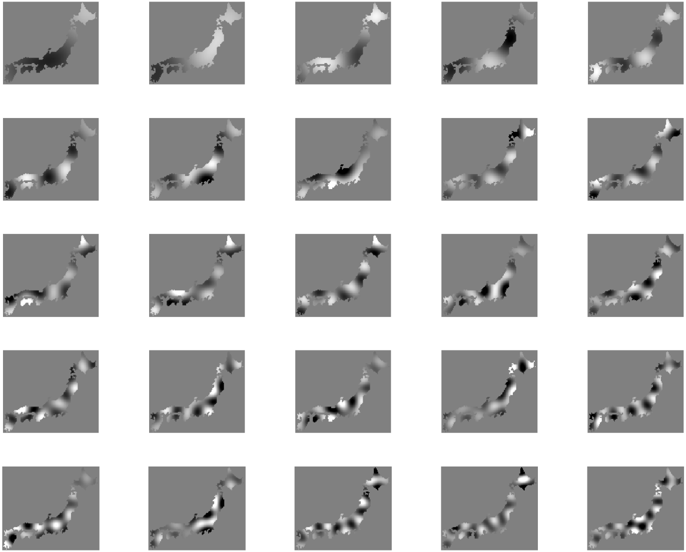


(a) What data?

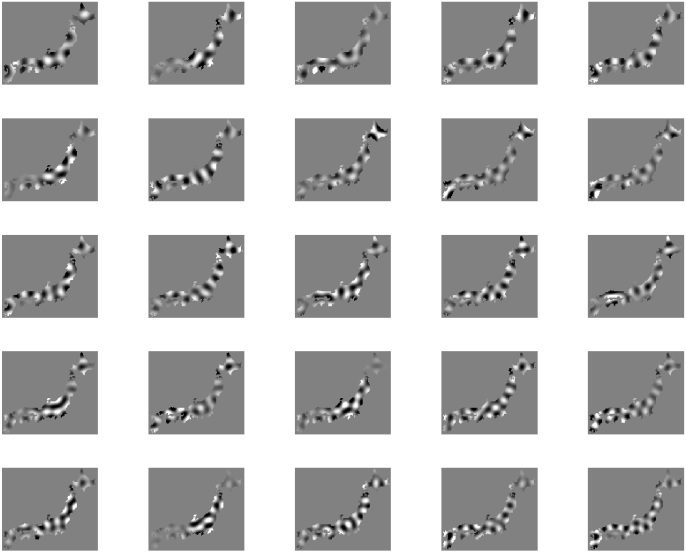


(b) $\chi_J \cdot$ Barbara

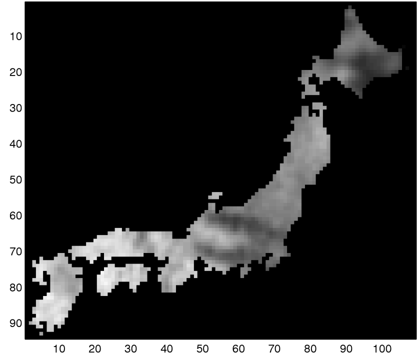
First 25 Basis Functions



Next 25 Basis Functions

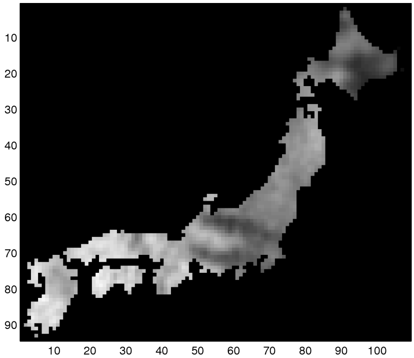


Reconstruction with Top 100 Coefficients

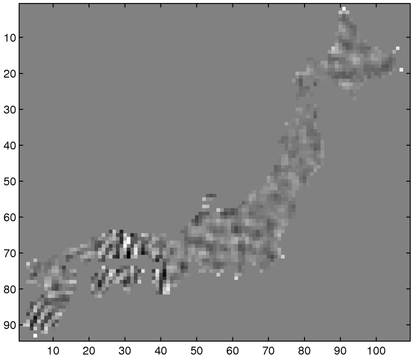


(a) Reconstruction

Reconstruction with Top 100 Coefficients

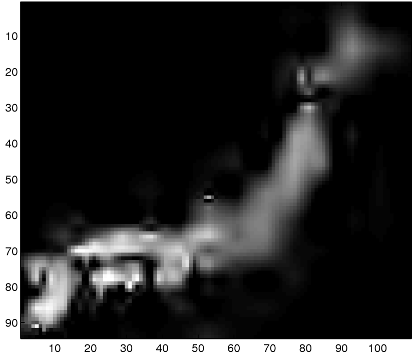


(a) Reconstruction



