Volume 0 (1981), Number 0 pp. 1–29

Spectral Mesh Processing

Hao Zhang Oliver van Kaick Ramsay Dyer

Graphics, Usability and Visualization (GrUVi) Lab, School of Computing Science, Simon Fraser University, Canada

Abstract

Spectral methods for mesh processing and analysis rely on the eigenvalues, eigenvectors, or eigenspace projections derived from appropriately defined mesh operators to carry out desired tasks. Early work in this area can be traced back to the seminal paper by Taubin in 1995, where spectral analysis of mesh geometry based on a combinatorial Laplacian aids our understanding of the low-pass filtering approach to mesh smoothing. Over the past fifteen years, the list of applications in the area of geometry processing which utilize the eigenstructures of a variety of mesh operators in different manners have been growing steadily. Many works presented so far draw parallels from developments in fields such as graph theory, computer vision, machine learning, graph drawing, numerical linear algebra, and high-performance computing. This paper aims to provide a comprehensive survey on the spectral approach, focusing on its power and versatility in solving geometry processing problems and attempting to bridge the gap between relevant research in computer graphics and other fields. Necessary theoretical background is provided. Existing works covered are classified according to different criteria: the operators or eigenstructures employed, application domains, or the dimensionality of the spectral embeddings used. Despite much empirical success, there still remain many open questions pertaining to the spectral approach. These are discussed as we conclude the survey and provide our perspective on possible future research.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

1. Introduction

A great number of spectral methods have been proposed in the computing science literature in recent years, appearing in the fields of graph theory, computer vision, machine learning, visualization, graph drawing, high performance computing, and computer graphics. Generally speaking, a spectral method solves a problem by examining or manipulating the eigenvalues, eigenvectors, eigenspace projections, or a combination of these quantities, derived from an appropriately defined linear operator. More specific to the area of geometry processing and analysis, spectral methods have been developed to solve a diversity of problems including mesh compression, correspondence, parameterization, segmentation, sequencing, smoothing, symmetry detection, watermarking, surface reconstruction, and remeshing.

As a consequence of these developments, researchers are now faced with an extensive literature on spectral methods. It might be a laborious task for those new to the field to collect the necessary references in order to obtain an overview of the different methods, as well as an understanding of their similarities and differences. Furthermore, this is a topic that still instigates much interest, with many open problems deserving further investigation. Although introductory and short surveys which cover particular aspects of the spectral approach have been given before, e.g., by Gotsman [Got03] on spectral partitioning, layout, and geometry coding, and more recently by Lévy [Lé06] on a study of Laplace-Beltrami eigenfunctions, we believe a comprehensive survey is still called for. Our goal is to provide sufficient theoretical background, informative insights, as well as a thorough and up-to-date reference on the topic so as to draw interested researchers into this area and facilitate future research. Our effort should also serve to bridge the gap between past and on-going developments in several related disciplines.

The survey is organized as follows. We start with a historical account on the use of spectral methods. Section 3 offers an overview of the spectral approach, its general solution paradigm, and possible classifications. Section 4 motivates

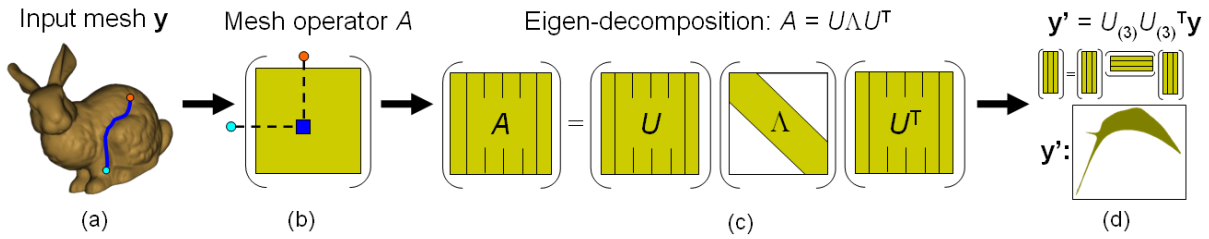


Figure 1: *Overview of the spectral approach to geometry processing. (a) An input mesh with certain relationship, e.g., geodesic distances between its primitives, is considered. (b) A linear mesh operator A is derived from this relationship. (c) Matrix A is eigendecomposed. (d) The eigenstructure is utilized in some way to serve the application as hand. Here, we construct the projection of the input mesh vertex coordinates* **y** *into the space spanned by the first three eigenvectors of A, where A is the graph Laplacian of the mesh (Section 2). The resulting structure lies in the plane and we show its boundary contour.*

the spectral approach through a few examples and mentions at a high level several natural applications. In Section 5, we provide some theoretical background with several theorems from linear algebra and other results that are frequently encountered in the literature covering spectral methods. Sections 6 and 7 survey existing operators used for spectral mesh processing and analysis, while Section 8 outlines how the different eigenstructures can be utilized to solve specific problems. Computational issues are addressed in Section 9. Section 10 finally provides a detailed survey of specific applications. Finally, we summarize and offer a few open questions for future consideration in Section 11.

2. A historical account

Historically, there have been three major threads underlying the development of spectral methods: spectral graph theory, a signal processing view relating to the classical Fourier analysis, and works in computer vision and machine learning, in particular those on kernel principal component analysis and spectral clustering. Spectral mesh processing draws inspirations from all these developments.

2.1. Spectral graph theory and the Fielder vector

Long before spectral methods came about in the computer graphics and geometry processing community, a great deal of knowledge from the field of spectral graph theory had been accumulated, following the pioneering work of Fielder [Fie73] in the 1970's. A detailed account of results from this theory can be found in the book by Chung [Chu97], two survey papers by Mohar [MP93, Moh97], as well as other graph theory texts, e.g., [Bol98].

The focus in spectral graph theory has been to derive relationships between the eigenvalues of the Laplacian or adjacency matrices of a graph and various fundamental properties of the graph, e.g., its diameter and connectivity [Chu97]. Given a graph $G = (V, E)$ with n vertices, the graph Laplacian $K = K(G)$ is an $n \times n$ matrix where

$$K_{ij} = \begin{cases} -1 & \text{if } (i,j) \in E, \\ d_i & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases}$$

and d_i is the degree or valence of vertex i.

In multidimensional calculus the Laplacian is a second-order differential operator frequently encountered in physics, e.g., in the study of wave propagation, heat diffusion, electrostatics, and fluid mechanics. In Riemannian geometry, the Laplace operator can be generalized to operate on functions defined on surfaces. The resulting Laplace-Beltrami operator is of particular interest in geometry processing. It has long been known that the graph Laplacian can be seen as a combinatorial version of the Laplace-Beltrami operator [Moh97]. Thus the interplay between spectral Riemannian geometry [Cha84] and spectral graph theory has been a subject of much study [Chu97].

One major development stemming from spectral graph theory that has found many practical applications involves the use of the *Fielder vector*, the eigenvector of a graph Laplacian corresponding to the smallest non-zero eigenvalue. These applications include graph layout [DPS02, Kor03], image segmentation via normalized cut [SM00], graph partitioning for parallel computing [AKY99], as well as sparse matrix reordering [BPS93] in numerical linear algebra. For the most part, these works had not received a great deal of attention in the graphics community until recently. For example, Fielder vectors have be used for mesh sequencing [IL05] and segmentation [ZL05, LZ07].

2.2. The signal processing view

Treating the mesh vertex coordinates as a 3D signal defined over the underlying mesh graph, Taubin [Tau95] first introduced the use of mesh Laplacian operators for discrete geometry processing in his SIGGRAPH 1995 paper. What had motivated this development were not results from spectral graph theory but an analogy between spectral analy-

sis with respect to the mesh Laplacian and the classical discrete Fourier analysis. Such an analysis was then applied over the irregular grids characterizing general meshes. Specifically, mesh smoothing was carried out via low-pass filtering. Subsequently, projections of a mesh signal into the eigenspaces of particular mesh Laplacians have been studied for different problems, e.g., implicit mesh fairing [DMSB99, KR05, ZF03], geometry compression [KG00], and mesh watermarking [OTMM01, OMT02]. A summary of the filtering approach to mesh processing was given by Taubin [Tau00]. Mesh Laplacian operators also allow us to define *differential coordinates* to represent mesh geometry, which is useful in applications such as mesh editing and shape interpolation; these works have been surveyed by Sorkine [Sor05] in her state-of-the-art report.

While mesh filtering [Tau95, DMSB99, ZF03] can be efficiently carried out in the spatial domain via convolution, methods which require explicit eigenvector computation, e.g., geometry compression [KG00] or mesh watermarking [OTMM01], had suffered from the high computational cost. One remedy proposed was to partition the mesh into smaller patches and perform spectral processing on a per patch basis [KG00]. Another approach is to convert each patch into one having regular connectivity so that the classical Fourier transform, which admits fast computations, may be performed [KG01]. Similarly, one may also choose to perform regular resampling geometrically over each patch and conduct Fourier analysis [PG01]. However, artifacts emerging at the artificially introduced patch boundaries may occur and it would still be desirable to perform global spectral analysis over the whole mesh surface seamlessly. Recently, efficient schemes for eigenvector computation, e.g., with the use of multi-grid methods [KCH02], spectral shift [DBG*06, VL08], and eigenvector approximation via the Nyström method [FBCM04], have fueled renewed interests in spectral mesh processing.

2.3. Works in computer vision and machine learning

At the same time, developments in fields such as computer vision and machine learning on spectral techniques have started to exert more influence on the computer graphics community. These inspiring developments include spectral graph matching and point correspondence from computer vision, dating back to the works of Umeyama [Ume88] and Shapiro and Brady [SB92] in the late 1980's and early 1990's. Extremal properties of the eigenvectors known from linear algebra provided the theoretical background. These techniques have been extended to the correspondence between 3D meshes, e.g., [JZvK07].

The method of spectral clustering [vL06] from machine learning, along with its variants, has received increased attention in the geometry processing community, e.g., for problems such as mesh segmentation [LZ04, LZ07] and surface reconstruction from point clouds [KSO04]. Central to the idea of spectral clustering is a transformation of input data from its original domain to a spectral domain, resulting in an embedding that is constructed using a set of eigenvectors of an appropriately defined linear operator; see Figure 2. Such an idea also underlines the closely related concepts of isomaps [TL00], locally linear embedding [RS00], Laplacian eigenmaps [BN03], and kernel principal component analysis (kernel PCA) [SSrM98]. It turns out that the kernel view can be used to unify these concepts [HLMS04], all as means for dimensionality reduction on manifolds.

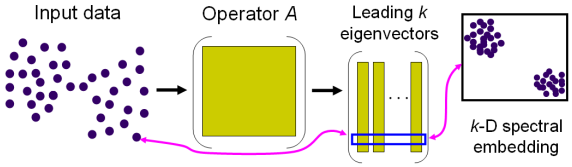


Figure 2: *Construction of a spectral embedding. The operator A is defined by a Gaussian of the pairwise Euclidean distances between the input points.*

Efforts on unifying related concepts using the spectral approach continue in the machine learning community, e.g., using learning eigenfunctions to link spectral clustering with kernel PCA [BDLR*04]. Also important are works which focus on explaining the success of spectral methods [ST96, NJW02] as well as discovering their limitations [vLBB05]. While at the same time, the geometry processing community has fulfilled the promise of the spectral approach, in particular the use of spectral embeddings, in a variety of applications including planar [ZKK02, MTAD08] and spherical [Got03] mesh parameterization, shape correspondence [JZvK07] and retrieval [EK03], quadrilateral remeshing [DBG*06], global intrinsic symmetry detection [OSG08], and mesh segmentation [LZ07, dGGV08].

3. Overview of the spectral approach

Most spectral methods have a basic framework in common, which can be roughly divided into three steps; see Figure 1 for an illustration. Note that throughout the paper, we only require the input mesh to be a 2-manifold embedded in 3D; the mesh can possibly have boundaries.

1. A matrix M which represents a discrete linear operator based on the structure of the input mesh is constructed, typically as a discretization of some continuous operator. This matrix can be seen as incorporating pairwise relations between mesh elements. That is, each entry M_{ij} possesses a value that represents the relation between the vertices (faces or other primitives) i and j of the mesh. The pairwise relations, sometimes called affinities, can take into account only the mesh connectivity or combine topological and geometric information.

2. An eigendecomposition of the matrix M is performed, that is, its eigenvalues and eigenvectors are computed.
3. Resulting structures from the decomposition are employed in a problem-specific manner to obtain a solution. In the example shown in Figure 1, the application is mesh segmentation. We see that the spectral transform from 3D mesh data to 2D contour data, while still preserving certain salient geometric features (the three tips of the 2D contour correspond to the two ears and the tail of the bunny), simplifies the processing task [LZ07].

The above framework leads to a few possible classifications of spectral methods.

- **Based on the operator used:**

  Depending on whether the matrix M should be defined by the geometry of the input mesh or only its connectivity, one can classify linear mesh operators used for spectral analysis as either *combinatorial* or *geometric*.

  It is also possible to distinguish between matrices which encode graph *adjacency* and matrices which approximate the *Laplacian* operator [Bol98, Chu97, Moh97]. In graph-theoretic terminology, the adjacency matrix is sometimes said to model the *Lagrangian* of a graph [Bol98]. Note here that for a given graph G and a scalar function $\mathbf{v}$ defined on the vertices of G, the Lagrangian $f_G(\mathbf{v}) = \langle A\mathbf{v}, \mathbf{v} \rangle$, where $\langle , \rangle$ is the conventional dot product and A is the adjacency matrix. One possible extension of the graph Laplacian operator is to the class of *discrete Schrödinger operators*, e.g., see [BHL*04, DGLS01]. The precise definition of these and other operators mentioned in this section will be given in Sections 6 and 7.

  Both the graph adjacency and the Laplacian matrices can also be extended to incorporate higher-order neighborhood information. That is, relationships between all pairs of mesh elements are modeled instead of only considering element pairs that are adjacent in a mesh graph. A particularly important class of such operators are the so-called *Gram matrices*, e.g., see [STWCK05]. These matrices play a crucial role in several techniques from machine learning, including spectral clustering [vL06] and kernel-based methods [SS02], e.g., kernel PCA [SSM98].

- **Based on the eigenstructures used:**

  In graph theory, the focus has been placed on the eigenvalues of graph adjacency or Laplacian matrices. Many results are known which relate these eigenvalues to graph-theoretical properties [Chu97]. While from a theoretical point of view, it is of interest to obtain various bounds on the graph invariants from the eigenvalues. Several practical applications simply rely on the eigenvalues of appropriately defined graphs to characterize geometric shapes, e.g., [JZ07, RWP06, SMD*05, SSGD03].

  Indeed, eigenvalues and eigenspace projections are primarily used to derive shape descriptors (or signatures) for shape matching and retrieval, where the latter, obtained by projecting a mesh representation along the appropriate eigenvectors, mimics the behavior of Fourier descriptors [ZR72] in the classical setting.

  Eigenvectors, on the other hand, are most frequently used to derive a *spectral embedding* of the input data, e.g., a mesh shape. Often, the new (spectral) domain is more convenient to operate on, e.g., it is low-dimensional, while the transform still retains as much information about the input data as possible. This issue, along with the use of eigenvalues and Fourier descriptors for shape characterization, will be discussed further in Sections 8 and 10.

- **Based on the dimensionality of the eigenstructure:**

  Such a classification is the most relevant to the use of eigenvectors for constructing spectral embeddings. One-dimensional embeddings typically serve as solutions to ordering or sequencing problems, where some specific optimization criterion is to be met. In many instances, the optimization problem is NP-hard and the use of an eigenvector provides a good heuristic [DPS02, MP93]. Of particular importance is the Fiedler vector [Fie73]. For example, it has been used by the well-known normalized cut algorithm for image segmentation [SM00].

  Two-dimensional spectral embeddings have been used for graph drawing [KCH02] and mesh flattening [ZSGS04, ZKK02], and three-dimensional embeddings have been applied to spherical mesh parameterization [Got03]. Generally speaking, low-dimensional embeddings can be utilized to facilitate solutions to several geometric processing problems, including mesh segmentation [LZ04, ZL05, LZ07] and correspondence [JZvK07]. These works are inspired by the use of the spectral approach for clustering [vL06] and graph matching [SB92, Ume88].

4. Motivation

In this section, we motivate the use of the spectral approach for mesh processing and analysis from several perspectives. These discussions naturally reveal which classes of problems are suitable for the spectral approach. Several examples are presented to better illustrate the ideas.

4.1. "Harmonic" behavior of Laplacian eigenvectors

One of the main reasons that combinatorial and geometric Laplacians are often considered for spectral mesh processing is that their eigenvectors possess similar properties as the classical Fourier basis functions. By representing mesh geometry using a discrete signal defined over the manifold mesh surface, it is commonly believed that a "Fourier transform" of such a signal can be obtained by an eigenspace projection of the signal along the eigenvectors of a mesh Laplacian. This stipulation was first applied by Taubin [Tau95] to develop a signal processing framework for mesh fairing.

