DIGITALCOMMONS
@WAYNESTATE

Wayne State University

Wayne State University Dissertations

1-1-2015

Shape Analysis Using Spectral Geometry

Jiaxi Hu
Wayne State University,

Follow this and additional works at: http://digitalcommons.wayne.edu/oa_dissertations

Recommended Citation
Hu, Jiaxi, "Shape Analysis Using Spectral Geometry" (2015). *Wayne State University Dissertations.* Paper 1143.

This Open Access Dissertation is brought to you for free and open access by DigitalCommons@WayneState. It has been accepted for inclusion in Wayne State University Dissertations by an authorized administrator of DigitalCommons@WayneState.

SHAPE ANALYSIS USING SPECTRAL GEOMETRY

by

JIAXI HU

DISSERTATION

Submitted to the Graduate School

of Wayne State University,

Detroit, Michigan

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

2015

MAJOR: COMPUTER SCIENCE

Approved by:

_____

Advisor Date

_____

_____

_____

© COPYRIGHT BY

JIAXI HU

2015

All Rights Reserved

DEDICATION

I dedicate this dissertation work to my family. Special thanks to my loving parents, Duanyang Hu and Xianzhen Xiong. My wife Wenwen Shi. My daughter, my little angle, Shimo Hu.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor, Dr. Jing Hua. I would not finish this dissertation without his support and encouragement. His wisdom, knowledge and commitment to the highest standards inspired and motivated me. I enjoy this wonderful science journey he guides me.

I wish to thank my committee members, Dr. Ming Dong, Dr. Abhilash Pandya, and Dr. Robert Reynolds, who were more than generous with their expertise, precious time, and valuable guidance.

I also thank all my colleagues in Graphics and Imaging Laboratory, Dr. Guangyu Zou, Dr.Yunhao Tan, Dr. Vahid Taimouri, Dr. Zhaoqiang Lai, Dr. Chang Liu, Dr. Darshan Pai, Hai Jin, Song Wang, Xiaodong Jiang, Shidi Wang, Xinyu Zhang, and Yan Yan.

Last but not least, I deeply acknowledge the love and encouragement from my parents, my wife, and my daughter.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Shape analysis is a fundamental problem in many research fields such as computer graphics, vision, image processing, robotics, and so on. Computer scientists and engineers consider shape as an attribute to describe an object [45, 34]. In the past decade, massive 3D shapes are produced with advanced technologies. Traditional computer aided design (CAD) creates a lot of manufacturing 3D models [56]. Laser scanning generates 3D point clouds or surface meshes [41]. Similarly, structure light camera generates depth image. For example, kinect from Microsoft reduced the cost of this technique and made it available for daily use. MRI or CT scanning produces intensity-based volume data. Although 3D shape data is represented with various formats, those formats can be converted to each other. For example, surfaces can be reconstructed from point clouds or extracted from the isovalues of the intensity-based volume. In this work, we focus on the shapes represented with 2 manifold boundary surfaces which are differentiable. In practice, these manifolds are discretized into triangle or tetrahedron meshes. The increasing 3D shape data demands a variety of shape analysis methods. Considering shape as general data, there exist basic analyses, e.g., matching, indexing, retrieval, registration, and mapping. On top of these basic ones, high level understanding is also desired, including pose analysis, 4D time-varying motion, etc.

Traditional shape analysis starts from the original spatial properties of shapes, e.g., curvature, diameter, and geodesic distance. There are also more advanced shape representations. For example moments describe a shape with a set of integrations of different orders [13, 62, 16, 50]. An extended Gaussian image is built with the orientation and area information of a convex polygon mesh for shape representation. This extended Gaussian image can describe a convex mesh uniquely [23, 66]. Shape distributions measure properties based on distance, angle, area, and volume measurements between random surface points. The similarity between the

objects is measured by a pseudo-metric that measures distances between distributions [53, 52]. Geometric hashing represents a shape with a set of its local interest features (points, lines, or other suitable features) [38, 40, 73, 20]. A shape can be described with another kind of data structure, such as vector or graph [68, 27, 44, 71, 63]. Graph-based approaches analyze a 3D shape by transforming it to a graph, such as B-Rep graph, Reeb graph and skeletal graph and convert shape analysis into graph problems [14, 15, 22, 4, 8, 4]. Spherical harmonics represents a shape by a 2D histogram of radius and frequency [71, 63, 32, 33, 51]. It decomposes a model into a collection of spherical functions on the concentric spheres, then calculates the Fourier transforms of these spherical functions. Shape histograms [1] describe a shape by the partitions of the 3D space. 3D space can be decomposed into disjoint cells in different ways. These traditional shape analysis approaches are often challenged by Euclidean transformations, irregular mesh samplings, and non-linear deformations.

Another category of shape analysis work is based on geometric mapping. Shape mapping is a powerful tool to reduce the complexity of arbitrary manifolds onto canonical domains such as unit cube or sphere, where regular analysis, e.g., image-based processing, can be applied directly [37]. Among this category, Functional methods typically start with defining certain penalty functions, such that the minima are assumed at desired results. The mapping is then achieved using optimization methods. Conformal mapping provides a unique mapping by preserving local angle geometries. Conformal methods possess several unique advantages, e.g., exact angle preserving, guarantee of solution existence, efficient algorithm, and a rich continuous theory in parallel. At the mean time, conformal mapping introduces large area distortions. In order to reduce the area distortions, additional process are applied. Gu and Yau punctured small holes at the tip of long appendages [19]. Cone singularities were introduced with non-vanishing Gaussian curvature in [35, 3]. Surface cuts were repeatedly augmented according to the geometric stretches generated through the course of tentative parameterizations [18]. Zou et al. [76] presented a practical method to compute a group of analytic global 2D area-preserving

mapping mathematically with *Lie advection*. A manifold can not be mapped to another domain without any distortion. Thus, different mapping methods have been proposed to preserve certain local geometries [18, 43]. That is the dilemma of the mapping-based approaches for shape analysis.

The spectrum-based approach is inspired by the Fourier transform in signal processing, where the time variant signals can be projected on to functional bases. Early shape spectrum work is applied on graphs [48, 49, 46]. Considering discrete meshes are also graphs, Laplacian matrix is defined on vertices and connections, weights may also applied. The eigenvalues are defined as the spectra of graphs, and the eigenfunctions are the orthogonal bases. This spectrum has a lot of similarities with Fourier transform. The graphs are then projected onto those bases and analyzed in the spectral domain. Karni and Gotsman [31] used the projections of geometry on the eigenfunctions for mesh compression and smoothing. Jain and Zhang [28] extended it for shape registration in the spectral domain. The Laplace spectrum focuses on the connection of graph, instead of the intrinsic geometry of the manifolds. Only using the connectivity of the graph may lead to highly distorted mappings [75].

