Spectral Processing

Part I
Although relatively recent in the context of geometry processing, spectral methods have already experienced a large development in the field of spectral graph theory.
Although relatively recent in the context of geometry processing, spectral methods have already experienced a large development in the field of spectral graph theory.

Treating the vertex coordinates as a signal defined over the graph structure supplied by the underlying mesh, G. Taubin first introduced the Spectral Theory in the context of geometry processing.
Spectral methods strongly rely on handling the eigendecomposition of a matrix, called *Laplacian Matrix*, which incorporates the adjacency and/or geometrical information derived from the mesh.

Let's stop a moment and review the definition and some properties of eigenvectors and eigenvalues.
Short Review of Eigenvalues and Eigenvectors

Let $A$ be an $n \times n$ matrix. A vector $u \in \mathbb{R}^n$ is an eigenvector of $A$ iff there is an scalar $\lambda$, called eigenvalue associated to $u$, such that:

$$Au = \lambda u$$
Let $A$ be an $n \times n$ matrix. A vector $u \in \mathbb{R}^n$ is an eigenvector of $A$ iff there is a scalar $\lambda$, called eigenvalue associated to $u$, such that:

$$Au = \lambda u$$

Let $U_i = \{\alpha u_i \mid \alpha \in \mathbb{R}\}$, where $u_i$ is an eigenvector of $A$, and

$$V = U_1 \oplus U_2 \cdots U_m$$

If we consider $A$ a linear operator $A : V \subset \mathbb{R}^n \to V$ then the subspaces $U_i$ are invariant under $A$. 
Short Review of Eigenvalues and Eigenvectors

The eigenvalues and eigenvectors associate to a matrix $A$ can be complex numbers, even if $A$ is a real matrix.

Example: $A(x,y) = (-y,x)$
Short Review of Eigenvalues and Eigenvectors

**Theorem:** If \( \{u_1, \ldots, u_m\} \) is a set of eigenvectors associated to distinct eigenvalues \( \lambda_1, \ldots, \lambda_m \) then \( \{u_1, \ldots, u_m\} \) is linearly independent.
Short Review of Eigenvalues and Eigenvectors

**Theorem:** If \( \{u_1, \ldots, u_m\} \) is a set of eigenvectors associated to distinct eigenvalues \( \lambda_1, \ldots, \lambda_m \) then \( \{u_1, \ldots, u_m\} \) is linearly independent.

**Corollary:** An \( n \times n \) matrix \( A \) has at most \( n \) distinct eigenvalues.
Short Review of Eigenvalues and Eigenvectors

If \( \{v_1, \ldots, v_n\} \) is a basis for the space \( V \subset \mathbb{R}^m \) and \( T : V \subset \mathbb{R}^n \rightarrow V \) is a linear operator then the matrix representation of \( T \) with respect to \( \{v_1, \ldots, v_n\} \) is given by:

\[
Tv_k = a_{1k}v_1 + \cdots + a_{nk}v_n, \quad k = 1, \ldots, n
\]

that is,

\[
\begin{bmatrix}
  a_{11} & \cdots & a_{1n} \\
  \vdots & & \vdots \\
  a_{n1} & \cdots & a_{nn}
\end{bmatrix}
\]
Short Review of Eigenvalues and Eigenvectors

If the matrix representation of a linear operator $T$ is diagonal, that is:

$$
\begin{bmatrix}
\lambda_1 & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \lambda_n
\end{bmatrix}
$$

then there is a base $\{v_1, \ldots, v_n\}$ such that:

$$
Tv_i = \lambda_i v_i
$$
Short Review of Eigenvalues and Eigenvectors

If the matrix representation of a linear operator $T$ is diagonal, that is:

$$
\begin{bmatrix}
\lambda_1 & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \lambda_n
\end{bmatrix}
$$

then there is a base $\{v_1, \ldots, v_n\}$ such that:

$$
Tv_i = \lambda_i v_i
$$

Thus a linear operator $T : V \to V$ has a diagonal matrix representation if only if $V$ has a basis consisting of eigenvectors of $T$. 
Short Review of Eigenvalues and Eigenvectors

If an operator $T : V \rightarrow V$ is represented by a symmetric matrix ($T$ is self-adjoint operator) then:

1. The eigenvalues of $T$ are real
2. The set of eigenvectors forms an orthogonal basis for $V$
Short Review of Eigenvalues and Eigenvectors