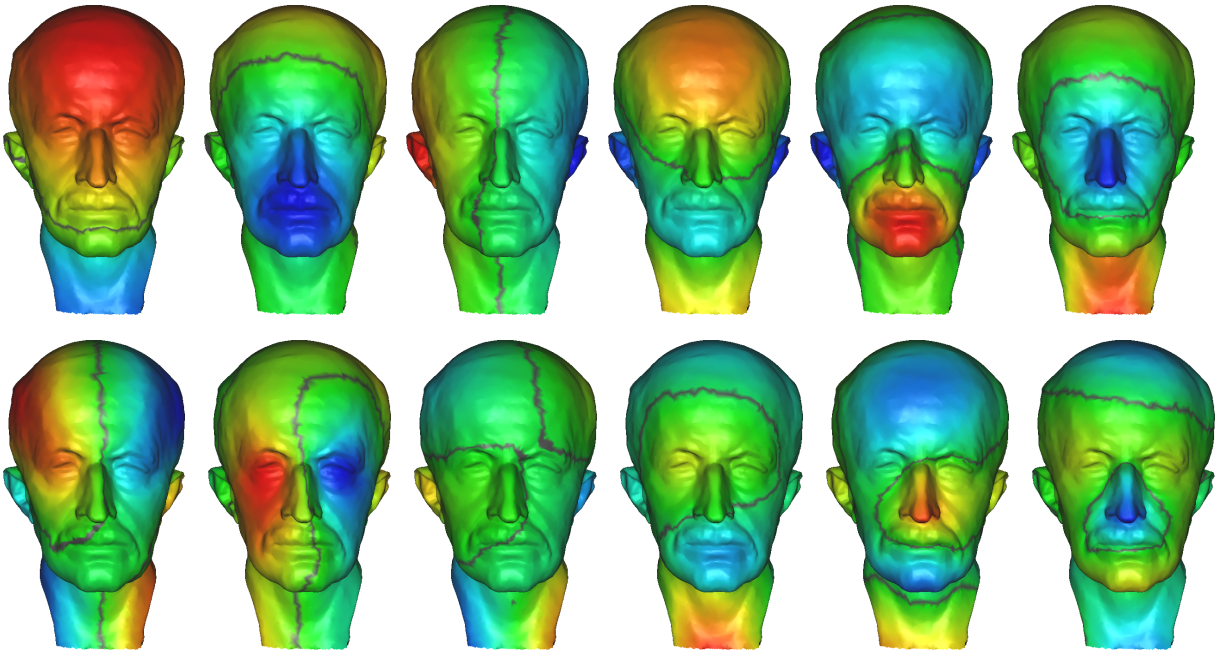


Figure 3: *Color plots of the first 12 eigenvectors of the graph Laplacian for the Max Planck mesh. Vertices whose corresponding eigenvector entry is zero, which are part of a nodal set, are shown in gray (nodal sets are discussed in Section 6.6.2).*

Indeed, the classical Fourier transform of a periodic 1D signal can be seen as the decomposition of the signal into a linear combination of the eigenvectors of the Laplacian operator. It is worth noting here that this statement still holds if we replace the Laplacian operator by any *circulant matrix* [Jai89]. A combinatorial mesh Laplacian is then adopted to conduct Fourier analysis on a mesh signal.

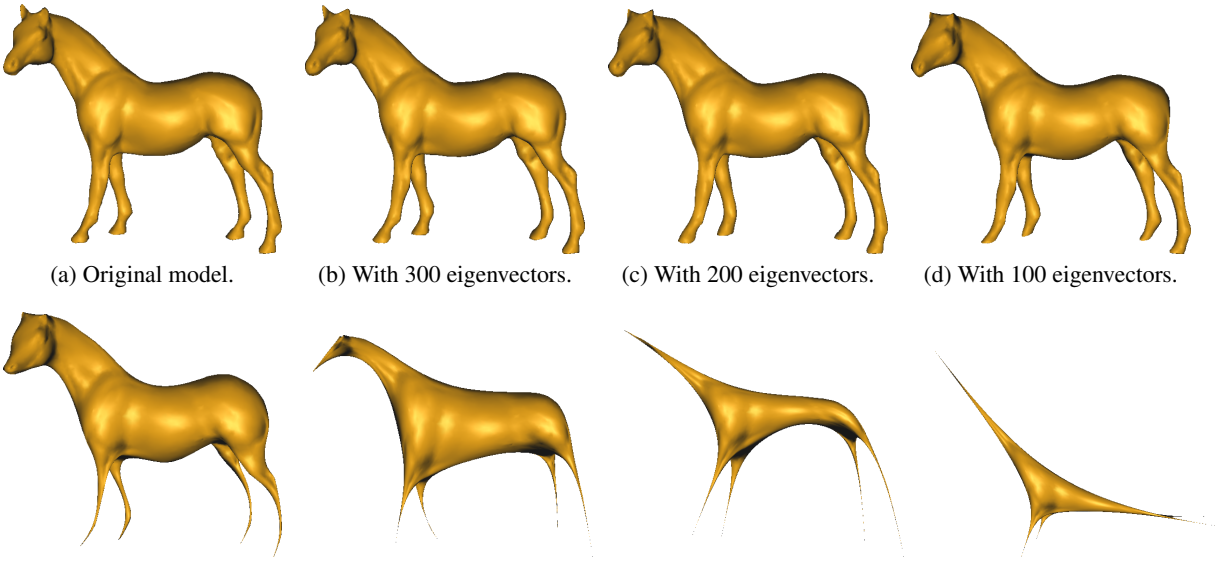
An important *distinction* between the mesh case and the classical Fourier transform however is that while the latter uses a fixed set of basis functions, the eigenvectors which serve as "Fourier-like" bases for mesh signal processing would change depending on mesh connectivity, geometry, and which type of Laplacian operator is adopted. Nevertheless, the eigenvectors of the mesh Laplacians all appear to exhibit "*harmonic behavior*", loosely referring to their oscillatory nature. They are seen as the *vibration modes* or the *harmonics* of the mesh surface with their corresponding eigenvalues as the associated *frequencies* [Tau95]. Here we recall that in the classical setting, harmonic functions are solutions to the Laplace equation with Dirichlet boundary conditions. With the above analogy, mesh fairing can then be carried out via low-pass filtering. This approach and subsequent developments have been described in detail in [Tau00].

In Figure 3, we give color plots of the first 12 eigenvectors of the combinatorial graph Laplacian of the Max Planck mesh, where the entries of an eigenvector are color-mapped. As we can see, the harmonic behavior of the eigenvectors is evident. Although the filtering approach proposed by Taubin does not fall strictly into the category of spectral methods since neither the eigenvalues nor the eigenvectors of the mesh Laplacian are explicitly computed, the resemblance to classical Fourier analysis implies that any application which utilizes the Fourier transform can be applied in the mesh setting, e.g., JPEG-like geometry compression [KG00]. In Figure 4, we show a horse model (with 7,502 vertices and 15,000 faces) reconstructed using a few spectral coefficients derived from the graph Laplacian.

4.2. Modeling of global characteristics

Although each entry in a linear mesh operator may encode only local information, it is widely held that the eigenvalues and eigenvectors of the operator can reveal meaningful global information about the mesh shape. This is hardly surprising from the perspective of spectral graph theory, where many results are known which relate extremal properties of a graph, e.g., its diameter and Cheeger constant, with the eigenvalues of the graph Laplacian.

As Chung stated in her book [Chu97], results from spectral theory suggest that the Laplacian eigenvalues are closely related to almost all major graph invariants. Thus if a matrix models the structures of a shape, either in terms of topology or geometry, then we would expect its set of eigenvalues to provide an adequate characterization of the shape. Indeed, this has motivated the use of graph spectra for shape match-



(a) Original model. (b) With 300 eigenvectors. (c) With 200 eigenvectors. (d) With 100 eigenvectors.

(e) With 50 eigenvectors. (f) With 10 eigenvectors. (g) With 5 eigenvectors. (h) With 3 eigenvectors.

Figure 4: *The horse model shown in (a) is reconstructed in (b)-(h) using the indicated number of eigenvectors of the graph Laplacian. The original model has 7,502 vertices and 15,000 faces.*

ing and retrieval in computer vision [SMD*05, SSGD03] and geometry processing [JZ07, RWP06]. The eigenvalues serve as compact global shape descriptors. They are sorted by their magnitudes so as to establish a correspondence for computing the similarity distance between two shapes. In this context, it is not necessary to consider what particular characteristic an individual eigenvalue reveals.

Compared with eigenvalues, eigenvectors provide a more refined shape characterization which also tends to have a global nature. For instance, it can be shown [SFYD04] that pairwise distances between points given by the spectral embeddings derived from the graph Laplacian model the so-called commute-time distances [Lov93], a global measure related to the behavior of random walks on a graph. Eigenvectors also possess extremal properties, highlighted by the Courant-Fischer theorem (given in Section 5), which enable spectral techniques to provide high-quality results for several NP-hard global optimization problems, including normalized cuts [SM97] and the linear arrangement problem [DPS02], among others [MP93].

4.3. Structure revelation

Depending on the requirement of the problem at hand, the operator we use to compute the spectral embeddings can be made to incorporate any intrinsic measure on a shape in order to obtain useful invariance properties, e.g., with respect to part articulation. In Figure 5, we show 3D spectral embeddings of a few human and hand models obtained from an operator derived from geodesic distances over the mesh

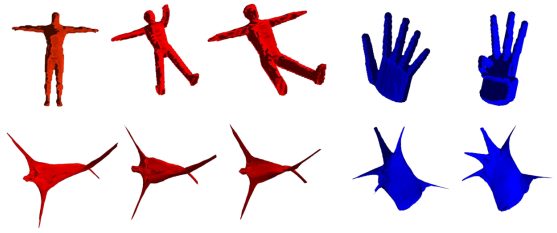


Figure 5: *Spectral embeddings (bottom row) of some articulated 3D shapes (top row) from the McGill 3D shape benchmark database [McG]. Since the mesh operator is constructed from geodesic distances, the embeddings are normalized with respect to shape bending.*

surfaces. As geodesic distance is bending-tolerant, the resulting embeddings are normalized with respect to bending and can facilitate shape retrieval under part articulation [EK03, JZ07]. Recently, such embeddings have been exploited to detect global intrinsic symmetries in a shape [OSG08]. To better handle moderate stretchings in a shape, Liu et al. [LZSCO09] propose to augment geodesic distances and normal variations by a volume-based part-aware surface distance to derive spectral embeddings for shape analysis.

Generally speaking, with an appropriately chosen linear operator, the resulting spectral embedding can better reveal, single out, or even exaggerate useful underlying structures in the input data. The above example shows that via a transformation into the spectral domain, certain intrinsic shape

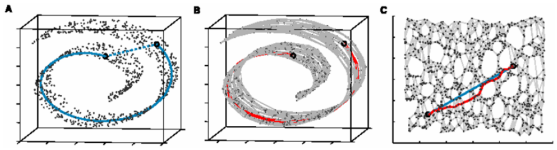


Figure 6: *Unfolding of a "Swiss roll" dataset which preserves distances measured over the surface of the roll. These images are taken from [TL00].*

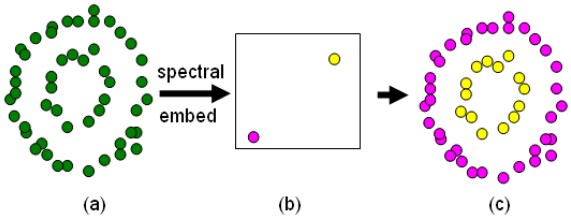


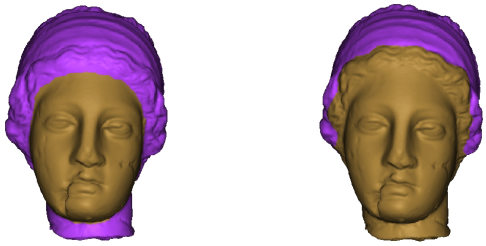
Figure 7: *Result of spectral clustering, shown in (c), on the 2-ring data set (a). (b) shows the 2D spectral embedding.*

structures, e.g., part composition of the shape, are better revealed by removing the effects of other features, e.g., those resulting from shape bending. In other instances, the spectral approach can present the nonlinear structures in an input data in a high-dimensional feature space so that they become much easier to handle. In particular, the nonlinear structures may be "unfolded" into linear ones so that methods based on linear transformations and linear separators, e.g., PCA and k-means clustering, can be applied. An illustration of such an "unfolding" is given in Figure 6. This concept is often referred to as the "kernel trick" in the machine learning literature [HLMS04], whereby linear classifiers can be used to solve non-linear problems.

One classical example to illustrate the "kernel trick" at work is the clustering of the 2-ring data set shown in Figure 7(a). Although to a human observer, the data should clearly be clustered into an outer and an inner ring via a circular separator, conventional clustering methods such as k-means or support vectors would fail. However, by constructing an operator using a Gaussian kernel (applying a Gaussian to the pairwise Euclidean distances between the input points) and then spectrally embedding the data into the 2D domain, we arrive at the set shown in Figure 7(b). This set is trivial to cluster via k-means to obtain the desired result in (c), as there is a clear linear separator. This is an instance of the spectral clustering method [vL06].

4.4. Dimensionality reduction

Typically, the dimensionality of the linear operator used for spectral mesh analysis is equal to the size of the input mesh, which can become quite large. By properly selecting a small number of leading eigenvectors of the operator to construct



(a) Optimal cut. (b) Result of line search.

Figure 8: *First cut on the Igea model (example taken from [LZ07]). (a) The best cut present in the mesh face sequence. (b) Result from line search based on part salience.*

an embedding, the dimensionality of the problem is effectively reduced while the global characteristics of the original data set are still retained. In fact, the extremal properties of the eigenvectors ensure that the spectral embeddings are "information-preserving"; this is suggested by a theorem due to Eckart and Young [EY36], which we give in Section 5 as Theorem 5.5. Furthermore, there is evidence that the cluster structures in the input data may be enhanced in a low-dimensional embedding space, as hinted by the Polarization Theorem [BH03] (Theorem 5.6 in Section 5).

Some of the advantages of dimensionality reduction include computational efficiency and problem simplification. One such example is image segmentation using normalized cuts [SM00]. In a recursive setting, each iteration of the segmentation algorithm corresponds to a line search along a 1D embedding obtained by the Fiedler vector of a weighted graph Laplacian. The same idea has been applied to mesh segmentation [ZL05, LZ07] where the simplicity of the line search allows the incorporation of any efficiently computable (but not necessarily easy to optimize) search criteria, e.g., part salience [HS97]. In Figure 8(a), we show the best cut present in the mesh face sequence obtained using the 1D spectral embedding technique given by Liu and Zhang [LZ07]. This example reflects the ability of spectral embeddings to reveal, in only one dimension, meaningful global shape characteristics for a model that is difficult to segment. However, line search based on part salience does not always return the best result, as shown in (b). This is due to the inability of the part salience measure to capture the most meaningful cut.

5. Theoretical background

In this section, we list a few theorems from linear algebra related to eigenstructures of general matrices, as well as a few useful results concerning spectral embeddings. These include the Spectral Theorem, the Courant-Fischer Theorem, the Ky-Fan theorem [Bha97], and the Polarization Theorem [BH03]. These theorems are stated here without proofs. Proofs of some of these theorems can be found in the associ-

ated references. The Spectral and Courant-Fischer theorems are well known in linear algebra, whose proofs can be found in many standard linear algebra texts, e.g., [Mey00, TB97].

Let M be an $n \times n$ diagonalizable matrix with eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$ and associated eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$. By definition,

$$M\mathbf{v}_i = \lambda_i \mathbf{v}_i \text{ and } \mathbf{v}_i \neq 0, \text{ for } i \in \{1, \ldots, n\}.$$

The set of eigenvalues $\lambda(M) = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ is known as the *spectrum* of the matrix.

When the matrix M is generalized to a linear operator acting on a Hilbert space, the eigenvectors become *eigenfunctions* of the operator. For our purpose, we will focus on real symmetric matrices, whose counterpart in functional analysis are compact self-adjoint operators. The main advantage offered by symmetric matrices is that they possess real eigenvalues whose eigenvectors form an orthogonal basis, that is, $\mathbf{v}_i^\mathsf{T}\mathbf{v}_j = 0$ for $i \neq j$. The eigendecomposition of a real symmetric matrix is described by the Spectral Theorem:

Theorem 5.1 (The Spectral Theorem) Let S be a real symmetric matrix of dimension n. Then we have

$$S = V\Lambda V^\mathsf{T} = \sum_{i=1}^{n} \lambda_i \mathbf{v}_i \mathbf{v}_i^\mathsf{T},$$

the eigendecomposition of S, where $V = [\mathbf{v}_1\ \mathbf{v}_2\ \ldots\ \mathbf{v}_n]$ is the matrix of eigenvectors of S and Λ is the diagonal matrix of the eigenvalues of S. The eigenvalues of S are real and its eigenvectors are orthogonal, i.e., $V^\mathsf{T}V = I$, where M^T denotes the transpose of a matrix M and I is the identity matrix.

One of the most fundamental theorems which characterize eigenvalues and eigenvectors of a symmetric matrix is the Courant-Fischer theorem. It reveals certain extremal property of eigenvectors, which has frequently motivated the use of eigenvectors and the embeddings they define for solving a variety of optimization problems.

Theorem 5.2 (Courant-Fischer) Let S be a real symmetric matrix of dimension n. Then its eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$ satisfy the following,

$$\lambda_i = \min_{\substack{\mathcal{V} \subset \mathbb{R}^n \\ \dim \mathcal{V} = i}} \max_{\substack{\mathbf{v} \in \mathcal{V} \\ \|\mathbf{v}\|_2 = 1}} \mathbf{v}^\mathsf{T} S \mathbf{v}$$

where $\mathcal{V}$ is a subspace of $\mathbb{R}^n$ with the given dimension. When considering only the smallest eigenvalue of S, we have

$$\lambda_1 = \min_{\|\mathbf{v}\|_2 = 1} \mathbf{v}^\mathsf{T} S \mathbf{v}.$$

Similarly, the largest eigenvalue

$$\lambda_n = \max_{\|\mathbf{v}\|_2 = 1} \mathbf{v}^\mathsf{T} S \mathbf{v}.$$

The unit length constraint can be removed if the quadratic form $\mathbf{v}^\mathsf{T} S \mathbf{v}$ in Theorem 5.2 is replaced with the well-known *Rayleigh quotient* $\mathbf{v}^\mathsf{T} S \mathbf{v} / \mathbf{v}^\mathsf{T}\mathbf{v}$. Another way of characterizing the eigenstructures is the following result, which can be seen as a corollary of the Courant-Fischer Theorem.

Theorem 5.3 Let S be a real symmetric matrix of dimension n. Then its eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$ satisfy the following,

$$\lambda_i = \min_{\substack{\|\mathbf{v}\|_2 = 1 \\ \mathbf{v}^\mathsf{T}\mathbf{v}_k = 0,\, k < i}} \mathbf{v}^\mathsf{T} S \mathbf{v}$$

where $\mathbf{v}_1, \ldots, \mathbf{v}_{i-1}$ are the eigenvectors of S corresponding to eigenvalues $\lambda_1, \ldots, \lambda_{i-1}$, respectively.

Another useful theorem which relates the sum of the partial spectrum of a symmetric matrix to the respective eigenvectors is also known.