From the view of computational geometry, the geometry of the shape can be represented with the differentiable manifold. Reuter defined shape spectrum as the family of eigenvalues of the Laplace-Beltrami operator on a manifold [59, 58, 60, 57]. Nearly at the same time, Reuter [59] and Lévy [42] found out shape spectrum as the family of eigenvalues of Laplace-Beltrami operator on a manifold, and used it as a global shape descriptor. Without further notation, shape spectrum in this work refers to the Laplace-Beltrami spectrum. Laplace-Beltrami eigenfunctions are also tools to understand the geometry. Rustamov [61] proposed a modified shape distribution based on eigenfunctions and eigenvalues. It is proved that the shape spectrum is invariant to those spatial translation, rotation, scaling, and isometric deformations. The spectrum is also stable to the different triangulations and near isometric deformation. The shape spectrum describes the similarities among shapes. The shape spectrum has a lot of great

properties for shape analysis. It is invariant to Euclidean transformations and isometric deformations. On discrete triangle meshes, the spectrum is invariant to different triangulations. It carries the intrinsic geometry of the manifold behind various representations. However, by definition, the spectrum depends on the global geometry. It will change a lot while the geometry changes. Analysis and experiments showed it is stable among near isometric deformations and minor noises. Greater non-isometric deformation breaks the connections among different objects. This often restricts the shape spectrum to global shape descriptor or same object analysis.

In order to solve those research problems, we propose the shape analysis approaches by employing the shape spectrum in differential geometry. The shape spectra contain the intrinsic geometry information of the original shape and reveal the relationship across shapes. There are three major contributions presented in this differential shape analysis work.

- We propose a method to extract salient spectral geometric features in the spectral domain derived from the Laplace-Beltrami operator, which is invariant to Euclidean transformations and isometric deformations. Describing and matching shapes with their salient features also conform to the procedure of "coarse-to-fine" multiresolution analysis. The features are extracted with local "frequencies" identified, which imply spatial scales of local support regions defining the features. That is to say, each salient feature finds its local support. The salient spectral features are very stable and distinctive. The shape representation built upon them may achieve a higher level of shape description. It can be easily applied to tasks such as shape matching and shape retrieval.

- We present a registration-free shape motion analysis method based on Laplace-Beltrami spectral domain. Surface mesh vertices belonging to the same semantic part on different pose surfaces will be mapped to the same coordinates in the geometry spectral domain, while they carry different spatial properties under different poses. The analysis of the spatial property variation in the geometry spectral domain is able to quantify the geome-

try behaviors of every point during the pose changes, consequently, classifying a point to a rigid part or an articulated part in the spectral domain. The shape is then decomposed into parts with different geometric semantics. The skeleton can be generated automatically based on eigenfunctions of the shape. The behavior of the skeleton is constrained by the surface properties and classified surface semantics, which also represents the semantics of that skeleton.

- We prove that eigenvalues are continuous functions of scalar factors applied on the conformal metric. The derivatives of the eigenvalues are analytically expressed with those of the scalar field defined on the original manifold. The discrete counterpart on the triangle meshes also follows the same behavior. In this case, the scalar field turns into a scale vector on the mesh, whose value is sampled at each vertex. More specifically, the analytic expression between the eigenvalues and the scale vector can be reformed into a linear system. With smoothness and local bound constraints, the linear system is consequently solvable as a quadratic problem. Then, the spectra can be controlled with local scale vectors. The approach closes the gap that shape spectrum is not invariant to non-isometric deformations. The scale field is the solution of a quadratic problem. On triangle meshes, the scalar scale factors are represented with a scale vector defined on each vertex.

The dissertation is organized as follows:

- Chapter 2: Gives definitions and detailed description of shape spectrum and conformal mapping which derived from the second order differential operators. This provides a mathematical foundation on which the remainder of the thesis dissertation will be based.

- Chapter 3: Introduces local salient spectral features extracted in the spectral domain. This section presents the detection, descriptor and matching of keypoints.

- Chapter 4: Shows the pose analysis in the spectral domain. By transferring the shapes

from the spatial domain to the spectral domain, all the Euclidean transformations and isometric deformations are filtered out. In the spectral domain, different poses are understood based on meaningful parts.

- Chapter 5: Invents an analytic method to align spectrum among different shapes with non-isometric deformations. We prove that eigenvalues are continuous functions of scale factors applied on the conformal metric. On discrete triangle meshes, the theorem is also valid. The derivatives of the eigenvalues and the scalar factors satisfy a linear system. Thus, the scale filed is the solution of a quadratic problem. By applying such scale field, spectra are aligned among shapes with non-isometric deformations.

- Chapter 6: Gives a summary of the dissertation.

CHAPTER 2

BACKGROUND

In this work, a shape is represented with a differentiable manifold in the computational geometry. On such a manifold, the differential operator is defined based on the local geometry. Within one shape, the operator contains intrinsic geometry information. The differential operator will introduce shape spectrum on the original manifolds. In this chapter, we are going to briefly review the definition of the shape spectrum and the numerical computations.

2.1 Laplace Shape Spectrum

In this section, we will review the theory of Laplacian spectrum and describe how to compute it on a triangle mesh.

Let $f \in C^2$ be a real function defined on a Riemannian manifold M. The Laplace-Beltrami operator Δ is defined as,

$$\Delta f = \nabla \cdot (\nabla f), \tag{2.1}$$

where ∇f being the gradient of f and $\nabla\cdot$ the divergence on the manifold M [7]. The Laplace-Beltrami operator is linear differential and can be calculated in local coordinates. Let ψ be a local parametrization of a submanifold of M such that, $\psi : R^n \rightarrow R^{n+k}$, $g_{ij} =< \partial_i\psi, \partial_j\psi >$, $G = (g_{ij})$, $W = \sqrt{\det G}$, and $(g^{ij}) = G^{-1}$, where $i, j = 1, 2, ..., n$, $<,>$ is the dot product and $\det$ is the determinant. The Laplace-Beltrami operator then is defined on the submanifold as $\Delta f = \frac{1}{W}\sum_{i,j}\partial_i(g^{ij}W\partial_j f)$. If $M \subset R^2$, the Laplace-Beltrami operator reduces to the Laplacian:

$$\Delta f = \frac{\partial^2 f}{(\partial x)^2} + \frac{\partial^2 f}{(\partial y)^2}. \tag{2.2}$$

Consider the Laplacian eigenvalue equation:

$$\Delta f = -\lambda f, \tag{2.3}$$

where λ is a real scalar. The solution will be a family of nonnegative scalar $\{\lambda_i\}$ and a corresponding real function $\{f_i\}$ for $i = 0, 1, 2, \ldots$. The spectrum is defined to be the eigenvalues arranged increasingly as $0 \leq \lambda_0 \leq \lambda_1 \leq \lambda_2 \leq \cdots \leq +\infty$. In the case of a close manifold or a open manifold with Neumann boundary condition, the first eigenvalue λ_0 will always be zero, and f_0 is a constant function. The spectrum is an isometric invariant because it only depends on the gradient and divergence which are dependent only on the Riemannian structure of the manifold. After the normalization of the eigenvalues, shape can be matching regardless of the scales. The inner product of the functions on M is defined with an integral all over the manifold as

$$< f, g >= \int_M fgd\sigma. \tag{2.4}$$

The Laplace-Beltrami operator is Hermitian, so the eigenvectors corresponding to its different eigenvalues are orthogonal:

$$< f_i, f_j >= \int_M f_i f_j d\sigma = 0, \tag{2.5}$$

where $i \neq j$. According to the definition of the eigenvalue problem, if there exists a solution f_i, αf_i is also a solution, where α is a nonzero scalar. Thus, f_i is usually normalized as

$$< f_i, f_i >= \int_M f_i^2 d\sigma = 1 \textbf{ for } i = 0, 1, 2, \ldots. \tag{2.6}$$