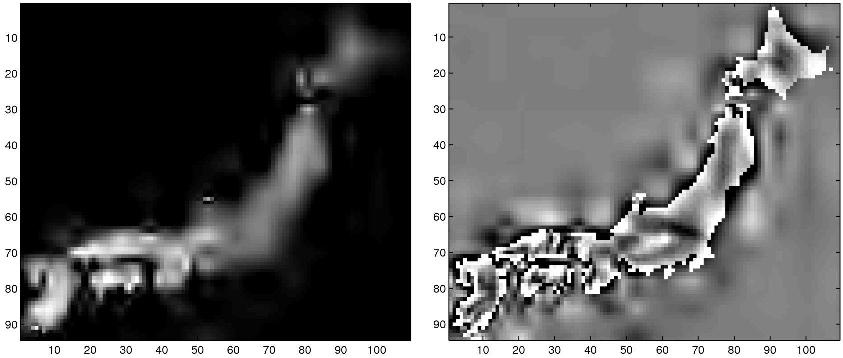
(b) Error

Reconstruction with Top 100 2D Wavelets (Symmlet 8)



(a) Reconstruction

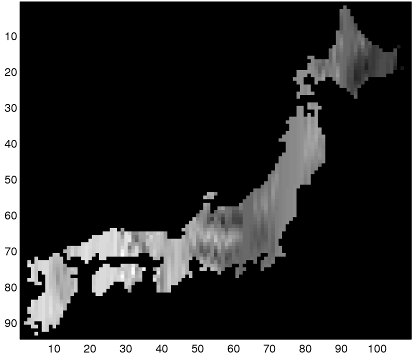
Reconstruction with Top 100 2D Wavelets (Symmlet 8)



(a) Reconstruction

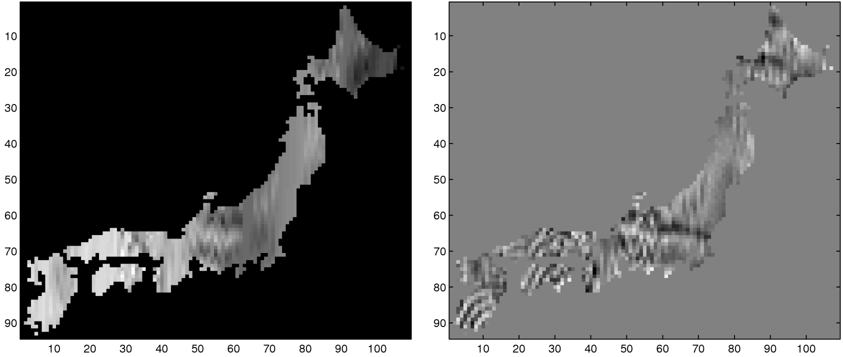
(b) Error

Reconstruction with Top 100 1D Wavelets (Symmlet 8)



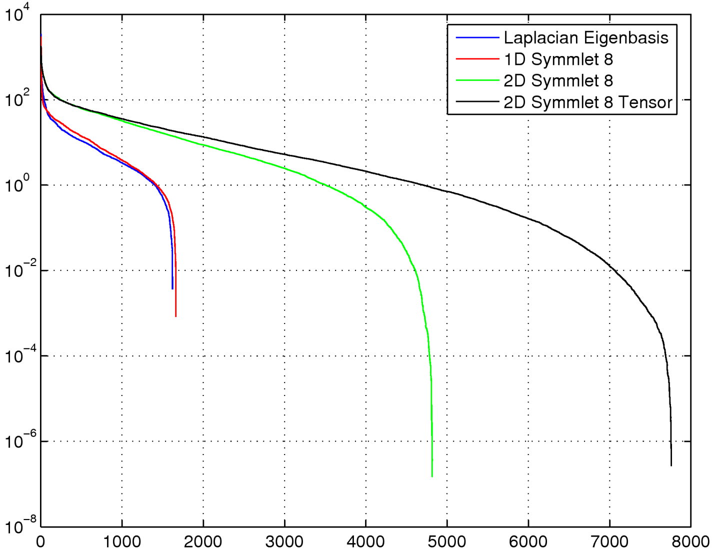
(a) Reconstruction

Reconstruction with Top 100 1D Wavelets (Symmlet 8)