Theorem 5.4 (Ky-Fan) Let S be a real symmetric matrix with eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$. Then

$$\sum_{i=1}^{k} \lambda_i = \min_{\substack{U \in \mathbb{R}^{n \times k} \\ U^\mathsf{T}U = I_k}} \text{tr}\,(U^\mathsf{T} S U),$$

where $\text{tr}(M)$ denotes the trace of a matrix M and I_k is the $k \times k$ identity matrix.

One may also interpret Theorem 5.4 as saying that a set of k orthogonal vectors which minimizes the matrix trace in the theorem is given by the k eigenvectors corresponding to the k smallest eigenvalues. Clearly, if U consists of the k eigenvectors of S corresponding to eigenvalues $\lambda_1, \ldots, \lambda_k$, then we have $\text{tr}\,(U^\mathsf{T} S U) = \sum_{i=1}^{k} \lambda_i$.

Taking an alternative view, we will see that the set of leading eigenvectors of a symmetric matrix plays a role in low-rank approximation of matrices, as given by a theorem due to Eckart and Young [EY36]. This result is useful in studying the properties of spectral embeddings, e.g., [BH03, dST04].

Theorem 5.5 (Eckart-Young) Let S be a real, symmetric and positive semi-definite matrix of dimension n and let $S = V\Lambda V^\mathsf{T}$ be the eigendecomposition of S. Suppose that the eigenvalues, given along the diagonal of Λ, are in descending order. Let $X = V\Lambda^{1/2}$ be the matrix of eigenvectors that are scaled by the square root of their respective eigenvalues. Denote by $X_{(k)} \in \mathbb{R}^{n \times k}$ a truncated version of X, i.e., its columns consist of the k leading columns of X. Then

$$X_{(k)} = \text{argmin}_{\substack{U \in \mathbb{R}^{n \times k} \\ \text{rank}(U) = k}} \|S - UU^\mathsf{T}\|_F,$$

where $\text{rank}(M)$ denotes the rank of a matrix M and $\|\cdot\|_F$ is the Frobenius norm.

Theorem 5.5 states that the outer product of the k largest eigenvectors of S (eigenvectors corresponding to the largest eigenvalues), when scaled using the square root of their respective eigenvalues, provides the *best rank-k approximation* of S. As a related result, we mention an interesting theorem which suggests that the clustering structures in a data set are somewhat *exaggerated* as the dimensionality of the spectral embedding decreases. This is the Polarization theorem due to Brand and Huang [BH03].

Theorem 5.6 (Polarization Theorem) Denote by $S_{(k)} = X_{(k)}X_{(k)}^{\mathsf{T}}$ the best rank-k approximation of S with respect to the Frobenius norm, where $X_{(k)}$ is as defined in Theorem 5.5. As S is projected to successively lower ranks $S_{(n-1)}, S_{(n-2)}, \ldots, S_{(2)}, S_{(1)}$, the sum of squared angle-cosines,

$$s_k = \sum_{i \neq j} (\cos \theta_{ij}^{(k)})^2 = \sum_{i \neq j} \left(\frac{\mathbf{x}_i^{(k)\mathsf{T}} \mathbf{x}_j^{(k)}}{\|\mathbf{x}_i^{(k)}\|_2 \cdot \|\mathbf{x}_j^{(k)}\|_2} \right)^2$$

is strictly increasing, where $\mathbf{x}_i^{(k)}$ is the i-th row of $X_{(k)}$.

This theorem states that as the dimensionality of the representation is reduced, the distribution of the cosines migrates away from 0 towards two poles $+1$ or -1, such that the angles migrate from $\theta_{ij} = \pi/2$ to $\theta_{ij} \in \{0, \pi\}$.

6. Mesh Laplacian operators

The mesh Laplacians are the most commonly used operators for spectral mesh processing. In this section we discuss these operators. In Section 7 we examine other operators that have been adopted for spectral analysis, most of which can be viewed as extensions of the Laplacian operators described here. These operators and their applications appear mostly in the fields of computer vision and machine learning.

We group mesh Laplacian operators into two categories. On the one hand are operators that have been extensively studied in graph theory [Chu97]. These operators are determined by the connectivity of the graph that is the 1-skeleton of the mesh and they do not explicitly encode geometric information. We refer to such operators as *combinatorial mesh Laplacians*. Although these operators are based solely upon topological information, their eigenfunctions generally exhibit a remarkable conformity to the mesh geometry. This is a manifestation of the fact that meshes are usually constructed in such a way that the connectivity implicitly encodes geometric information [IGG01]. Nonetheless, the eigenfunctions of these operators are inherently sensitive to changes in mesh connectivity.

The other category of mesh Laplacians represents discretizations of the Laplace-Beltrami operator from Riemannian geometry [Ros97, Cha84]. Since these operators do explicitly encode geometric information we refer to them as *geometric mesh Laplacians*. While the combinatorial Laplacians are meaningfully defined on general meshes, the geometric mesh Laplacians require a manifold triangle mesh. The eigenfunctions of these operators exhibit robustness with respect to changes in mesh connectivity [DZM07].

Despite their distinct heritage, both categories of mesh Laplacians can be encompassed in a single mathematical definition. We present this definition and several fundamental properties in Section 6.2. Subsequent subsections develop each of the two categories and their properties.

6.1. Notation

A triangle mesh with n vertices is represented as $\mathscr{M} = (\mathscr{G}, P)$, where $\mathscr{G} = (V, E)$ models the mesh graph, with V denoting the set of mesh vertices and $E \subseteq V \times V$ the set of edges. $P \in \mathbb{R}^{n \times 3}$ represents the geometry of the mesh, given by an array of 3D vertex coordinates. Each vertex $i \in V$ has an associated position vector, denoted by $\mathbf{p}_i = [x_i\ y_i\ z_i]$; it corresponds to the i-th row of P. The set of 1-ring neighbours of i is $N(i) = \{j \in V \mid (i, j) \in E\}$.

Matrices are denoted by upper-case letters (e.g., M), vectors by lower-case bold (e.g., $\mathbf{v}$), and scalars or functions by lower-case roman (e.g., s). The i-th element of a vector $\mathbf{v}$ is denoted by v_i, and the (i, j)-th element of a matrix M by M_{ij}.

6.2. Mesh Laplacians: overview and properties

Mesh Laplacian operators are linear operators that act on functions defined on a mesh. These functions are specified by their values at the vertices. Thus if a mesh $\mathscr{M}$ has n vertices, then functions on $\mathscr{M}$ will be represented by vectors with n components and a mesh Laplacian will be described by an $n \times n$ matrix.

Loosely speaking, a mesh Laplacian operator locally takes the difference between the value of a function at a vertex and a weighted average of its values at the first-order or immediate neighbour vertices. Although we will discuss generalizations, for introductory purposes a Laplacian, L, will have a local form given by

$$(L\mathbf{f})_i = b_i^{-1} \sum_{j \in N(i)} w_{ij} \left(f_i - f_j \right). \tag{1}$$

The edge weights, w_{ij}, are symmetric: $w_{ij} = w_{ji}$. The factor b_i^{-1} is a positive number. Its expression as an inverse will appear natural in subsequent developments.

A Laplacian satisfying equation (1) is called a *first order Laplacian* because its definition at a given vertex involves only the one-ring neighbours. On a manifold triangle mesh, the matrix of such an operator will be sparse, with an average of seven nonzero entries per row.

6.2.1. Zero row sum

An important property imposed by equation (1) is the *zero row sum*. If $\mathbf{f}$ is a constant vector, i.e., one all of whose components are the same, then $\mathbf{f}$ lies in the kernel of L, since $L\mathbf{f} = \mathbf{0}$ for an operator L with zero row sum. This implies that the constant vectors are eigenvectors of L with eigenvalue zero, and allows the identification of a *DC component* of the spectral projection to be discussed in Section 8.3. It is known [MP93, Moh97] that the multiplicity of the zero eigenvalue equals the number of connected components in the graph.

6.2.2. Eigenvector orthogonality

An operator that is locally expressed by (1) can be factored into the product of a diagonal and a symmetric matrix

$$L = B^{-1}S, \tag{2}$$

where B^{-1} is a diagonal matrix whose diagonal entries are the b_i^{-1}'s and S is a symmetric matrix whose diagonal entries are given by $S_{ii} = \sum_{j \in N(i)} w_{ij}$ and whose off diagonal entries are $-w_{ij}$. Although L itself is not symmetric in general, it is similar to the symmetric matrix $O = B^{-1/2}SB^{-1/2}$ since

$$L = B^{-1}S = B^{-1/2}B^{-1/2}SB^{-1/2}B^{1/2} = B^{-1/2}OB^{1/2}.$$

Thus L and O have the same real eigenvalues. And if $\mathbf{v}$ is an eigenvector of O with eigenvalue λ, then $\mathbf{u} = B^{-1/2}\mathbf{v}$ is an eigenvector of L with the same eigenvalue. As mentioned in Section 9.1, these observations can be exploited to facilitate the computation of eigenvectors of L.

The eigenvectors of O are mutually orthogonal, since O is symmetric. This is not generally true for L. However, if we define a scalar product by

$$\langle \mathbf{f}, \mathbf{g} \rangle_B = \mathbf{f}^\mathsf{T} B \mathbf{g}, \tag{3}$$

then the eigenvectors of L are orthogonal with respect to that product:

$$\langle \mathbf{u}_i, \mathbf{u}_j \rangle_B = \mathbf{u}_i^\mathsf{T} B \mathbf{u}_j = \mathbf{v}_i^\mathsf{T} \mathbf{v}_j = \delta_{ij}.$$

Thus although a Laplacian satisfying equation (1) is not symmetric in general, for most applications the properties, such as orthogonality of the eigenvectors, that motivate the desire for a symmetric matrix can be recovered by using the appropriate scalar product.

The above arguments apply whenever B^{-1} is symmetric positive definite, not just diagonal, since in this case $B^{-1/2}$ is well defined. Thus the comments are quite general, since *any* matrix L, symmetric or not, which has real eigenvalues and a complete set of eigenvectors can be written in the form (2): If $L = X\Lambda X^{-1}$ is the eigendecomposition of L, then we have the positive definite $B^{-1} = XX^\mathsf{T}$ and $S = (XX^\mathsf{T})^{-1}L = (XX^\mathsf{T})^{-1}X\Lambda X^{-1} = (X^{-1})^\mathsf{T}\Lambda X^{-1}$ is symmetric.

The inner product (3), renders a matrix of the form (2) self-adjoint. If this inner product is employed, then theorems which demand a symmetric matrix can be applied. For example, the Courant-Fisher theorem becomes

$$\lambda_i = \min_{\substack{\mathcal{V} \subset \mathbb{R}^n \\ \dim \mathcal{V} = i}} \max_{\substack{\mathbf{v} \in \mathcal{V} \\ \|\mathbf{v}\|_B = 1}} \langle \mathbf{v}, L\mathbf{v} \rangle_B,$$

where $\|\mathbf{v}\|_B = \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle_B}$. To see this, transform L into the basis of its eigenvectors. The Courant-Fisher theorem applies to the resulting diagonal matrix, Λ. Replacing Λ with $X^{-1}LX$ and $\mathbf{v}$ with $X^{-1}\mathbf{v}$ yields the above expression.

6.2.3. Positive semi-definiteness

Equation (1) does not guarantee that L is positive semidefinite, but such a property is desirable in a Laplacian operator: the zero eigenvalue associated with the constant (zero frequency) eigenvectors should be the smallest one.

Suppose that the weights w_{ij}'s are non-negative. Then L is positive semi-definite with respect to the appropriate inner product (3). Indeed, it is straightforward to show that

$$\langle \mathbf{f}, L\mathbf{f} \rangle_B = \mathbf{f}^\mathsf{T} S \mathbf{f} = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2 \geq 0. \tag{4}$$

However, some important Laplacians, specifically the cotangent operator in Section 6.5, may have negative weights in S, yet they can still be shown to be positive semi-definite.

6.2.4. Mesh Laplacians: no free lunch

We have highlighted here the characteristic properties of mesh Laplacians that are important for most spectral processing applications. Mesh Laplacians are ubiquitous in geometry processing, not just when spectral methods are employed. Depending on the application, different properties may be deemed fundamental. In an interesting recent work, Wardetzky et al. [WMKG07] listed several properties, including the ones above, which may be naturally expected of a mesh Laplacian. They then went on to demonstrate that a certain four of these properties cannot be simultaneously satisfied by any one operator on all triangle meshes.

6.3. Combinatorial mesh Laplacians

On a mesh $\mathscr{M} = (V, E, P)$, a combinatorial mesh Laplacian is completely defined by the graph associated with the mesh; the geometry component, P, plays no role.

6.3.1. Graph Laplacian

The adjacency matrix W of $\mathscr{M}$ is given by

$$W_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

The degree matrix D is defined as

$$D_{ij} = \begin{cases} d_i = |N(i)| & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

d_i is said to be the *degree* of vertex i. W and D are $n \times n$ matrices, where $n = |V|$.

We define the *graph Laplacian* matrix K as

$$K = D - W.$$

Referring to equation (1), K corresponds to setting $b_i = 1$ and $w_{ij} = W_{ij}$ for all i, j. The operator K is also known as the *Kirchoff operator* [OTMM01], as it has been encountered in the study of electrical networks by Kirchoff. In that context, the (weighted) adjacency matrix W is referred as the *conductance matrix* [GM00].

6.3.2. Tutte Laplacian

Another operator that has been applied as a combinatorial mesh Laplacian was used by Taubin [Tau95] in his signal processing approach to mesh fairing. It has also been used in the context of planar graph drawing [Kor03], first studied by Tutte [Tut63]. Following the terminology used by Gotsman et al. [Got03], we call this operator the *Tutte Laplacian*. It is defined as

$$T = D^{-1}K,$$

thus

$$T_{ij} = \begin{cases} 1 & \text{if } i = j, \\ -1/d_i & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

In other words we take $b_i^{-1} = d_i^{-1}$ in equation (1).

Of course, T, while possessing useful properties, is not a symmetric matrix. This fact has been responsible for the creation of several associated combinatorial operators.

6.3.3. Normalized graph Laplacian

In the literature, there is no consensus as to what should be called a graph Laplacian. In Chung's book [Chu97], for example, the following symmetrized version of T,

$$Q = D^{-1/2}KD^{-1/2},$$

with

$$Q_{ij} = \begin{cases} 1 & \text{if } i = j, \\ -1/\sqrt{d_i d_j} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise,} \end{cases}$$

is called a graph Laplacian. In this paper, we call Q the *normalized graph Laplacian*. Since Q is similar to $T = D^{-1/2}QD^{1/2}$, it has the same spectrum. However, it is not a Laplacian as defined by equation (1). In particular it does not have a zero row sum. For spectral processing, the main utility of Q is to provide a symmetric matrix to facilitate computation of the eigenvectors of T, as described in Section 9.1.

6.3.4. Other symmetrized graph Laplacians

Another way to obtain a symmetric version from the initial non-symmetric Laplacian T is by applying the simple transformation $T' = \frac{1}{2}(T + T^\intercal)$, as suggested by Lévy [Lév06]. Zhang [Zha04] proposes a new symmetric operator which approximates the Tutte Laplacian. A common drawback of these suggestions is that their geometric significance is unclear. More recent works tend to prefer to treat the symmetry issue by exploiting an appropriate inner product, as described in Section 6.2.2; see [VL08] for example.

The Tutte Laplacian T can also be "symmetrized" into a second-order operator $T'' = T^\intercal T$, where the non-zero entries of the matrix extend to the second-order neighbors of a vertex [Zha04]. The eigendecomposition of T'' is related to the singular value decomposition of T: the nonzero singular values of T are the square roots of the nonzero eigenvalues of T'' [TB97].

6.3.5. Weighted variations

Finally, it is trivial to extend the above definitions to weighted graphs, where the graph adjacency matrix W would be defined by $W_{ij} = w(e_{ij}) = w_{ij}$, for some edge weight $w : E \to \mathbb{R}^+$, whenever $(i,j) \in E$. Then, it is necessary to define the diagonal entries of the degree matrix D as $D_{ii} = \sum_{j \in N(i)} w_{ij}$.

6.4. Comments on combinatorial Laplacians

Zhang [Zha04] has examined various matrix-theoretic properties of the three (unweighted) combinatorial Laplacians K, T and Q. While the eigenvectors of the three operators appear qualitatively similar, it was shown that depending on the application, there are some subtle differences between them. For example, unlike K and T, the eigenvector of Q corresponding to the smallest (zero) eigenvalue is not a constant vector, where we assume that the mesh in question is connected. It follows that the normalized graph Laplacian Q cannot be used for low-pass filtering as done in [Tau95].

In the context of spectral mesh compression, Ben-Chen and Gotsman [BCG05] demonstrate that if a specific distribution of geometries is assumed, then the graph Laplacian K is optimal in terms of capturing the most spectral power for a given number of leading eigenvectors. This result is based on the idea that the spectral decomposition of a mesh signal of a certain class is equivalent to its PCA, when this class is equipped with the specific probability distribution.

However, it has been shown [ZB04, Zha04] that although optimal for a specific singular multivariate Gaussian distribution, the graph Laplacian K tends to exhibit more sensitivity towards vertex degrees, resulting in artifacts in meshes reconstructed from a truncated spectrum, as shown in Figure 9. In comparison, the Tutte Laplacian appears to possess more desirable properties in this regard and as well as when they are applied to spectral graph drawing [Kor03].

6.4.1. Graph Laplacian and Laplace-Beltrami operator

Following Mohar [Moh97], we show below that the graph Laplacian K can be seen as a combinatorial analogue of the Laplace-Beltrami operator defined on a manifold.

Let us first define an *oriented incidence matrix* R of a mesh graph G as follows. Orient each of the m edges of G in an arbitrary manner. Then $R \in \mathbb{R}^{n \times m}$ is an oriented vertex-edge incidence matrix where

$$R_{ie} = \begin{cases} -1 & \text{if } i \text{ is the initial vertex of edge } e, \\ +1 & \text{if } i \text{ is the terminal vertex of edge } e. \end{cases}$$

It is not hard to show that $K = RR^\intercal$ regardless of the assignment of edge orientations [Moh97].