Another weak version of equation 2.3 can also be derived as for any test function $\varphi \in C^1$

$$\int_M \nabla\varphi \cdot \nabla f d\sigma = \lambda \int_M \varphi f d\sigma, \tag{2.7}$$

if f is a solution of equation 2.3, which is necessary but not sufficient. This weak version is proved with Greens formula on a closed manifold or a open manifold with Neumann boundary

condition as

$$\begin{aligned} \Delta f &= -\lambda f \\ \varphi \Delta f &= -\lambda \varphi f \\ \int_M \varphi \Delta f d\sigma &= -\lambda \int_M \varphi f d\sigma \\ -\int_M \nabla \varphi \cdot \nabla f d\sigma &= -\lambda \int_M \varphi f d\sigma. \end{aligned} \tag{2.8}$$

A given function f on the surface can be expanded as:

$$f = c_1 f_1 + c_2 f_2 + c_3 f_3 + \cdots, \tag{2.9}$$

where the coefficients are

$$c_i =< f, f_i >= \int_M f f_i d\sigma. \tag{2.10}$$

2.2 Finite Element Method Computation

In order to solve the Laplace eigenvalue problem numerically, the manifold is discretized. equation 2.7 can be solved with the finite element method [59]. Assume there exist n sample points on the manifold M. Each function is defined on those n digital sample points. Differential operations are also defined on such points. If a set of n linear independent functions $\{\varphi_i\}$ are chosen, which means

$$< \varphi_i, \varphi_j >= \int_M \varphi_i \varphi_j d\sigma = 0 \text{ if } i \neq j, \tag{2.11}$$

any function on M, including eigenfunction f, is a linear combination of $\{\varphi_i\}$ as

$$f = c_1 \varphi_1 + c_2 \varphi_2 + c_3 \varphi_3 + \cdots + c_n \varphi_n. \tag{2.12}$$

Eigenfunction f is uniquely represented with coefficient vector c and satisfies equation 2.7. If we consider each φ_i is a test function, then we get

$$\sum_{j=1}^{n} c_j \int_M \nabla\varphi_i \cdot \nabla\varphi_j d\sigma = \lambda \sum_{j=1}^{n} c_j \int_M \varphi_i \varphi_j d\sigma, \text{ for } i = 1, 2, \ldots, n. \tag{2.13}$$

These n independent linear equations can be put into a linear system as

$$\begin{aligned} A \cdot c &= \lambda B \cdot c \\ A_{ij} &= \int_M \nabla\varphi_i \cdot \nabla\varphi_j d\sigma \\ B_{ij} &= \int_M \varphi_i \varphi_j d\sigma. \end{aligned} \tag{2.14}$$

Depending on different representations, $\nabla\varphi_i$ and the integral can be calculated on the finite elements. The weak version of the eigenvalue problem is converted into a generalized eigenvalue problem in matrix form. The family of the solution represents the eigenvalues and eigenfunctions.

2.3 Discrete Laplace-Beltrami Operator

Another numerical approach is discrete differential operator [47]. In our framework, 2D manifold data is approximated with discrete triangle meshes. A triangle mesh is defined with $M = (V, E, F)$, where $V = \{\mathbf{p}_i\}$ denotes the set of vertices, $E = \{\mathbf{e}_{ij}\}$ the edge set, and $F = \{\mathbf{f}_{ijk}\}$ the face set with $1 \leq i, j, k \leq n = |V|$. $\mathbf{p}_i$ also denotes the position of the vertex in R^3, $\mathbf{e}_{ij}$ the edge vector connects vertices $\mathbf{p}_i$ and $\mathbf{p}_j$ as $\mathbf{e}_{ij} = \mathbf{p}_j - \mathbf{p}_i$. The 1-ring neighbors of $\mathbf{p}_i$ are denoted as $N_1(i)$. All triangular faces assume counterclockwise orientation. Each triangular face represents the local manifold. We can define property on each element, e.g., vertex, edge, and face, which is a spatial average around such element. The properties on the vertices are considered as discrete samplings on the manifold. The discrete operators are also defined on each vertex. In this work, the neighborhood of a vertex $\mathbf{p}_i$ is approximated with

its Voronoi area. Figure 2.1 demonstrates the Voronoi area of the vertex $\mathbf{p}_i$ within its $N_1(i)$.

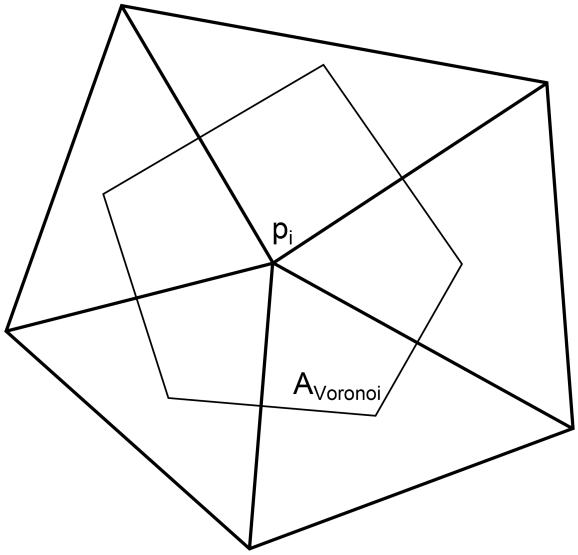


Figure 2.1: Voronoi area of the vertex $\mathbf{p}_i$ within its one-ring neighborhood.

A discrete Laplace-Beltrami operator K_i, also known as mean curvature normal operator, is defined with the average value over the Voronoi area. Suppose we have the integral of the Laplace-Beltrami value, then the Laplace-Beltrami operator at vertex $\mathbf{p}_i$ is represented as

$$K_i(g) = \frac{\int_{A_{Voronoi}} K(g) dA}{A_{Voronoi}}, \tag{2.15}$$

where $g(p) \in C^2$ is a scalar function defined on the triangle mesh M. The calculation of Voronoi areas on triangle meshes is trivial by definition. The geometry on each face is piece-wise linear, so the Laplace-Beltrami operator turns into Laplacian on the parameter space, such that

$$K(g) = -\Delta g = -g_{uu} - g_{vv}, \tag{2.16}$$

where u and v are the parameters. According to Gauss's Theorem, the area integral of Laplacian can be calculated with a line integral on the boundary

$$\int_{A_{Voronoi}} K(g)dA = -\int_{A_{Voronoi}} \Delta g dA = -\int_{\partial A_{Voronoi}} \nabla_{uv} g \cdot \mathbf{n} dl. \tag{2.17}$$

where $\mathbf{n}$ is the unit normal vector of the boundary. The boundary is piecewise line segment