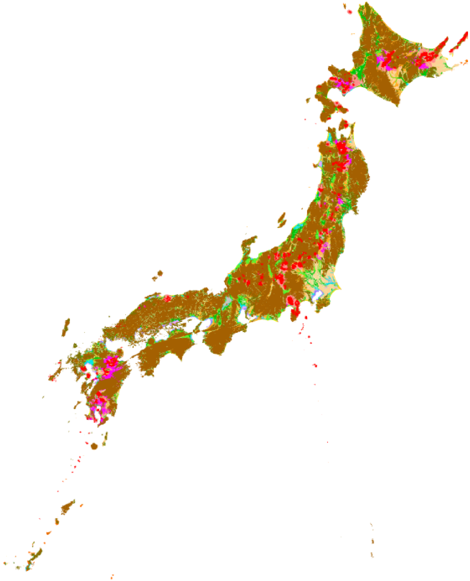


(a) Reconstruction (b) Error

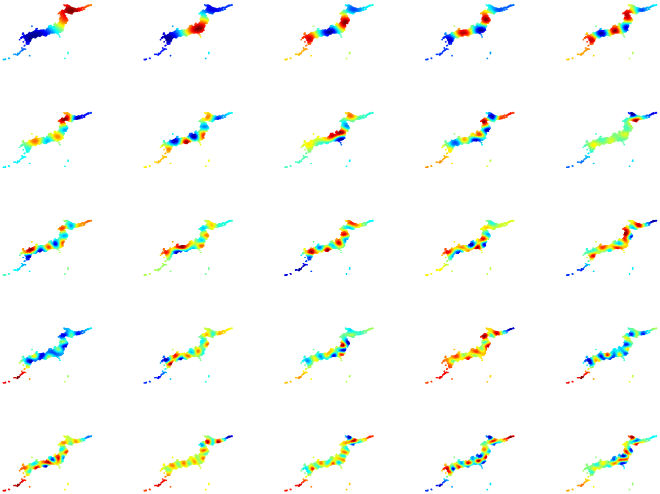
Comparison of Coefficient Decay



A Real Challenge: Kernel matrix is of 387924×387924.



First 25 Basis Functions via the FMM-based algorithm



Conjecture on the Coefficient Decay Rate

Conjecture (NS 2007)

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 $m \to \infty$.

Outline

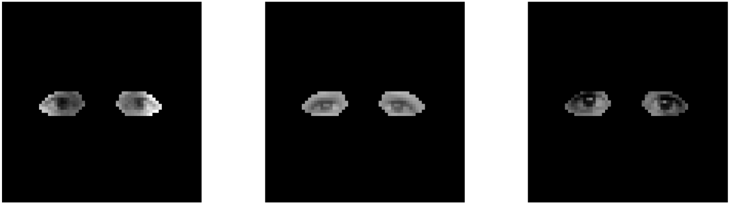
1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Comparison with PCA

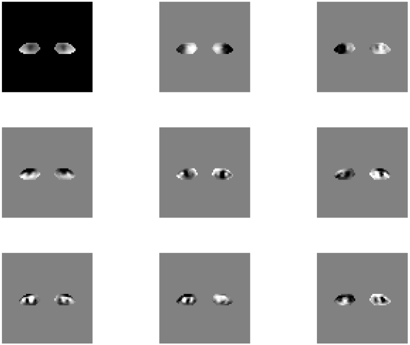
- Consider a stochastic process living on a domain Ω.
- PCA/Karhunen-Loève Transform is often used.
- PCA/KLT incorporate geometric information of the measurement (or pixel) location through the data correlation, i.e., implicitly.
- Our Laplacian eigenfunctions use explicit geometric information through the harmonic kernel $\varphi(\mathbf{x}, \mathbf{y})$.

Comparison with PCA: Example

- *"Rogue's Gallery"* dataset from Larry Sirovich
- 72 training dataset; 71 test dataset
- Left & right eye regions