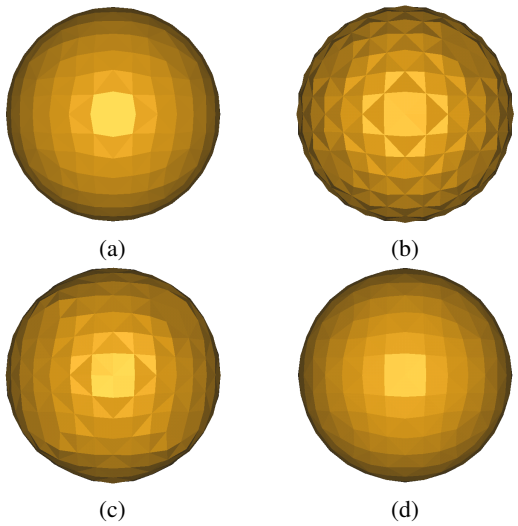


Figure 9: *Comparing results of spectral compression of the sphere (a) using 100 out of 379 spectral coefficients: (b) graph Laplacian K, (c) Tutte Laplacian T, (d) second-order Tutte Laplacian $T'' = T^\intercal T$. Visible artifacts result from using the graph Laplacian K, likely due to variations in the vertex degrees; the sphere mesh has 4-8 connectivity.*

The Laplace-Beltrami operator Δ on a Riemannian manifold is a second-order differential operator which can be defined as the divergence of the gradient [Ros97]. Given a smooth real scalar function ϕ defined over the manifold,

$$\Delta(\phi) = \mathrm{div}(\mathrm{grad}(\phi)).$$

Now let $G = (V, E)$ be the graph of a triangulation of the Riemannian manifold. Consider the scalar function $f : V \to \mathbb{R}$ which is a restriction of ϕ to V. Let R be an oriented incidence matrix corresponding to G, imposing an orientation on the edges of G. Consider the operator $R^\intercal$ which acts on functions f and returns a real-valued discrete function acting on the set of oriented edges,

$$(R^\intercal f)(e) = f(e^+) - f(e^-),$$

where e^+ and e^- are the terminal and initial vertices of the oriented edge e, respectively. One can view the above as a natural analogue of the gradient of ϕ along edge e. It follows that $K = RR^\intercal$ provides an analogue of the divergence of the gradient, giving a combinatorial version of the Laplace-Beltrami operator.

Recently these insights have been developed in considerably more detail within the emerging framework of the discrete exterior calculus which we discuss briefly in Section 6.5.1. In this context, $R^\intercal$ is related to the discrete differential operator, d, and R plays the role of the discrete co-differential, ∂, when the geometry of the triangles is ignored. The Laplacian (Laplace-deRham operator) is defined by $\Delta = \partial d + d\partial$, but the second term vanishes on 0-forms, i.e., functions. However, the Laplacian that is defined by means of the discrete exterior calculus belongs to the family of geometric mesh Laplacians.

6.5. Geometric mesh Laplacians

Although the graph Laplacian can be viewed as a discrete analogue of the Laplace-Beltrami operator, a geometric mesh Laplacian is constructed at the outset as a discrete approximation to the Laplace-Beltrami operator (Laplacian) on a smooth surface. On a C^∞ surface without boundary, $\mathscr{S}$, the Laplacian is a self-adjoint positive semi-definite operator $\Delta_\mathscr{S} : C^\infty(\mathscr{S}) \to C^\infty(\mathscr{S})$. An important, and even defining, property of $\Delta_\mathscr{S}$ is that

$$\int_\mathscr{S} f \Delta_\mathscr{S} g \, da = \int_\mathscr{S} \nabla f \cdot \nabla g \, da, \tag{5}$$

from which the self-adjoint nature of the Laplacian is an immediate consequence. On a surface with boundary, if von Neumann boundary conditions are imposed, constraining the functions to those whose gradient vanishes at the boundary, the Laplacian remains self-adjoint. Choosing $g = f$ yields

$$\int_\mathscr{S} f \Delta_\mathscr{S} f \, da = \int_\mathscr{S} \|\nabla f\|^2 \, da, \tag{6}$$

and establishes the positive semi-definite property. The right hand side of equation (6) defines the *Dirichlet* energy of the function f.

Given a triangle mesh $\mathscr{M}$ that approximates $\mathscr{S}$, we want an operator $\mathrm{L}_\mathscr{M}$ on $\mathscr{M}$ that will play the role that $\Delta_\mathscr{S}$ plays on $\mathscr{S}$. Let $\mathfrak{A}(\mathscr{M})$ denote the n dimensional vector space of functions on $\mathscr{M}$. These functions are defined by their values at the vertices and we let the function values on the faces of the mesh be given by barycentric interpolation so that $\mathfrak{A}(\mathscr{M})$ represents piecewise linear (continuous) functions on $\mathscr{M}$. To emphasize this viewpoint we drop, in this subsection, the convention of using boldface to represent elements of $\mathfrak{A}(\mathscr{M})$.

Within this setting a candidate, C, for $\mathrm{L}_\mathscr{M}$ is defined by

$$[Cf]_i = \sum_{j \in N(i)} \frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ij})(f_i - f_j), \tag{7}$$

where the angles α_{ij} and β_{ij} are subtended by the edge (i, j), as shown in Figure 10. In reference to (1), C is obtained by setting $b_i = 1$ for all i and $w_{ij} = \frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ij})$ for all i, j. If (i, j) is a boundary edge, the $\cot\beta_{ij}$ term vanishes. This corresponds to imposing von Neumann boundary conditions [VL08].

The expression (7) was obtained in [PP93] by noting that on a triangular face T with vertices $\mathbf{p}_i, \mathbf{p}_j, \mathbf{p}_k$, we have

$$\nabla\varphi_k \cdot \nabla\varphi_j = -\frac{1}{2a_T}\cot\angle\mathbf{p}_i, \tag{8}$$

where φ_j is the nodal linear basis function ("hat function") centred at $\mathbf{p}_j$ and a_T is the area of T.

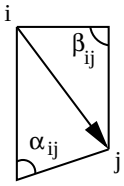


Figure 10: *Angles involved in the calculation of cotangent weights for geometric mesh Laplacians.*

In analogy with equation (6), C satisfies

$$f^{\mathsf{T}}Cf = \sum_{T\in\mathcal{F}} \|\nabla f|_T\|^2 a_T = \int_{\mathscr{M}} \|\nabla f\|^2 \, da \tag{9}$$

for piecewise linear functions $f \in \mathfrak{A}(\mathscr{M})$. Also, C is a symmetric matrix so it possesses the important self-adjoint property of the Laplacian.

The expression on the right hand side of equation (7) can be arrived at in several different ways. Let Ω_i be a neighbourhood of $\mathbf{p}_i$ in $\mathscr{M}$ such that $\partial\Omega_i$ intersects at the midpoint each of the edges linking $\mathbf{p}_i$ with its one-ring neigbours. A natural choice for such a cell has its boundary defined by straight lines connecting the barycentres of the triangles adjacent to $\mathbf{p}_i$ with the midpoints of the adjacent edges. We refer to cells constructed in this way as *barycells*.

The rhs of equation (7) can then be seen to be the total flux of the gradient of the piecewise linear function f that crosses $\partial\Omega_i$. Such a computation can be found in [MDSB02] for example. By the divergence theorem it follows that $[Cf]_i$ is actually the integral of the Laplacian of f over Ω_i.

This points to a weakness in C as a representative of the Laplacian: as an operator $\mathfrak{A}(\mathscr{M}) \to \mathfrak{A}(\mathscr{M})$, C alone yields nodal values that represent the integral of $\mathrm{L}_{\mathscr{M}} f$ over a neighbourhood, rather than a point sample.

The solution proposed in [MDSB02] is to divide by the area of the local neighbourhood thus yielding values that are local spatial averages of the Laplacian. If D is the diagonal matrix whose entries are $|\Omega_i|$, the area of Ω_i, then the Laplacian proposed is $Y = D^{-1}C$. Then

$$[Yf]_i = \frac{1}{|\Omega_i|} \sum_{j\in N(i)} \frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ji})(f_i - f_j). \tag{10}$$

However, Y is not a symmetric matrix, so the self-adjoint character of the Laplacian is apparently sacrificed. By modifying the definition of the scalar product in $\mathfrak{A}(\mathscr{M})$ as described in Section 6.2.2, the situation is salvaged. For $f, g \in \mathfrak{A}(\mathscr{M})$ we define

$$\langle f, g\rangle_D = f^{\mathsf{T}} D g. \tag{11}$$

Then $\langle Yf, g\rangle_D = \langle f, Yg\rangle_D$ and the eigenfunctions of Y are orthogonal with respect to this inner product.

Now instead of equation (9) we have

$$\langle f, Yf\rangle_D = \sum_{T\in\mathcal{F}} \|\nabla f|_T\|^2 a_T = \int_{\mathscr{M}} \|\nabla f\|^2 \, da. \tag{12}$$

For a smooth surface $\mathscr{S}$ the usual scalar product on $C^\infty(\mathscr{S})$ is given by

$$\langle f, g\rangle = \int_{\mathscr{S}} fg \, da. \tag{13}$$

If we interpret $\langle f, g\rangle_D$ as an approximation to this integral, then equation (12) is again in analogy with equation (6) and in a sense the analogy is closer.

However, there is something aesthetically wanting about using an approximation to the integral as a scalar product. Members of $\mathfrak{A}(\mathscr{M})$ are viewed as piecewise linear functions. As such we are able to integrate them analytically. Indeed for $f, g \in \mathfrak{A}(\mathscr{M})$ we have

$$\int_{\mathscr{M}} fg \, da = \int_{\mathscr{M}} \sum_i f_i\varphi_i \sum_j g_j\varphi_j \, da = f^{\mathsf{T}} B g, \tag{14}$$

where B is the *mass matrix* encountered in finite element analysis. It is a sparse matrix defined by $B_{ij} = \int_{\mathscr{M}} \varphi_i\varphi_j \, da$. If $\mathbf{p}_i$ and $\mathbf{p}_j$ are neighbours, then B_{ij} is $1/12$ the area of the triangles adjacent to edge (i, j). The diagonal entries B_{ii} are $1/6$ the area of the triangles adjacent to $\mathbf{p}_i$. All other entries are zero.

Thus another candidate for $\mathrm{L}_{\mathscr{M}}$ is suggested: $F = B^{-1}C$. This matrix is self-adjoint with respect to the inner product $\langle f, g\rangle_B = f^{\mathsf{T}} B g$, which also renders its eigenvectors orthogonal. The generalized eigenvalue problem that yields the spectral decomposition of F (c.f. Section 9.1) is exactly the equation that results when the eigenfunctions of the Laplace-Beltrami operator are computed via the finite element method with linear elements. For this reason we refer to F as the FEM operator.

The sum of all the entries in B or D is equal to the surface area of $\mathscr{M}$. Note that when the matrix D is defined using barycells as the Ω_i, then the diagonal entries of D are just the sums of the corresponding rows in B. Thus in this context the operator defined by Y represents the lumped mass approximation that is sometimes employed in finite element methods.

Although the FEM Laplacian, F, has been presented as an improvement upon Y and C, experiments indicate that Y produces eigenvalues and eigenfunctions which are more robust with respect to variations in the mesh used to represent a surface [DZM07]. As mentioned in [SF73], this may be due to the dampening effect of the lumped mass approximation cancelling errors in the stiffness matrix C.

The geometric Laplacians we have introduced are based on the cotan formula (7) and represent the most popular discrete approximations to the Laplace-Beltrami operator cur-

rently used for geometry processing. There are other geometric Laplacians, as presented in [Fuj95], [Flo03] and [Xu04a] for example, but these have not seen use in spectral geometry processing, although a Laplacian similar to the one presented by Fujiwara was used by Karni and Gotsman [KG00] to assess errors for spectral compression.

6.5.1. The discrete exterior calculus

In differential geometry, differential forms give rise to the powerful mathematical framework of the exterior calculus. This framework is further enriched with the introduction of the metric tensor in Riemannian geometry, and yields a Laplacian operator which acts on differential forms. This operator is sometimes referred to as the *Laplace-deRham operator*, or the *Hodge Laplacian*. Functions on a manifold are 0-forms and, when applied to functions, the Laplace-deRham operator is equivalent to the Laplace-Beltrami operator. A formal development of the subject can be found in [Ros97, Cha84].

The discrete exterior calculus is an emerging mathematical framework that is formulated from first principles in the discrete setting. It is modeled on its differential counterpart, but it should not be considered as a discretization of the continuous theory. A persuasive argument for this viewpoint, together with a gentle introduction to the subject is provided in [DKT06]. More comprehensive expositions can be found in the original work of Hirani [Hir03], and in a later preprint [DHLM05].

The discrete exterior calculus provides yet another derivation of the cotan weights that appear in equation (7). The Laplacian operator computed in this way has the same form as Y in equation (10), but the cell Ω_i in this context is the *circumcentric dual cell* of $\mathbf{p}_i$. It will have sides that connect the circumcentres of the triangles incident to $\mathbf{p}_i$. On general triangle meshes these abstract dual cells may have sides with negative length and may even have a negative area. However, if the triangle mesh is intrinsically Delaunay, then the cells are simply the Voronoi cells of the vertices. A derivation of this Laplacian can be found in [VL08, DHLM05]. A nice exposition presented in [WBH*07] ties this viewpoint in with the presentation of the geometric Laplacians given above.

In closely related work, Glickenstein [Gli05] focuses on the orthogonal duality structures that can be defined on $\mathscr{M}$ when the vertices are given distinct weights. The Laplacian of the discrete exterior calculus is thus presented as a particular case of a family of mesh Laplacians; the case when all the vertices are equally weighted. The dual cells studied by Glickenstein are related to the power diagram of a set of weighted vertices in the same way as the circumcentric dual cells are related to the Voronoi diagram.

6.6. Spectral properties of mesh Laplacians

We present here some notable properties of the eigenvectors of a mesh Laplacian operator. We begin by noting in Section 6.6.1 that the order in which the mesh vertices are indexed has no consequence on the eigenvectors. This observation applies to any mesh operator, not just the Laplacians. We then discuss in Section 6.6.2 a property that is particular to eigenfunctions of a Laplacian, and in Section 6.6.3 the characteristics of the Fiedler vector.

6.6.1. Mesh indexing independence and symmetries

We represent a function on a mesh $\mathscr{M}$ as a vector of values at the vertices. The order in which we choose to list the vertices is arbitrary, but it determines the matrix representation of a linear operator that acts on mesh functions. However, the spectral decomposition of the linear operator is not affected by this choice.

Indeed, for any permutation of the n indices of $\mathscr{M}$, there is an associated permutation matrix P which is obtained by performing the same permutation on the columns of the $n \times n$ identity matrix. Note that P is an orthogonal matrix. If $\mathbf{f}$ is a function represented with the original ordering of the vertices, then $\tilde{\mathbf{f}} = P\mathbf{f}$ will be its representation in the new ordering. A linear operator L is represented by a matrix and it transforms as

$$\tilde{L} = PLP^{\mathsf{T}}.$$

Now if

$$L\mathbf{v} = \lambda\mathbf{v},$$

then

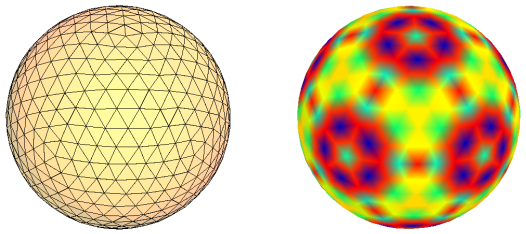
$$\tilde{L}\tilde{\mathbf{v}} = (PLP^{\mathsf{T}})(P\mathbf{v}) = PL\mathbf{v} = P\lambda\mathbf{v} = \lambda\tilde{\mathbf{v}}. \qquad (15)$$

We see that the transformed eigenvector becomes an eigenvector of the transformed operator and the eigenvalue remains the same.

Equation (15) has interesting implications with respect to symmetries of $\mathscr{M}$. If P leaves L invariant, i.e. $PLP^{\mathsf{T}} = L$, we say that P represents a *symmetry* of $\mathscr{M}$ with respect to L. In this case, Equation (15) implies that $\mathbf{v}$ and $\tilde{\mathbf{v}}$ are both eigenvectors of L with the same eigenvalue. If the corresponding eigenvalue has multiplicity one, then $P\mathbf{v} = \mathbf{v}$. Thus $\mathbf{v}$ captures the symmetry expressed by P. On highly symmetric meshes, such eigenvectors can have attractive visualizations, as shown in Figure 11.

6.6.2. Nodal domains

An interesting property of the Laplacians is the relation between their eigenfunctions and the number of nodal domains that they possess. A *nodal set* associated with an eigenfunction is defined as the set composed of points at which the eigenfunction takes on the value zero, and it partitions the surface into a set of *nodal domains*, each taking on positive or negative values in the eigenfunction. Examples of these structures are shown in Figure 3. The nodal sets and domains are bounded by the following theorem [JNT01]:



(a) (b)

Figure 11: *(a) The spherical mesh is created by projecting the vertices of a subdivided icosahedron onto the sphere. (b) An eigenvector associated with an eigenvalue of multiplicity one must be invariant under all transformations which leave the operator invariant. In this case the symmetry group of the icosahedron is comprised entirely of such transformations.*

Theorem 6.1 (Courant's Nodal Domain Theorem) Let the eigenfunctions of the Laplace operator be labelled in increasing order. Then, the i-th eigenfunction can have at most i nodal domains, i.e., the zero set of the i-th eigenfunction can separate the domain into at most i connected components.

This theorem only gives an upper bound for the number of nodal domains. The direct relation between a specific eigenfunction and its nodal domains is not clear. One possible application of nodal domains is pointed out by Dong et al. for spectral mesh quadrangulation [DBG*06], explained in Section 10.2.1. By carefully selecting a suitable eigenfunction, they take advantage of the partitioning given by the nodal domains and remesh an input surface.

Discrete analogues of Courant's Nodal Domain Theorem are known [DGLS01]. In fact, these results are applicable to a larger class of discrete operators, called the discrete Schrödinger operators, which we define in Section 7.1.

