HAL
open science

3D Shape Registration Using Spectral Graph Embedding and Probabilistic Matching

Avinash Sharma, Radu Horaud, Diana Mateus

To cite this version:

Avinash Sharma, Radu Horaud, Diana Mateus. 3D Shape Registration Using Spectral Graph Embedding and Probabilistic Matching. Olivier Lezoray and Leo Grady. Image Processing and Analysing With Graphs: Theory and Practice, CRC Press, pp.441-474, 2012. inria-00590273v2

HAL Id: inria-00590273

https://inria.hal.science/inria-00590273v2

Submitted on 10 Aug 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Chapter 8

3D Shape Registration Using Spectral Graph Embedding and Probabilistic Matching

AVINASH SHARMA

INRIA Grenoble Rhône-Alpes
655 avenue de l'Europe
38330 Montbonnot Saint-Martin, France
Avinash.Sharma@inrialpes.fr

RADU HORAUD

INRIA Grenoble Rhône-Alpes
655 avenue de l'Europe
38330 Montbonnot Saint-Martin, France
Radu.Horaud@inrialpes.fr

DIANA MATEUS

Institut für Informatik
Technische Universität München
Garching b. München Germany
mateus@in.tum.de

Abstract

In this book chapter we address the problem of 3D shape registration and we propose a novel technique based on spectral graph theory and probabilistic matching. Recent advancement in shape acquisition technology has led to the capture of large amounts of 3D data. Existing real-time multi-camera 3D acquisition methods provide a frame-wise reliable visual-hull or mesh representations for real 3D animation sequences The task of 3D shape analysis involves tracking, recognition, registration, etc. Analyzing 3D data in a single framework is still a challenging task considering the large variability of the data gathered with different acquisition devices. 3D shape registration is one such challenging shape analysis task. The main contribution of this chapter is to extend the spectral graph matching methods to very large graphs by combining spectral graph matching with Laplacian embedding. Since the embedded representation of a graph is obtained by dimensionality reduction we claim that the existing spectral-based methods are not easily applicable. We discuss solutions for the exact and inexact graph isomorphism problems and recall the main spectral properties of the combinatorial graph Laplacian; We provide a novel analysis of the commute-time embedding that allows us to interpret the latter in terms of the PCA of a graph, and to select the appropriate dimension of the associated embedded metric space; We derive a unit hyper-sphere normalization for the commute-time embedding that allows us to register two shapes with different samplings; We propose a novel method to find the eigenvalue-eigenvector ordering and the eigenvector sign using the eigensignature (histogram) which is invariant to the isometric shape deformations and fits well in the spectral graph matching framework, and we present a probabilistic shape matching formulation using an expectation maximization point registration algorithm which alternates between aligning the eigenbases and finding a vertex-to-vertex assignment.

8.1 Introduction

In this chapter we discuss the problem of 3D shape registration. Recent advancement in shape acquisition technology has led to the capture of large amounts of 3D data. Existing real-time multi-camera 3D acquisition methods provide a frame-wise reliable visual-hull or mesh representations for real 3D animation sequences [1, 2, 3, 4, 5, 6]. The task of 3D shape analysis involves tracking, recognition, registration, etc. Analyzing 3D data in a single framework is still a challenging task considering the large variability of the data gathered with different acquisition devices. 3D shape registration is one such challenging shape analysis task. The major difficulties in shape registration arise due to: 1) variation in the shape acquisition techniques, 2) local deformations in non-rigid shapes, 3) large acquisition discrepancies (*e.g.*, holes, topology change, surface acquisition noise), 4) local scale change.

Most of the previous attempts of shape matching can be broadly categorized as *extrinsic* or *intrinsic*

approaches depending on how they analyze the properties of the underlying manifold. Extrinsic approaches mainly focus on finding a global or local rigid transformation between two 3D shapes.

There is large set of approaches based on variations of Iterative Closest Point (ICP) algorithm [**?**, **?**, 7] that falls in the category of extrinsic approaches. However, the majority of these approaches compute rigid transformations for shape registration and are not directly applicable to non-rigid shapes. Intrinsic approaches are a natural choice for finding dense correspondences between articulated shapes, as they embed the shape in some canonical domain which preserves some important properties of the manifold, *e.g.*, geodesics and angles. Intrinsic approaches are preferable over extrinsic as they provide a global representation which is invariant to non-rigid deformations that are common in the real-world 3D shapes.

Interestingly, mesh representation also enables the adaptation of well established graph matching algorithms that use eigenvalues and eigenvectors of graph matrices, and are theoretically well investigated in the framework of *Spectral Graph Theory* (SGT) *e.g.*, [8, 9]. Existing methods in SGT are mainly theoretical results applied to small graphs and under the premise that eigenvalues can be computed exactly. However, spectral graph matching does not easily generalize to very large graphs due to the following reasons: 1) eigenvalues are approximately computed using eigen-solvers, 2) eigenvalue multiplicity and hence ordering change are not well studied, 3) exact matching is intractable for very large graphs. It is important to note that these methods mainly focus on exact graph matching while majority of the real-world graph matching applications involve graphs with different cardinality and for which only a subgraph isomorphism can be sought.

The main contribution of this work is to extend the spectral graph methods to very large graphs by combining spectral graph matching with *Laplacian Embedding*. Since the embedded representation of a graph is obtained by dimensionality reduction we claim that the existing SGT methods (*e.g.*, [8]) are not easily applicable. The major contributions of this work are the following: 1) we discuss solutions for the exact and inexact graph isomorphism problems and recall the main spectral properties of the combinatorial graph Laplacian, 2) we provide a novel analysis of the commute-time embedding that allows us to interpret the latter in terms of the PCA of a graph, and to select the appropriate dimension of the associated embedded metric space, 3) we derive a unit hyper-sphere normalization for the commute-time embedding that allows us to register two shapes with different samplings, 4) we propose a novel method to find the eigenvalue-eigenvector ordering and the eigenvector signs using the eigensignatures (histograms) that are invariant to the isometric shape deformations and which fits well in the spectral graph matching framework, 5) we present a probabilistic shape matching formulation using *expectation maximization* for point registration algorithm which alternates between aligning the eigenbases and finding a vertex-to-vertex assignment.

The existing graph matching methods that use intrinsic representations are: [10, 11, 12, 13, 14, 15,

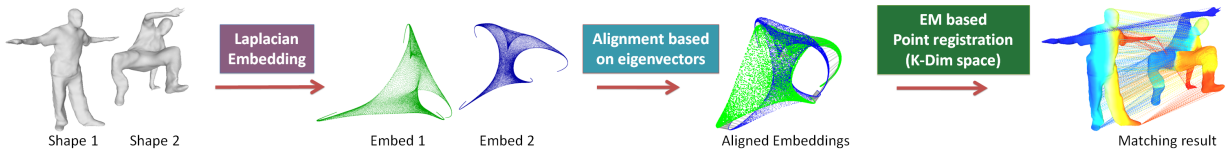


Figure 8.1: Overview of the proposed method. First, a Laplacian embedding is obtained for each shape. Next, these embeddings are aligned to handle the issue of sign flip and ordering change using the histogram matching. Finally, an Expectation-Maximization based point registration is performed to obtain dense probabilistic matching between two shapes.

16, 17]. There is another class of methods that allows to combine intrinsic (geodesics) and extrinsic (appearance) features and which were previously successfully applied for matching features in pairs of images [**?**, **?**, **?**, 19, 20, 21, 22, 23, 24, 25]. Some recent approaches apply hierarchical matching to find dense correspondences [26, 27, 28]. However, many of these graph matching algorithms suffer from the problem of either computational intractability or a lack of proper distance metric (*w.r.t.* underlying manifold structure) as the Euclidean metric is not directly applicable while computing distances on non-rigid shapes. A recent benchmarking of shape matching methods was performed in [29]. Recently, a few methods proposed a diffusion framework for the task of shape registration [30, 31, 32].

In this chapter we present an intrinsic approach for unsupervised 3D shape registration first proposed in [14, 33]. In the first step, dimensionality reduction is performed using the graph Laplacian which allows us to embed a 3D shape in an isometric subspace invariant to non-rigid deformations. This leads to an embedded point cloud representation where each vertex of the underlying graph is mapped to a point in a K-dimensional metric space. Thus, the problem of non-rigid 3D shape registration is transformed into a K-dimensional point registration task. However, before point registration, the two eigen spaces need to be correctly aligned. This alignment is critical for the spectral matching methods because the two eigen spaces are defined up to the signs and the ordering of the eigenvectors of their Laplacian matrices. This is achieved by a novel matching method that uses histograms of eigenvectors as eigensignatures. In the final step, a point registration method based on a variant of the expectation-maximization (EM) algorithm [34] is applied in order to register two sets of points associated with the Laplacian embeddings of the two shapes. The proposed algorithm alternates between the estimation of an orthogonal transformation matrix associated with the alignment of the two eigen spaces and the computation of probabilistic vertex-to-vertex assignment. Figure 1.1 presents the overview of the proposed method. According to the results summarized in [29], this method is one among the best performing unsupervised shape matching algorithms.

Chapter Overview: Graph matrices are introduced in section 1.2. The problem of exact graph isomorphism and existing solutions are discussed in section 1.3. Section 1.4 deals with dimensionality reduction using the graph Laplacian in order to obtain embedded representations for 3D shapes. In the same section we discuss the PCA of graph embeddings and propose a unit hyper-sphere normalization for these embeddings along with a method to choose the embedding dimension. Section 1.5 introduces the formulation of maximum subgraph isomorphism before presenting a two-step method for 3D shape registration. In the first step Laplacian embeddings are aligned using histogram matching while in the second step we briefly discuss an EM point registration method to obtain probabilistic shape registration. Finally we present shape matching results in section 1.6 and conclude with a brief discussion in section 1.7.

8.2 Graph Matrices

A shape can be treated as a connected *undirected weighted graph* $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V}(\mathcal{G}) = \{v_1, \ldots, v_n\}$ is the vertex set, $\mathcal{E}(\mathcal{G}) = \{e_{ij}\}$ is the edge set. Let $\mathbf{W}$ be the weighted adjacency matrix of this graph. Each $(i, j)^{\text{th}}$ entry of $\mathbf{W}$ matrix stores weight w_{ij} whenever there is an edge $e_{ij} \in \mathcal{E}(\mathcal{G})$ between graph vertices v_i and v_j and 0 otherwise with all the diagonal elements set to 0 . We use the following notations: The degree d_i of a graph vertex $d_i = \sum_{i \sim j} w_{ij}$ ($i \sim j$ denotes the set of vertices v_j which are adjacent to v_i), the *degree matrix* $\mathbf{D} = \text{diag}[d_1 \ldots d_i \ldots d_n]$, the $n \times 1$ vector $\mathbb{1} = (1 \ldots 1)^\top$ (the constant vector), the $n \times 1$ *degree vector* $\mathbf{d} = \mathbf{D}\mathbb{1}$, and the *graph volume* $\text{Vol}(\mathcal{G}) = \sum_i d_i$.

In spectral graph theory, it is common [35, 36] to use the following expression for the edge weights:

$$w_{ij} = e^{-\frac{\text{dist}^2(v_i, v_j)}{\sigma^2}}, \tag{8.1}$$

where $\text{dist}(v_i, v_j)$ denotes any distance metric between two vertices and σ is a free parameter. In the case of a *fully connected graph*, matrix $\mathbf{W}$ is also referred to as the *similarity matrix*. The *normalized weighted adjacency matrix* writes $\tilde{\mathbf{W}} = \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$. The *transition* matrix of the non-symmetric reversible Markov chain associated with the graph is $\tilde{\mathbf{W}}_R = \mathbf{D}^{-1}\mathbf{W} = \mathbf{D}^{-1/2}\tilde{\mathbf{W}}\mathbf{D}^{1/2}$.

8.2.1 Variants of the Graph Laplacian Matrix

We can now build the concept of the *graph Laplacian operator*. We consider the following variants of the Laplacian matrix [37, 36, 38]:

- The *unnormalized Laplacian* which is also referred to as the *combinatorial Laplacian* $\mathbf{L}$,
- the *normalized Laplacian* $\tilde{\mathbf{L}}$, and

- the *random-walk Laplacian* $\tilde{\mathbf{L}}_R$ also referred to as the *discrete Laplace operator*.

In more detail we have:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \tag{8.2}$$

$$\tilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2} = \mathbf{I} - \tilde{\mathbf{W}} \tag{8.3}$$

$$\tilde{\mathbf{L}}_R = \mathbf{D}^{-1}\mathbf{L} = \mathbf{I} - \tilde{\mathbf{W}}_R \tag{8.4}$$

with the following relations between these matrices:

$$\mathbf{L} = \mathbf{D}^{1/2}\tilde{\mathbf{L}}\mathbf{D}^{1/2} = \mathbf{D}\tilde{\mathbf{L}}_R \tag{8.5}$$

$$\tilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2} = \mathbf{D}^{1/2}\tilde{\mathbf{L}}_R\mathbf{D}^{-1/2} \tag{8.6}$$

$$\tilde{\mathbf{L}}_R = \mathbf{D}^{-1/2}\tilde{\mathbf{L}}\mathbf{D}^{1/2} = \mathbf{D}^{-1}\mathbf{L}. \tag{8.7}$$

8.3 Spectral Graph Isomorphism

Let $\mathcal{G}_A$ and $\mathcal{G}_B$ be two *undirected weighted graphs* with the same number of nodes, n, and let $\mathbf{W}_A$ and $\mathbf{W}_B$ be their adjacency matrices. They are real-symmetric matrices. In the general case, the number r of distinct eigenvalues of these matrices is smaller than n. The standard spectral methods only apply to those graphs whose adjacency matrices have n distinct eigenvalues (each eigenvalue has multiplicity one), which implies that the eigenvalues can be ordered.