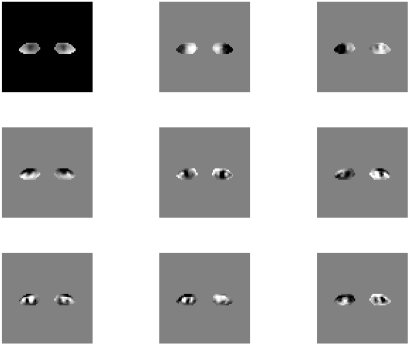


Comparison with PCA: Basis Vectors

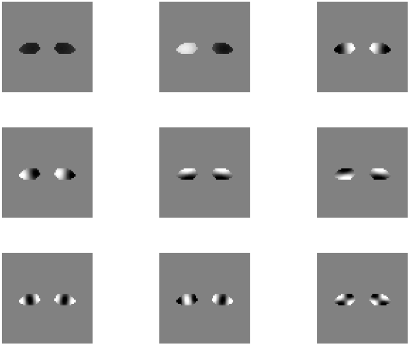


(a) KLB/PCA 1:9

Comparison with PCA: Basis Vectors

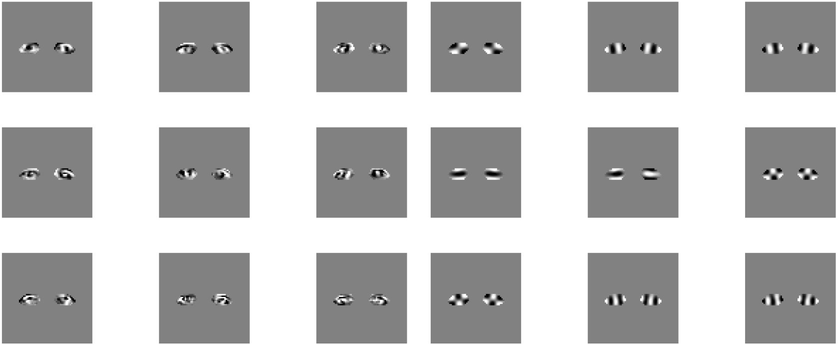


(a) KLB/PCA 1:9



(b) Laplacian Eigenfunctions 1:9

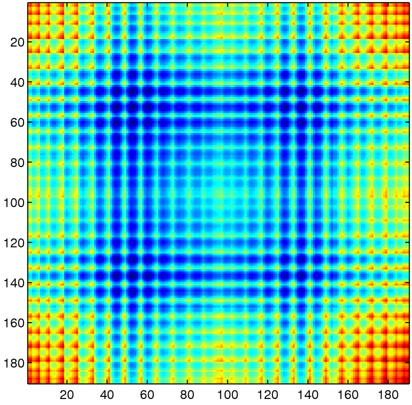
Comparison with PCA: Basis Vectors . . .



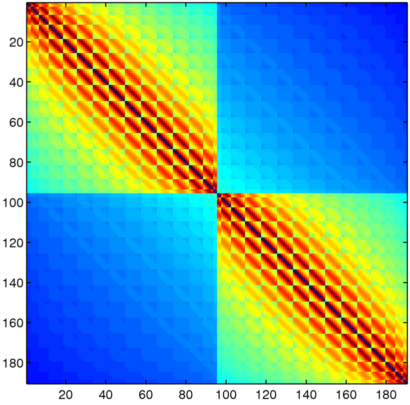
(a) KLB/PCA 10:18

(b) Laplacian Eigenfunctions 10:18

Comparison with PCA: Kernel Matrix

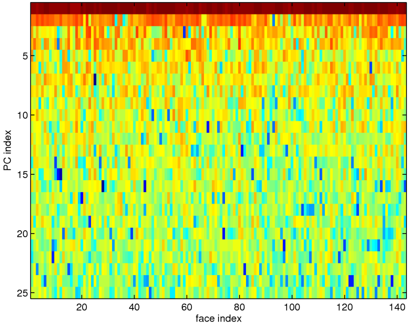


(a) Covariance

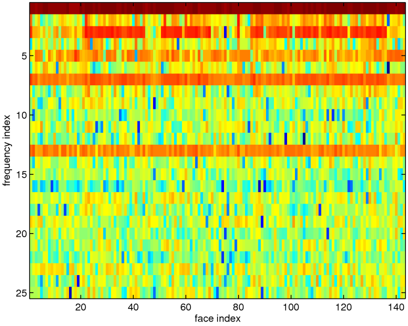


(b) Harmonic kernel

Comparison with PCA: Energy Distribution over Coordinates

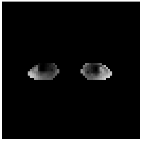


(a) KLB/PCA

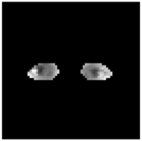


(b) Laplacian Eigenfunctions

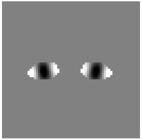
Comparison with PCA: Basis Vector #7 . . .



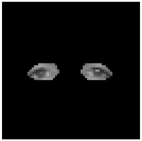
c_7:large



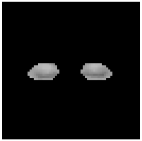
c_7:large



φ_7

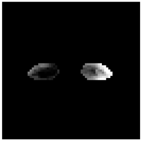


c_7:small

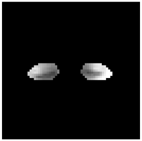


c_7:small

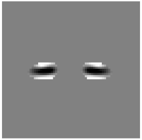
Comparison with PCA: Basis Vector #13 . . .