6.6.3. The Fiedler vector

By Theorem 5.3 and Equation (4), we can characterize the Fiedler vector $\mathbf{v}_2(K)$ of a connected graph, the eigenvector associated with the smallest non-zero eigenvalue of K, as follows,

$$\mathbf{v}_2(K) = \text{argmin}_{\mathbf{u}^\tau \mathbf{1}=0,\ \|\mathbf{u}\|_2=1} \sum_{i,j=1}^{n} w_{ij}(u_i - u_j)^2.$$

This extremal property of the Fiedler vector reveals its usefulness in providing a heuristic solution to the NP-hard *minimum linear arrangement* (MLA) problem. MLA seeks a permutation $\pi : V \to \{1,2,\ldots,n\}$ of the vertices of a graph $\mathcal{G} = (V,E)$ so as to minimize

$$\sum_{i,j=1}^{n} w_{ij}|\pi(i) - \pi(j)|.$$

Another example is the well-known *normalized cut* (NCut) problem, which is also NP-hard. Given a graph $\mathcal{G} = (V,E)$, with edge weights $w : E \to \mathbb{R}^+$, NCut seeks a bipartition of V into disjoint subsets A and B which minimizes the normalized cut criterion,

$$NCut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)},$$

where $cut(A,B) = \sum_{i\in A, j\in B} w_{ij}$ defines the graph cut, $assoc(A,V) = \sum_{i\in A, k\in V} w_{ik}$ is the total connection from nodes in A to all the nodes in the graph, and $assoc(B,V)$ is similarly defined. It has been shown that when relaxing NCut into the real value domain, the Fiedler vector of the Tutte Laplacian for $\mathcal{G}$ provides a solution [SM00].

7. Other operators for spectral methods

The spectral characteristics and sparsity of the mesh Laplacian operators make them ideally suited for many spectral mesh processing applications. However, many other operators have also demonstrated their utility, in particular in the fields of computer vision and machine learning where the input data can take an abstract form and do not reside on an apparent surface. We now present a few such examples.

7.1. Discrete Schrödinger operator

In quantum mechanics, the Schrödinger operator and Schrödinger equation play a central role as the latter models how the quantum state of a physical system changes in time. The discrete Schrödinger operator is defined by supplementing the discrete Laplacian with a *potential* function, which is again a term arising from the study of electrical networks. The potential function is a real function, taking on both negative and positive values, defined on the vertices of a graph. Specifically, for a given graph $\mathcal{G} = (V,E)$, H is a discrete Schrödinger operator for $\mathcal{G}$ if

$$H_{ij} = \begin{cases} \text{a negative real number} & \text{if } (i,j) \in E, \\ \text{any real number} & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

Such operators have been considered by Yves Colin de Verdière [dV90] in the study of a particular spectral graph invariant, as well as by Davies et al. al. [DGLS01] who have proved discrete analogues of Courant's nodal domain theorem [JNT01].

A special sub-class of discrete Schrödinger operators, those having exactly one negative eigenvalue, have drawn particular attention. Lovász and Schrijver [LS99] have proved that if a graph $\mathcal{G}$ is planar and 3-connected, then any matrix M in that special sub-class for $\mathcal{G}$ with co-rank 3 (dimensionality of the null-space of M) admits a valid null-space embedding on the unit sphere. The null-space embedding is obtained by the three eigenvectors corresponding to the zero eigenvalue of M. This result has

subsequently been utilized by Gotsman et al. [Got03] to construct valid spherical mesh parameterizations.

7.2. Higher-order operators

In the fields of computer vision and machine learning, spectral methods usually employ a different operator, the so-called *affinity matrix* [SM00, Wei99]. Each entry W_{ij} of an affinity matrix W represents a numerical relation, the affinity, between two data points i and j, e.g., pixels in an image, vertices in a mesh, or two face models in the context of face recognition. Note that the affinity matrix differs from the Laplacian in that affinities between all data pairs are defined. Therefore this matrix is not sparse in general. In practice, this non-sparse structure implies more memory requirements and more expensive computations.

7.2.1. Gram matrices

A particularly important class of affinity matrices are the so-called *Gram matrices*, which are frequently encountered in machine learning. By definition, an $n \times n$ Gram matrix is a matrix of inner products for a given set of n-dimensional vectors. Specifically, let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ be such a set of vectors, then their associated Gram matrix is given by $G \in \mathbb{R}^{n \times n}$ where $G_{ij} = \langle \mathbf{v}_i, \mathbf{v}_j \rangle$. If we denote by V the $n \times k$ matrix whose columns are the $\mathbf{v}_i$'s, then $G = VV^{\mathrm{T}}$.

The use of Gram matrices in machine learning is typically associated with the application of the "kernel trick" [HLMS04]. In this context, the Gram matrix is derived by applying a kernel function to pairwise distances between a set of data points. For example, when a Gaussian kernel is used, we obtain a Gram matrix W with

$$W_{ij} = e^{-||x_i - x_j||^2 / 2\sigma^2}$$

where the Gaussian width σ is a free parameter. Furthermore, a normalized affinity matrix N can be obtained,

$$N = D^{-1/2} W D^{-1/2},$$

where the degree matrix D is defined as before.

While some authors advocate the use of the normalized affinity matrix, e.g., for spectral clustering [NJW02, vLBB05], others employ the unnormalized W; subtle differences between the two have been studied [vLBB05]. In practice, different ways of defining the affinities exist for mesh processing. One possibility is to use vertex-to-vertex distances in the graph implied by the mesh connectivity [LZvK06, JZvK07]. This is often a way to approximate geodesic distances. Other approaches have also been proposed, e.g., refining the graph distances by considering the more global *traversal distances*. These consist in defining the affinity between two vertices as the number of paths in the graph between these two elements [SFYD04].

7.2.2. Dissimilarity-based multidimensional scaling

Multidimensional scaling or MDS is a set of related techniques often employed in data visualization for exploring similarities or dissimilarities in data [CC94]. In classical MDS, low-dimensional spectral embeddings, typically 2D, are constructed to facilitate visualization of high-dimensional data. Given some $n \times n$ pairwise dissimilarity distance matrix M, e.g., one which measures squared geodesic distances between mesh vertices [ZKK02], *double centering* and normalization results in the matrix,

$$B = -\frac{1}{2} JMJ,$$

where

$$J = I - \frac{1}{n} \mathbf{1}\mathbf{1}^{\mathsf{T}}$$

and $\mathbf{1}$ is the column vector of 1's. It can be shown that the Euclidean distances between points in the spectral embedding obtained by the eigenvectors of B closely approximate the distances in M. The related theory is given in part by Theorem 5.5. MDS, in combination with geodesic distances, has been used to obtain bending-invariant shape signatures [EK03] and low-distortion texture mapping [ZKK02].

7.2.3. Non-sparse Laplacian

The affinity matrix, which is analogous to a graph adjacency matrix, can also be used to define a *non-sparse Laplacian*, for lack of a better term. The non-sparse Laplacian is given by $D - W$, where D and W are as defined in previous sections. Basically, this operator can be seen as giving the overall structure of a Laplacian to the affinity matrix. Shi and Malik [SM00] employ a normalized version of this operator to perform image segmentation based on normalized cuts.

In much of the machine learning literature, the non-sparse Laplacian with W defined as in Section 7.2.1 is called the graph Laplacian, e.g., in [BN05]. In this latter work, convergence of the non-sparse Laplacian to the Laplace-Beltrami operator is proven. Specifically, the non-sparse (graph) Laplacian is defined on a cloud of points sampled near a manifold. Under certain conditions and as the sampling rate goes to infinity, the non-sparse Laplacian can be shown to approach the Laplace-Beltrami operator of the underlying manifold.

8. Use of different eigenstructures

The eigendecomposition of a linear mesh operator provides a set of eigenvalues and eigenvectors, which can be directly used by an application to accomplish different tasks. Moreover, the eigenvectors can also be used as a basis onto which a signal defined on a triangle mesh is projected. The resulting coefficients can be further analyzed or manipulated. In this section, we expand our discussion on these issues.

8.1. Use of eigenvalues

Drawing analogies from discrete Fourier analysis, one would treat the eigenvalues of a mesh Laplacian as measuring the frequencies of their corresponding eigenfunctions [Tau95]. However, it is not easily seen what the term frequency means exactly in the context of eigenfunctions that oscillate irregularly over a manifold. Furthermore, since different meshes generally possess different operators and thus different eigenbases, using the magnitude of the eigenvalues to pair up corresponding eigenvectors between the two meshes for shape analysis, e.g., correspondence, is unreliable [JZ06]. Despite of these issues, much empirical success has been obtained using eigenvalues as global shape descriptors for graph [SMD*05] and shape matching [JZ07]. These applications are described in more detail in Section 10.1.

Besides directly employing the eigenvalues as graph or shape descriptors, spectral clustering methods use the eigenvalues to scale the corresponding eigenvectors so as to obtain some form of normalization. Caelli and Kosinov [CK04] scale the eigenvectors by the squares of the corresponding eigenvalues, while Jain and Zhang [JZ07] provide justification for using the square root of the eigenvalues as a scaling factor. The latter choice is consistent with the scaling used in spectral clustering [NJW02], normalized cuts [SM00], and multidimensional scaling [CC94].

8.2. Use of eigenvectors

Eigenvectors are typically used to obtain an embedding of the input shape in the spectral domain. After obtaining the eigendecomposition of a specific operator, the coordinates of vertex i in a k-dimensional embedding are given by the i-th row of matrix $V_k = [\mathbf{v}_1, \ldots, \mathbf{v}_k]$, where $\mathbf{v}_1, \ldots, \mathbf{v}_k$ are the first k eigenvectors from the spectrum (possibly after scaling). Whether the eigenvectors should be in ascending or descending order of eigenvalues depends on the operator that is being used. In the case of Gram matrices, eigenvectors corresponding to the largest eigenvalues are used to compute spectral embeddings. While for the various Laplacian operators, the opposite end of the spectrum is considered.

For example, spectral clustering makes use of such embeddings. Ng et al. [NJW02] present a method where the entries of the first k eigenvectors corresponding to the largest eigenvalues of a normalized affinity matrix (see Section 7.2.1) are used to obtain the transformed coordinates of the input points. Additionally, the embedded points are projected onto the unit k-sphere. Points that possess high affinities tend to be grouped together in the spectral domain, where a simple clustering algorithm, such as k-means, can reveal the final clusters. Furthermore, the ability of the spectral methods to unfold nonlinearity in the input data has been demonstrated via numerous examples, including data sets similar to the one shown in Figure 7.

8.3. Use of eigenprojections

If a mesh operator possesses a set of orthogonal eigenvectors, given by the columns of matrix V, then any discrete function defined on the mesh vertices, given by a vector $\mathbf{x}$, can be transformed into the spectral domain by

$$\tilde{\mathbf{x}} = V^\mathsf{T}\mathbf{x},$$

where $\tilde{\mathbf{x}}$ contains the obtained spectral coefficients. This can be seen as a change of basis for $\mathbf{x}$. As well, the transform is energy-preserving in that the Euclidean 2-norm is preserved: $\|\tilde{\mathbf{x}}\|_2 = \|\mathbf{x}\|_2$. The inverse transform is obtained by

$$\mathbf{x} = V\tilde{\mathbf{x}}.$$

These spectral transforms are closely related to the Fourier transform that is the foundation of signal processing theory. In geometry processing, the signal considered is often the embedding function that specifies the 3D coordinates of each vertex. This signal is commonly referred to as the geometry of the mesh. Thus the geometry signal is an $n \times 3$ matrix P whose ith row is the transpose of the position vector of the ith vertex.

The resulting coefficients $\tilde{P} = V^\mathsf{T}P$ are then a representation of the mesh geometry in the spectral domain. A rotation of the mesh yields a corresponding rotation of the spectral coefficients. In other words, the spectral transform commutes with rotations. Indeed, if R is a 3×3 rotation matrix, then $P' = PR^\mathsf{T}$ is the geometry of the rotated mesh and we have

$$\tilde{P}' = V^\mathsf{T}P' = V^\mathsf{T}PR^\mathsf{T} = \tilde{P}R^\mathsf{T}.$$

If the operator is a Laplacian, translations of the mesh in the spatial domain do not affect the spectral coefficients since constant signals lie in the kernel of the operator.

As in the case of Fourier analysis, the intuition is that when the signal is transformed into the spectral domain, it might be easier to carry out certain tasks because of the relation of the coefficients to low and high frequency information. For example, the projections of P with respect to the eigenvectors of the graph Laplacian can be used for mesh compression [KG00]. That is, a set of the transformed coefficients from the high-frequency end of the spectrum can be removed without affecting too much the approximation quality of the mesh, when it is reconstructed by the inverse transform.

For spectral watermarking of meshes [OTMM01] however, it is the low-frequency end of the spectrum that is to be modulated. This way, the watermark is less perceptible and the watermarked mesh can become resilient against such attacks as smoothing. We elaborate more on these in Section 10.3. The observation that the human eye is less sensitive to low-frequency errors in geometric shape was first made by Sorkine et al. [SCOT03] in their work on high-pass quantization for mesh compression. This work, along with related developments, can be found in [Sor05].

9. Efficient computations

Since spectral methods all require the construction of a (possibly non-sparse) mesh operator and its eigendecomposition, efficiency is a concern for large meshes. In this section, we survey several speed-up techniques.

For some operators, such as the FEM operator, the generalized eigenvalue problem, discussed in Section 9.1, is useful for computing the spectrum without the need to construct an explicit representation of the operator. When it comes to actual computation, methods that either compute a good approximation of the eigenvectors or that exploit the structure of a specific type of matrices have been proposed. When using a mesh Laplacian, it is natural to exploit its sparsity. One method proposed to accomplish this makes use of a multi-scale approach, as described in Section 9.2. An alternative, which allows one to compute a specific set of eigenvectors, is to introduce a spectral shift into the operator. This can be combined with iterative methods to speed up the computation, as described in Section 9.3. On the other hand, to compute the eigendecomposition of dense affinity matrices, the Nyström method [FBCM04] can be employed. It is based on computing approximate eigenvectors given by a number of sampled elements; see Section 9.4. Thus, it also avoids the construction of the full matrix describing the operator.

9.1. The generalized eigenvalue problem

Numerical eigensolvers are usually much more efficient at producing a spectral decomposition if the matrix is symmetric. If the matrix has the form (2), with B a diagonal matrix, then the solver can be given the symmetric matrix $O = B^{-1/2}SB^{-1/2}$ which is similar to L and thus has the same eigenvalues. As described in Section 6.2.2, the eigenvectors of O will need to be adjusted by a factor of $B^{-1/2}$ to obtain eigenvectors of L.

However, if B is not diagonal, as is the case with the mass matrix associated with the FEM operator, $F = B^{-1}C$, then computing $B^{-1/2}$ is not a viable option. In fact, there is no need to compute B^{-1} at all. Instead, one solves the generalized eigenvalue problem,

$$C\mathbf{f} = \lambda B\mathbf{f}, \tag{16}$$

and the matrix F is never explicitly constructed. Equation (16) can be handled by most popular solvers and it is equivalent to $F\mathbf{f} = \lambda\mathbf{f}$.

9.2. Exploiting sparsity

Koren et al. [KCH02] propose ACE, or *Algebraic multigrid Computation of Eigenvectors*, a multi-scale method to accelerate the computation of eigenvectors of Laplacians. The method proceeds in two steps: *coarsening* and *refinement*. An initial high-dimensional problem is progressively reduced to lower and lower dimensions by applying the coarsening step, which creates less complex instances of the problem. The exact solution of one of these low-dimensional versions of the problem is then computed. Furthermore, the refinement step progressively translates the solution of the problem in lower dimensions to higher ones, usually performing some adjustments to the solution, until a solution to the problem in the original dimension is obtained.

However, the design of the coarsening and refinement steps is usually application-dependent, since both steps rely on exploiting a special feature of the problem being solved and need to preserve the essence of the initial problem. In the case of Laplacian operators, the sparsity of the related matrices is what allows to speed up the computation of eigenvectors by means of a multi-scale method.

To carry out the coarsening and refinement steps, the key concept introduced by Koren et al. [KCH02] is that of an interpolation matrix A, which is an $n \times m$ matrix that interpolates m-dimensional vectors y into n-dimensional ones x, given by $x = Ay$. The interpolation matrix is employed to obtain a coarsened Laplacian matrix given by $L^c = A^\intercal LA$, where L is the original Laplacian. The same interpolation matrix is then used for the refinement step, computing the eigenvectors of the problem at higher and higher resolutions.

The interpolation matrix is created either by contracting edges on the underlying graph, or by performing a weighted interpolation of a node from several nodes in the coarser version of the problem. The contractions or weighted interpolations are what define the entries of the interpolation matrix, which can be very sparse, when computed by the contractions, or less sparse but conveying a more accurate interpolation, when weighted interpolations are used. Determining which method should be preferentially used depends mainly on whether the underlying graphs are homogeneous or not.

9.3. Spectral shift and iterative methods

Iterative algorithms compute the eigenvectors of large sparse matrices in a more efficient manner [TB97]. However, these methods only allow to obtain the leading eigenvectors of a matrix. In order to compute eigenvectors associated to a specific set of eigenvalues, it is necessary to modify the eigenproblem being solved. Dong et al. [DBG*06] and Vallet and Lévy [VL08] accomplish that by utilizing a *spectral shift*.

The original problem $L\mathbf{v} = \lambda\mathbf{v}$ is modified to the form

$$(L - \sigma I)\mathbf{v} = (\lambda - \sigma)\mathbf{v}$$

so that when this eigenproblem is solved, the eigenvectors that are obtained correspond to eigenvalues close to σ. This is valid due to the fact that, if $\mathbf{v}$ is an eigenvector of L with associated eigenvalue λ, then it is also an eigenvector of $L - \sigma I$ with associated eigenvalue $\lambda - \sigma$.

Moreover, to compute the eigenvectors associated with the smallest eigenvalues instead of the leading ones, Vallet and Lévy [VL08] also resort to the idea of swapping the

spectrum of a matrix by inverting it. This comes from the observation that the leading eigenvalues of the eigenproblem $L^{-1}\mathbf{v} = (1/\lambda)\mathbf{v}$ are the smallest eigenvalues of $L\mathbf{v} = \lambda\mathbf{v}$, with the same set of eigenvectors $\mathbf{v}$. This is combined with the spectral shift to obtain the *shift-invert* spectral transform, which can be used to split the computation of the eigenvectors of large matrices into multiple bands. Each band can be computed in linear time, and this process can be further accelerated by the use of out-of-core factorization methods.