Graph isomorphism [39] can be written as the following minimization problem:

$$\mathbf{P}^\star = \arg\min_{\mathbf{P}} \|\mathbf{W}_A - \mathbf{P}\mathbf{W}_B\mathbf{P}^\top\|_F^2 \tag{8.8}$$

where $\mathbf{P}$ is an $n \times n$ permutation matrix (see appendix A.1) with $\mathbf{P}^\star$ as the desired vertex-to-vertex permutation matrix and $\|\bullet\|_F$ is the Frobenius norm defined by (see appendix A.2):

$$\|\mathbf{W}\|_F^2 = \langle \mathbf{W}, \mathbf{W} \rangle = \sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}^2 = \mathrm{tr}(\mathbf{W}^\top\mathbf{W}) \tag{8.9}$$

Let:

$$\mathbf{W}_A = \mathbf{U}_A\mathbf{\Lambda}_A\mathbf{U}_A^\top \tag{8.10}$$

$$\mathbf{W}_B = \mathbf{U}_B\mathbf{\Lambda}_B\mathbf{U}_B^\top \tag{8.11}$$

be the eigen-decompositions of the two matrices with n eigenvalues $\mathbf{\Lambda}_A = \mathrm{diag}[\alpha_i]$ and $\mathbf{\Lambda}_B = \mathrm{diag}[\beta_i]$ and n orthonormal eigenvectors, the column vectors of $\mathbf{U}_A$ and $\mathbf{U}_B$.

8.3.1 An Exact Spectral Solution

If there exists a vertex-to-vertex correspondence that makes (1.8) equal to 0, we have:

$$\mathbf{W}_A = \mathbf{P}^\star \mathbf{W}_B \mathbf{P}^{\star\top}. \tag{8.12}$$

This implies that the adjacency matrices of the two graphs should have the same eigenvalues. Moreover, if the eigenvalues are non null and, the matrices $\mathbf{U}_A$ and $\mathbf{U}_B$ have full rank and are uniquely defined by their n orthonormal column vectors (which are the eigenvectors of $\mathbf{W}_A$ and $\mathbf{W}_B$), then $\alpha_i = \beta_i, \forall i,\ 1 \leq i \leq n$ and $\mathbf{\Lambda}_A = \mathbf{\Lambda}_B$. From (1.12) and using the eigen-decompositions of the two graph matrices we obtain:

$$\mathbf{\Lambda}_A = \mathbf{U}_A^\top \mathbf{P}^\star \breve{\mathbf{U}}_B \mathbf{\Lambda}_B \breve{\mathbf{U}}_B^\top \mathbf{P}^{\star\top} \mathbf{U}_A = \mathbf{\Lambda}_B, \tag{8.13}$$

where the matrix $\breve{\mathbf{U}}_B$ is defined by:

$$\breve{\mathbf{U}}_B = \mathbf{U}_B \mathbf{S}. \tag{8.14}$$

Matrix $\mathbf{S} = \mathrm{diag}[s_i]$, with $s_i = \pm 1$, is referred to as a sign matrix with the property $\mathbf{S}^2 = \mathbf{I}$. Post multiplication of $\mathbf{U}_B$ with a sign matrix takes into account the fact that the eigenvectors (the column vectors of $\mathbf{U}_B$) are only defined up to a sign. Finally we obtain the following permutation matrix:

$$\mathbf{P}^\star = \mathbf{U}_B \mathbf{S} \mathbf{U}_A^\top. \tag{8.15}$$

Therefore, one may notice that there are as many solutions as the cardinality of the set of matrices $\mathbf{S}_n$, i.e., $|\mathbf{S}_n| = 2^n$, and that *not all of these solutions correspond to a permutation matrix*. This means that there exist some matrices $\mathbf{S}^\star$ that exactly make $\mathbf{P}^\star$ a permutation matrix. Hence, all those permutation matrices that satisfy (1.15) are solutions of the exact graph isomorphism problem. Notice that once the permutation has been estimated, one can write that the rows of $\mathbf{U}_B$ can be aligned with the rows of $\mathbf{U}_A$:

$$\mathbf{U}_A = \mathbf{P}^\star \mathbf{U}_B \mathbf{S}^\star. \tag{8.16}$$

The rows of $\mathbf{U}_A$ and of $\mathbf{U}_B$ can be interpreted as isometric embeddings of the two graph vertices: A vertex v_i of $\mathcal{G}_A$ has as coordinates the ith row of $\mathbf{U}_A$. This means that the spectral graph isomorphism problem becomes a point registration problem, where graph vertices are represented by points in $\mathbb{R}^n$. To conclude, the exact graph isomorphism problem has a spectral solution based on: (i) the eigen-decomposition of the two graph matrices, (ii) the ordering of their eigenvalues, and (iii) the choice of a sign for each eigenvector.

8.3.2 The Hoffman-Wielandt Theorem

The Hoffman-Wielandt theorem [40, 41] is the fundamental building block of spectral graph isomorphism. The theorem holds for normal matrices; Here, we restrict the analysis to real symmetric matrices, although the generalization to Hermitian matrices is straightforward:

Theorem 1

(Hoffman and Wielandt) If $\mathbf{W}_A$ and $\mathbf{W}_B$ are real-symmetric matrices, and if α_i and β_i are their eigenvalues arranged in increasing order, $\alpha_1 \leq \ldots \leq \alpha_i \leq \ldots \leq \alpha_n$ and $\beta_1 \leq \ldots \leq \beta_i \leq \ldots \leq \beta_n$, then

$$\sum_{i=1}^{n}(\alpha_i - \beta_i)^2 \leq \|\mathbf{W}_A - \mathbf{W}_B\|_F^2. \tag{8.17}$$

Proof: The proof is derived from [9, 42]. Consider the eigen-decompositions of matrices $\mathbf{W}_A$ and $\mathbf{W}_B$, (1.10), (1.11). Notice that for the time being we are free to prescribe the ordering of the eigenvalues α_i and β_i and hence the ordering of the column vectors of matrices $\mathbf{U}_A$ and $\mathbf{U}_B$. By combining (1.10) and (1.11) we write:

$$\mathbf{U}_A\mathbf{\Lambda}_A\mathbf{U}_A^\top - \mathbf{U}_B\mathbf{\Lambda}_B\mathbf{U}_B^\top = \mathbf{W}_A - \mathbf{W}_B \tag{8.18}$$

or, equivalently:

$$\mathbf{\Lambda}_A\mathbf{U}_A^\top\mathbf{U}_B - \mathbf{U}_A^\top\mathbf{U}_B\mathbf{\Lambda}_B = \mathbf{U}_A^\top(\mathbf{W}_A - \mathbf{W}_B)\mathbf{U}_B. \tag{8.19}$$

By the unitary-invariance of the Frobenius norm (see appendix A.2) and with the notation $\mathbf{Z} = \mathbf{U}_A^\top\mathbf{U}_B$ we obtain:

$$\|\mathbf{\Lambda}_A\mathbf{Z} - \mathbf{Z}\mathbf{\Lambda}_B\|_F^2 = \|\mathbf{W}_A - \mathbf{W}_B\|_F^2, \tag{8.20}$$

which is equivalent to:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \beta_j)^2 z_{ij}^2 = \|\mathbf{W}_A - \mathbf{W}_B\|_F^2. \tag{8.21}$$

The coefficients $x_{ij} = z_{ij}^2$ can be viewed as the entries of a doubly-stochastic matrix $\mathbf{X}$: $x_{ij} \geq 0, \sum_{i=1}^{n} x_{ij} = 1, \sum_{j=1}^{n} x_{ij} = 1$. Using these properties, we obtain:

$$\begin{aligned}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \beta_j)^2 z_{ij}^2 &= \sum_{i=1}^{n}\alpha_i^2 + \sum_{j=1}^{n}\beta_j^2 - 2\sum_{i=1}^{n}\sum_{j=1}^{n} z_{ij}^2\alpha_i\beta_j \\ &\geq \sum_{i=1}^{n}\alpha_i^2 + \sum_{j=1}^{n}\beta_j^2 - 2\max_{Z}\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} z_{ij}^2\alpha_i\beta_j\right\}.\end{aligned} \tag{8.22}$$

Hence, the minimization of (1.21) is equivalent to the maximization of the last term in (1.22). We can modify our maximization problem to admit all the doubly-stochastic matrices. In this way we seek an extremum over a convex compact set. The maximum over this compact set is larger than or equal to our maximum:

$$\max_{Z\in O_n}\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} z_{ij}^2\alpha_i\beta_j\right\} \leq \max_{X\in\mathcal{D}_n}\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}\alpha_i\beta_j\right\} \tag{8.23}$$

where O_n is the set of orthogonal matrices and $\mathcal{D}_n$ is the set of doubly stochastic matrices (see appendix A.1). Let $c_{ij} = \alpha_i\beta_j$ and hence one can write that the right term in the equation above as the dot-product of two matrices:

$$\langle\mathbf{X},\mathbf{C}\rangle = \mathrm{tr}(\mathbf{X}\mathbf{C}) = \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}c_{ij}. \tag{8.24}$$

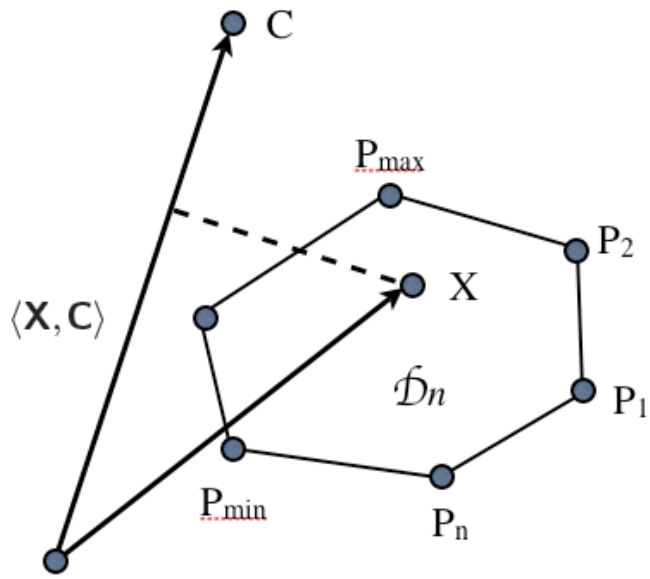


Figure 8.2: This figure illustrates the maximization of the dot-product $\langle \mathbf{X}, \mathbf{C} \rangle$. The two matrices can be viewed as vectors of dimension n^2. Matrix $\mathbf{X}$ belongs to a compact convex set whose extreme points are the permutation matrices $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_n$. Therefore, the projection of this set (i.e., $\mathcal{D}_n$) onto $\mathbf{C}$ has projected permutation matrices at its extremes, namely $\langle \mathbf{P}_{\min}, \mathbf{X} \rangle$ and $\langle \mathbf{P}_{\max}, \mathbf{X} \rangle$ in this example.