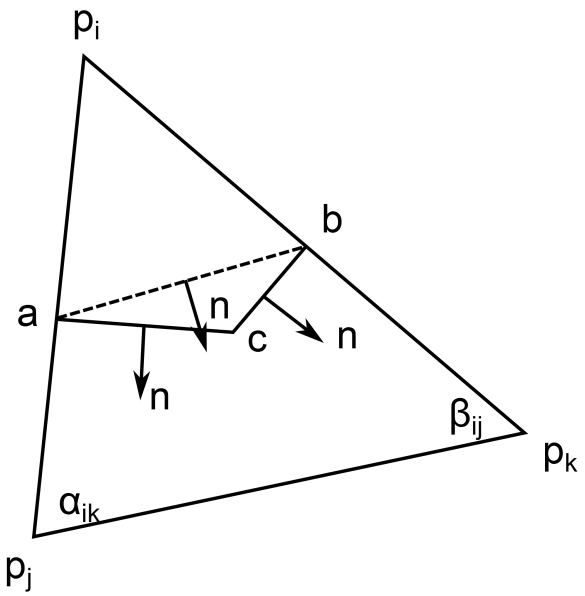


Figure 2.2: The boundary of a Voronoi area consists of the piecewise line segment in each face.

in the one-ring neighbor faces, and the gradient is analytic and constant within each face. Figure 2.2 demonstrates the piecewise linear boundary of the Voronoi area of $\mathbf{p}_i$ in the face $\mathbf{f}_{ijk}$. As we know, the gradient in each face is a constant vector, the line segment integral on the segments ac and cb is equivalent of that on the dot straight line ab in the face. That is, in this face $\mathbf{f}$,

$$\int_{\partial A_{Voronoi} \cap \mathbf{f}} \nabla_{uv} g \cdot \mathbf{n} dl = \nabla_{uv} g \cdot \mathbf{n}_{ab} \parallel \mathbf{e}_{ab} \parallel . \tag{2.18}$$

a and b are the middle point of $\mathbf{e}_i j$ and $\mathbf{e}_i k$ by definition, so $\mathbf{e}_{ab}$ and $\mathbf{e}_{jk}$ are parallel and $\parallel \mathbf{e}_{jk} \parallel = \frac{1}{2} \parallel \mathbf{e}_{jk} \parallel$. The normal direction of a line segment in a face is retrieved by rotating the directed edge 90° on the face plane. This rotation can be derived from vector cross production.

Denote $\mathbf{N}$ as the unit normal of the face $\mathbf{f}_{ijk}$ as

$$\mathbf{N} = \frac{1}{2A_f}\mathbf{e}_{ki} \times \mathbf{e}_{ij} = \frac{1}{2A_f}\mathbf{e}_{ij} \times \mathbf{e}_{jk} = \frac{1}{2A_f}\mathbf{e}_{jk} \times \mathbf{e}_{ki}, \tag{2.19}$$

where A_f is the area of $\mathbf{f}_{ijk}$. Thus the rotation above is derived from cross product $\mathbf{N}$ from the right. And equation 2.18 turns into

$$\int_{\partial A_{Voronoi} \cap \mathbf{f}} \nabla_{uv} g \cdot \mathbf{n} dl = \frac{1}{2}\nabla_{uv} g \cdot (\mathbf{e}_{jk} \times \mathbf{N}). \tag{2.20}$$

The face is piecewise linear and the gradient on it is constant which is expressed as

$$\nabla_{uv} g = \frac{1}{2A_f}(g_i \mathbf{e}_{jk} + g_j \mathbf{e}_{ki} + g_k \mathbf{e}_{ij}) \times \mathbf{N}, \tag{2.21}$$

where g_i, g_j, and g_k are the values of function g at those vertices. Combining equation 2.20 and 2.21, we have

$$\int_{\partial A_{Voronoi} \cap \mathbf{f}} \nabla_{uv} g \cdot \mathbf{n} dl = \frac{1}{4A_f}((g_j - g_i)\mathbf{e}_{ik} \cdot \mathbf{e}_{kj} + (g_k - g_i)\mathbf{e}_{ij} \cdot \mathbf{e}_{jk}), \tag{2.22}$$

or

$$\int_{\partial A_{Voronoi} \cap \mathbf{f}} \nabla_{uv} g \cdot \mathbf{n} dl = \frac{1}{2}(\cot \beta_{ij}(g_j - g_i) + \cot \alpha_{ik}(g_k - g_i)), \tag{2.23}$$

where α_{ik} and β_{ij} are the angles demonstrated in Figure 2.2. The sum of equation 2.23 of all the faces with in the one-ring neighborhood leads to the discrete Laplace-Beltrami operator K_i on vertex $\mathbf{p}_i$

$$K_i(g) = \frac{1}{2A_i} \sum_{\mathbf{p}_j \in N_1(\mathbf{p}_i)} (\cot \alpha_{ij} + \cot \beta_{ij})(g_i - g_j), \tag{2.24}$$

where α_{ij} and β_{ij} are the two angles opposite to the edge in the two triangles sharing the edge i, j, and A_i is the Voronoi area of $\mathbf{p}_i$. For the whole vertices of a triangle mesh, a Laplace-

Beltrami matrix can be constructed as:

$$L_{ij} = \begin{cases} -\frac{\cot\alpha_{ij}+\cot\beta_{ij}}{2A_i} & \text{if } i,j \text{ are adjacent,} \\ \sum_k \frac{\cot\alpha_{ik}+\cot\beta_{ik}}{2A_i} & \text{if } i=j, \\ 0 & \text{otherwise,} \end{cases} \tag{2.25}$$

where α_{ij}, β_{ij}, and A_i are the same as in equation 2.24 for certain i and j. Then, the spectrum problem equation 2.3 turns into the following eigenvalue problem:

$$L\mathbf{v} = \lambda\mathbf{v}. \tag{2.26}$$

where $\mathbf{v}$ is n dimensional vector. Each entry of $\mathbf{v}$ represents the function value at one of n vertices on the mesh. The equation above can be represented as a generalized eigenvalue problem which is much easier to solve numerically by constructing a sparse matrix W and a diagonal matrix S such that,

$$W_{ij} = \begin{cases} -\frac{\cot\alpha_{ij}+\cot\beta_{ij}}{2} & \text{if } i,j \text{ are adjacent,} \\ \sum_k \frac{\cot\alpha_{ik}+\cot\beta_{ik}}{2} & \text{if } i=j, \\ 0 & \text{otherwise,} \end{cases}$$

and $S_{ii} = A_i$. Thus, the Laplace Matrix L is decomposed as $L = S^{-1}W$ and the generalized eigenvalue problem is presented as:

$$W\mathbf{v} = \lambda S\mathbf{v}. \tag{2.27}$$

As defined above, W is symmetric and S is symmetric positive-define. All the eigenvalues and eigenvectors are real, and the eigenvectors corresponding to different eigenvalue are orthogonal in terms of S dot product:

$$< \mathbf{u}, \mathbf{w} >_S = \mathbf{u}^T S\mathbf{w} \tag{2.28}$$

where $\mathbf{u}$ and $\mathbf{w}$ are eigenvectors of equation 2.27. equation 2.5, equation 2.9 and equation 2.10 can be reduced, respectively, to

$$< \mathbf{v}_i, \mathbf{v}_j >_S = 0, i \neq j, \tag{2.29}$$

$$\mathbf{v} = \sum_{i=1}^{n} \mathbf{v}_i c_i, \tag{2.30}$$

and

$$c_i = < \mathbf{v}, \mathbf{v}_i >_S . \tag{2.31}$$

Under this setting, the spectrum, $\{0, \lambda_1, \lambda_2, \lambda_3, \cdots, \lambda_{n-1}\}$, is the family of eigenvalues of the generalized eigenvalue problem defined above.