9.4. Nyström approximation

In order to compute the eigendecomposition of large affinity matrices, one technique that can be used to approximate the leading eigenvectors of sampled matrices is the Nyström method [FBCM04]. Given a set of mesh vertices $\mathcal{Z}$ of size n, whose affinities are given in the matrix $W \in \mathbb{R}^{n\times n}$, the first step in Nyström's method is to divide the set $\mathcal{Z}$ into a subset of samples $\mathcal{X}$ of size l, with $l \ll n$, and a subset $\mathcal{Y}$ of size m, which contains the remaining points.

Next, the affinities between the points in the subsets $\mathcal{X}$ and $\mathcal{Y}$ are stored in the matrices $A \in \mathbb{R}^{l\times l}$ and $C \in \mathbb{R}^{m\times m}$, respectively. The *cross-affinities* between points of $\mathcal{X}$ and $\mathcal{Y}$ are stored in matrix $B \in \mathbb{R}^{l\times m}$. Thus, the matrix W can be written in the following block form

$$W = \begin{bmatrix} A & B \\ B^\intercal & C \end{bmatrix}.$$

After obtaining the eigendecomposition of the matrix A, given by $A = U\Lambda U^\intercal$, the columns of $\bar{U}$, expressed below, are the approximation for the l leading eigenvectors of W, that is, the l eigenvectors related to the largest eigenvalues. $\bar{U}$ is given by Nyström's method as

$$\bar{U} = \begin{bmatrix} U \\ B^\intercal U \Lambda^{-1} \end{bmatrix}.$$

Therefore, only the affinities A between the sampled points and the cross-affinities B need to be computed in order to obtain the approximation. Moreover, the original matrix W can be reconstructed by using the approximated eigenvectors. The approximation of W is given by

$$\bar{W} = \bar{U}\Lambda\bar{U}^\intercal = \begin{bmatrix} A & B \\ B^\intercal & B^\intercal A^{-1} B \end{bmatrix}.$$

The quality of this approximation is given by the quantity $\|W - \bar{W}\|$, which is equivalent to computing only the norm of the *Schur complement* $\|C - B^\intercal A^{-1} B\|$. However, it is expensive to directly compute this quantity since it requires the large matrix C. Methods that compute an indirect quality measure should usually be employed.

The overall complexity of this method is $O(l^3)$ for computing the eigendecomposition of matrix A, and $O(ml^2)$ for obtaining the approximated eigenvectors via extrapolation. Therefore, the problem of computing the leading eigenvectors is reduced from $O(n^3)$ to only $O(ml^2 + l^3)$, recalling that $l \ll n$. In practical applications such as spectral image segmentation [FBCM04], spectral mesh segmentation [LJZ06, ZL05], and spectral shape correspondence [JZvK07] and retrieval [JZ07], l can be as small as less than 1% of n while still ensuring satisfactory results.

Nevertheless, there are a few issues which emerge with the use of Nyström's method. First of all, the approximated eigenvectors are not orthogonal. Fowlkes et al. [FBCM04] present two techniques for re-orthogonalization, depending on whether the affinity matrix A is positive definite or indefinite. However, this step may introduce additional numerical errors. Moreover, the accuracy of the eigenvectors obtained by Nyström's method is determined by the sampling technique employed. Different schemes were proposed, e.g., random sampling [FBCM04], max-min farthest point sampling [dST04], and greedy sampling based on maximizing the trace of the matrix $B^\intercal A^{-1} B$ [LJZ06]. However, these schemes all judge the quality of a sampling by the approximation quality of the eigenvectors obtained, measured by standard matrix norms, and they do not take into consideration the application at hand.

10. Applications

In this section, we survey applications which apply the spectral approach. Although our focus will be on spectral methods for mesh processing and analysis, highly relevant and representative problems and techniques from other fields will also be covered for completeness.

Let us first list in Table 1 the relevant references grouped by applications. Most of these references will be discussed in detail in subsequent sections. Others have been discussed in other parts of the paper, where appropriate.

10.1. Use of eigenvalues

Although most applications in the field of geometry processing employ eigenvectors to accomplish different tasks, eigenvalues have been successfully used to address certain problems, such as graph and shape indexing.

- **Graph indexing:**

  The use of graph spectra for indexing is well known in computer vision and machine learning, e.g., see a recent comparative study in [ZW05]. However, one should note the existence of *iso-spectral graphs*, graphs that are topologically different yet possessing the same spectra [CK04]. Recently, Shokoufandeh et al. [SMD*05] make use of eigenvalues for indexing graphs that represent hierarchical structures, such as *shock graphs* that define image silhouettes. The eigenvalues provide a measure indicating which graphs are similar and should be compared with a more expensive matching algorithm. Basically, the

Application	References
Clustering	[NJW02], [VM03], [BN03], [KVV00], [vLBB05], [vL06], [BH03], [MP04]
Graph drawing	[Hal70], [PST00], [KCH02] [DPS02], [Kor03]
Graph indexing	[SMD*05]
Graph matching	[Ume88], [SB92], [CK04]
Graph partitioning	[ST96], [SM97], [PF98], [Wei99], [AKY99],
Matrix reordering	[BPS93]
Mesh compression	[KG00], [KG01], [ZB04], [Zha04], [BCG05]
Mesh parameterization	[GGS03], [ZSGS04] [MTAD08]
Mesh reordering	[IL05], [LZvK06]
Mesh segmentation	[LZ04], [KLT05], [ZL05] [dGGV08]
Mesh smoothing	[VL08]
Remeshing	[DBG*06]
Shape correspondence	[LH05], [JZ06], [JZvK07] [MCBH07]
Shape indexing	[EK03], [RWP06], [JZ07] [Rus07]
Surface reconstruction	[KSO04]
Symmetry detection	[OSG08]
Texture mapping	[ZKK02]
Watermarking	[OTMM01], [OMT02]

Table 1: *Applications addressed by spectral methods.*

index is defined as the sum of eigenvalues of the adjacency matrix of the graph, which allows to obtain a low-dimensional index. However, in order to also reflect local properties of the graph, one term corresponding to the sum of eigenvalues is stored for each subtree of the graph.

- **Shape indexing:**

  For 3D shape indexing, Jain and Zhang [JZ07] propose to use the leading eigenvalues of an affinity matrix constructed using approximated geodesic distances over the shape surfaces. The similarity between two models is given by the χ^2-distance between the selected k eigenvalues of the two models, where k is usually very small, for example, 6. This comparison between the first eigenvalues intuitively corresponds to comparing the variation of the models in each of the first k nonlinear principal components. As argued before, the bending invariance of geodesic distances should facilitate the retrieval of articulated shapes. This has indeed been confirmed by their experiments on the McGill articulated shape database [McG]. The simple eigenvalue-based descriptor did outperform two of the best shape descriptors, the spherical harmonics descriptor [MKR03] and the light field descriptor [CTSO03], even when they are applied to the bending-normalized spectral embeddings.

  Reuter et al. [RWP06] elect to use the eigenvalues of the Laplace-Beltrami operator for shape indexing. Recently, Rustamov [Rus07] also relies on the Laplace-Beltrami operator and constructs spectral embeddings using its eigenvectors. Normalization with respect to bending is also achieved by using this operator and a slightly modified version of the shape distribution descriptor [OFCD02] is used for indexing and classification.

10.2. Use of eigenvectors

In this section, we survey eigenvector-based methods and classify them according to the dimensionality of the spectral embeddings used.

10.2.1. 1D embedding

Embedding in one dimension consists in producing a linear sequence of mesh elements based on the order given by the entries of one eigenvector. The Fiedler vector has been extensively used in a number of different applications to obtain a linear ordering of mesh vertices or faces. However, other applications also select different eigenvectors to obtain a linear sequencing of mesh elements.

- **Sparse matrix reordering:**

  Barnard et al. [BPS93] use the Fiedler vector to reduce the *envelope* of sparse matrices. By reordering a matrix and optimizing its envelope, the locality of its elements is increased and the the resulting matrix will become more "banded". Several numerical algorithms can improve their performance when applied to a reordered matrix. The Fiedler vector is selected due to its property of minimizing the 2-sum in a continuous relaxation of the problem [MP93].

- **Mesh sequencing:**

  Isenburg and Lindstrom [IL05] introduce the concept of *streaming meshes*. The idea is to process very large meshes that do not fit in main memory by streaming its components, i.e., transferring blocks of vertices and faces in an incremental manner from the hard disk to main memory and back. In this sense, it is highly desirable that the order in which the vertices and faces are traversed preserves neighboring relations the most, so that only a small number of mesh elements have to be maintained simultaneously in main memory. Therefore, one of the possibilities to obtain such a streaming sequence is also to employ the Fiedler vector to order the vertices and faces of the mesh, which provides a linear sequence that heuristically minimizes *vertex separation*.

  In this context of obtaining a good ordering of mesh elements, Liu et al. [LZvK06] investigate how the embeddings given by an eigenvector of an affinity matrix differ from the ones given by the Laplacian matrix or other

possible heuristics. These embeddings are evaluated in terms of various measures related to the interdependence of mesh elements, the best known ones being *span* and *width* of vertices and faces. Experiments show that the embeddings given by the affinity matrix provide a better trade-off between these two measures, when compared to other approaches.

- **Image segmentation:**

  In the case of image segmentation, instead of the Fiedler vector, the eigenvector related to the largest eigenvalue of an affinity matrix has also been used in a great number of works, since it essentially provides a heuristic method for obtaining satisfactory graph partitions [ST96]. Weiss [Wei99] presents a unified view of four well-known methods that follow this approach, such as the work of Shi and Malik [SM00] and Perona and Freeman [PF98].

  These methods firstly define an affinity matrix W based on the distances between pixels in the image, which are seen as nodes in a graph. Next, the eigenvector related to the eigenvalue that is largest in magnitude or, equivalently, the second smallest eigenvector, in the case of the method of Shi and Malik [SM00], is computed. In the latter case, the matrix considered is the non-sparse Laplacian $D-W$. Finally, the entries of this specific eigenvector are used to convey a partition of the pixels into two groups. Either the signs of the entries of the eigenvector or a thresholding of these entries is used to obtain a binary partition of the pixels. The essence of this approach is that this specific eigenvector is a continuous solution to the discrete problem of minimizing the normalized cut between two sets of nodes in a graph.

- **Spectral clustering for surface reconstruction:**

  Kolluri et al. [KSO04] follow basically the same approach for the reconstruction of surfaces from point clouds. After constructing a Delaunay tetrahedralization based on the input points, the tetrahedra are divided into two sets by a spectral graph partitioning method, which provides the indication of which tetrahedra are inside of the original object and which are outside. Finally, this labeling of tetrahedra defines a watertight surface. The partitioning of the tetrahedra is also given by the signs of the entries of the smallest eigenvector of a *pole matrix*, which is similar to a Laplacian.

- **Mesh segmentation:**

  In a different setting, Zhang and Liu [ZL05] propose a mesh segmentation approach based on a recursive 2-way spectral clustering method. An affinity matrix encodes distances between mesh faces, which are a combination of geodesic and angular distances, that provide information for a visually meaningful segmentation. Next, only the first two largest eigenvectors are computed. This provides a one-dimensional embedding of faces given by the quotient of the entries of the two eigenvectors. Finally, the most salient cut in this embedding is located, given by a part salience measure. The cut provides a segmentation of the faces into two parts. This process is recursively repeated in order to obtain a hierarchical binary partitioning of the mesh.

- **Spectral mesh quadrangulation:**

  Dong et al. [DBG*06] propose to use one specific eigenvector of the geometric Laplacian to guide a remeshing process. Firstly, a suitable eigenfunction of the mesh has to be selected, which possesses a desired number of critical points. The critical points are points of minima, maxima, or saddle points. Next, the *Morse-Smale complex* is extracted from the mesh based on this eigenvector. The complex is obtained by connecting critical points with lines of steepest ascend/descend, and partitions the mesh into rectangular patches, which are then refined and parameterized, conveying a quadrangular remeshing of the original model. One of the key points of this method is in selecting an eigenfunction that provides an adequate partition of the mesh. This is achieved by computing all the eigenvectors of a simplified version of the mesh, choosing from these eigenvectors the most suitable for the remeshing task, and then computing the corresponding eigenvector in the full-resolution mesh by applying a spectral shift to the underlying matrix.

10.2.2. 2D and 3D embeddings

Instead of using only one eigenvector given by the eigendecomposition of a specific operator, the next possible step is to use two or three eigenvectors, to obtain a planar or three-dimensional embedding of the mesh.

- **Graph drawing:**

  Methods that provide such embeddings have been successfully applied in the field of graph drawing, where the main goal is to obtain a disposition of nodes and edges on a plane or volume which looks organized and is aesthetically pleasing. Early graph drawing algorithms already proposed to use the Laplacian matrix, as in the work of Tutte [Tut63], whose method actually dates back to the work of Fáry [Far48]. However, in Tutte's method, the eigenvectors of the Laplacian matrix are not computed. Instead, the positions of the nodes of a graph are obtained by solving a linear system based on this matrix.

  Hall [Hal70] later proposed to use the eigenvectors of the Laplacian matrix to embed the nodes of a graph in a space of arbitrary dimension. The entries of the k eigenvectors related to the first smallest non-zero eigenvalues are used as the coordinates of a node (a k-dimensional embedding). This method has been recently applied in different domains to provide planar embeddings of graphs. For example, Pisanski and Shawe-Taylor [PST00] use this method to obtain pleasing drawings of symmetrical graphs, such as fullerene molecules in chemistry. Koren et al. [KCH02, Kor03] employ the ACE algorithm

(Section 9.2) to accelerate Hall's method in order to obtain drawings of large graphs.

- **Planar mesh parameterization via MDS:**

  Zigelman et al. [ZKK02] use a variant of MDS to obtain a planar mesh embedding and then map a given texture on this planar surface. In this work, a geodesic distance matrix is computed by fast marching. Next, the matrix is centered and its eigendecomposition is computed. The two eigenvectors which are related to the two largest eigenvalues are used to provide a planar embedding of the mesh. The flattening thus obtained heuristically minimizes distortions, which is desirable for texture mapping. However, the absence of triangle fold-overs is not guaranteed.

  Following a similar approach, Zhou et al. [ZSGS04] employ MDS based on geodesic distances to obtain a parameterization and then a chartification of a given mesh. By growing patches around representative points, which are determined according to the spectral embedding, the mesh is divided into charts. The representatives that are selected are points that are well-spaced over the mesh and that are also points of minima or maxima, according to their coordinates in the spectral embedding.

- **MDS for mesh segmentation:**

  MDS is also used by Katz et al. [KLT05] in mesh segmentation, due to its potential of obtaining a *pose-invariant* embedding [EK03]. Three eigenvectors are selected to obtain a 3D embedding. Next, feature points are located in this normalized space, which guide the segmentation algorithm that partitions the mesh into meaningful parts.

- **Spherical mesh parameterization:**

  Gotsman et al. [GGS03] describe how a spherical parameterization of a mesh can be obtained from the eigenvectors of *Colin de Verdière* matrices. Each graph has a class of these matrices associated with it. By using the entries of three eigenvectors of such a matrix as the coordinates of the mesh vertices, a valid spherical embedding is obtained. By solving a non-linear system based on a Laplacian or a similar operator, a Colin de Verdière matrix is generated and the three eigenvectors that give the valid embedding (which are associated to repeated eigenvalues) are simultaneously computed.

- **Spectral conformal parameterization:**

  Mullen et al. [MTAD08] have brought spectral techniques to the task of efficiently computing a quality conformal parameterization of a surface mesh patch. A problem with previous linear methods for conformal parameterization is that a couple of boundary vertices needed to be fixed in order to avoid the trivial solution when minimizing the conformal energy. The quality of the resulting parameterization can depend significantly on the choice of these constraint vertices. The insight of Mullen et al. is that the Fiedler vector of a well crafted generalized eigenvalue problem yields an appropriate solution without the need to explicitly constrain specific boundary points. They find such a Fiedler vector from the equation

$$L_C \mathbf{u} = \lambda B \mathbf{u},$$

  where $\mathbf{u}$ is the parameterization being sought, L_C is the quadratic form for the conformal energy, and B is a degenerate diagonal binary matrix whose nonzero entries correspond to the boundary vertices. An appeal to the Courant-Fisher theorem (Theorem 5.3) reveals that the Fiedler vector seeks to minimize the conformal energy, while appropriately constraining the boundary vertices.

- **Mesh segmentation:**

  Recently, Liu and Zhang [LZ07] proposed an algorithm for mesh segmentation via recursive bisection where at each step, a sub-mesh embedded in 3D is spectrally projected to 2D and then a contour is extracted from the planar embedding. Two operators are used in combination to compute the projection: the well-known graph Laplacian and a geometric operator designed to emphasize concavity. The two embeddings reveal distinctive shape semantics of the 3D model and complement each other in capturing the structural or geometrical aspects of a segmentation. Transforming the shape analysis problem to the 2D domain also facilitates segmentability analysis and sampling, where the latter is needed to identify two samples residing on different parts of the sub-mesh. These two samples are used by the Nyström method in the construction of a 1D face sequence for finding an optimal cut, as in [ZL05].

10.2.3. Higher-dimensional embedding

Since a set of n eigenvectors can be obtained from the eigendecomposition of an $n \times n$ matrix, it is natural to intend to use more eigenvectors simultaneously to extract more information from the eigendecomposition.

- **Clustering and mesh segmentation:**

  One of the most well-known techniques in this regard is spectral clustering [BN03, KVV00, NJW02]. Interested readers should refer to the recent survey by von Luxburg [vL06] and the comparative study by Verma and Meilă [VM03]. The approach by Ng et al. [NJW02] has already been outlined in Section 8.2. Other approaches differ only slightly from the core solution paradigm, e.g., in terms of the operator used and the dimensionality of the embedding. Some works, e.g., [AKY99, NJW02, VM03], seem to suggest that clustering based on multiple eigenvectors tends to produce better results compared with recursive approaches using individual eigenvectors.