Therefore, this expression can be interpreted as the projection of $\mathbf{X}$ onto $\mathbf{C}$, see figure 1.2. The Birkhoff theorem (appendix A.1) tells us that the set $\mathcal{D}_n$ of doubly stochastic matrices is a compact convex set. We obtain that the extrema (minimum and maximum) of the projection of $\mathbf{X}$ onto $\mathbf{C}$ occur at the projections of one of the extreme points of this convex set, which correspond to permutation matrices. Hence, the maximum of $\langle \mathbf{X}, \mathbf{C} \rangle$ is $\langle \mathbf{P}_{\max}, \mathbf{X} \rangle$ and we obtain:

$$\max_{X \in \mathcal{D}_n} \left\{ \sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} \alpha_i \beta_j \right\} = \sum_{i=1}^{n} \alpha_i \beta_{\pi(i)}. \tag{8.25}$$

By substitution in (1.22) we obtain:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} (\alpha_i - \beta_j)^2 z_{ij}^2 \geq \sum_{i=1}^{n} (\alpha_i - \beta_{\pi(i)})^2. \tag{8.26}$$

If the eigenvalues are in increasing order then the permutation that satisfies theorem 1.17 is the identity matrix, i.e., $\pi(i) = i$. Indeed, let's assume that for some indices k and $k+1$ we have: $\pi(k) = k+1$ and $\pi(k+1) = k$. Since $\alpha_k \leq \alpha_{k+1}$ and $\beta_k \leq \beta_{k+1}$, the following inequality holds:

$$(\alpha_k - \beta_k)^2 + (\alpha_{k+1} - \beta_{k+1})^2 \leq (\alpha_k - \beta_{k+1})^2 + (\alpha_{k+1} - \beta_k)^2 \tag{8.27}$$

and hence (1.17) holds. ■

Corollary 1.1

The inequality (1.17) becomes an equality when the eigenvectors of $\mathbf{W}_A$ *are aligned with the eigenvectors of* $\mathbf{W}_B$ *up to a sign ambiguity:*

$$\mathbf{U}_B = \mathbf{U}_A \mathbf{S}. \tag{8.28}$$

Proof: Since the minimum of (1.21) is achieved for $\mathbf{X} = \mathbf{I}$ and since the entries of $\mathbf{X}$ are z_{ij}^2, we have that $z_{ii} = \pm 1$, which corresponds to $\mathbf{Z} = \mathbf{S}$. ■

Corollary 1.2

If $\mathbf{Q}$ *is an orthogonal matrix, then*

$$\sum_{i=1}^{n} (\alpha_i - \beta_i)^2 \leq \|\mathbf{W}_A - \mathbf{Q}\mathbf{W}_B\mathbf{Q}^\top\|_F^2. \tag{8.29}$$

Proof: Since the eigen-decomposition of matrix $\mathbf{Q}\mathbf{W}_B\mathbf{Q}^\top$ is $(\mathbf{Q}\mathbf{U}_B)\mathbf{\Lambda}_B(\mathbf{Q}\mathbf{U}_B)^\top$ and since it has the same eigenvalues as $\mathbf{W}_B$, the inequality (1.29) holds and hence corollary 1.2. ■

These corollaries will be useful in the case of spectral graph matching methods presented below.

8.3.3 Umeyama's Method

The exact spectral matching solution presented in section 1.3.1 finds a permutation matrix satisfying (1.15). This requires an exhaustive search over the space of all possible 2^n matrices. Umeyama's method presented in [8] proposes a relaxed solution to this problem as outlined below.

Umeyama [8] addresses the problem of *weighted graph matching* within the framework of spectral graph theory. He proposes two methods, the first for *undirected weighted graphs* and the second for *directed weighted graphs*. The adjacency matrix is used in both cases. Let's consider the case of undirected graphs. The eigenvalues are (possibly with multiplicities):

$$\mathbf{W}_A: \quad \alpha_1 \leq \ldots \leq \alpha_i \leq \ldots \leq \alpha_n \tag{8.30}$$

$$\mathbf{W}_B: \quad \beta_1 \leq \ldots \leq \beta_i \leq \ldots \leq \beta_n. \tag{8.31}$$

Theorem 2

(Umeyama) If $\mathbf{W}_A$ *and* $\mathbf{W}_B$ *are real-symmetric matrices with* n *distinct eigenvalues (that can be ordered),* $\alpha_1 < \ldots < \alpha_i < \ldots < \alpha_n$ *and* $\beta_1 < \ldots < \beta_i < \ldots < \beta_n$*, the minimum of :*

$$J(\mathbf{Q}) = \|\mathbf{W}_A - \mathbf{Q}\mathbf{W}_B\mathbf{Q}^\top\|_F^2 \tag{8.32}$$

is achieved for:

$$\mathbf{Q}^{\star} = \mathbf{U}_A \mathbf{S} \mathbf{U}_B^{\top} \tag{8.33}$$

and hence (1.29) becomes an equality:

$$\sum_{i=1}^{n} (\alpha_i - \beta_i)^2 = \|\mathbf{W}_A - \mathbf{Q}^{\star}\mathbf{W}_B\mathbf{Q}^{\star\top}\|_F^2. \tag{8.34}$$

Proof: The proof is straightforward. By corollary 1.2, the Hoffman-Wielandt theorem applies to matrices $\mathbf{W}_A$ and $\mathbf{Q}\mathbf{W}_B\mathbf{Q}^{\top}$. By corollary 1.1, the equality (1.34) is achieved for:

$$\mathbf{Z} = \mathbf{U}_A^{\top}\mathbf{Q}^{\star}\mathbf{U}_B = \mathbf{S} \tag{8.35}$$

and hence (1.33) holds. ■

Notice that (1.33) can be written as:

$$\mathbf{U}_A = \mathbf{Q}^{\star}\mathbf{U}_B\mathbf{S} \tag{8.36}$$

which is a *relaxed* version of (1.16): The permutation matrix in the exact isomorphism case is replaced by an orthogonal matrix.

A Heuristic for Spectral Graph Matching: Let us consider again the exact solution outlined in section 1.3.1. Umeyama suggests a heuristic in order to avoid exhaustive search over all possible 2^n matrices that satisfy (1.15). One may easily notice that:

$$\|\mathbf{P} - \mathbf{U}_A\mathbf{S}\mathbf{U}_B^{\top}\|_F^2 = 2n - 2\mathrm{tr}(\mathbf{U}_A\mathbf{S}(\mathbf{P}\mathbf{U}_B)^{\top}). \tag{8.37}$$

Using Umeyama's notations, $\bar{\mathbf{U}}_A = [|u_{ij}|], \bar{\mathbf{U}}_B = [|v_{ij}|]$ (the entries of $\bar{\mathbf{U}}_A$ are the absolute values of the entries of $\mathbf{U}_A$), one may further notice that:

$$\mathrm{tr}(\mathbf{U}_A\mathbf{S}(\mathbf{P}\mathbf{U}_B)^{\top}) = \sum_{i=1}^{n}\sum_{j=1}^{n} s_j u_{ij} v_{\pi(i)j} \leq \sum_{i=1}^{n}\sum_{j=1}^{n} |u_{ij}||v_{\pi(i)j}| = \mathrm{tr}(\bar{\mathbf{U}}_A\bar{\mathbf{U}}_B^{\top}\mathbf{P}^{\top}). \tag{8.38}$$

The minimization of (1.37) is equivalent to the maximization of (1.38) and the maximal value that can be attained by the latter is n. Using the fact that both $\mathbf{U}_A$ and $\mathbf{U}_B$ are orthogonal matrices, one can easily conclude that:

$$\mathrm{tr}(\bar{\mathbf{U}}_A\bar{\mathbf{U}}_B^{\top}\mathbf{P}^{\top}) \leq n. \tag{8.39}$$

Umeyama concludes that when the two graphs are isomorphic, the optimum permutation matrix maximizes $\mathrm{tr}(\bar{\mathbf{U}}_A\bar{\mathbf{U}}_B^{\top}\mathbf{P}^{\top})$ and this can be solved by the Hungarian algorithm [43].

When the two graphs are not exactly isomorphic, theorem 1 and theorem 2 allow us to relax the permutation matrices to the group of orthogonal matrices. Therefore with similar arguments as above we obtain:

$$\mathrm{tr}(\mathbf{U}_A\mathbf{S}\mathbf{U}_B^{\top}\mathbf{Q}^{\top}) \leq \mathrm{tr}(\bar{\mathbf{U}}_A\bar{\mathbf{U}}_B^{\top}\mathbf{Q}^{\top}) \leq n. \tag{8.40}$$

The permutation matrix obtained with the Hungarian algorithm can be used as an initial solution that can then be improved by some hill-climbing or relaxation technique [8].

The spectral matching solution presented in this section is not directly applicable to large graphs. In the next section we introduce the notion of dimensionality reduction for graphs which will lead to a tractable graph matching solution.

8.4 Graph Embedding and Dimensionality Reduction

For *large* and *sparse* graphs, the results of section 1.3 and Umeyama's method (section 1.3.3) hold only *weakly*. Indeed, one cannot guarantee that all the eigenvalues have multiplicity equal to one: the presence of symmetries causes some of eigenvalues to have an algebraic multiplicity greater than one. Under these circumstances and due to numerical approximations, it might not be possible to properly order the eigenvalues. Moreover, for very large graphs with thousands of vertices it is not practical to compute all its eigenvalue-eigenvector pairs. This means that one has to devise a method that is able to match shapes using a small set of eigenvalues and eigenvectors.

One elegant way to overcome this problem, is to reduce the dimension of the eigenspace, along the line of spectral dimensionality reductions techniques. The eigendecomposition of graph Laplacian matrices (introduced in section 1.2.1) is a popular choice for the purpose of dimensionality reduction [35].

8.4.1 Spectral Properties of the Graph Laplacian

The spectral properties of the Laplacian matrices introduced in section 1.2.1 have been thoroughly studied. They are summarized in table 1.1. We derive some subtle properties of the combinatorial Laplacian which

Laplacian	Null space	Eigenvalues	Eigenvectors
$\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$	$\mathbf{u}_1 = \mathbb{1}$	$0 = \lambda_1 < \lambda_2 \leq \ldots \leq \lambda_n$	$\mathbf{u}_{i>1}^\top \mathbb{1} = 0, \mathbf{u}_i^\top \mathbf{u}_j = \delta_{ij}$
$\tilde{\mathbf{L}} = \tilde{\mathbf{U}}\mathbf{\Gamma}\tilde{\mathbf{U}}^\top$	$\tilde{\mathbf{u}}_1 = \mathbf{D}^{1/2}\mathbb{1}$	$0 = \gamma_1 < \gamma_2 \leq \ldots \leq \gamma_n$	$\tilde{\mathbf{u}}_{i>1}^\top \mathbf{D}^{1/2}\mathbb{1} = 0, \tilde{\mathbf{u}}_i^\top \tilde{\mathbf{u}}_j = \delta_{ij}$
$\tilde{\mathbf{L}}_R = \mathbf{T}\mathbf{\Gamma}\mathbf{T}^{-1}, \mathbf{T} = \mathbf{D}^{-1/2}\tilde{\mathbf{U}}$	$\mathbf{t}_1 = \mathbb{1}$	$0 = \gamma_1 < \gamma_2 \leq \ldots \leq \gamma_n$	$\mathbf{t}_{i>1}^\top \mathbf{D}\mathbb{1} = 0, \mathbf{t}_i^\top \mathbf{D}\mathbf{t}_j = \delta_{ij}$

Table 8.1: Summary of the spectral properties of the Laplacian matrices. Assuming a connected graph, the null eigenvalue (λ_1, γ_1) has multiplicity one. The first non null eigenvalue (λ_2, γ_2) is known as the Fiedler value and its multiplicity is, in general, equal to one. The associated eigenvector is denoted the Fiedler vector [37].

will be useful for the task of shape registration. In particular, we will show that the eigenvectors of the

combinatorial Laplacian can be interpreted as directions of maximum variance (principal components) of the associated embedded shape representation. We note that the embeddings of the normalized and random-walk Laplacians have different spectral properties which make them less interesting for shape registration, i.e., Appendix A.3.

The combinatorial Laplacian. Let $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ be the spectral decomposition of the combinatorial Laplacian with $\mathbf{U}\mathbf{U}^\top = \mathbf{I}$. Let $\mathbf{U}$ be written as:

$$\mathbf{U} = \begin{bmatrix} u_{11} & \dots & u_{1k} & \dots & u_{1n} \\ \vdots & & \vdots & & \vdots \\ u_{n1} & \dots & u_{nk} & \dots & u_{nn} \end{bmatrix} \tag{8.41}$$

Each column of $\mathbf{U}$, $\mathbf{u}_k = (u_{1k} \dots u_{ik} \dots u_{nk})^\top$ is an eigenvector associated with the eigenvalue λ_k. From the definition of $\mathbf{L}$ in (1.2) (see [35]) one can easily see that $\lambda_1 = 0$ and that $\mathbf{u}_1 = \mathbb{1}$ (a constant vector). Hence, $\mathbf{u}_{k\geq 2}^\top \mathbb{1} = 0$ and by combining this with $\mathbf{u}_k^\top \mathbf{u}_k = 1$, we derive the following proposition:

Proposition 1

The components of the non-constant eigenvectors of the combinatorial Laplacian satisfy the following constraints:

$$\textstyle\sum_{i=1}^n u_{ik} = 0, \quad \forall k, 2 \leq k \leq n \tag{8.42}$$

$$-1 < u_{ik} < 1, \quad \forall i,k, 1 \leq i \leq n, 2 \leq k \leq n. \tag{8.43}$$

Assuming a connected graph, λ_1 has multiplicity equal to one [36]. Let's organize the eigenvalues of $\mathbf{L}$ in increasing order: $0 = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_n$. We prove the following proposition [37]:

Proposition 2

For all $k \leq n$, we have $\lambda_k \leq 2\max_i(d_i)$, where d_i is the degree of vertex i.