CHAPTER 3

SALIENT SPECTRAL GEOMETRIC FEATURES FOR SHAPES

In this chapter, we introduce a new method for extracting salient features from surfaces. This method extracts salient geometric feature points in the Laplace-Beltrami spectral domain instead of usual spatial domains. Simultaneously, a spatial region is determined as a local support of each feature point, which is correspondent to the "frequency" where the feature point is identified. The local shape descriptor of a feature point is the Laplace-Beltrami spectrum of the spatial region associated to the point, which are stable and distinctive. The method leads to the salient spectral geometric features invariant to spatial transforms such as translation, rotation, and scaling. The properties of the discrete Laplace-Beltrami operator make them invariant to isometric deformations and mesh triangulations as well. With the scale information transformed from the "frequency", the local supporting region always remain the same ratio to the original model no matter how it is scaled. That means the spatial region is scale invariant as well. Therefore, both global and partial matching can be achieved with these salient feature points.

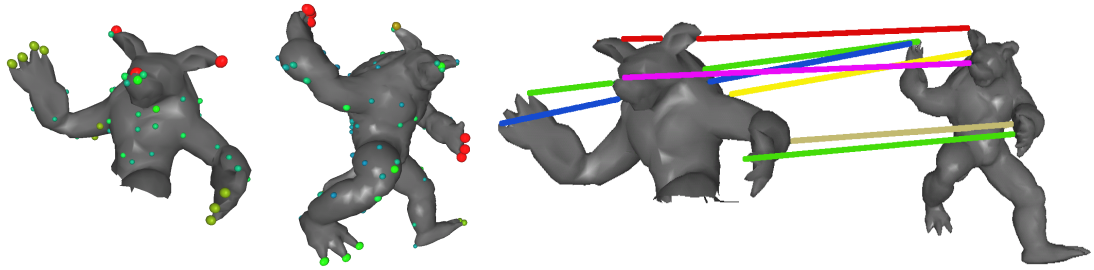


Figure 3.1: Matching with salient spectral geometric features. The highlighted feature points are extracted in the spectral domain generated directly form the triangles meshes. Notice that the two meshes are different in position, orientation, scale, pose, number of vertices, and triangulation. Furthermore, the left shape is only a "part" of the whole model. In order to clearly see the matching result, only some of the matched points are displayed.

3.1 Related Work

There are a number of well studied global shape representations such as moment [62][13][50], extended Gaussian image [23][66], shape distribution [53][26][52], and shape harmonics [33]. These work showed great power in shape analysis. But matching with these global representations usually requires the data to be aligned or normalized. It can be determined whether two shapes are similar or not or how similar they are with those global representations. However, it is difficult to obtain more matching details directly, such as part level similarity, correspondence or registration. These global representations also perform poorly in a "part to whole" matching. A part or a sub-shape is always considered as a quite different shape from its original shape by these methods. To decide a proper scale of the part to the whole shape is also a difficult task.

A part to whole matching is also considered as partial matching which is more general than global one. Partial matching decides if a shape is a part of another one and where it should be located. It is often applied with matching local features. Gal et al. [17] proposed a partial matching method based on salient local features extracted from 3D surfaces. The salient features are extracted locally with an area growing algorithm following an empirical formula, and the descriptors are defined on the quadratic fitted surfaces based on the original meshes. There are also more rigorous scale space-based methods for extracting salient features [39][25][77]. Graph-based approach is another important solution to shape matching. For example Reeb graph [22] and skeletal graph [68][27]. represent a shape with a graph and turn the matching problem into the graph problem. A part to whole matching can be handled here with sub-graph searching. However, the graph extracted from a shape is sensitive to topology. The tiny change of topology may result quite different graphs.

In this work, we analyze shapes based on their spectrums. Shape spectrum is a new topic in computer graphics the recent years. Reuter [59] defined shape spectrum as the family of eigenvalues of Laplace-Beltrami operator on a manifold, and used it as a global shape descrip-

tor. Lévy [42] pointed out that Laplace-Beltrami eigenfunctions are "tools" to understand the geometry of shapes and discussed the properties of those eigenfunctions of Laplace-Beltrami operator. Rustamov [61] proposed a modified shape distribution base on eigenfunctions and eigenvalues. Karni and Gotsman [31] used the projections of geometry on the eigenfunctions for mesh compression. Laplace-Beltrami spectrum is showing more and more power in shape analysis. It is proved to have many good invariant properties [59]. In this chapter, we propose to extract salient geometric features in the domain of spectrum.

3.2 Salient Spectral Feature Extraction

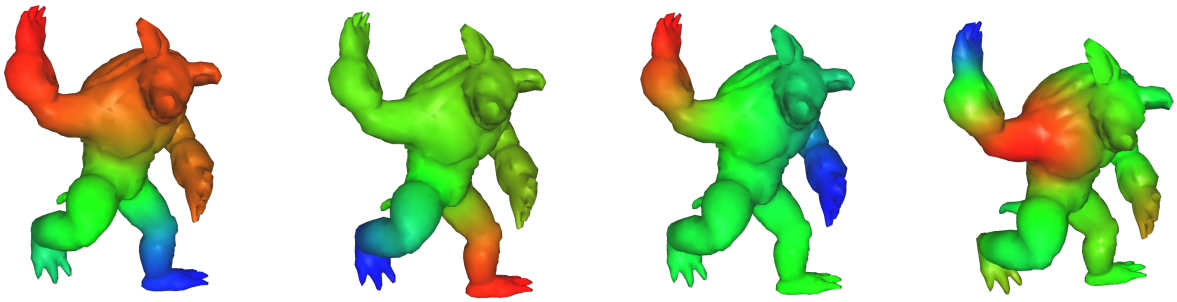
Given a 3D triangle mesh, we first analyze its Laplacian spectrum and represent the shape by a set of salient feature points with scale information in its spectrum. With their associated scales, local Laplacian spectrums are calculated and assigned as local shape descriptors to the feature points.

Salient Feature Point Detection

In this section, we will describe how to extract salient feature points based on the geometry and eigenfunctions since there are much shape information in the eigenvalues and eigenfunctions. Figure 3.2 illustrates some eigenfunctions on the armadillo model. The eigenfunctions contain very symmetric and meaningful information. Figure 3.3 shows isometric properties of eigenfunctions. They are the 5th eigenfunctions on different meshes. The three meshes are generated from the same shape with different poses. The sampling rates are also different. In Figure 3.3, (a) has 1,000 vertices, (b) has 1,500, while (c) has 3,000. We can see that the eigenfunctions are independent to poses and triangulations.