Given a basis \( \{v_1, \ldots, v_m\} \) of a subspace \( V \subset \mathbb{R}^n \), the projection \( \hat{x} \) of a vector \( x \in \mathbb{R}^n \) onto \( V \) can be computed by solving the following linear system:

\[
\begin{bmatrix}
\langle v_1, v_1 \rangle & \cdots & \langle v_1, v_m \rangle \\
\vdots & \ddots & \vdots \\
\langle v_m, v_1 \rangle & \cdots & \langle v_m, v_m \rangle
\end{bmatrix}
\begin{bmatrix}
\alpha_1 \\
\vdots \\
\alpha_m
\end{bmatrix}
= \begin{bmatrix}
\langle x, v_1 \rangle \\
\vdots \\
\langle x, v_m \rangle
\end{bmatrix}
\]

\[\hat{x} = \sum_{i=1}^{m} \alpha_i v_i\]

For further details see:

Three main steps are involved in most mesh processing methods:

1. Construction of the matrix $M$ that incorporates the pairwise relations between mesh elements (adjacencies and geometry).

2. The eigendecomposition of $M$.

3. Handling of the eigendecomposition towards obtaining the desired mesh processing.
Given a triangulated surface $S$, an usual way to define the matrix $M$ is as follows:

$$m_{ij} = \begin{cases} 
-w_{ij} & i \neq j \text{ and } j \in N(i) \\
{w_{ii}^*} & i = j \\
0 & \text{otherwise}
\end{cases}$$

where $w_{ij} = w_{ji}$ is a weight assigned to the edge connecting the vertices $v_i$ and $v_j$, $N(i)$ are the indexes of neighbours of vertex $i$, and $w_{ii}^* = \sum_{j \in N(i)} w_{ij}$
Given a triangulated surface $S$, an usual way to define the matrix $M$ is as follows:

$$m_{ij} = \begin{cases} 
-w_{ij} & i \neq j \text{ and } j \in N(i) \\
 w_{ii}^* & i = j \\
0 & \text{otherwise}
\end{cases}$$

where $w_{ij} = w_{ji}$ is a weight assigned to the edge connecting the vertices $v_i$ and $v_j$, $N(i)$ are the indexes of neighbors of vertex $i$, and $w_{ii}^* = \sum_{j \in N(i)} w_{ij}$.
The Laplacian operator is defined as follows:

\[ L = B^{-1}M \]

where \( B \) is a diagonal matrix. Although \( L \) is not symmetric in general, it is similar to the symmetric matrix \( O = B^{-1/2}MB^{-1/2} \), thus \( L \) has real eigenvalues (the same as \( O \)), being \( u_i = B^{-1/2}v_i \) its eigenvectors (\( v_i \) being the eigenvectors of \( O \)).
The Laplacian operator is defined as follows:

\[ L = B^{-1}M \]

where \( B \) is a diagonal matrix. Although \( L \) is not symmetric in general, it is similar to the symmetric matrix \( O = B^{-1/2}MB^{-1/2} \), thus \( L \) has real eigenvalues (the same as \( O \)), being \( u_i = B^{-1/2}v_i \) its eigenvectors (\( v_i \) being the eigenvectors of \( O \)).

Let \( f : S \to \mathbb{R} \) be a function that assigns a real scalar to each vertex of the surface \( S \) (notice that \( f \) can be thought as a vector in \( \mathbb{R}^m \), where \( m \) is the number of vertices in \( S \)). Then, apply the Laplacian operator in \( f \) is equivalent to:

\[ (Lf)_i = b_i^{-1} \sum_{j \in \text{viz}(i)} w_{ij}(f_i - f_j) \]

where \( b_i \) is the \( i^{th} \) diagonal element of \( B \) and \( f_k \) is the value of \( f \) in the vertex \( v_k \).
Examples: [Taubin 1995]: Signal Processing for to mesh fairing (combinatorial Laplacian)
[Tutte 1963]: graph drawing

\[ b_i = |N(i)| \quad \text{(number of elements in the neighborhood of } v_i) \]
\[ w_{ij} = 1 \]

If \( S \) is a polygon the Laplacian operator defined from \( b_i = |N(i)| \) and \( w_{ij} = 1 \) is:

\[
(Lf)_i = \frac{1}{2}(f_{i-1} - f_i) + \frac{1}{2}(f_{i+1} - f_i)
\]

\[
L = \frac{1}{2}
\begin{bmatrix}
2 & -1 & -1 \\
-1 & 2 & -1 \\
-1 & 2 & -1 \\
\vdots \\
-1 & -1 & 2
\end{bmatrix}
\]
Examples:  