Proof: The largest eigenvalue of $\mathbf{L}$ corresponds to the maximization of the Rayleigh quotient, or

$$\lambda_n = \max_{\mathbf{u}} \frac{\mathbf{u}^\top \mathbf{L}\mathbf{u}}{\mathbf{u}^\top \mathbf{u}}. \tag{8.44}$$

We have $\mathbf{u}^\top \mathbf{L}\mathbf{u} = \sum_{e_{ij}} w_{ij}(u_i - u_j)^2$. From the inequality $(a-b)^2 \leq 2(a^2+b^2)$ we obtain:

$$\lambda_n \leq \frac{2\sum_{e_{ij}} w_{ij}(u_i^2 + u_j^2)}{\sum_i u_i^2} = \frac{2\sum_i d_i u_i^2}{\sum_i u_i^2} \leq 2\max_i(d_i). \quad \blacksquare \tag{8.45}$$

This ensures an upper limit on the eigenvalues of $\mathbf{L}$. By omitting the zero eigenvalue and associated eigenvector, we can rewrite $\mathbf{L}$ as:

$$\mathbf{L} = \sum_{k=2}^{n} \lambda_k \mathbf{u}_k \mathbf{u}_k^\top. \tag{8.46}$$

Each entry u_{ik} of an eigenvector $\mathbf{u}_k$ can be interpreted as a real-valued function that projects a graph vertex v_i onto that vector. The mean and variance of the set $\{u_{ik}\}_{i=1}^n$ are therefore a measure of how the graph *spreads* when projected onto the k-th eigenvector. This is clarified by the following result:

Proposition 3

The mean $\overline{u}_k$ and the variance σ_{u_k} of an eigenvector $\mathbf{u}_k$. For $2 \leq k \leq n$, and $1 \leq i \leq n$ we have

$$\overline{u}_k = \sum_{i=1}^{n} u_{ik} = 0 \tag{8.47}$$

$$\sigma_{u_k} = \frac{1}{n}\sum_{i=1}^{n}(u_{ik} - \overline{u}_k)^2 = \frac{1}{n} \tag{8.48}$$

Proof: These results can be easily obtained from $\mathbf{u}_{k\geq 2}^\top \mathbb{1} = 0$ and $\mathbf{u}_k^\top \mathbf{u}_k = 1$. ■

These properties will be useful while aligning two Laplacian embeddings and thus registering two 3D shapes.

8.4.2 Principal Component Analysis of a Graph Embedding

The Moore-Penrose pseudo-inverse of the Laplacian can be written as:

$$\begin{aligned} \mathbf{L}^\dagger &= \mathbf{U}\mathbf{\Lambda}^{-1}\mathbf{U}^\top \\ &= (\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top)^\top(\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top) \\ &= \mathbf{X}^\top\mathbf{X} \end{aligned} \tag{8.49}$$

where $\mathbf{\Lambda}^{-1} = \mathrm{diag}(0, 1/\lambda_2, \ldots, 1/\lambda_n)$.

The symmetric semi-definite positive matrix $\mathbf{L}^\dagger$ is a *Gram* matrix with the same eigenvectors as those of the graph Laplacian. When omitting the null eigenvalue and associated constant eigenvector, $\mathbf{X}$ becomes a $(n-1) \times n$ matrix whose columns are the coordinates of the graph's vertices in an *embedded (or feature) space*, i.e., $\mathbf{X} = [\mathbf{x}_1 \ldots \mathbf{x}_j \ldots \mathbf{x}_n]$. It is interesting to note that the entries of $\mathbf{L}^\dagger$ may be viewed as *kernel* dot-products, or a Gram matrix [44]. The Gram-matrix representation allows us to embed the graph in an Euclidean feature-space where each vertex v_j of the graph is a feature point represented as $\mathbf{x}_j$.

The left pseudo-inverse operator of the Laplacian $\mathbf{L}$, satisfying $\mathbf{L}^\dagger\mathbf{L}\mathbf{u} = \mathbf{u}$ for any $\mathbf{u} \perp \mathrm{null}(\mathbf{L})$, is also called the *Green function* of the heat equation. Under the assumption that the graph is connected and thus $\mathbf{L}$

has an eigenvalue $\lambda_1 = 0$ with multiplicity 1, we obtain:

$$\mathbf{L}^{\dagger} = \sum_{k=2}^{n} \frac{1}{\lambda_k} \mathbf{u}_k \mathbf{u}_k^{\top}. \tag{8.50}$$

The Green function is intimately related to random walks on graphs, and can be interpreted probabilistically as follows. Given a Markov chain such that each graph vertex is the state, and the transition from vertex v_i is possible to any adjacent vertex $v_j \sim v_i$ with probability w_{ij}/d_i, the expected number of steps required to reach vertex v_j from v_i, called the *access* or *hitting time* $O(v_i, v_j)$. The expected number of steps in a round trip from v_i to v_j is called the *commute-time distance*: $\mathrm{CTD}^2(v_i, v_j) = O(v_i, v_j) + O(v_j, v_i)$. The commute-time distance [45] can be expressed in terms of the entries of $\mathbf{L}^{\dagger}$:

$$\begin{aligned}
\mathrm{CTD}^2(v_i, v_j) &= \mathrm{Vol}(\mathcal{G})(\mathbf{L}^{\dagger}(i,i) + \mathbf{L}^{\dagger}(j,j) - 2\mathbf{L}^{\dagger}(i,j)) \\
&= \mathrm{Vol}(\mathcal{G}) \left(\sum_{k=2}^{n} \frac{1}{\lambda_k} \mathbf{u}_{ik}^2 + \sum_{k=2}^{n} \frac{1}{\lambda_k} \mathbf{u}_{jk}^2 - 2 \sum_{k=2}^{n} \frac{1}{\lambda_k} \mathbf{u}_{ik} \mathbf{u}_{jk} \right) \\
&= \mathrm{Vol}(\mathcal{G}) \sum_{k=2}^{n} \left(\lambda_k^{-1/2} (\mathbf{u}_{ik} - \mathbf{u}_{jk}) \right)^2 \\
&= \mathrm{Vol}(\mathcal{G}) \|\mathbf{x}_i - \mathbf{x}_j\|^2,
\end{aligned} \tag{8.51}$$

where the volume of the graph, $\mathrm{Vol}(\mathcal{G})$ is the sum of the degrees of all the graph vertices. The CTD function is positive-definite and sub-additive, thus defining a *metric* between the graph vertices, referred to as *commute-time* (or *resistance*) *distance* [46]. The CTD is inversely related to the number and length of paths connecting two vertices. Unlike the shortest-path (geodesic) distance, CTD captures the connectivity structure of the graph volume rather than a single path between the two vertices. The great advantage of the commute-time distance over the shortest geodesic path is that it is robust to topological changes and therefore is well suited for characterizing complex shapes. Since the volume is a graph constant, we obtain:

$$\mathrm{CTD}^2(v_i, v_j) \propto \|\mathbf{x}_i - \mathbf{x}_j\|^2. \tag{8.52}$$

Hence, the Euclidean distance between any two feature points $\mathbf{x}_i$ and $\mathbf{x}_j$ is the commute time distance between the graph vertex v_i and v_j.

Using the first K non-null eigenvalue-eigenvector pairs of the Laplacian $\mathbf{L}$, the *commute-time embedding* of the graph's nodes corresponds to the column vectors of the $K \times n$ matrix $\mathbf{X}$:

$$\mathbf{X}_{K \times n} = \mathbf{\Lambda}_K^{-1/2} (\mathbf{U}_{n \times K})^{\top} = [\mathbf{x}_1 \ldots \mathbf{x}_j \ldots \mathbf{x}_n]. \tag{8.53}$$

From (1.43) and (1.53) one can easily infer lower and upper bounds for the i-th coordinate of $\mathbf{x}_j$:

$$-\lambda_i^{-1/2} < x_{ji} < \lambda_i^{-1/2}. \tag{8.54}$$

The last equation implies that the graph embedding stretches along the eigenvectors with a factor that is inversely proportional to the square root of the eigenvalues. Theorem 3 below characterizes the smallest non-null K eigenvalue-eigenvector pairs of $\mathbf{L}$ as the directions of maximum variance (the principal components) of the commute-time embedding.

Theorem 3

The largest eigenvalue-eigenvector pairs of the pseudo-inverse of the combinatorial Laplacian matrix are the principal components of the commute-time embedding, i.e., the points $\mathbf{X}$ *are zero-centered and have a diagonal covariance matrix.*

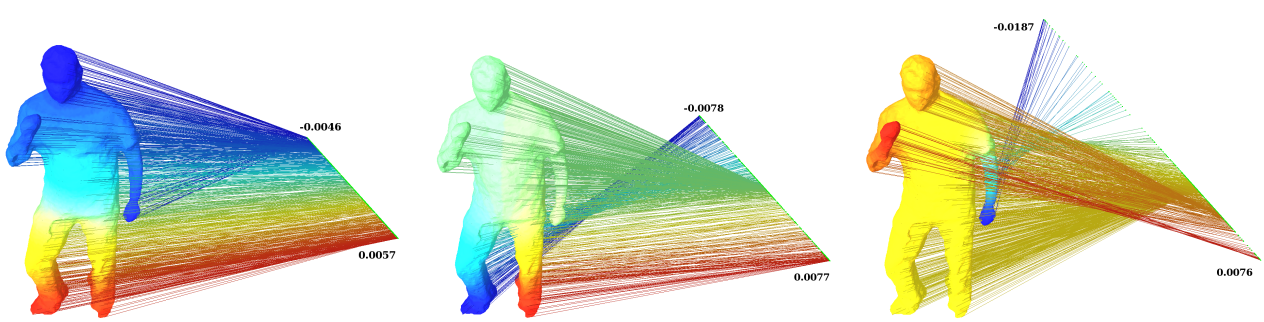
Proof: Indeed, from (1.47) we obtain a zero-mean while from (1.53) we obtain a diagonal covariance matrix:

$$\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i = \frac{1}{n}\mathbf{\Lambda}^{-\frac{1}{2}}\begin{pmatrix} \sum_{i=1}^{n}\mathbf{u}_{i2} \\ \vdots \\ \sum_{i=1}^{n}\mathbf{u}_{ik+1} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \tag{8.55}$$

$$\mathbf{\Sigma}_X = \frac{1}{n}\mathbf{X}\mathbf{X}^\top = \frac{1}{n}\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top\mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}} = \frac{1}{n}\mathbf{\Lambda}^{-1} \tag{8.56}$$

■.

Figure 1.3 shows the projection of graph (in this case 3D shape represented as meshes) vertices on eigenvectors.



(a) (b) (c)

Figure 8.3: This is an illustration of the concept of the PCA of a graph embedding. The graph's vertices are projected onto the second, third and fourth eigenvectors of the Laplacian matrix. These eigenvectors can be viewed as the principal directions of the shape.

8.4.3 Choosing the Dimension of the Embedding

A direct consequence of theorem 3 is that the embedded graph representation is centered and the eigenvectors of the combinatorial Laplacian are the directions of maximum variance. The *principal* eigenvectors correspond to the eigenvectors associated with the K *largest* eigenvalues of the $\mathbf{L}^{\dagger}$, i.e., $\lambda_2^{-1} \geq \lambda_3^{-1} \geq \ldots \geq \lambda_K^{-1}$. The variance along vector $\mathbf{u}_k$ is λ_k^{-1}/n. Therefore, the total variance can be computed from the trace of the $\mathbf{L}^{\dagger}$ matrix :

$$\mathrm{tr}(\mathbf{\Sigma}_X) = \frac{1}{n}\mathrm{tr}(\mathbf{L}^{\dagger}). \tag{8.57}$$

A standard way of choosing the principal components is to use the *scree diagram*:

$$\theta(K) = \frac{\sum_{k=2}^{K+1} \lambda_k^{-1}}{\sum_{k=2}^{n} \lambda_k^{-1}}. \tag{8.58}$$

The selection of the first K principal eigenvectors therefore depends on the spectral fall-off of the inverses of the eigenvalues. In spectral graph theory, the dimension K is chosen on the basis of the existence of an eigengap, such that $\lambda_{K+2} - \lambda_{K+1} > t$ with $t > 0$. In practice it is extremely difficult to find such an eigengap, in particular in the case of sparse graphs that correspond to a discretized manifold. Instead, we propose to select the dimension of the embedding in the following way. Notice that (1.58) can be written as $\theta(K) = A/(A+B)$ with $A = \sum_{k=2}^{K+1} \lambda_k^{-1}$ and $B = \sum_{k=K+2}^{n} \lambda_k^{-1}$. Moreover, from the fact that the λ_k's are arranged in increasing order, we obtain $B \leq (n-K-1)\lambda_{K+1}^{-1}$. Hence:

$$\theta_{\min} \leq \theta(K) \leq 1, \tag{8.59}$$

with

$$\theta_{\min} = \frac{\sum_{k=2}^{K+1} \lambda_k^{-1}}{\sum_{k=2}^{K} \lambda_k^{-1} + (n-K)\lambda_{K+1}^{-1}}. \tag{8.60}$$

This lower bound can be computed from the K smallest non null eigenvalues of the combinatorial Laplacian matrix. Hence, one can choose K such that the sum of the first K eigenvalues of the $\mathbf{L}^{\dagger}$ matrix is a good approximation of the total variance, e.g., $\theta_{\min} = 0.95$.