c_{13}:large

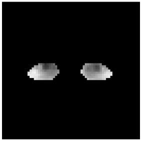


c_{13}:large



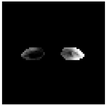
φ_{13}

c_{13}:small

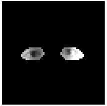


c_{13}:small

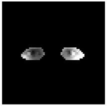
Asymmetry Detector



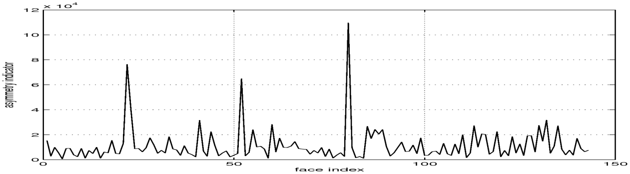
Eyes #80



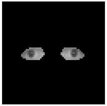
Eyes #22



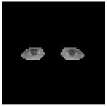
Eyes #52



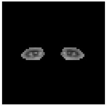
Asymmetry detector



Eyes #5



Eyes #84



Eyes #59

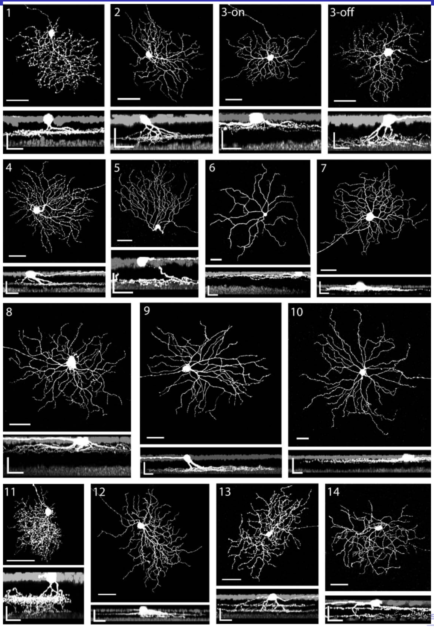
Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Clustering Mouse Retinal Ganglion Cells

- Objective: To understand how the structural/geometric properties of mouse retinal ganglion cells (RGCs) relate to the cell types and their functionality
- Why mouse? $\Longrightarrow$ great possibilities for genetic manipulation
- Data: 3D images of dendrites/axons of RGCs
- State of the Art: Process each image via specialized software to extract geometric/morphological parameters (totally 14 such parameters) followed by a conventional clustering algorithm
- These parameters include: somal size; dendric field size; total dendrite length; branch order; mean internal branch length; branch angle; mean terminal branch length, etc. $\Longrightarrow$ takes half a day per cell with a lot of human interactions!

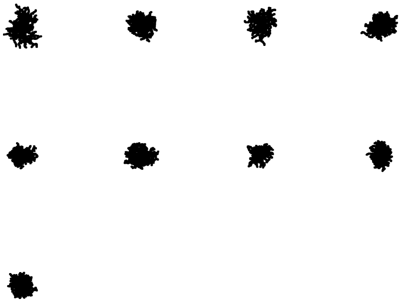
Clustering Mouse Retinal Ganglion Cells . . . 3D Data



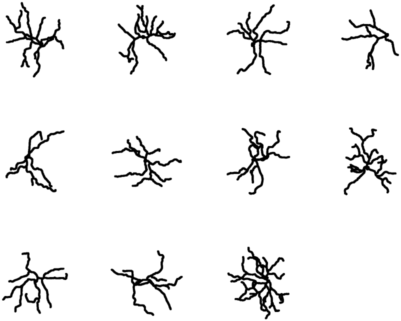
Preliminary Study on Mouse Retinal Ganglion Cells

- Use 2D plane projection data instead of full 3D
- Compute the smallest k Laplacian eigenvalues using our method (i.e., the largest k eigenvalues of $\mathcal{K}$) for each image
- Construct a feature vector per image
- Possible feature vectors reflecting geometric information:
  $\mathbf{F}_1 = (\lambda_1, \dots, \lambda_k)^T$; $\mathbf{F}_2 = (\mu_1, \dots, \mu_k)^T$; $\mathbf{F}_3 = (\lambda_1/\lambda_2, \dots, \lambda_1/\lambda_k)^T$; $\mathbf{F}_4 = (\mu_1/\mu_2, \dots, \mu_1/\mu_k)^T$.
- Do visualization and clustering

Preliminary Study on Mouse RGCs . . .