[Taubin 1995]: Signal Processing for to mesh fairing  
[Tutte 1963]: graph drawing

\[ b_i = |N(i)| \quad \text{(number of elements in the neighborhood of } v_i) \]
\[ w_{ij} = 1 \]

IF \( S \) is a polygon the Laplacian operator defined from \( b_i = |N(i)| \) and \( w_{ij} = 1 \) is:

\[
(Lf)_i = \frac{1}{2}(f_{i-1} - f_i) + \frac{1}{2}(f_{i+1} - f_i)
\]

\[
L = \frac{1}{2}
\begin{bmatrix}
2 & -1 & & -1 \\
-1 & 2 & -1 & \\
& & \ddots & \\
-1 & & -1 & 2
\end{bmatrix}
\]

\[
B^{-1} M
\]

Eigenvalues are:

\[
\lambda_j = 1 - \cos(2\pi \lfloor j/2 \rfloor / n)
\]
\[
0 \leq \lambda_1 \leq \ldots \lambda_n \leq 2
\]

Eigenvectors are:

\[
(u_i)_h = \begin{cases} 
\sqrt{1/n} & \quad i = 1 \\
\sqrt{2/n \sin(2\pi \lfloor j/2 \rfloor / n)} & \quad i \text{ is even} \\
\sqrt{2/n \cos(2\pi \lfloor j/2 \rfloor / n)} & \quad i \text{ is odd}
\end{cases}
\]
Examples:
[Taubin 1995]: Signal Processing for mesh fairing
[Tutte 1963]: graph drawing

\[ b_i = |N(i)| \quad \text{(number of elements in the neighborhood of } v_i) \]
\[ w_{ij} = 1 \]

If \( S \) is a polygon the Laplacian operator defined from \( b_i = |N(i)| \) and \( w_{ij} = 1 \) is:

\[
(Lf)_i = \frac{1}{2}(f_{i-1} - f_i) + \frac{1}{2}(f_{i+1} - f_i)
\]

\[
L = \frac{1}{2} \begin{bmatrix}
2 & -1 & -1 \\
-1 & 2 & -1 \\
& \vdots & \\
-1 & -1 & 2
\end{bmatrix}
\]

\[
B^{-1}M
\]

Eigenvalues are:
\[
\lambda_j = 1 - \cos \left( \frac{2\pi [j/2]}{n} \right)
\]
\[
0 \leq \lambda_1 \leq \ldots \lambda_n \leq 2
\]

Equivalent to the Fourier Basis

Eigenvectors are:
\[
(u_i)_h = \begin{cases}
\sqrt{\frac{1}{n}} & i = 1 \\
\sqrt{\frac{2}{n}} \sin \left( \frac{2\pi \left[ j/2 \right]}{n} \right) & i \text{ is even} \\
\sqrt{\frac{2}{n}} \cos \left( \frac{2\pi \left[ j/2 \right]}{n} \right) & i \text{ is odd}
\end{cases}
\]
Examples: [Pinkall and Polthier 1993]: Geometrical information

\[(L f)_i = \sum_{j \in N(i)} \frac{1}{2} (\cot(\alpha_{ij}) + \cot(\beta_{ij}))(f_i - f_j)\]

\[b_i = 1; \quad w_{ij} = \frac{1}{2} (\cot(\alpha_{ij}) + \cot(\beta_{ij}))\]

What do the weights as defined above mean?
What do the weights as defined above mean?

It aims at approximating the Laplace-Beltrami operator which can be derived from the mean curvature normal operator:

\[
K(x) = 2\kappa_H(x)\vec{n}(x)
\]

\[
K(x_i) = \frac{1}{2|x_i^*|} \sum_{j \in N(x_i)} (\cot(\alpha_{ij}) + \cot(\beta_{ij}))(x_i - x_j)
\]
Examples: [Vallet and Levy 2008]: Manifold Harmonics

\[(L f)_i = \sum_{j \in N(i)} \frac{(\cot(\alpha_{ij}) + \cot(\beta_{ij}))}{|v_i^*|} (f_i - f_j)\]

where \(|\nu_i^*|\) is the barycentric dual area.
Examples: [Vallet and Levy 2008]: Manifold Harmonics

\[
(Lf)_i = \sum_{j \in N(i)} \frac{(\cot(\alpha_{ij}) + \cot(\beta_{ij}))}{|v_i^*|} (f_i - f_j)
\]

where $|v_i^*|$ is the barycentric dual area.