8.4.4 Unit Hyper-sphere Normalization

One disadvantage of the standard embeddings is that, when two shapes have large difference in sampling the embeddings will differ by a significant scale factor. In order to avoid this we can *normalize* the embedding such that the vertex coordinates lie on a unit sphere of dimension K, which yields:

$$\hat{\mathbf{x}}_i = \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|}. \tag{8.61}$$

In more detail, the k-th coordinate of $\hat{\mathbf{x}}_i$ writes as:

$$\hat{\mathbf{x}}_{ik} = \frac{\lambda_k^{-\frac{1}{2}} \mathbf{u}_{ik}}{\left(\sum_{l=2}^{K+1} \lambda_l^{-\frac{1}{2}} \mathbf{u}_{il}^2\right)^{1/2}}. \tag{8.62}$$

8.5 Spectral Shape Matching

In the previous sections we discussed solutions for the exact and inexact graph isomorphism problems, we recalled the main spectral properties of the combinatorial graph Laplacian, and we provided a novel analysis of the commute-time embedding that allows to interpret the latter in terms of the PCA of a graph, and to select the appropriate dimension $K \ll n$ of the associated embedded metric space. In this section we address the problem of 3D shape registration and we illustrate how the material developed above can be exploited in order to build a robust algorithm for spectral shape matching.

Let's consider two shapes described by two graphs, $\mathcal{G}_A$ and $\mathcal{G}_B$ where $|\mathcal{V}_A| = n$ and $|\mathcal{V}_B| = m$. Let $\mathbf{L}_A$ and $\mathbf{L}_B$ be their corresponding graph Laplacians. Without loss of generality, one can choose the same dimension $K \ll \min(n,m)$ for the two embeddings. This yields the following eigen decompositions:

$$\mathbf{L}_A = \mathbf{U}_{n\times K} \mathbf{\Lambda}_K (\mathbf{U}_{n\times K})^\top \tag{8.63}$$

$$\mathbf{L}_B = \mathbf{U}'_{m\times K} \mathbf{\Lambda}'_K (\mathbf{U}'_{m\times K})^\top. \tag{8.64}$$

For each one of these graphs, one can build two *isomorphic* embedded representations, as follows:

- An *unnormalized Laplacian embedding* that uses the K rows of $\mathbf{U}_{n\times K}$ as the Euclidean coordinates of the vertices of $\mathcal{G}_A$ (as well as the K rows of $\mathbf{U}'_{m\times K}$ as the Euclidean coordinates of the vertices of $\mathcal{G}_B$), and
- A *normalized commute-time embedding* given by (1.61), i.e., $\hat{\mathbf{X}}_A = [\hat{\mathbf{x}}_1 \ldots \hat{\mathbf{x}}_j \ldots \hat{\mathbf{x}}_n]$ (as well as $\hat{\mathbf{X}}_B = [\hat{\mathbf{x}}'_1 \ldots \hat{\mathbf{x}}'_j \ldots \hat{\mathbf{x}}'_m]$). We recall that each column $\hat{\mathbf{x}}_j$ (and respectively $\hat{\mathbf{x}}'_j$) is a K-dimensional vector corresponding to a vertex v_j of $\mathcal{G}_A$ (and respectively v'_j of $\mathcal{G}_B$).

8.5.1 Maximum Subgraph Matching and Point Registration

Let's apply the graph isomorphism framework of Section 1.3 to the two graphs. They are embedded into two congruent spaces of dimension $\mathbb{R}^K$. If the smallest K non-null eigenvalues associated with the two embeddings are distinct and can be ordered, i.e.:

$$\lambda_2 < \ldots < \lambda_k < \ldots < \lambda_{K+1} \tag{8.65}$$

$$\lambda'_2 < \ldots < \lambda'_k < \ldots < \lambda'_{K+1} \tag{8.66}$$

then, the Umeyama method could be applied. If one uses the unnormalized Laplacian embeddings just defined, (1.33) becomes:

$$\mathbf{Q}^{\star} = \mathbf{U}_{n\times K}\mathbf{S}_K(\mathbf{U}'_{m\times K})^{\top} \tag{8.67}$$

Notice that here the sign matrix $\mathbf{S}$ defined in 1.33 became a $K \times K$ matrix denoted by $\mathbf{S}_K$. We now assume that the eigenvalues $\{\lambda_2,\ldots,\lambda_{K+1}\}$ and $\{\lambda'_2,\ldots,\lambda'_{K+1}\}$ *cannot be reliably ordered.* This can be modeled by multiplication with a $K \times K$ permutation matrix $\mathbf{P}_K$:

$$\mathbf{Q} = \mathbf{U}_{n\times K}\mathbf{S}_K\mathbf{P}_K(\mathbf{U}'_{m\times K})^{\top} \tag{8.68}$$

Pre-multiplication of $(\mathbf{U}'_{m\times K})^{\top}$ with $\mathbf{P}_K$ permutes its rows such that $\mathbf{u}'_k \rightarrow \mathbf{u}'_{\pi(k)}$. Each entry q_{ij} of the $n \times m$ matrix $\mathbf{Q}$ can therefore be written as:

$$q_{ij} = \sum_{k=2}^{K+1} s_k u_{ik} u'_{j\pi(k)} \tag{8.69}$$

Since both $\mathbf{U}_{n\times K}$ and $\mathbf{U}'_{m\times K}$ are column-orthonormal matrices, the dot-product defined by (1.69) is equivalent to the cosine of the angle between two K-dimensional vectors. This means that each entry of $\mathbf{Q}$ is such that $-1 \leq q_{ij} \leq +1$ and that two vertices v_i and v'_j are matched if q_{ij} is close to 1.

One can also use the normalized commute-time coordinates and define an equivalent expression as above:

$$\hat{\mathbf{Q}} = \hat{\mathbf{X}}^{\top}\mathbf{S}_K\mathbf{P}_K\hat{\mathbf{X}}' \tag{8.70}$$

with:

$$\hat{q}_{ij} = \sum_{k=2}^{K+1} s_k \hat{x}_{ik} \hat{x}'_{j\pi(k)} \tag{8.71}$$

Because both sets of points $\hat{\mathbf{X}}$ and $\hat{\mathbf{X}}'$ lie on a K-dimensional unit hyper-sphere, we also have $-1 \leq \hat{q_{ij}} \leq +1$.

It should however be emphasized that the rank of the $n \times m$ matrices $\mathbf{Q}, \hat{\mathbf{Q}}$ is equal to K. Therefore, these matrices cannot be viewed as *relaxed permutation matrices* between the two graphs. In fact they define many-to-many correspondences between the vertices of the first graph and the vertices of the second graph, this being due to the fact that the graphs are embedded on a low-dimensional space. This is one of the main differences between our method proposed in the next section and the Umeyama method, as well as many other subsequent methods, that use all eigenvectors of the graph. As it will be explained below, our formulation leads to a shape matching method that will alternate between aligning their eigenbases and finding a vertex-to-vertex assignment.

It is possible to extract a one-to-one assignment matrix from $\mathbf{Q}$ (or from $\hat{\mathbf{Q}}$) using either dynamic programming or an assignment method technique such as the Hungarian algorithm. Notice that this assignment is conditioned by the choice of a sign matrix $\mathbf{S}_K$ and of a permutation matrix $\mathbf{P}_K$, i.e., $2^K K!$ possibilities,

and that not all these choices correspond to a valid sub-isomorphism between the two graphs. Let's consider the case of the normalized commute-time embedding; there is an equivalent formulation for the unnormalized Laplacian embedding. The two graphs are described by two sets of points, $\hat{\mathbf{X}}$ and $\hat{\mathbf{X}}'$, both lying onto the K-dimensional unity hyper-sphere. The $K \times K$ matrix $\mathbf{S}_K\mathbf{P}_K$ transforms one graph embedding onto the other graph embedding. Hence, one can write $\hat{\mathbf{x}}_i = \mathbf{S}_K\mathbf{P}_K\hat{\mathbf{x}}'_j$ if vertex v_i matches v_j. More generally Let $\mathbf{R}_K = \mathbf{S}_K\mathbf{P}_K$ and let's extend the domain of $\mathbf{R}_K$ to all possible orthogonal matrices of size $K \times K$, namely $\mathbf{R}_K \in O_K$ or the *orthogonal group* of dimension K. We can now write the following criterion whose minimization over $\mathbf{R}_K$ guarantees an optimal solution for registering the vertices of the first graph with the vertices of the second graph:

$$\min_{R_K} \sum_{i=1}^{n} \sum_{j=1}^{m} \hat{q}_{ij} \|\hat{\mathbf{x}}_i - \mathbf{R}_K\hat{\mathbf{x}}'_j\|^2 \tag{8.72}$$

One way to solve minimization problems such as (1.72) is to use a point registration algorithm that alternates between (i) estimating the $K \times K$ orthogonal transformation $\mathbf{R}_K$, which aligns the K-dimensional coordinates associated with the two embeddings, and (ii) updating the assignment variables $\hat{q}_{ij}$. This can be done using either ICP-like methods (the $\hat{q}_{ij}$'s are binary variables), or EM-like methods (the $\hat{q}_{ij}$'s are posterior probabilities of assignment variables). As we just outlined above, matrix $\mathbf{R}_K$ belongs to the orthogonal group O_K. Therefore this framework differs from standard implementations of ICP and EM algorithms that usually estimate a 2-D or 3-D *rotation* matrix which belong to the *special orthogonal group*.

It is well established that ICP algorithms are easily trapped in local minima. The EM algorithm recently proposed in [34] is able to converge to a good solution starting with a rough initial guess and is robust to the presence of outliers. Nevertheless, the algorithm proposed in [34] performs well under *rigid transformations* (rotation and translation), whereas in our case we have to estimate a more general orthogonal transformation that incorporates both rotations and reflections. Therefore, before describing in detail an EM algorithm well suited for solving the problem at hand, we discuss the issue of estimating an initialization for the transformation aligning the K eigenvectors of the first embedding with those of the second embedding and we propose a practical method for initializing this transformation (namely, matrices $\mathbf{S}_K$ and $\mathbf{P}_K$ in (1.70)) based on comparing the histograms of these eigenvectors, or *eigensignatures*.

8.5.2 Aligning Two Embeddings Based on Eigensignatures

Both the unnormalized Laplacian embedding and the normalized commute-time embedding of a graph are represented in a metric space spanned by the eigenvectors of the Laplacian matrix, namely the n-dimensional vectors $\{\mathbf{u}_2, \ldots, \mathbf{u}_k, \ldots, \mathbf{u}_{K+1}\}$, where n is the number of graph vertices. They correspond to *eigenfunctions* and each such eigenfunction maps the graph's vertices onto the real line. More precisely, the k-th

eigenfunction maps a vertex v_i onto u_{ik}. Propositions 1 and 3 revealed interesting statistics of the sets $\{u_{1k},\ldots,u_{ik},\ldots,u_{nk}\}_{k=2}^{K+1}$. Moreover, theorem 3 provided an interpretation of the eigenvectors in terms of principal directions of the embedded shape. One can therefore conclude that the probability distribution of the components of an eigenvector have interesting properties that make them suitable for comparing two shapes, namely $-1 < u_{ik} < +1$, $\overline{u}_k = 1/n\sum_{i=1}^{n} u_{ik} = 0$, and $\sigma_k = 1/n\sum_{i=1}^{n} u_{ik}^2 = 1/n$. This means that one can build a histogram for each eigenvector and that all these histograms share the same bin width w and the same number of bins b [47]:

$$w_k = \frac{3.5\sigma_k}{n^{1/3}} = \frac{3.5}{n^{4/3}} \tag{8.73}$$

$$b_k = \frac{\sup_i u_{ik} - \inf_i u_{ik}}{w_k} \approx \frac{n^{4/3}}{2}. \tag{8.74}$$

We claim that these histograms are eigenvector signatures which are invariant under graph isomorphism. Indeed, let's consider the Laplacian $\mathbf{L}$ of a shape and we apply the isomorphic transformation $\mathbf{PLP}^\top$ to this shape, where $\mathbf{P}$ is a permutation matrix. If $\mathbf{u}$ is an eigenvector of $\mathbf{L}$, it follows that $\mathbf{Pu}$ is an eigenvector of $\mathbf{PLP}^\top$ and therefore, while the order of the components of $\mathbf{u}$ are affected by this transformation, their frequency and hence their probability distribution remain the same. Hence, one may conclude that such a histogram may well be viewed as an *eigensignature*.

We denote with $H\{\mathbf{u}\}$ the histogram formed with the components of $\mathbf{u}$ and let $C(H\{\mathbf{u}\},H\{\mathbf{u}'\})$ be a similarity measure between two histograms. From the eigenvector properties just outlined, it is straightforward to notice that $H\{\mathbf{u}\} \neq H\{-\mathbf{u}\}$: These two histograms are mirror symmetric. Hence, the histogram is not invariant to the sign of an eigenvector. Therefore one can use the eigenvectors' histograms to estimate both the permutation matrix $\mathbf{P}_K$ and the sign matrix $\mathbf{S}_K$ in (1.70). The problem of finding one-to-one assignments $\{\mathbf{u}_k \leftrightarrow s_k\mathbf{u}'_{\pi(k)}\}_{k=2}^{K+1}$ between the two sets of eigenvectors associated with the two shapes is therefore equivalent to the problem of finding one-to-one assignments between their histograms.