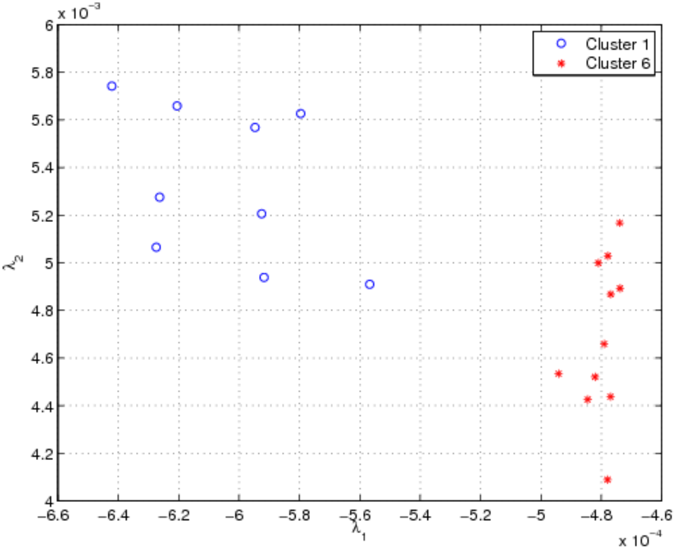


(a) Cluster 1

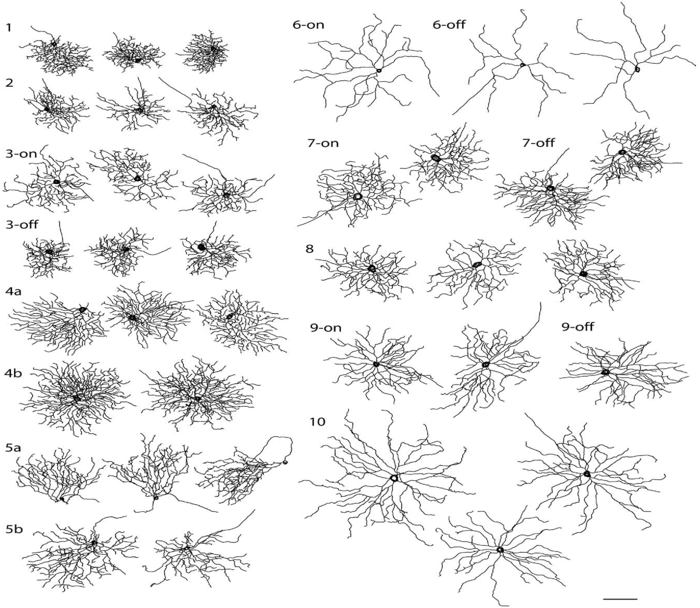


(b) Cluster 6

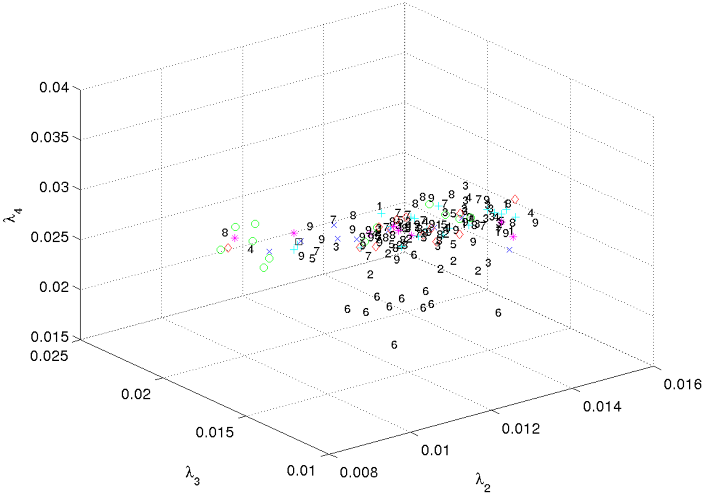
Crossplot of the First Two Laplacian Eigenvalues



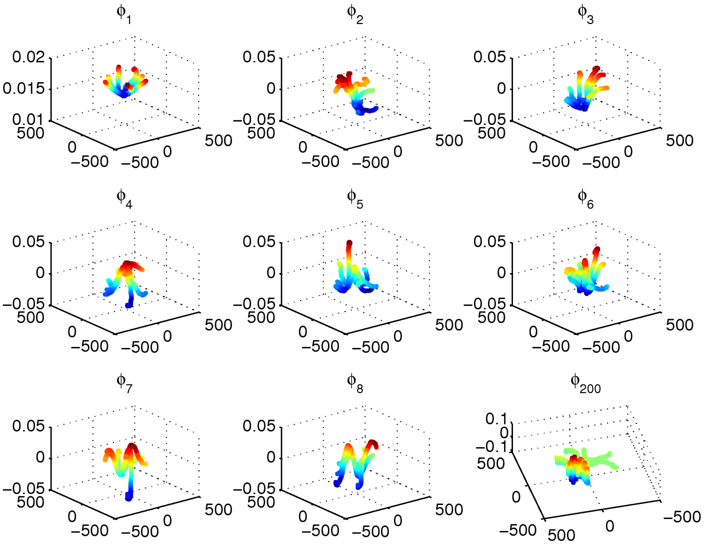
Clustering Results by the Manually Intensive Method