  Although the reasons for the empirical success of spectral clustering are still not fully understood, Ng et al. [NJW02] provide an analysis in terms of matrix perturbation theory to show that the algorithm is expected to work well even in situations where the cluster structure in the input data is far from an ideal case. There are

other possible interpretations of spectral clustering, e.g., in terms of graph cuts or random walks [vL06].

The ubiquity of the clustering problem makes spectral clustering an extremely useful technique. Besides the work of Kolluri et al. [KSO04] mentioned in Section 10.2.1, Liu and Zhang [LZ04] perform mesh segmentation via spectral clustering. Basically, an affinity matrix is constructed as in [ZL05]. Next, the eigenvectors given by the eigendecomposition of this matrix guide a clustering method, which provides patches of faces that define the different segments of the mesh returned by the segmentation algorithm. It is shown by example that it can be advantageous to perform segmentation in the spectral domain, e.g., in terms of higher-quality cut boundaries as evidenced by the Polarization Theorem (Theorem 5.6 in Section 5). The downside however is the computational cost. In a follow-up work [LJZ06], Nyström approximation is applied to speed-up spectral mesh segmentation.

Recently, de Goes et al. [dGGV08] present a hierarchical segmentation method for articulated bodies. Their approach relies on the diffusion distance, which is a multi-scale metric based on the heat kernel (Section 2.1) and computed from the eigenvectors of the Laplace-Beltrami operator. The diffusion distance is used to compute a bijection between medial structures and segments of the model. The medial structures yield a means to further refine the segmentation in an iterative manner and provide a full hierarchy of segments for the shape.

Huang et al. [HWAG09] also perform hierarchical shape decomposition via spectral analysis. However, the operator they use encapsulates shape geometry beyond the static setting. The idea is to define a certain deformation energy and use the eigenvectors of the Hessian of the deformation energy to characterize the space of possible deformations of a given shape. The *eigenmodes* corresponding to the low-end of the spectrum of the Hessian capture low-energy or in their formulation, more rigid deformations, called "typical" deformations. The optimal shape decomposition they compute is one whose optimal articulated (piecewise rigid) deformation defined on the parts of the decomposition best conforms to the basis vectors of the space of typical deformations. As a result, their method tends to identify parts of a shape which would likely remain rigid during the "typical" deformations.

- **Shape correspondence and retrieval:**

  Elad and Kimmel [EK03] use MDS to compute bending-invariant signatures for meshes. Geodesic distances between mesh vertices are computed by fast marching. The resulting spectral embedding effectively normalizes the mesh shapes with respect to translation, rotation, as well as bending transformations. The similarity between two shapes is then given by the Euclidean distance between the moments of the first few eigenvectors, usually less than 15, and these similarity distances can be used for shape classification.

  Inspired by works in computer vision on spectral point correspondence [SB92], Jain and Zhang [JZvK07] rely on higher-dimensional embeddings based on the eigenvectors of an affinity matrix to obtain point correspondence between two mesh shapes. The first k eigenvectors of the affinity matrix encoding the geodesic distances between pairs of vertices are used to embed the model in a k-dimensional space; typically $k = 5$ or 6. After this process is performed for two models, the two embeddings are non-rigidly aligned via thin-plate splines and the correspondence between the two shapes is given by the proximity of the vertices after such alignment. Any measure for the cost of a correspondence, e.g., sum of distances between corresponding vertices, can be used as a similarity distance for shape retrieval.

  One of the key observations made in [JZvK07] is the presence of "eigenvector switching" due to non-uniform scaling of the shapes. Specifically, the eigenmodes of similar shapes do not line up with respect to the magnitude of their corresponding eigenvalues, as illustrated in Figure 12. As a result, it is unreliable to sort the eigenvectors according to the magnitude of their respective eigenvalues, as has been done in all works on spectral correspondence so far. Jain and Zhang [JZvK07] rely on a heuristic to "unswitch" the eigenmodes and thin-plate splines to further align the shapes in the spectral domain [JZvK07]. Recent work of Mateus et al. [MCBH07] addresses the issue using an alignment by the EM algorithm instead.

  The method by Leordeanu and Hebert [LH05] focuses on the global characteristics of correspondence computation and aims at finding *consistent correspondences* between two sets of shape or image features, where the spectral approach has also found its utility. They build a graph whose nodes represent possible feature pairings and edge weights measure how agreeable the pairings are. The principal eigenvector of an affinity matrix W, one corresponding to the largest eigenvalue, is inspected to detect how strongly the graph nodes belong to the main cluster of W. The idea is that correct feature correspondences are likely to establish links among each other and thus form a strongly connected cluster.

- **Graph matching:**

  Generally speaking, graphs are commonly used to model shape structures, e.g., skeletal graphs [SSGD03], shock graphs, or Reeb graphs [HSKK01]. The subsequent graph matching problem is well studied in the computer vision community, where a number of spectral approaches have been proposed, e.g., [Ume88, CK04], adding a geometric flavor to the problem. As a basic framework, a graph adjacency matrix, which may only encode topological information, is eigendecomposed, whereby the graph nodes are

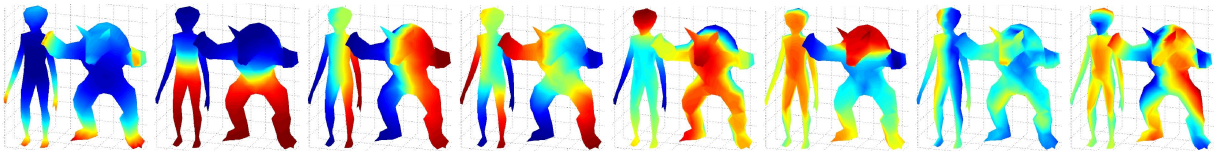


Figure 12: *Eigenvector plots for two similar shapes, both with 252 vertices. The entries in an eigenvector are color-mapped. As we can see, there is an "eigenvector switching" occurring between the fifth and sixth eigenvectors. Such "switching" is difficult to detect from the magnitude of the eigenvalues. The first 8 eigenvalues for the two shapes are [205.6, 11.4, 4.7, 3.8, 1.8, 0.4, 0.26, 0.1] and [201.9, 10.9, 6.3, 3.4, 1.8, 1.2, 0.31, 0.25], respectively.*

mapped into a low-dimensional vector space. The matching problem is solved in the embedding space.

- **Global intrinsic symmetry detection:**

  Ovsjanikov et al. [OSG08] propose an approach to detect the intrinsic symmetries of a shape which are invariant up to isometry preserving transformations. They show that if the shape is embedded into the signature space defined by the eigenfunctions of the Laplace-Beltrami operator, then the intrinsic symmetries are transformed into extrinsic Euclidean symmetries (rotations or reflections). However, it is possible to restrict the search of symmetries only to reflections, avoiding the search of rotational symmetries, a task that can be hard in high-dimensional space. This result allows to obtain the intrinsic symmetries by first computing the eigenvalues of the operator, then embedding the shape into the signature space, and finally finding point-to-point correspondences of symmetric points.

10.3. Use of eigenprojections

Instead of directly using the entries of the eigenvectors to provide an embedding for a given model, the eigenvectors can also be used as a basis to transform signals defined on the vertices of the mesh. One example of such a signal is the actual geometry of the mesh (the 3D coordinates of its vertices). The set of eigenvectors given by the eigendecomposition can be used to project these signals into the spectral space, where a specific problem might be easier to solve.

- **Geometry compression:**

  Karni and Gotsman [KG00] propose an approach to compress the geometry of triangle meshes. Firstly, the set of eigenvectors of the Tutte Laplacian is computed. Next, the mesh vertex coordinates are projected into the spectral space spanned by the computed eigenvectors. Part of the coefficients obtained by this transformation is eliminated in order to reduce the storage space required for mesh geometry. The coefficients related to the eigenvectors associated to larger eigenvalues are firstly removed, which would correspond to high frequency detail, when following an analogy with Fourier analysis.

  The main drawback of this method is that many eigenvectors need to be computed. Karni and Gotsman propose to partition the mesh into smaller sets of vertices. Although that alleviates the problem of computing the eigenvectors for large matrices, it still requires a good partitioning of the mesh for the efficiency of the compression, and artifacts along the partition boundaries are evident when higher compression rates are employed.

- **Watermarking:**

  Ohbuchi et al. [OTMM01, OMT02] also employ the eigenprojection approach, but to insert watermarks into triangle meshes. In this method, the eigenvectors of the *Kirchhoff* operator are used as the basis for the projection. After transforming the geometry of the mesh and obtaining the spectral coefficients, a watermark is inserted into the model by modifying coefficients at the low-frequency end of the spectrum. In this way, modifications on the geometry of the mesh are well-spread over the model and less noticeable than if they were directly added to the vertex coordinates. This method also requires the computation of eigenvectors of the Laplacian operator, which is prohibitive in the case of large meshes. Mesh partitioning is again used to address this problem.

- **Fourier descriptors:**

  2D Fourier descriptors have been quite successful as a means to characterize 2D shapes. Using eigendecomposition with respect to the the mesh Laplacians, one can compute analogous Fourier-like descriptors to describe mesh geometry. However, we have not seen such mesh Fourier descriptors being proposed for shape analysis so far. There have been methods, e.g., [VS01], which rely on 3D Fourier descriptors for 3D shape retrieval. In this case, the mesh shapes are first voxelized and 3D Fourier descriptors are extracted from the resulting volumetric data. We suspect that the main difficulties with the use of mesh Fourier descriptors for shape matching include computational costs and the fact that when the eigenmodes vary between the two mesh shapes to be matched, it becomes doubtful whether their associated eigenspace projections can serve as reliable shape descriptors. Also, even when the shapes are very similar, eigenvector switching, as depicted in Figure 12, can occur when the eigenvectors are ordered by the magnitude of their eigenvalues.

11. Summary and discussions

In this paper, we describe, motivate, and classify spectral methods for mesh processing and analysis. Related and representative developments from other fields, e.g., computer vision and machine learning, are also covered. Necessary theoretical background and illustrative examples are both provided to facilitate understanding of the various concepts. Finally, we give a detailed survey of specific spectral methods developed to solve a diversity of problems.

From a theoretical standpoint, we are still missing an adequate sampling theory for signals defined over 2-manifolds. We envision this theory to be one whose results and analysis tools resemble those from the theory of sampling and reconstruction in the regular setting using Fourier transforms. Fundamental questions concerning the proper definition of concepts such as frequencies, band-limited surfaces, shift-invariance, etc., should be addressed.

Take the concept of frequency for example. Our general belief is that eigenvalues of the mesh Laplacian represent (squared) frequencies. However, eigenvalues are only able to indicate global properties of the manifold or global properties of the associated eigenfunctions. The variability of eigenfunctions having the same or similar eigenvalues implies that eigenvalues alone cannot provide sufficient characterization of their related eigenfunctions. This has been the case when we seek a proper ordering of the eigenvectors in order to facilitate robust spectral shape correspondence [JZvK07]. The situation described here differs from the classical case of the two dimensional Fourier transform where the eigenfunctions are catalogued by two frequency values (corresponding to the x and y directions), and the canonical Fourier basis functions resolve the ambiguity inherent in decomposing an eigenspace corresponding to a degenerate eigenvalue.

Ideally, we would like to find additional characteristic measures for the eigenfunctions. This, for example, might help us more easily in locating the right eigenvector for deriving a high-quality surface quadrangulation automatically [DBG*06]. As Lévy [Lé06] has eloquently put it, Laplace-Beltrami eigenfunctions (or eigenfunctions of other geometric mesh operators) appear to "understand" geometry. However, it is not necessarily easy to interpret what the eigenfunctions are presenting to us. A better understanding of the eigenvectors and how they relate to the shape of the underlying manifolds would certainly spark new research and allow for improvements in the spectral methods.

Other theoretical studies concerning mesh operators and their eigenstructures include convergence analyses for geometric mesh Laplacians, e.g., Hildebrandt et al. [HPW06] and Xu [Xu04b], analyses on the sensitivity of the eigenstructures against shape or connectivity variations, e.g., Dyer et al. [DZM07], as well as studies on sampling for Nyström approximation. In this setting, as for the development of a sampling theory, we are concerned with the interplay between the continuous and the discrete settings. Also of interest is the robustness of the eigenstructures of different operators under topological changes.

A generalization of the mesh Laplacian operators to Schrödinger operators introduces a new class of possible operators. However, it is not clear how to easily construct specific instances of these operators in an efficient manner, e.g., the Colin de Verdière matrices for spherical mesh parameterization [GGS03], or to explicitly design such an operator having the property that it is optimal for a specific application, e.g., compression or segmentation. Recent development on discrete exterior calculus [Hir03] may also shed light on what other possible discrete operators can be suitable for mesh processing.

Another wide avenue for further research is the study of the theoretical aspects of spectral clustering algorithms. First of all, the reason for the good results obtained by these algorithms is still not completely understood. Fortunately there exist a number of studies and analyses which elucidate certain properties responsible for the exceptional behavior of these algorithms, e.g., [vL06]. These studies might serve as a starting point to explain the functioning of spectral clustering and lead to ideas for more complete explanations. Additionally, other aspects, such as how to select the dimensionality of the spectral embeddings or how to construct affinity matrices more suitable for specific applications, e.g., for proper handling of part stretching in shape characterization, still require further attention.

Acknowledgments: This research is supported in part by an NSERC Discovery Grant (No. 611370) and an MITACS research grant (No. 699127). Mesh models used in the paper come from various sources; these include the McGill 3D Shape Benchmark Database, the AIM@SHAPE shape repository, Cyberware Inc., and the Stanford 3D Scanning Repository, with credit to the Stanford Computer Graphics Laboratory. The bunny, horse, and Max Planck models used in the paper were produced by the QSlim mesh simplification program written by Michael Garland. We would like to thank Rong Liu for help in producing some of the images in the paper. Finally, the authors would like to extend their gratitude to the editor, for his hard work and patience, and the three anonymous reviewers for their thorough, well-thought, and sometimes inspiring comments on the paper.

References

[AKY99] ALPERT C. J., KAHNG A. B., YAO S. Z.: Spectral partitioning with multiple eigenvectors. *Discrete Applied Mathematics 90* (1999), 3–26.

[BCG05] BEN-CHEN M., GOTSMAN C.: On the optimality of spectral compression of mesh data. *ACM Trans. on Graphics 24*, 1 (2005), 60–80.

[BDLR*04] BENGIO Y., DELALLEAU O., LE ROUX N., PAIEMENT J.-F., VINCENT P., OUIMET M.: Learning eigenfunctions links spectral embedding and kernel PCA. *Neural Computation 16*, 10 (2004), 2197–2219.

[BH03] BRAND M., HUANG K.: A unifying theorem for spectral embedding and clustering. In *Proc. of Int. Conf. on AI and Stat.* (Key West, Florida, 2003).

[Bha97] BHATIA R.: *Matrix Analysis*. Springer-Verlag, 1997.

[BHL*04] BIYIKOĞLU T., HORDIJK W., LEYDOLD J., PISANSKI T., STADLER P. F.: Graph Laplacians, nodal domains, and hyperplane arrangements. *Linear Algebra and Its Applications 390* (2004), 155–174.

[BN03] BELKIN M., NIYOGI P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. *Neural Computation 15*, 6 (2003), 1373–1396.

[BN05] BELKIN M., NIYOGI P.: Towards a theoretical foundation for Laplacian-based manifold methods. In *Proc. of Conference on Learning Theory, Lecture Notes on Comput. Sci.* (2005), vol. 3559, pp. 486–500.

[Bol98] BOLLOBÁS B.: *Modern Graph Theory*. Springer, 1998.

[BPS93] BARNARD S. T., POTHEN A., SIMON H. D.: A spectral algorithm for envelope reduction of sparse matrices. In *Proc. of ACM Conference on Supercomputing* (1993), pp. 493–502.

[CC94] COX T. F., COX M. A. A.: *Multidimensional Scaling*. Chapman & Hall, 1994.

[Cha84] CHAVEL I.: *Eigenvalues in Riemannian Geometry*. Academic Press, 1984.

[Chu97] CHUNG F. R. K.: *Spectral Graph Theory*. AMS, 1997.

[CK04] CAELLI T., KOSINOV S.: An eigenspace projection clustering method for inexact graph matching. *IEEE Trans. Pat. Ana. & Mach. Int. 26*, 4 (2004), 515–519.

[CTSO03] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On visual similarity based 3D model retrieval. *Computer Graphics Forum 22*, 3 (2003), 223–232.

[DBG*06] DONG S., BREMER P.-T., GARLAND M., PASCUCCI V., HART J. C.: Spectral surface quadrangulation. In *SIGGRAPH* (2006), pp. 1057–1066.

[dGGV08] DE GOES F., GOLDENSTEIN S., VELHO L.: A hierarchical segmentation of articulated bodies. *Computer Graphics Forum (Symposium on Geometry Processing) 27*, 5 (2008), 1349–1356.

[DGLS01] DAVIES E. B., GLADWELL G. M. L., LEYDOLD J., STADLER P. F.: Discrete nodal domain theorems. *Lin. Alg. Appl. 336* (2001), 51–60.

[DHLM05] DESBRUN M., HIRANI A., LEOK M., MARSDEN J.: Discrete exterior calculus. (preprint, arXiv:math.DG/0508341), 2005.

[DKT06] DESBRUN M., KANSO E., TONG Y.: Discrete differential forms for computational modeling. In *SIGGRAPH Courses Notes* (2006), pp. 39–54.

[DMSB99] DESBRUN M., MEYER M., SCHRÖDER P., BARR A. H.: Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH* (1999), pp. 317–324.

[DPS02] DÍAZ J., PETIT J., SERNA M.: A survey of graph layout problems. *ACM Computing Survey 34*, 3 (2002), 313–356.

[dST04] DE SILVA V., TENENBAUM B.: *Sparse Multidimensional Scaling using Landmark Points*. Tech. rep., Stanford University, June 2004.

[dV90] DE VERDIÈRE Y. C.: Sur un nouvel invariant des graphes et un critére de planarité. *Journal of Combinatorial Theory, Series B 50* (1990), 11–21.

[DZM07] DYER R., ZHANG H., MÖLLER T.: *An Investigation of the Spectral Robustness of Mesh Laplacians*. Tech. rep., Simon Fraser University, June 2007.