Let $\mathbf{A}_K$ be an assignment matrix between the histograms of the first shape and the histograms of the second shape. Each entry of this matrix is defined by:

$$a_{kl} = \sup[C(H\{\mathbf{u}_k\},H\{\mathbf{u}'_l\});C(H\{\mathbf{u}_k\},H\{-\mathbf{u}'_l\})] \tag{8.75}$$

Similarly, we define a matrix $\mathbf{B}_K$ that accounts for the *sign assignments*:

$$b_{kl} = \begin{cases} +1 & \text{if} \quad C(H\{\mathbf{u}_k\},H\{\mathbf{u}'_l\}) \geq C(H\{\mathbf{u}_k\},H\{-\mathbf{u}'_l\}) \\ -1 & \text{if} \quad C(H\{\mathbf{u}_k\},H\{\mathbf{u}'_l\}) < C(H\{\mathbf{u}_k\},H\{-\mathbf{u}'_l\}) \end{cases} \tag{8.76}$$

Extracting a permutation matrix $\mathbf{P}_K$ from $\mathbf{A}_K$ is an instance of the bipartite maximum matching problem and the Hungarian algorithm is known to provide an optimal solution to this assignment problem [43].

Moreover, one can use the estimated $\mathbf{P}_K$ to extract a sign matrix $\mathbf{S}_K$ from $\mathbf{B}_K$. Algorithm 1 estimates an alignment between two embeddings.

Algorithm 1 *Alignment of Two Laplacian Embeddings*

input : Histograms associated with eigenvectors $\{\mathbf{u}_k\}_{k=2}^{K+1}$ and $\{\mathbf{u}'_k\}_{k=2}^{K+1}$.

output : A permutation matrix $\mathbf{P}_K$ and a sign matrix $\mathbf{S}_K$.

1: Compute the assignment matrices $\mathbf{A}_K$ and $\mathbf{B}_K$.

2: Compute $\mathbf{P}_K$ from $\mathbf{A}_K$ using the Hungarian algorithm.

3: Compute the sign matrix $\mathbf{S}_K$ using $\mathbf{P}_K$ and $\mathbf{B}_K$.

Figure 1.4 illustrates the utility of the histogram of eigenvectors as eigensignatures for solving the problem of sign flip and change in eigenvector ordering by computing histogram matching. It is interesting to observe that a threshold on the histogram matching score (1.75) allows us to discard the eigenvectors with low similarity cost. Hence, starting with large K obtained using (1.60), we can limit the number of eigenvectors to just a few, which will be suitable for EM based point registration algorithm proposed in the next section.

8.5.3 An EM Algorithm for Shape Matching

As explained in section 1.5.1, the maximum subgraph matching problem reduces to a point registration problem in K dimensional metric space spanned by the eigenvectors of graph Laplacian where two shapes are represented as point clouds. The initial alignment of Laplacian embeddings can be obtained by matching the histogram of eigenvectors as described in the previous section. In this section we propose an EM algorithm for 3D shape matching that computes a probabilistic vertex-to-vertex assignment between two shapes. The proposed method alternates between the step to estimate an orthogonal transformation matrix associated with the alignment of the two shape embeddings and the step to compute a point-to-point probabilistic assignment variable.

The method is based on a parametric probabilistic model, namely maximum likelihood with missing data. Let us consider the Laplacian embedding of two shapes, i.e., (1.53) : $\hat{\mathbf{X}} = \{\hat{\mathbf{x}}_i\}_{i=1}^n, \hat{\mathbf{X}}' = \{\hat{\mathbf{x}}'_j\}_{j=1}^m$, with $\hat{\mathbf{X}}, \hat{\mathbf{X}}' \subset \mathbb{R}^K$. Without loss of generality, we assume that the points in the first set, $\hat{\mathbf{X}}$ are cluster centers of a Gaussian mixture model (GMM) with n clusters and an additional uniform component that accounts for outliers and unmatched data. The matching $\hat{\mathbf{X}} \leftrightarrow \hat{\mathbf{X}}'$ will consist in fitting the Gaussian mixture to the set $\hat{\mathbf{X}}'$.

Let this Gaussian mixture undergo a $K \times K$ transformation $\mathbf{R}$ (for simplicity, we omit the index K) with $\mathbf{R}^\top \mathbf{R} = \mathbf{I}_K, \det(\mathbf{R}) = \pm 1$, more precisely $\mathbf{R} \in O_K$, the group of orthogonal matrices acting on $\mathbb{R}^K$. Hence, each cluster in the mixture is parametrized by a prior p_i, a cluster mean $\mu_i = \mathbf{R}\hat{\mathbf{x}}_i$, and a covariance matrix

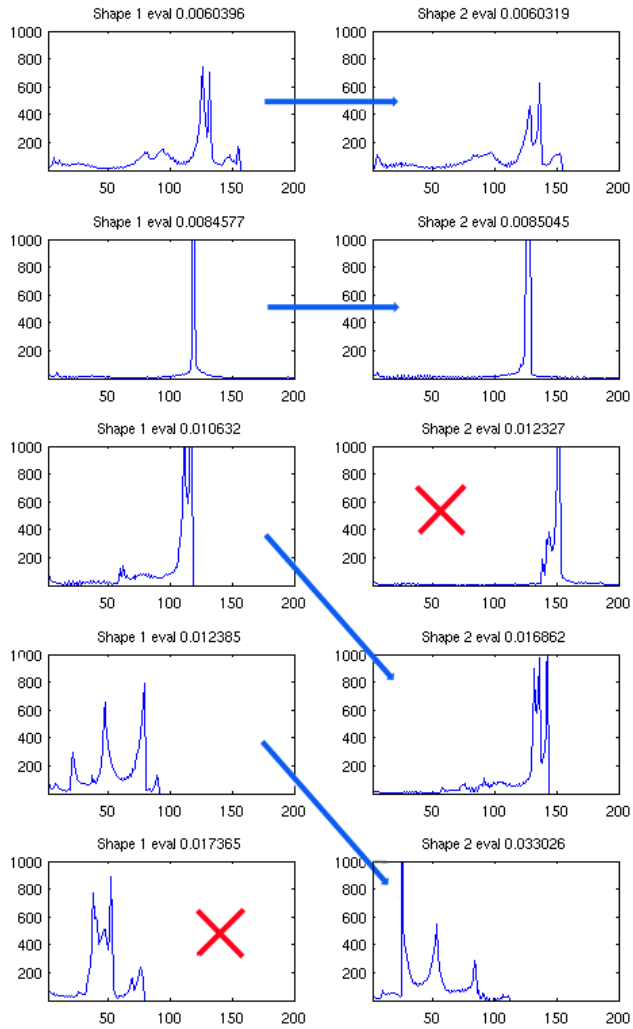


Figure 8.4: An illustration of applicability of eigenvector histogram as eigensignature to detect sign flip and eigenvector ordering change. The blue line shows matched eigenvector pairs and the red-cross depicts discarded eigenvectors.

$\mathbf{\Sigma}_i$. It will be assumed that all the clusters in the mixture have the same priors, $\{p_i = \pi_{\text{in}}\}_{i=1}^n$, and the same isotropic covariance matrix, $\{\mathbf{\Sigma}_i = \sigma \mathbf{I}_K\}_{i=1}^n$. This parametrization leads to the following *observed-data log-likelihood* (with $\pi_{\text{out}} = 1 - n\pi_{\text{in}}$ and $\mathcal{U}$ is the uniform distribution):

$$\log P(\hat{\mathbf{X}}') = \sum_{j=1}^{m} \log \left(\sum_{i=1}^{n} \pi_{\text{in}} \mathcal{N}(\hat{\mathbf{x}}'_j | \mu_i, \sigma) + \pi_{\text{out}} \mathcal{U} \right) \tag{8.77}$$

It is well known that the direct maximization of (1.77) is not tractable and it is more practical to maximize the *expected complete-data log-likelihood* using the EM algorithm, where "complete-data" refers to both the

observed data (the points $\hat{\mathbf{X}}'$) and the missing data (the data-to-cluster assignments). In our case, the above expectation writes (see [34] for details):

$$\mathcal{E}(\mathbf{R},\sigma) = -\frac{1}{2}\sum_{j=1}^{m}\sum_{i=1}^{n}\alpha_{ji}(\|\hat{\mathbf{x}}'_j - \mathbf{R}\hat{\mathbf{x}}_i\|^2 + k\log\sigma), \tag{8.78}$$

where α_{ji} denotes the posterior probability of an assignment: $\hat{\mathbf{x}}'_j \leftrightarrow \hat{\mathbf{x}}_i$:

$$\alpha_{ji} = \frac{\exp(-\|\hat{\mathbf{x}}'_j - \mathbf{R}\hat{\mathbf{x}}_i\|^2/2\sigma)}{\sum_{q=1}^{n}\exp(-\|\hat{\mathbf{x}}'_j - \mathbf{R}\hat{\mathbf{x}}_q\|^2/2\sigma) + \emptyset\sigma^{k/2}}, \tag{8.79}$$

where $\emptyset$ is a constant term associated with the uniform distribution $\mathcal{U}$. Notice that one easily obtains the posterior probability of a data point to remain unmatched, $\alpha_{jn+1} = 1 - \sum_{i=1}^{n}\alpha_{ij}$. This leads to the shape matching procedure outlined in Algorithm 2.

Algorithm 2 *EM for shape matching*

input : Two embedded shapes $\hat{\mathbf{X}}$ and $\hat{\mathbf{X}}'$;

output : Dense correspondences $\hat{\mathbf{X}} \leftrightarrow \hat{\mathbf{X}}'$ between the two shapes;

1: *Initialization:* Set $\mathbf{R}^{(0)} = \mathbf{S}_K\mathbf{P}_K$ choose a large value for the variance $\sigma^{(0)}$;

2: *E-step:* Compute the current posteriors $\alpha_{ij}^{(q)}$ from the current parameters using (1.79);

3: *M-step:* Compute the new transformation $\mathbf{R}^{(q+1)}$ and the new variance $\sigma^{(q+1)}$ using the current posteriors:

$$\mathbf{R}^{(q+1)} = \arg\min_{\mathbf{R}}\sum_{i,j}\alpha_{ij}^{(q)}\|x'_j - \mathbf{R}x_i\|^2$$

$$\sigma^{(q+1)} = \sum_{i,j}\alpha_{ij}^{(q)}\|\hat{\mathbf{x}}'_j - \mathbf{R}^{(q+1)}\hat{\mathbf{x}}_i\|^2/k\sum_{i,j}\alpha_{ij}^{(q)}$$

4: *MAP:* Accept the assignment $\hat{\mathbf{x}}'_j \leftrightarrow \hat{\mathbf{x}}_i$ if $\max_i \alpha_{ij}^{(q)} > 0.5$.

8.6 Experiments and Results

We have performed several 3D shape registration experiments to evaluate the proposed method. In the first experiment, 3D shape registration is performed on 138 high-resolution (10K-50K vertices) triangular meshes from the publicly available TOSCA dataset [29]. The dataset includes 3 shape classes (human, dog, horse) with simulated transformations. Transformations are split into 9 classes (isometry, topology, small and big holes, global and local scaling, noise, shot noise, sampling). Each transformation class appears in five different strength levels. An estimate of average geodesic distance to ground truth correspondence was computed for performance evaluation (see [29] for details).

We evaluate our method in two settings. In the first setting SM1 we use the commute-time embedding (1.53) while in the second setting SM2 we use the unit hyper-sphere normalized embedding (1.61).

	Strength									
Transform	1		≤ 2		≤ 3		≤ 4		≤ 5	
	SM1	SM2	SM1	SM2	SM1	SM2	SM1	SM2	SM1	SM2
Isometry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Topology	6.89	5.96	7.92	6.76	7.92	7.14	8.04	7.55	8.41	8.13
Holes	7.32	5.17	8.39	5.55	9.34	6.05	9.47	6.44	12.47	10.32
Micro holes	0.37	0.68	0.39	0.70	0.44	0.79	0.45	0.79	0.49	0.83
Scale	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Local scale	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Sampling	11.43	10.51	13.32	12.08	15.70	13.65	18.76	15.58	22.63	19.17
Noise	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Shot noise	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average	2.88	2.48	3.34	2.79	3.71	3.07	4.08	3.37	4.89	4.27

Table 8.2: 3D shape registration error estimates (average geodesic distance to ground truth correspondences) using proposed spectral matching method with commute-time embedding (SM1) and unit hyper-sphere normalized embedding (SM2).

Table 1.2 shows the error estimates for dense shape matching using proposed spectral matching method. In the case of some transforms, the proposed method yields zero error because the two meshes had identical triangulations. Figure 1.5 shows some matching results. The colors emphasize the correct matching of body parts while we show only 5% of matches for better visualization. In Figure 1.5(e) the two shapes have large difference in the sampling rate. In this case the matching near the shoulders is not fully correct since we used the commute-time embedding.