Plot of $(\lambda_2, \lambda_3, \lambda_4)$ of All the RGCs



Laplacian Eigenfunctions on a Mouse RGC



Challenges of Mouse Retinal Ganglion Cells

- Interpretation of our eigenvalues are not yet fully understood compared to the usual Dirichlet-Laplacian case that have been well studied: the Payne-Pólya-Weinberger inequalities; the Faber-Krahn inequalities; the Ashbaugh-Benguria results, etc. For $\Omega \in \mathbb{R}^d$,

$$\lambda_1^{(D)}(\Omega) \geq \left(\frac{|\mathcal{B}_1^d|}{|\Omega|}\right)^2 \lambda_1^{(D)}(\mathcal{B}_1^d), \quad \frac{\lambda_{k+1}^{(D)}(\Omega)}{\lambda_k^{(D)}(\Omega)} \leq \frac{\lambda_2^{(D)}(\mathcal{B}_1^d)}{\lambda_1^{(D)}(\mathcal{B}_1^d)}, \quad k = 1, 2, 3.$$

  Note the related work on "Shape DNA" by Reuter et al. (2005), and classification of tree leaves by Khabou et al. (2007).
- Original 3D data should be used instead of projected 2D data.
- Heat propagation on the dendrites may give us interesting and useful information; after all the dendrites are network to disseminate information via chemical reaction-diffusion mechanism.
- Construct actual graphs based on the connectivity and analyze them directly via spectral graph theory and diffusion maps $\Longrightarrow$ the Cheeger constant of a graph is related to the time to transmit "information" among its nodes! (T. Sunada)

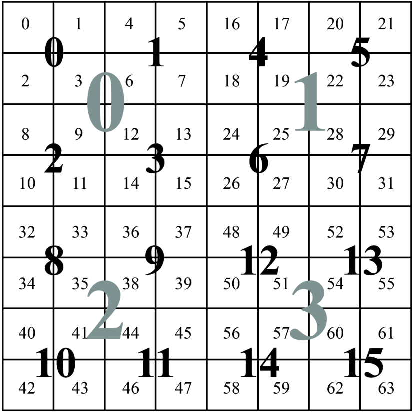
Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

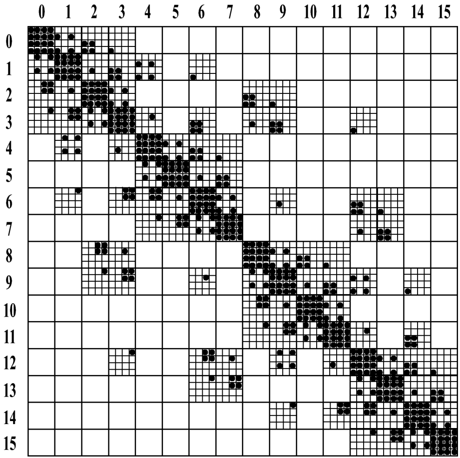
A Possible Fast Algorithm for Computing φ_j's

- Observation: our kernel function $K(\mathbf{x}, \mathbf{y})$ is of special form, i.e., the fundamental solution of Laplacian used in potential theory.
- Idea: Accelerate the matrix-vector product $K\varphi$ using the Fast Multipole Method (FMM).
- Convert the kernel matrix to the tree-structured matrix via the FMM whose submatrices are nicely organized in terms of their ranks. (Computational cost: our current implementation costs $O(N^2)$, but can achieve $O(N \log N)$ via the randomized SVD algorithm of Martinsson-Rokhlin-Tygert.)
- Construct $O(N)$ matrix-vector product module fully utilizing rank information (See also the work of Bremer (2007) and the "HSS" algorithm of Chandrasekaran et al. (2006)).
- Embed that matrix-vector product module in the Krylov subspace method, e.g., Lanczos iteration.
  (Computational cost: $O(N)$ for each eigenvalue/eigenvector).