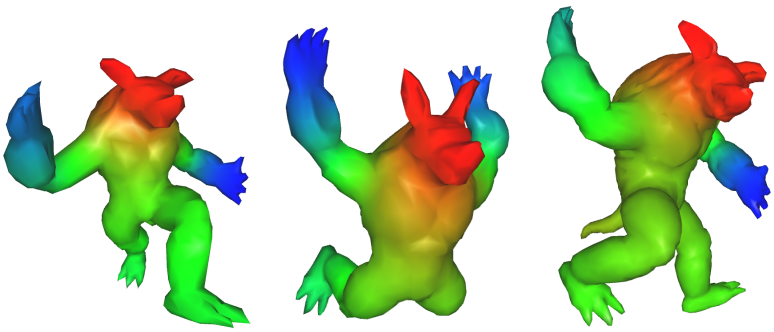
Consider the geometric reconstruction problem. Let matrix P be the position matrix consisting with the $\{x, y, z\}$ coordinates of each vertex:

$$P = [\mathbf{x}, \mathbf{y}, \mathbf{z}], \tag{3.1}$$



(a) (b) (c) (d)

Figure 3.2: The 2nd, 3rd, 4th, and 10th eigenfunctions on the shape. Red color indicates larger value while blue color denotes smaller value.



(a) (b) (c)

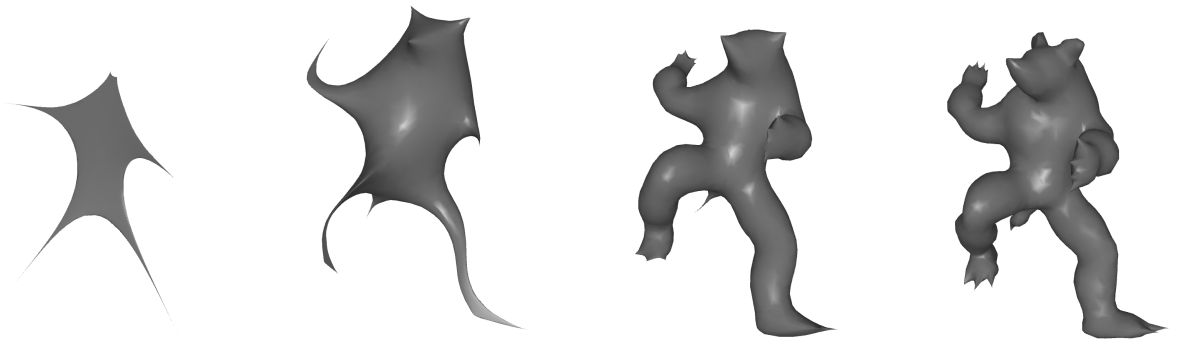
Figure 3.3: The 5th eigenfunctions on the model with different poses. The eigenfunctions are isometric invariant. The three shapes, from left to right, have 1000, 1500, and 3000 vertices, respectively.

where $\mathbf{x} = [x_1, x_2, x_3, ..., x_n]^T$, $\mathbf{y} = [y_1, y_2, y_3, ..., y_n]^T$, and $\mathbf{z} = [z_1, z_2, z_3, ..., z_n]^T$ are coordinate vectors. Then the expansion with equation 2.30 and equation 2.31 can be rewritten in matrix form as:

$$P = VC^T, \tag{3.2}$$

where V is the eigenvector matrix $V = [v_1, v_2, ..., v_n]$ and $C = P^T SV$. Let $A_{1-p,1-q}$ denote a sub matrix consisting of 1-p rows and 1-q columns of matrix A. Then the first k eigenfunctions reconstruction is represented as:

$$P(k) = V_{1-n,1-k} C^T_{1-3,1-k}. \tag{3.3}$$



(a) 5 (b) 20 (c) 100 (d) 400

Figure 3.4: Geometric reconstruction with first 5, 20, 100, and 400 eigenfunctions, respectively.

The reconstructed mesh is represented by the coordinate $P(k)$ with the same connections. Figure 3.4 shows a reconstruction process. The eigenfunctions, corresponding with smaller eigenvalues, represent lower frequency information. As more and more eigenfunctions added up, more details of the mesh are presented. New salient features come out with new eigenfunctions, which means that features are contained by their eigenfunctions within the corresponding frequencies. We define the geometry energy of a vertex i corresponding with the kth eigenvalue

as:

$$E(i,k) = \| V(i,k) \times C_{1-3,k} \|_2 . \tag{3.4}$$

We pick the maxima in E as the feature points, which means more geometry energy is added locally in both spatial and spectral neighborhood. If $E(i,k)$ is larger than those of its neighboring vertices within several neighbor frequencies, it will be picked up as a salient feature point with a scale factor, $sf = 1/\sqrt{\lambda_k^2}$. Notice that, one vertex may be picked several times with different scale factors corresponding with different eigenvalues. See Figure 3.5 for an example.

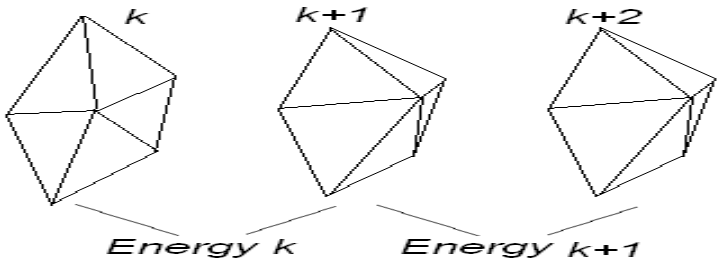


Figure 3.5: Geometry energy between neighboring eigenfunction reconstructions. In this illustrative example, the vertex in the middle of the one ring neighborhood receives largest geometry energy when $(k+1)$*th* eigenfunction is added for reconstruction. Thus it is considered as a maxima at the "frequency" of λ_k

.

Shape Descriptor Construction

In previous sections, we have described how feature points are extracted from an original mesh with scale information factors. The next step is to find a local descriptor for each feature point, which is invariant to translation, rotation, scaling, and isometric deformation and also distinctive enough for similarity measure. We propose to use the Laplace-Beltrami spectrum of the spatial local region defined by the identified scale (i.e., the local support of the corresponding salient feature point). However, most of these regions are open boundary sub-surfaces and properties of those eigenfunctions of Laplace-Beltrami operator. Rustamov [61] mentioned Laplace matrix could meet some problems with open boundary surface. In order to solve this

problem, we attach another surface patch to the open boundary region patch. The attached patch has exactly the same shape as the original patch, but opposite normal at every point. Then, an open boundary path turns into a water-tight surface and equation 2.27 can be applied on it without any problem.