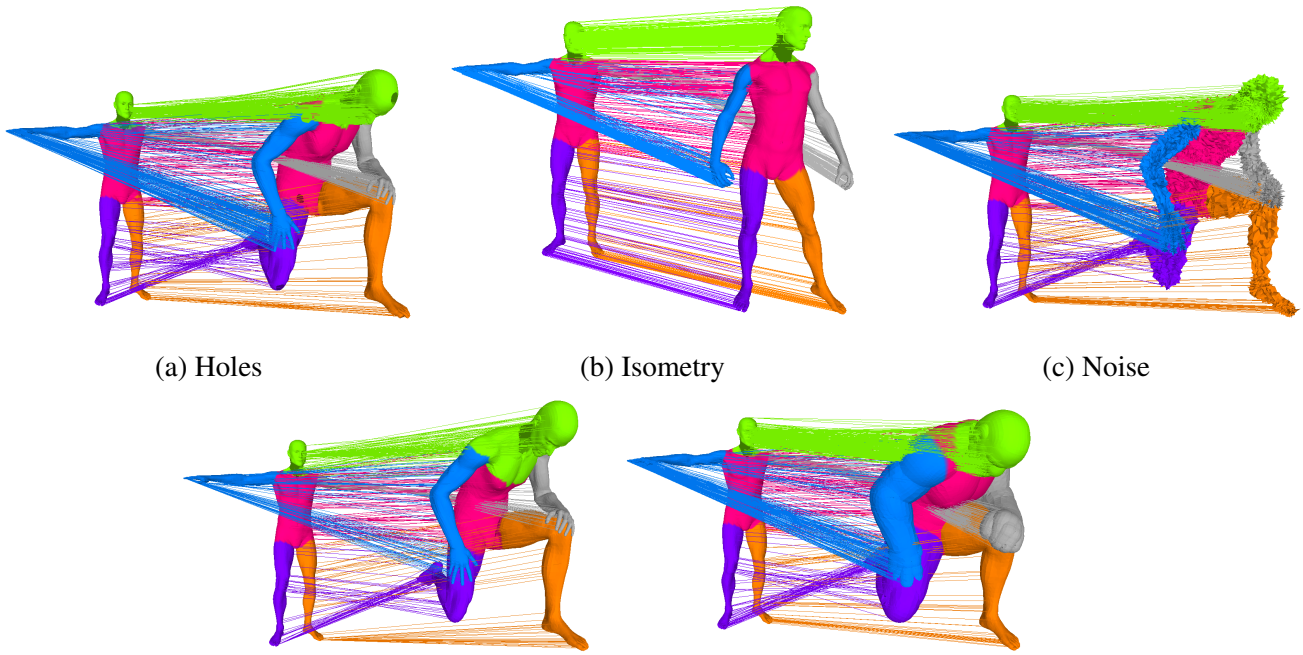
Table 1.3 summarizes the comparison of proposed spectral matching method (SM1 and SM2) with generalized multidimensional scaling (GMDS) based matching algorithm introduced in [17] and the Laplace-Beltrami matching algorithm proposed in [10] with two settings LB1 (uses graph Laplacian) and LB2 (uses cotangent weights). GMDS computes correspondence between two shapes by trying to embed one shape into another with minimum distortion. LB1 and LB2 algorithms combines the surface descriptors based on the eigendecomposition of the Laplace-Beltrami operator and the geodesic distances measured on the shapes when calculating the correspondence quality. The above results in a quadratic optimization problem formulation for correspondence detection, and its minimizer is the best possible correspondence. The proposed method clearly outperform the other two methods with minimum average error estimate computed over all the transformations in the dataset.

	Strength				
Method	1	≤ 2	≤ 3	≤ 4	≤ 5
LB1	10.61	15.48	19.01	23.22	23.88
LB2	15.51	18.21	22.99	25.26	28.69
GMDS	39.92	36.77	35.24	37.40	39.10
SM1	2.88	3.34	3.71	4.08	4.89
SM2	2.48	2.79	3.07	3.37	4.27

Table 8.3: Average shape registration error estimates over all transforms (average geodesic distance to ground truth correspondences) computed using proposed methods (SM1 and SM2), GMDS [17] and LB1, LB2 [10].

	Strength		
Transform	1	≤ 3	≤ 5
Isometry	SM1,SM2	SM1,SM2	SM1,SM2
Topology	SM2	SM2	SM2
Holes	SM2	SM2	SM2
Micro holes	SM1	SM1	SM1
Scale	SM1,SM2	SM1,SM2	SM1,SM2
Local scale	SM1,SM2	SM1,SM2	SM1,SM2
Sampling	LB1	SM2	LB2
Noise	SM1,SM2	SM1,SM2	SM1,SM2
Shot noise	SM1,SM2	SM1,SM2	SM1,SM2
Average	SM1,SM2	SM1,SM2	SM1,SM2

Table 8.4: 3D shape registration performance comparison: The proposed methods (SM1 and SM2) performed best by providing minimum average shape registration error over all the transformation classes with different strength as compare to GMDS [17] and LB1, LB2 [10] methods.



(a) Holes (b) Isometry (c) Noise

(e) Sampling (f) Local scale

Figure 8.5: 3D shape registration in the presence of different transforms.

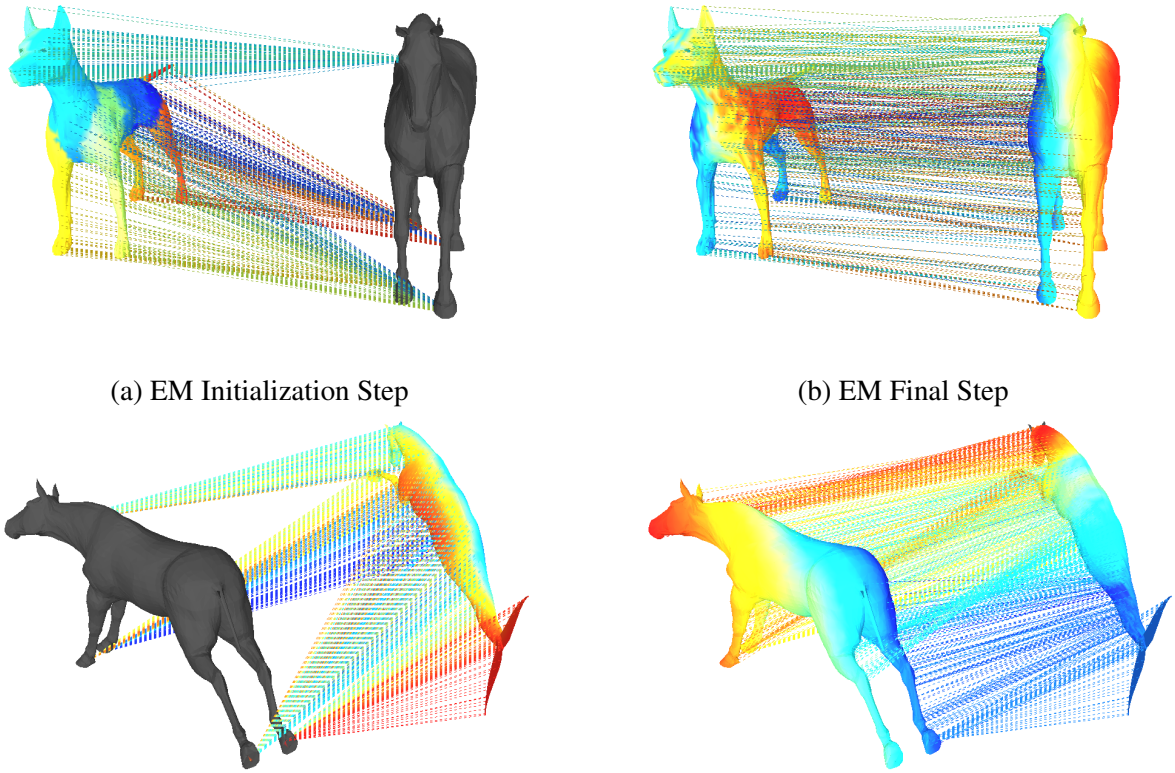
In table 1.4, we show a detailed comparison of proposed method with other methods. For a detailed quantitative comparison refer to [29]. The proposed method inherently uses diffusion geometry as opposed to geodesic metric used by other two methods and hence outperform them.

In the second experiment we perform shape registration on two different shapes with similar topology. In Figure 1.6, results of shape registration on different shapes is presented. Figure 1.6(a),(c) shows the initialization step of EM algorithm while Figure 1.6(b),(d) shows the dense matching obtained after EM convergence.

Finally, we show shape matching results on two different human meshes captured with multi-camera system at MIT [5] and University of Surrey [2] in Figure 1.7

8.7 Discussion

This chapter describes a 3D shape registration approach that computes dense correspondences between two articulated objects. We address the problem using spectral matching and unsupervised point registration method. We formally introduce graph isomorphism using the Laplacian matrix, and we provide an analysis of the matching problem when the number of nodes in the graph is very large, *i.e.* of the order of $O(10^4)$. We show that there is a simple equivalence between graph isomorphism and point registration under the



(a) EM Initialization Step

(b) EM Final Step

(c) EM Initialization Step

(d) EM Final Step

Figure 8.6: 3D shape registration performed on different shapes with similar topology.

(a) Original Meshes

(b) Dense Matching

Figure 8.7: 3D shape registration performed on two real meshes captured from different sequence.

group of orthogonal transformations, when the dimension of the embedding space is much smaller than the cardinality of the point-sets.

The eigenvalues of a large sparse Laplacian cannot be reliably ordered. We propose an elegant alternative to eigenvalue ordering, using eigenvector histograms and alignment based on comparing these histograms. The point registration that results from eigenvector alignment yields an excellent initialization for the EM algorithm, subsequently used only to refine the registration.

However, the method is susceptible to large topology changes that might occur in the multi-camera shape acquisition setup due to self-occlusion (originated from complex kinematics poses) and shadow effects. This is because Laplacian embedding is a global representation and any major topology change will lead to large changes in embeddings causing failure of this method. Recently, a new shape registration method proposed in [32] provides robustness to the large topological changes using the heat kernel framework.

Appendix A

A.1 Permutation and Doubly-stochastic Matrices

A matrix $\mathbf{P}$ is called a *permutation* matrix if exactly one entry in each row and column is equal to 1, and all other entries are 0. Left multiplication of a matrix $\mathbf{A}$ by a permutation matrix $\mathbf{P}$ permutes the *rows* of $\mathbf{A}$, while right multiplication permutes the *columns* of $\mathbf{A}$.

Permutation matrices have the following properties: $\det(\mathbf{P}) = \pm 1$, $\mathbf{P}^\top = \mathbf{P}^{-1}$, the identity is a permutation matrix, and the product of two permutation matrices is a permutation matrix. Hence the set of permutation matrices $\mathbf{P} \in \mathcal{P}_n$ constitute a subgroup of the subgroup of orthogonal matrices, denoted by O_n, and $\mathcal{P}_n$ has finite cardinality $n!$.

A non-negative matrix $\mathbf{A}$ is a matrix such that all its entries are non-negative. A non-negative matrix with the property that all its row sums are $+1$ is said to be a *(row) stochastic matrix*. A *column stochastic matrix* is the transpose of a row stochastic matrix. A stochastic matrix $\mathbf{A}$ with the property that $\mathbf{A}^\top$ is also stochastic is said to be *doubly stochastic*: all row and column sums are $+1$ and $a_{ij} \geq 0$. The set of stochastic matrices is a compact convex set with the simple and important property that $\mathbf{A}$ is stochastic if and only if $\mathbf{A}\mathbb{1} = \mathbb{1}$ where $\mathbb{1}$ is the vector with all components equal to $+1$.

Permutation matrices are doubly stochastic matrices. If we denote by $\mathcal{D}_n$ the set of doubly stochastic matrices, it can be proved that $\mathcal{P}_n = O_n \cap \mathcal{D}_n$ [48]. The permutation matrices are the fundamental and prototypical doubly stochastic matrices, for Birkhoff's theorem states that any doubly stochastic matrix is a linear convex combination of finitely many permutation matrices [42]:

Theorem 4

(Birkhoff) A matrix $\mathbf{A}$ is a doubly stochastic matrix if and only if for some $N < \infty$ there are permutation matrices $\mathbf{P}_1, \ldots, \mathbf{P}_N$ and positive scalars $s_1, \ldots, s_N$ such that $s_1 + \ldots + s_N = 1$ and $\mathbf{A} = s_1\mathbf{P}_1 + \ldots + s_N\mathbf{P}_N$.

A complete proof of this theorem is to be found in [42][pages 526–528]. The proof relies on the fact that $\mathcal{D}_n$ is a compact convex set and every point in such a set is a convex combination of the extreme points

of the set. First it is proved that every permutation matrix is an extreme point of $\mathcal{D}_n$ and second it is shown that a given matrix is an extreme point of $\mathcal{D}_n$ if an only if it is a permutation matrix.