Tree-Structured Matrix via FMM

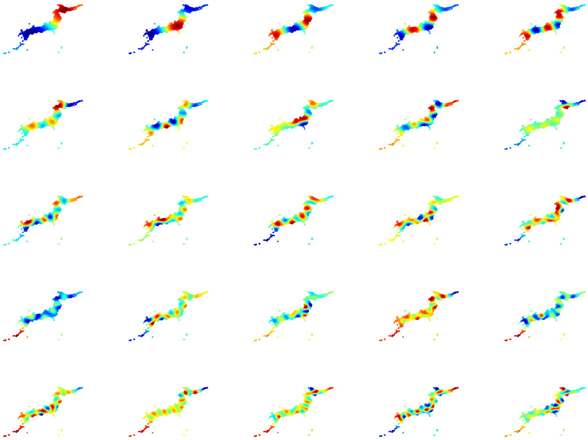


(a) Hierarchical indexing scheme

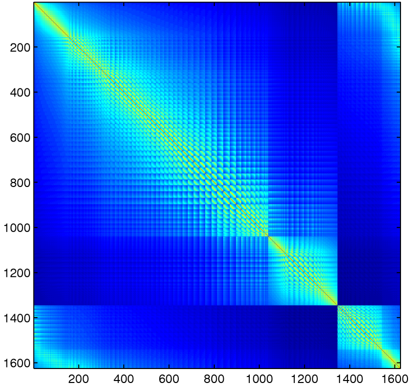


(b) Tree-Structured Matrix

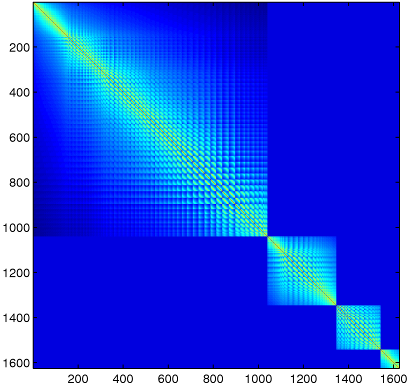
First 25 Basis Functions via the FMM-based algorithm



Splitting into Subproblems for Faster Computation

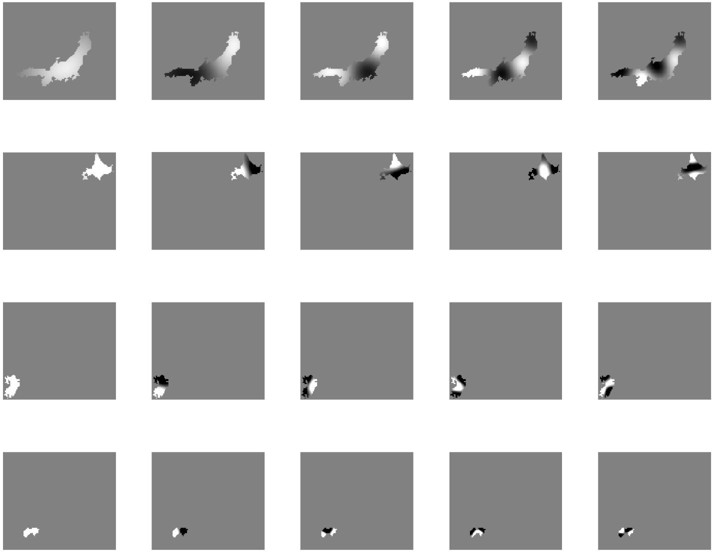


(a) Whole islands



(b) Separated islands

Eigenfunctions for Separated Islands



Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

Conclusions

- Allow object-oriented image analysis & synthesis
- Can get fast-decaying expansion coefficients
- Can decouple geometry/domain information and statistics of data
- Can extract geometric information of a domain through the eigenvalues
- $\exists$ A variety of applications: interpolation, extrapolation, local feature computation, solving heat equations on complicated domains ...
- Fast algorithms are the key for higher dimensions/large domains
- Connection to lots of interesting mathematics: spectral geometry, spectral graph theory, isoperimetric inequalities, Toeplitz operators, PDEs, potential theory, almost-periodic functions, ...
- Many things to be done:
  - Synthesize the Dirichlet-Laplacian eigenvalues/eigenfunctions from our eigenvalues/eigenfunctions
  - How about higher order, i.e., polyharmonic ?
  - Features derived from heat kernels ?
  - Improve our fast algorithm

Outline

1. Motivations
2. Laplacian Eigenfunctions
3. Integral Operators Commuting with Laplacian
4. Examples
   - 1D Example
   - 2D Example
   - 3D Example
5. Discretization of the Problem
6. Applications
   - Image Approximation
   - Statistical Image Analysis; Comparison with PCA
   - Clustering Mouse Retinal Ganglion Cells
7. Fast Algorithms for Computing Eigenfunctions
8. Conclusions
9. References/Acknowledgment

References

- Laplacian Eigenfunction Resource Page http://www.math.ucdavis.edu/~saito/lapeig/ contains
  - All the talk slides of the minisymposium "Laplacian Eigenfunctions and Their Applications, " which Mauro Maggioni and I organized for ICIAM 2007 at Zürich; and
  - My Course Note (elementary) on "Laplacian Eigenfunctions: Theory, Applications, and Computations"
- The following article is available at http://www.math.ucdavis.edu/~saito/publications/
  - N. Saito: "Data analysis and representation using eigenfunctions of Laplacian on a general domain," to appear *Applied & Computational Harmonic Analysis*, 2008.

Acknowledgment

- Raphy Coifman, Peter Jones (Yale)
- Dave Donoho (Stanford)
- Leo Chalupa, Julie Coombs (UCD, Neurobiology)
- John Hunter, Linh Lieu, Allen Xue, Zhihua Zhang (UCD, Math)
- NSF
- ONR

Thank you very much for your attention!