[EK03] ELAD A., KIMMEL R.: On bending invariant signatures for surfaces. *IEEE Trans. Pat. Ana. & Mach. Int. 25*, 10 (2003), 1285–1295.

[EY36] ECKART C., YOUNG G.: The approximation of one matrix by another of lower rank. *Psychometrika 1* (1936), 211–218.

[Far48] FARY I.: On straight line representations of planar graphs. *Acta Sci. Math. 11* (1948), 229–233.

[FBCM04] FOWLKES C., BELONGIE S., CHUNG F., MALIK J.: Spectral grouping using the Nyström method. *IEEE Trans. Pat. Ana. & Mach. Int. 26*, 2 (2004), 214–225.

[Fie73] FIEDLER M.: Algebraic connectivity of graphs. *Czech. Math. J. 23* (1973), 298–305.

[Flo03] FLOATER M. S.: Mean value coordinates. *Comput. Aided Geom. Des. 20*, 1 (2003), 19–27.

[Fuj95] FUJIWARA K.: Eigenvalues of Laplacians on a closed Riemannian manifold and its nets. *Proceedings of the AMS 123*, 8 (1995), 2585–2594.

[GGS03] GOTSMAN C., GU X., SHEFFER A.: Fundamentals of spherical parameterization for 3d meshes. *ACM Trans. on Graphics 22*, 3 (2003), 358–363.

[Gli05] GLICKENSTEIN D.: Geometric triangulations and discrete Laplacians on manifolds, 2005. arxiv:math.MG/0508188.

[GM00] GUATTERY S., MILLER G. L.: Graph embeddings and Laplacian eigenvalues. *SIAM J. Matrix Anal. Appl. 21*, 3 (2000), 703–723.

[Got03] GOTSMAN C.: On graph partitioning, spectral analysis, and digital mesh processing. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications* (2003), pp. 165–171.

[Hal70] HALL K. M.: An r-dimensional quadratic placement algorithm. *Manage. Sci. 17*, 8 (1970), 219–229.

[Hir03] HIRANI A. N.: *Discrete Exterior Calculus*. PhD thesis, Caltech, 2003.

[HLMS04] HAM J., LEE D. D., MIKA S., SCHÖLKOPF B.: A kernel view of the dimensionality reduction of manifolds. In *Proc. of Int. Conf. on Machine learning* (2004), pp. 47–54.

[HPW06] HILDEBRANDT K., POLTHIER K., WARDETZKY M.: On the Convergence of Metric and Geometric Properties of Polyhedral Surfaces. *Geometriae Dedicata 123*, 1 (2006), 89–112.

[HS97] HOFFMAN D. D., SINGH M.: Salience of visual parts. *Cognition 63* (1997), 29–78.

[HSKK01] HILAGA M., SHINAGAWA Y., KOHMURA T., KUNII T. L.: Topology matching for fully automatic similarity estimation of 3D shapes. In *SIGGRAPH* (2001), pp. 203–212.

[HWAG09] HUANG Q., WICKE M., ADAMS B., GUIBAS L. J.: Shape decomposition using modal analysis. *Computer Graphics Forum (Eurographics) 28*, 2 (2009), to appear.

[IGG01] ISENBURG M., GUMHOLD S., GOTSMAN C.: Connectivity shapes. In *Proc. of IEEE Visualization* (2001).

[IL05] ISENBURG M., LINDSTROM P.: Streaming meshes. In *Proc. of IEEE Visualization* (2005), pp. 231–238.

[Jai89] JAIN A. K.: *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.

[JNT01] JAKOBSON D., NADIRASHVILI N., TOTH J.: Geometric properties of eigenfunctions. *Russian Mathematical Surveys 56*, 6 (2001), 1085–1105.

[JZ06] JAIN V., ZHANG H.: Robust 3D shape correspondence in the spectral domain. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications* (2006), pp. 118–129.

[JZ07] JAIN V., ZHANG H.: A spectral approach to shape-based retrieval of articulated 3D models. *Computer Aided Design 39* (2007), 398–407.

[JZvK07] JAIN V., ZHANG H., VAN KAICK O.: Non-rigid spectral correspondence of triangle meshes. *Int. J. on Shape Modeling 13*, 1 (2007), 101–124.

[KCH02] KOREN Y., CARMEL L., HAREL D.: ACE: A fast multiscale eigenvector computation for drawing huge graphs. In *Proc. of IEEE Symposium on Information Visualization (InfoVis)* (2002), pp. 137–144.

[KG00] KARNI Z., GOTSMAN C.: Spectral compression of mesh geometry. In *SIGGRAPH* (2000), pp. 279–286.

[KG01] KARNI Z., GOTSMAN C.: 3D mesh compression using fixed spectral bases. In *Proc. of Graphics Interface* (June 2001), pp. 1–8.

[KLT05] KATZ S., LEIFMAN G., TAL A.: Mesh segmentation using feature point and core extraction. *The Visual Computer (Special Issue of Pacific Graphics) 21*, 8-10 (2005), 649–658.

[Kor03] KOREN Y.: On spectral graph drawing. In *Proc. of the International Computing and Combinatorics Conference* (2003), pp. 496–508.

[KR05] KIM B. M., ROSSIGNAC J.: Geofilter: Geometric selection of mesh filter parameters. *Computer Graphics Forum (Eurographics) 24*, 3 (2005), 295–302.

[KSO04] KOLLURI R., SHEWCHUK J. R., O'BRIEN J. F.: Spectral surface reconstruction from noisy point clouds. In *Proc. Eurographics Symp. on Geometry Processing* (2004), pp. 11–21.

[KVV00] KANNAN R., VEMPALA S., VETTA A.: On clustering – good, bad, and spectral. In *IEEE Symposium on Foundations of Computer Science* (2000), pp. 367–377.

[LÓ6] LÉVY B.: Laplace-Beltrami eigenfunctions: Towards an algorithm that understands geometry. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications* (2006), pp. 13–20.

[LH05] LEORDEANU M., HEBERT M.: A spectral technique for correspondence problems using pairwise constraints. In *Proc. Int. Conf. on Comp. Vis.* (October 2005), vol. 2, pp. 1482–1489.

[LJZ06] LIU R., JAIN V., ZHANG H.: Subsampling for efficient spectral mesh processing. In *Proc. of Computer Graphics International* (2006), pp. 172–184.

[Lov93] LOVÁSZ: Random walks on graphs: a survey. In *Combinatorics, Paul Erdös is eighty*, Keszthely, (Ed.), vol. 2. Budapest: János Bolyai Math. Soc., 1993, pp. 353–397.

[LS99] LOVÁSZ L., SCHRIJVER A.: On the null space of a Colin de Verdière matrix. *Annales de l'institut Fourier 49*, 3 (1999), 1017–1026.

[LZ04] LIU R., ZHANG H.: Segmentation of 3D meshes through spectral clustering. In *Prof. of Pacific Graphics* (2004), pp. 298–305.

[LZ07] LIU R., ZHANG H.: Mesh segmentation via spectral embedding and contour analysis. *Computer Graphics Forum (Eurographics) 26* (2007), 385–394.

[LZSCO09] LIU R., ZHANG H., SHAMIR A., COHEN-OR D.: A part-aware surface metric for shape analysis. *Computer Graphics Forum (Eurographics) 28*, 2 (2009), to appear.

[LZvK06] LIU R., ZHANG H., VAN KAICK O.: *An Investigation into Spectral Sequencing using Graph Distance*. Tech. Rep. TR 2006-08, School of Computing Science, SFU, May 2006.

[MCBH07] MATEUS D., CUZZOLIN F., BOYER E., HORAUD R.: Articulated shape matching by robust alignment of embedded representations. In *ICCV '07 Workshop on 3D Representation for Recognition* (2007).

[McG] McGill 3D Shape Benchmark: http://www.cim.mcgill.ca/~shape/benchMark/.

[MDSB02] MEYER M., DESBRUN M., SCHRÖDER P., BARR A. H.: Discrete differential-geometry operators for triangulated 2-manifolds. In *Proc. of VisMath* (2002), pp. 35–57.

[Mey00] MEYER C. D.: *Matrix analysis and applied linear algebra*. SIAM, 2000.

[MKR03] M. KAZHDAN T. F., RUSINKIEWICZ S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Proc. Eurographics Symp. on Geometry Processing* (2003), pp. 156–165.

[Moh97] MOHAR B.: Some applications of Laplacian eigenvalues of graphs. In *Graph Symmetry: Algebraic Methods and Applications*, Hahn G., Sabidussi G., (Eds.). Kluwer, 1997, pp. 225–275.

[MP93] MOHAR B., POLJAK S.: *Eigenvalues in Combinatorial Optimization*, vol. 50 of *IMA Volumes in Mathematics and Its Applications*. Springer-Verlag, 1993, pp. 107–151.

[MP04] MANOR L., PERONA P.: Self-tuning spectral clustering. In *Advances in Neural Information Processing Systems (NIPS)* (2004), vol. 17, pp. 1601–1608.

[MTAD08] MULLEN P., TONG Y., ALLIEZ P., DESBRUN M.: Spectral conformal parameterization. *Computer Graphics Forum (Symposium on Geometry Processing) 27*, 5 (2008), 1487–1494.

[NJW02] NG A. Y., JORDAN M. I., WEISS Y.: On spectral clustering: analysis and an algorithm. In *Advances in Neural Information Processing Systems (NIPS)* (2002), vol. 14, pp. 849–856.

[OFCD02] OSADA R., FUNKHOUSER T., CHAZELLE B., DOBKIN D.: Shape distributions. *ACM Trans. on Graphics 21*, 4 (2002), 807–832.

[OMT02] OHBUCHI R., MUKAIYAMA A., TAKAHASHI S.: A frequency-domain approach to watermarking 3D shapes. *Computer Graphics Forum 21*, 3 (2002), 373–382.

[OSG08] OVSJANIKOV M., SUN J., GUIBAS L.: Global intrinsic symmetries of shapes. *Computer Graphics Forum (Symposium on Geometry Processing) 27*, 5 (2008), 1341–1348.

[OTMM01] OHBUCHI R., TAKAHASHI S., MIYAZAWA T., MUKAIYAMA A.: Watermarking 3D polygonal meshes in the mesh spectral domain. In *Proc. of Graphics Interface* (2001), pp. 9–18.

[PF98] PERONA P., FREEMAN W.: A factorization approach to grouping. In *Proc. Euro. Conf. on Comp. Vis.* (1998), pp. 655–670.

[PG01] PAULY M., GROSS M.: Spectral processing of point-sampled geometry. In *SIGGRAPH* (2001), pp. 379–386.

[PP93] PINKALL U., POLTHIER K.: Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2*, 1 (1993), 15–36.

[PST00] PISANSKI T., SHAWE-TAYLOR J.: Characterizing graph drawing with eigenvectors. *Journal of Chemical Information and Computer Sciences 40*, 3 (2000), 567–571.

[Ros97] ROSENBERG S.: *The Laplacian on a Riemannian Manifold*. Cambridge University Press, 1997.

[RS00] ROWEIS S., SAUL L.: Nonlinear dimensionality reduction by locally linear embedding. *Science 290*, 5500 (2000), 2323–2326.

[Rus07] RUSTAMOV R. M.: Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In *Proc. Eurographics Symp. on Geometry Processing* (2007), pp. 225–233.

[RWP06] REUTER M., WOLTER F.-E., PEINECKE N.: Laplacian-Beltrami spectra as 'shape-DNA' for shapes and solids. *Computer Aided Geometric Design 38* (2006), 342–366.

[SB92] SHAPIRO L. S., BRADY J. M.: Feature-based correspondence: an eigenvector approach. *Image and Vision Computing 10*, 5 (1992), 283–288.

[SCOT03] SORKINE O., COHEN-OR D., TOLEDO S.: High-pass quantization for mesh encoding. In *Proc. Eurographics Symp. on Geometry Processing* (2003), pp. 42–51.

[SF73] STRANG G., FIX G. J.: *An Analysis of the Finite Element Method*. Prentice-Hall, 1973.

[SFYD04] SAERENS M., FOUSS F., YEN L., DUPONT P.: *The Principal Components Analysis of a Graph, and its Relationship to Spectral Clustering*, vol. 3201 of *Lecture Notes in Artificial Intelligence*. Springer, 2004, pp. 371–383.

[SM97] SHI J., MALIK J.: Normalized cuts and image segmentation. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition* (1997), pp. 731–737.

[SM00] SHI J., MALIK J.: Normalized cuts and image segmentation. *IEEE Trans. Pat. Ana. & Mach. Int. 22*, 8 (2000), 888–905.

[SMD*05] SHOKOUFHANDEH A., MACRINI D., DICKINSON S., SIDDIQI K., ZUCKER S.: Indexing hierarchi-

cal structures using graph spectra. *IEEE Trans. Pat. Ana. & Mach. Int. 27*, 7 (2005), 1125–1140.

[Sor05] SORKINE O.: Laplacian mesh processing. In *Eurographics State-of-the-Art Report* (2005).

[SS02] SCHÖLKOPF B., SMOLA A. J.: *Learning with Kernels*. MIT Press, 2002.

[SSGD03] SUNDAR H., SILVER D., GAGVANI N., DICKINSON S.: Skeleton based shape matching and retrieval. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications* (2003), pp. 130–142.

[SSM98] SCHÖLKOPF B., SMOLA A. J., MÜLLER K.-R.: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computations 10* (1998), 1299–1319.

[SSrM98] SCHOLKÖPF B., SMOLA E., ROBERT MÜLLER K.: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation 10* (1998), 1299–1319.

[ST96] SPIELMAN D. A., TENG S.-H.: Spectral partitioning works: Planar graphs and finite element meshes. In *IEEE Symposium on Foundation of Computer Science* (1996), pp. 96–105.

[STWCK05] SHAWE-TAYLOR J., WILLIAMS C. K. I., CRISTIANINI N., KANDOLA J.: On the eigenspectrum of the gram matrix and the generalization error of kernel-PCA. *IEEE Trans. on Information Theory 51*, 7 (2005), 2510–2522.

[Tau95] TAUBIN G.: A signal processing approach to fair surface design. In *SIGGRAPH* (1995), pp. 351–358.

[Tau00] TAUBIN G.: Geometric signal processing on polygonal meshes. In *Eurographics State-of-the-Art Report* (2000).

[TB97] TREFETHEN L. N., BAU D.: *Numerical Linear Algebra*. SIAM, 1997.

[TL00] TENENBAUM J. B., LANGFORD J. C.: A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500 (2000), 2319–2323.

[Tut63] TUTTE W. T.: How to draw a graph. *Proc. London Math. Society 13* (1963), 743–768.

[Ume88] UMEYAMA S.: An eigendecomposition approach to weighted graph matching problems. *IEEE Trans. Pat. Ana. & Mach. Int. 10*, 5 (1988), 695–703.

[vL06] VON LUXBURG U.: *A Tutorial on Spectral Clustering*. Tech. Rep. TR-149, Max Plank Institute for Biological Cybernetics, August 2006.

[VL08] VALLET B., LÉVY B.: Spectral geometry processing with manifold harmonics. *Computer Graphics Forum (Eurographics) 27*, 2 (2008), 251–260.

[vLBB05] VON LUXBURG U., BOUSQUET O., BELKIN M.: Limits of spectral clustering. In *Advances in Neural Information Processing Systems (NIPS)* (2005), vol. 17, pp. 857–864.

[VM03] VERMA D., MEILA M.: *A comparison of spectral clustering algorithms*. Tech. Rep. UW-CSE-03-05-01, University of Washington, 2003.

[VS01] VRANIĆ D. V., SAUPE D.: 3D shape descriptor based on 3D Fourier transform. In *Proc. EURASIP Conf. on Digital Signal Processing for Multimedia Communications and Services* (2001).

[WBH*07] WARDETZKY M., BERGOU M., HARMON D., ZORIN D., GRINSPUN E.: Discrete quadratic curvature energies. *Comput. Aided Geom. Des. 24*, 8-9 (2007), 499–518.

[Wei99] WEISS Y.: Segmentation using eigenvectors: A unifying view. In *Proc. Int. Conf. on Comp. Vis.* (1999), pp. 975–983.

[WMKG07] WARDETZKY M., MATHUR S., KÄLBERER F., GRINSPUN E.: Discrete Laplace operators: no free lunch. In *Proc. Eurographics Symp. on Geometry Processing* (2007), pp. 33–37.

[Xu04a] XU G.: Convergent discrete Laplace-Beltrami operators over triangular surfaces. In *Proc. of Geometric Modeling and Processing* (2004), pp. 195–204.

[Xu04b] XU G.: Discrete Laplace-Beltrami operators and their convergence. *Comput. Aided Geom. Des. 21*, 8 (2004), 767–784.

[ZB04] ZHANG H., BLOK H. C.: Optimal mesh signal transforms. In *Proc. of Geometric Modeling and Processing* (2004), pp. 373–379.

[ZF03] ZHANG H., FIUME E.: Butterworth filtering and implicit fairing of irregular meshes. In *Proc. of Pacific Graphics* (2003), pp. 502–506.

[Zha04] ZHANG H.: Discrete combinatorial Laplacian operators for digital geometry processing. In *Proc. SIAM Conf. on Geom. Design and Comp.* (2004), pp. 575–592.

[ZKK02] ZIGELMAN G., KIMMEL R., KIRYATI N.: Texture mapping using surface flattening via multidimensional scaling. *IEEE Trans. Vis. & Comp. Graphics 8*, 2 (2002), 198–207.

[ZL05] ZHANG H., LIU R.: Mesh segmentation via recursive and visually salient spectral cuts. In *Proc. of Vision, Modeling, and Visualization* (2005).

[ZR72] ZAHN C. T., ROSKIES R. Z.: Fourier descriptors for plane closed curves. *IEEE Trans. on Computers 21*, 3 (1972), 269–281.

[ZSGS04] ZHOU K., SNYDER J., GUO B., SHUM H.-Y.: Iso-charts: Stretch-driven mesh parameterization using spectral analysis. In *Proc. Eurographics Symp. on Geometry Processing* (2004).

[ZW05] ZHU P., WILSON R. C.: A study of graph spectra for comparing graphs. In *Proc. of British Machine Vision Conference* (2005).