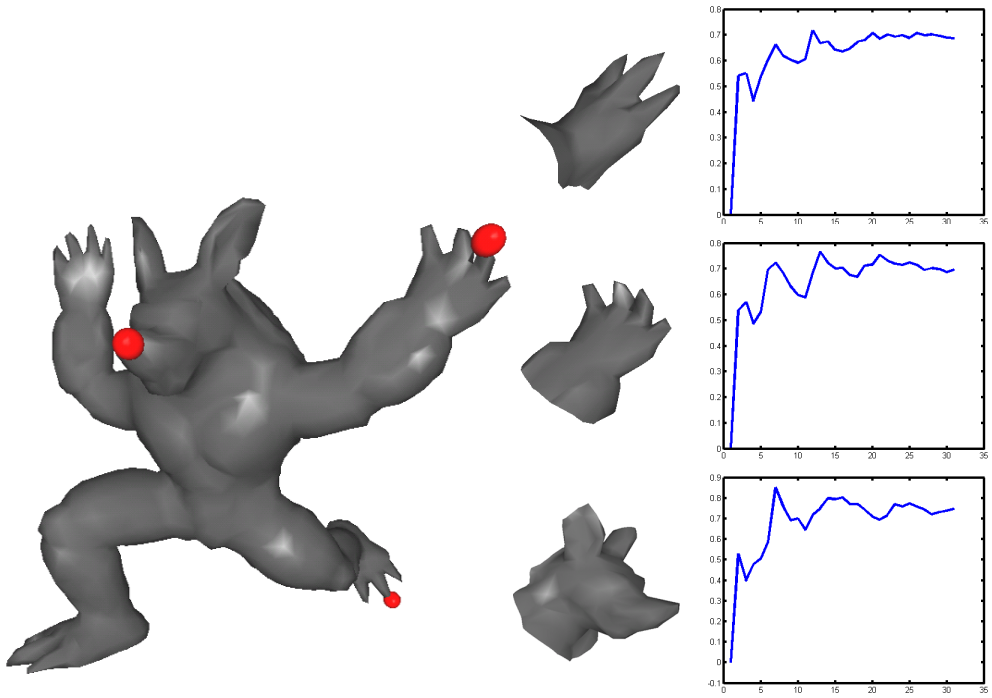
The algorithm procedure is as follow: First, a spatial local patch is extracted by drawing geodesic circle with the feature point as the center and $r \times 1/\sqrt{\lambda_k}$ as the radius, where r is a uniform, constant radius factor. Note that, because of the scaling factor, $1/\sqrt{\lambda_k}$, the shape of the local patch will remain the same despite the scaling of the mesh; Then, equation 2.27 is applied on the patch to obtain the spectrum of the patch; Finally, for similarity comparison, the spectrum is normalize by a fitting function $f(x) = \frac{4\pi}{A}x, x = 1, 2, ..., n$, where A is the area of the local patch [59]. Figure 3.6 illustrates how to construct a descriptor for a feature point. The histograms show the spectral values over the Eigenvectors in the local patches. The matching can be performed by comparing the Euclidean distance between two descriptors.

3.3 Correspondence and Matching

Given a 3D surface, we can now represent it with a set of salient spectral geometric features. In this section, we propose a method to build correspondence between two models with those features. The correspondence problem can be described as: If there are two set of features $\{p_i\}$ and $\{p'_{i'}\}$, try to find a mapping function $\phi(\cdot)$ such that $\phi(i) = i'$. The similarity between these two model relies on the mapping function ϕ. We denote the similarity as

$$Sim(\phi) = Sim_s(\phi) + Sim_p(\phi), \tag{3.5}$$

where $Sim_s(\phi)$ is the similarity calculated based on single feature to single feature mapping and $Sim_p(\phi)$ based on cluster to cluster mapping. Single to singe feature similarity can be based on the Euclidean distances between the spectrum descriptors. Let $f(i)$ denote the feature



(a) (b)

Figure 3.6: The local shape descriptors of the model in (a) are normalized spectra of local patches as shown in (b).

vector of the feature i, then $Sim_s(\phi)$ is defined as

$$Sim_s(\phi) = \omega_s \sum_{\phi(i)=i'} C(i, i'),$$
$$C(i, i') = exp(\frac{\| f(i) - f(i') \|^2}{2\sigma_s^2}). \tag{3.6}$$

Cluster to cluster feature similarity is base on the relative geodesic distances and scale factors. Let $g(i, j)$ denote the absolute geodesic distance between i and j based on the spatial coordinate and $sf(i)$ denote the scale factor of i, then $Sim_p(\phi)$ is defined as

$$Sim_p(\phi) = \omega_p \sum_{\phi(i)=i', \phi(j)=j'} H(i, j, i', j'),$$
$$H(i, j, i', j') = exp(\frac{(d_{pg}(i, j, i'j') + \beta d_{ps}(i, j, i'j'))^2}{2\sigma_p^2}), \tag{3.7}$$

where d_{pg} is the distance between relative geodesic distances

$$d_{pg}(i, j, i'j') = |\frac{g(i, j)}{sf(i)} - \frac{g(i', j')}{sf(i')}|, \tag{3.8}$$

and d_{ps} is the distance between scale ratios

$$d_{ps}(i, j, i'j') = |log(\frac{sf(j)}{sf(i)}) - log(\frac{sf(j')}{sf(i')})|. \tag{3.9}$$

ω_s, σ_s, ω_p, σ_p, and β are weight scalars. The goal of the correspondence algorithm is to find certain mapping function ϕ_c, which maximizes the similarity $Sim(\phi)$. If we define binary indicators variable $x(i, i')$ as

$$x(i, i') = \begin{cases} 1 & \text{if } \phi(i) = i' \text{ exists,} \\ 0 & \text{otherwise.} \end{cases} \tag{3.10}$$

then equation 3.5 can be represented with an Integer Quadratic Programming (IQP) problem as

$$Sim(\mathbf{x}) = \sum_{i,i',j,j'} H(i,j,i',j')x(i,i')x(j,j') + \sum_{i,i'} C(i,i')x(i,i'). \tag{3.11}$$

We also constrain a one-to-one mapping which means one feature in a model can not be assigned more than one correspondence in the other model. Consequently, we have $\sum_i x(i,i') \leq 1$ and $\sum_{i'} x(i,i') \leq 1$. These linear constraints can be encoded in one row of A and an entry of b. Therefore, our IQP problem can be formalized in the following matrix form:

$$\max\ Sim(\mathbf{x}) = \mathbf{x}'H\mathbf{x} + C\mathbf{x} \text{ subject to } A\mathbf{x} \leq \mathbf{b}. \tag{3.12}$$

We use the IQP solver proposed by Bemporad et al. [2] to solve the above optimization problem.

3.4 Experiments and Applications

Salient Spectral Geometric Features: In previous sections, we have introduced our method for extracting salient spectral geometric features from a surface. Figure 3.7 shows some examples of salient feature points. The vertices colored with red and green colors are feature points extracted in the spectral domain with our method. Each mesh in Figure 3.7 has 1,000 to 1,500 vertices. In order to find extrema in the spectral domain, each vertex is compared with its one-ring neighbors within three frequencies, the current, previous, and next ones. The extrema are extracted in the first 100 eigenfunctions. Redder color means the feature is found in a lower "frequency", which has large supporting region, while greener color corresponds to higher "frequency". Only lowest "frequency" is visualized for a vertex with multi "frequencies". The highlighted patches illustrate the local supports for some identified feature points. The experiments show that the salient features are very stable and invariant to Euclidean transforms and isometric deformations.

Shape Correspondence: A nature application with the salient features is partial matching.