As $w_{ij} \neq w_{ji}$ this operator is not symmetric, furthermore, it can not be written as $L = B^{-1}M$ where $M$ is a symmetric matrix.

\[
(Lf)_i = \sum_{j \in N(i)} \frac{(\cot(\alpha_{ij}) + \cot(\beta_{ij}))}{\sqrt{|v_i^*||v_j^*|}} (f_i - f_j)
\]

\[
b_i = 1; \quad w_{ij} = \frac{(\cot(\alpha_{ij}) + \cot(\beta_{ij}))}{\sqrt{|v_i^*||v_j^*|}}
\]
Comparison: [Vallet and Levy 2008]

A: combinatorial
B: cotangents
C: weighted cotangents
D: symmetrized cotangents
E: symmetric weights
Given the orthonormal basis $v_i$, we can compute the coefficients:

$$\alpha_i = \langle x, v_i \rangle$$

such that $x$ is projected on the eigenspace by:

$$x \sim \hat{x} = \sum \alpha_i v_i$$
In the context of Manifold Harmonics, Vallet and Levy propose a filtering scheme based on “transfer functions” acting on the coefficients:

\[ \hat{x}^F = \sum F(\omega) \alpha_i v_i = \sum F(\sqrt{\lambda_i}) \alpha_i v_i \]
There are other operators with similar structure to Laplacian operators:

- **Schrödinger operator**

\[
H_{ij} = \begin{cases} 
\text{negative real number} & \text{if } i \neq j \text{ and } j \in \text{viz}(i) \\
\text{any real number} & \text{if } i = j \\
0 & \text{otherwise}
\end{cases}
\]

- **Affinity Matrix**

\[
w_{ij} \text{ represents a numerical relation between } i \text{ and } j
\]

\[
w_{ij} = e^{-\text{dist}(i,j)/2\sigma^2}
\]
Spectral Processing

Part II
Eigenvector 1
Eigenvector 2
Eigenvector 11
Eigenvector 5
Nodal Domain: The *nodal set* of an eigenfunction is composed of points at which the eigenfunction is zero. The regions bounded by the nodal set are called *nodal domains*.

PS. The eigenfunction is built by interpolating the values of the eigenvector.

Courant's Nodal Theorem: Let the eigenvectors of the Laplace operator be labeled in increasing order in accordance with the corresponding eigenvalues. Then, the *k*-th eigenfunction can have at most *k* nodal domains, that is, the *k*-th eigenfunction can separate the surface into at most *k* connected components.
In the case of combinatorial Laplacian:

\[ \frac{f^T L f}{f^T f} = \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2 \]

\( L \) is positive semi-definite, so the smallest eigenvalue is zero with a constant associated eigenvector.
In the case of combinatorial Laplacian:

\[
\frac{f^T L f}{f^T f} = \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2 \quad \text{L is positive semi-definite, so the smallest eigenvalue is zero with a constant associated eigenvector.}
\]

\[
\frac{\|f\|^2}{\|f\|^2}
\]

Similar results can also be obtained for other Laplacians!
In the case of combinatorial Laplacian:

The second eigenvector (called Fiedler vector) is characterized by:

$$v_2(L) = \arg \min_u \sum_{i,j} w_{ij}(u_i - u_j)^2$$

Subject to: $\sum u_i = 0$ and $\sum u_i^2 = 1$

The solution of this problem is related with one-dimensional embedding of the graph and can be used for finding vertices ordering.
Computational Aspects:

Computing the eigendecomposition of matrix efficiently has been a challenge problem for many years. Much effort has been devoted to devise robust techniques that speed-up the eigendecomposition, as efficiency is a concern for large meshes.

In the context of geometry processing, eigendecomposition techniques aim at accelerating the computation by exploring the sparsity of the matrices together with the shift property of the eigenvalues.
Computational Aspects:

The numerical analysis literature describes efficient techniques to compute the largest eigenvalue and associated eigenvector of a given matrix, as for example, the iterative power method.

The following properties allow the computation of other elements of the spectrum:

**Shift Property:**
If $\lambda$ is an eigenvalue of $L$ with associated eigenvector $v$ then $v$ is also an eigenvector of $L - \sigma I$ with $\lambda - \sigma$ its associated eigenvalue.

**Inversion Property:**
If $\lambda$ is an eigenvalue of $L$ with associated eigenvector $v$ then $1/\lambda$ is an eigenvalue of $L^{-1}$ and $v$ is its associated eigenvector.

$$B = (L - \sigma I)^{-1}$$
Computational Aspects:

Available Libs:

1. ARPACK: Based on Arnoldi's Method

2. Sparse Factorization: TAUCS