A.2 The Frobenius Norm

The Frobenius (or Euclidean) norm of a matrix $\mathbf{A}_{n\times n}$ is an *entry-wise* norm that treats the matrix as a vector of size $1\times nn$. The standard norm properties hold: $\|\mathbf{A}\|_F > 0 \Leftrightarrow \mathbf{A} \neq 0$, $\|\mathbf{A}\|_F = 0 \Leftrightarrow \mathbf{A} = 0$, $\|c\mathbf{A}\|_F = c\|\mathbf{A}\|_F$, and $\|\mathbf{A}+\mathbf{B}\|_F \leq \|\mathbf{A}\|_F + \|\mathbf{B}\|_F$. Additionally, the Frobenius norm is *sub-multiplicative*:

$$\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F \tag{A.1}$$

as well as *unitarily-invariant*. This means that for any two orthogonal matrices $\mathbf{U}$ and $\mathbf{V}$:

$$\|\mathbf{UAV}\|_F = \|\mathbf{A}\|_F. \tag{A.2}$$

It immediately follows the following equalities:

$$\|\mathbf{UAU}^\top\|_F = \|\mathbf{UA}\|_F = \|\mathbf{AU}\|_F = \|\mathbf{A}\|_F. \tag{A.3}$$

A.3 Spectral Properties of the Normalized Laplacian

The normalized Laplacian Let $\tilde{\mathbf{u}}_k$ and γ_k denote the eigenvectors and eigenvalues of $\tilde{\mathbf{L}}$; The spectral decomposition is $\tilde{\mathbf{L}} = \tilde{\mathbf{U}}\mathbf{\Gamma}\tilde{\mathbf{U}}^\top$ with $\tilde{\mathbf{U}}\tilde{\mathbf{U}}^\top = \mathbf{I}$. The smallest eigenvalue and associated eigenvector are $\gamma_1 = 0$ and $\tilde{\mathbf{u}}_1 = \mathbf{D}^{1/2}\mathbb{1}$.

We obtain the following equivalent relations:

$$\textstyle\sum_{i=1}^n d_i^{1/2}\tilde{u}_{ik} = 0, \quad 2 \leq k \leq n \tag{A.4}$$

$$d_i^{1/2}|\tilde{u}_{ik}| < 1, \qquad 1 \leq i \leq n, 2 \leq k \leq n. \tag{A.5}$$

Using (1.5) we obtain a useful expression for the combinatorial Laplacian in terms of the spectral decomposition of the normalized Laplacian. Notice, however, that the expression below is NOT a spectral decomposition of the combinatorial Laplacian:

$$\mathbf{L} = (\mathbf{D}^{1/2}\tilde{\mathbf{U}}\mathbf{\Gamma}^{1/2})(\mathbf{D}^{1/2}\tilde{\mathbf{U}}\mathbf{\Gamma}^{1/2})^\top. \tag{A.6}$$

For a connected graph γ_1 has multiplicity 1: $0 = \gamma_1 < \gamma_2 \leq \ldots \leq \gamma_n$. As in the case of the combinatorial Laplacian, there is an upper bound on the eigenvalues (see [37] for a proof):

Proposition 4

For all $k \leq n$, we have $\mu_k \leq 2$.

We obtain the following spectral decomposition for the normalized Laplacian :

$$\tilde{\mathbf{L}} = \sum_{k=2}^{n} \gamma_k \tilde{\mathbf{u}}_k \tilde{\mathbf{u}}_k^\top. \tag{A.7}$$

The spread of the graph along the k-th normalized Laplacian eigenvector is given by $\forall (k,i), 2 \leq k \leq n, 1 \leq i \leq n$:

$$\bar{\tilde{u}}_k = \frac{1}{n} \sum_{i=1}^{n} \tilde{u}_{ik} \tag{A.8}$$

$$\sigma_{u_k} = \frac{1}{n} - \bar{\tilde{u}}_k^2. \tag{A.9}$$

Therefore, the projection of the graph onto an eigenvector $\tilde{\mathbf{u}}_k$ is not centered. By combining (1.5) and (A.7) we obtain an alternative representation of the combinatorial Laplacian in terms of the the spectrum of the normalized Laplacian, namely:

$$\mathbf{L} = \sum_{k=2}^{n} \gamma_k (\mathbf{D}^{1/2} \tilde{\mathbf{u}}_k)(\mathbf{D}^{1/2} \tilde{\mathbf{u}}_k)^\top. \tag{A.10}$$

Hence, an alternative is to project the graph onto the vectors $\mathbf{t}_k = \mathbf{D}^{1/2} \tilde{\mathbf{u}}_k$. From $\tilde{\mathbf{u}}_{k\geq 2}^\top \tilde{\mathbf{u}}_1 = 0$ we get that $\mathbf{t}_{k\geq 2}^\top \mathbb{1} = 0$. Therefore, the spread of the graph's projection onto $\mathbf{t}_k$ has the following mean and variance, $\forall (k,i), 2 \leq k \leq n, 1 \leq i \leq n$:

$$\bar{t}_k = \textstyle\sum_{i=1}^{n} d_i^{1/2} \tilde{u}_{ik} = 0 \tag{A.11}$$

$$\sigma_{t_k} = \tfrac{1}{n} \textstyle\sum_{i=1}^{n} d_i \tilde{u}_{ik}^2. \tag{A.12}$$

The random-walk Laplacian. This operator is not symmetric, however its spectral properties can be easily derived from those of the normalized Laplacian using (1.7). Notice that this can be used to transform a non-symmetric Laplacian into a symmetric one, as proposed in [49] and in [50].

Bibliography

[1] J.-S. Franco and E. Boyer, "Efficient Polyhedral Modeling from Silhouettes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, p. 414427, 2009.

[2] J. Starck and A. Hilton, "Surface capture for performance based animation," *IEEE Computer Graphics and Applications*, vol. 27, no. 3, pp. 21–31, 2007.

[3] G. Slabaugh, B. Culbertson, T. Malzbender, and R. Schafer, "A survey of methods for volumetric scene reconstruction from photographs," in *International Workshop on Volume Graphics*, 2001, pp. 81–100.

[4] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, pp. 519–528.

[5] D. Vlasic, I. Baran, W. Matusik, and J. Popovic, "Articulated mesh animation from multi-view silhouettes," *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 27, no. 3, pp. 97:1–97:9, 2008.

[6] A. Zaharescu, E. Boyer, and R. P. Horaud, "Topology-adaptive mesh deformation for surface evolution, morphing, and multi-view reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 4, pp. 823 – 837, April 2011.

[7] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *International Conference on 3D Digital Imaging and Modeling*, 2001, pp. 145–152.

[8] S. Umeyama, "An eigendecomposition approach to weighted graph matching problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 5, pp. 695–703, May 1988.

[9] J. H. Wilkinson, "Elementary proof of the Wielandt-Hoffman theorem and of its generalization," Stanford University, Tech. Rep. CS150, January 1970.

[10] A. Bronstein, M. Bronstein, and R. Kimmel, "Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching," *Proceedings of National Academy of Sciences*, vol. 103, pp. 1168–1172, 2006.

[11] S. Wang, Y. Wang, M. Jin, X. Gu, D. Samaras, and P. Huang, "Conformal geometry and its application on 3d shape matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 7, pp. 1209–1220, 2007.

[12] V. Jain, H. Zhang, and O. van Kaick, "Non-rigid spectral correspondence of triangle meshes," *International Journal of Shape Modeling*, vol. 13, pp. 101–124, 2007.

[13] W. Zeng, Y. Zeng, Y. Wang, X. Yin, X. Gu, and D. Samras, "3d non-rigid surface matching and registration based on holomorphic differentials," in *European Conference on Computer Vision*, 2008, pp. 1–14.

[14] D. Mateus, R. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer, "Articulated shape matching using Laplacian eigenfunctions and unsupervised point registration," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[15] M. R. Ruggeri, G. Patané, M. Spagnuolo, and D. Saupe, "Spectral-driven isometry-invariant matching of 3d shapes," *International Journal of Computer Vision*, vol. 89, pp. 248–265, 2010.

[16] Y. Lipman and T. Funkhouser, "Mobius voting for surface correspondence," *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 28, no. 3, pp. 72:1–72:12, 2009.

[17] A. Dubrovina and R. Kimmel, "Matching shapes by eigendecomposition of the Laplace-Beltrami operator," in *International Symposium on 3D Data Processing, Visualization and Transmission*, 2010.

[18] H. Qiu and E. R. Hancock, "Graph matching and clustering using spectral partitions," *Pattern Recognition*, vol. 39, pp. 22–34, January 2006.

[19] H. F. Wang and E. R. Hancock, "Correspondence matching using kernel principal components analysis and label consistency constraints," *Pattern Recognition*, vol. 39, pp. 1012–1025, June 2006.

[20] H. Qiu and E. R. Hancock, "Graph simplification and matching using commute times," *Pattern Recognition*, vol. 40, pp. 2874–2889, October 2007.

[21] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints," in *International Conference on Computer Vision*, 2005, pp. 1482–1489.

[22] O. Duchenne, F. Bach, I. Kweon, and J. Ponce, "A tensor based algorithm for high order graph matching," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1980–1987.

[23] L. Torresani, V. Kolmogorov, and C. Rother, "Feature correspondence via graph matching : Models and global optimazation," in *European Conference on Computer Vision*, 2008, pp. 596–609.

[24] R. Zass and A. Shashua, "Probabilistic graph and hypergraph matching," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[25] J. Maciel and J. P. Costeira, "A global solution to sparse correspondence problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 187–199, 2003.

[26] Q. Huang, B. Adams, M. Wicke, and L. J. Guibas, "Non-rigid registration under isometric deformations," *Computer Graphics Forum*, vol. 27, no. 5, pp. 1449–1457, 2008.

[27] Y. Zeng, C. Wang, Y. Wang, X. Gu, D. Samras, and N. Paragios, "Dense non-rigid surface registration using high order graph matching," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 382–389.

[28] Y. Sahillioglu and Y. Yemez, "3d shape correspondence by isometry-driven greedy optimization," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 453–458.

[29] A. M. Bronstein, M. M. Bronstein, U. Castellani, A. Dubrovina, L. J. Guibas, R. P. Horaud, R. Kimmel, D. Knossow, E. v. Lavante, M. D., M. Ovsjanikov, and A. Sharma, "Shrec 2010: robust correspondence benchmark," in *Eurographics Workshop on 3D Object Retrieval*, 2010.

[30] M. Ovsjanikov, Q. Merigot, F. Memoli, and L. Guibas, "One point isometric matching with the heat kernel," *Computer Graphics Forum (Proc. SGP)*, vol. 29, no. 5, pp. 1555–1564, 2010.

[31] A. Sharma and R. Horaud, "Shape matching based on diffusion embedding and on mutual isometric consistency," in *NORDIA workshop IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

[32] A. Sharma, R. Horaud, J. Cech, and E. Boyer, "Topologically-robust 3d shape matching based on diffusion geometry and seed growing," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011.

[33] D. Knossow, A. Sharma, D. Mateus, and R. Horaud, "Inexact matching of large and sparse graphs using laplacian eigenvectors," in *Graph-Based Representations in Pattern Recognition*, 2009, pp. 144–153.

[34] R. P. Horaud, F. Forbes, M. Yguel, G. Dewaele, and J. Zhang, "Rigid and articulated point registration with expectation conditional maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 587–602, 2011.

[35] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[36] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[37] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, 1997.

[38] L. Grady and J. R. Polimeni, *Discrete Calculus: Applied Analysis on Graphs for Computational Science*. Springer, 2010.

[39] C. Godsil and G. Royle, *Algebraic Graph Theory*. Springer, 2001.

[40] A. J. Hoffman and H. W. Wielandt, "The variation of the spectrum of a normal matrix," *Duke Mathematical Journal*, vol. 20, no. 1, pp. 37–39, 1953.

[41] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*. Oxford: Clarendon Press, 1965.

[42] R. A. Horn and C. A. Johnson, *Matrix Analysis*. Cambridge: Cambridge University Press, 1994.

[43] R. Burkard, *Assignment Problems*. Philadelphia: SIAM, Society for Industrial and Applied Mathematics, 2009.

[44] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, "A kernel view of the dimensionality reduction of manifolds," in *International Conference on Machine Learning*, 2004, pp. 47–54.

[45] H. Qiu and E. R. Hancock, "Clustering and embedding using commute times," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 11, pp. 1873–1890, 2007.

[46] C. M. Grinstead and L. J. Snell, *Introduction to Probability*. American Mathematical Society, 1998.

[47] D. W. Scott, "On optimal and data-based histograms," *Biometrika*, vol. 66, no. 3, pp. 605–610, 1979.

[48] M. M. Zavlanos and G. J. Pappas, "A dynamical systems approach to weighted graph matching," *Automatica*, vol. 44, pp. 2817–2824, 2008.

[49] J. Sun, M. Ovsjanikov, and L. Guibas, "A concise and provably informative multi-scale signature based on heat diffusion," in *SGP*, 2009.

[50] C. Luo, I. Safa, and Y. Wang, "Approximating gradients for meshes and point clouds via diffusion metric," *Computer Graphics Forum (Proc. SGP)*, vol. 28, pp. 1497–1508, 2009.