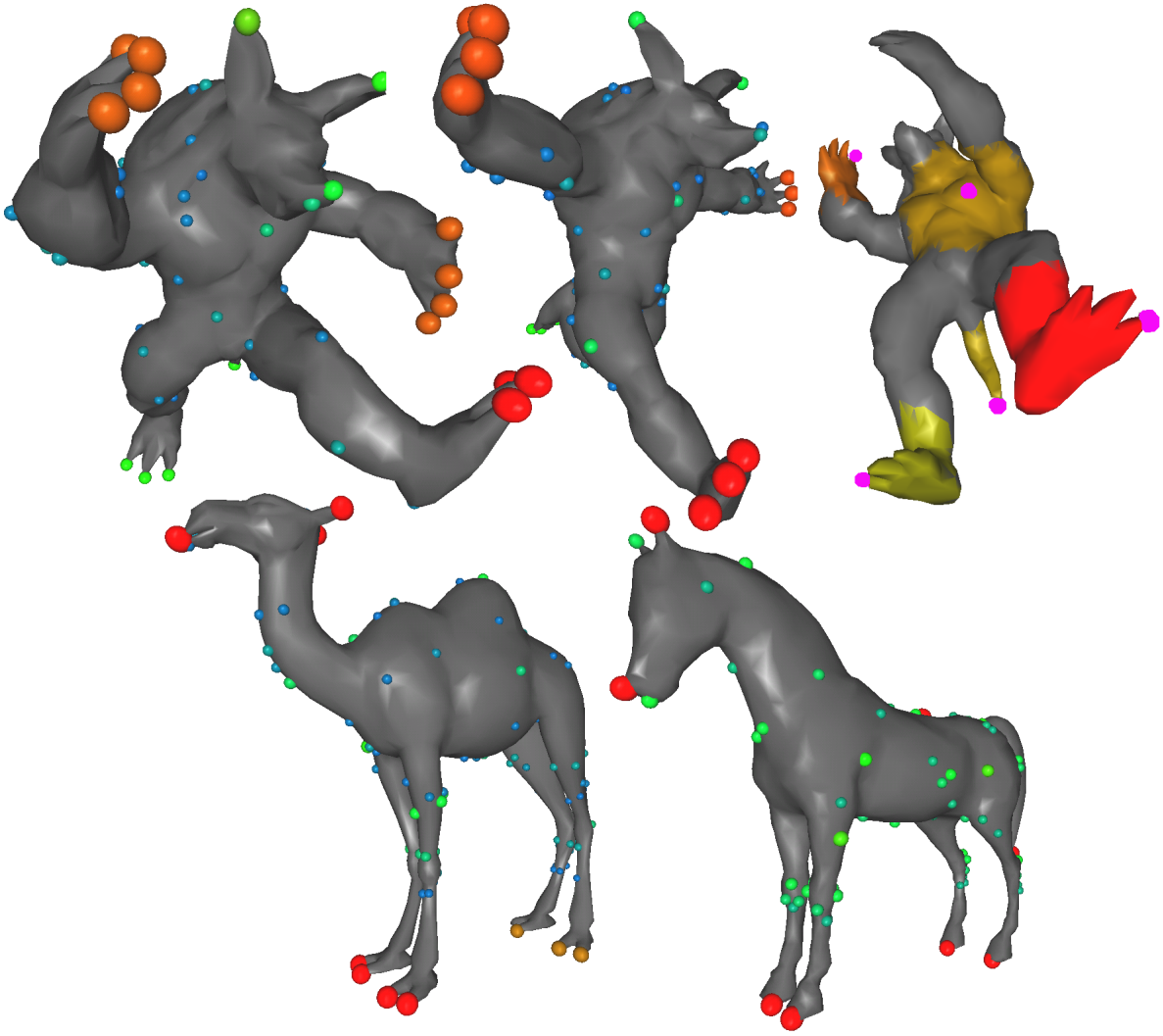
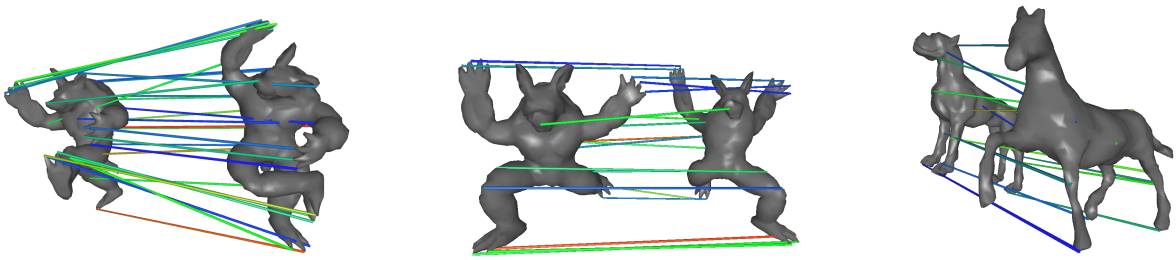


Figure 3.7: Salient feature points extracted in the spectral domain. Redder color means the feature is found in a lower "frequency", which has a large supporting region, while greener color corresponds to a higher "frequency". The highlighted patches illustrate the local supports for some of the feature points.



(a) (b) (c)

Figure 3.8: Examples of matching. (a) and (b) demonstrate correspondence between shapes form the same model with different poses. (c) shows correspondence between similar shapes.

Because each feature point is found with both the position and the scale factor, there is a local region to support the feature point. The ratio of the local support region to the entire surface is independent of the scale of the original surface. In our experiments, the radius constant r is set to be 1.7 and a geodesic circle patch is approximated with the graph shortest edge path on the mesh. Then, the spectrum of the local patch is calculated, and the descriptor consists of the eigenvalues of λ_1 to λ_{21} with normalization. Fig 3.1 and Figure 3.8 show some examples of matching and partial matching. The two meshes in the matching pairs are different in position, orientation, scale, pose, and triangulation. Our experiments demonstrated that the salient features are very powerful in matching of similar shapes. In Figure 3.8, (a) and (b) are shapes of the same armadillo model with different poses. We can see that even poses are quite different from each other but the correspondences are stable, even at very detailed levels. Note that, the mirrored matches could happen in our algorithm. Not only the different poses from the same model, but also similar shapes can have correct correspondences too. (c) illustrates a dog shape corresponds to a horse shape with their similar features, such as heads, knees, necks, and feet.

Shape Retrieval: Another application is 3D shape searching and retrieval in large databases. Not only globally, shape can also be searched by parts, which leads to a more powerful partial matching. We use SHREC '07 3D shape database. We extract 10 shapes from each category. Since our framework extracts stable and distinctive salient spectral geometric features, the

retrieval task is very straightforward by comparing the matching score of the IQP throughout the database, and then picking up the best ones as the query outputs. Figure 3.9 shows some results demonstrating the stability and accuracy of our retrieval. For the dataset that we use, our method outperforms the reported best result in the latest SHREC contest [70]. Figure 3.10 shows the precision/recall graph.

As we mentioned in the previous sections, another desirable property of the salient spectral geometric features is their powerful partial representation. Figure 3.11 shows how our method performs in retrieving shapes if only a part is given.

3.5 Summary

We have introduced a novel 3D shape representation with a set of salient feature points in the LaplaceBeltrami spectrum. The spectrum is defined as the family of eigenvalues of the LaplaceBeltrami operator on a manifold. The eigenvalues and eigenfunctions are invariant to translation, rotation, and scaling. They are also invariant to isometric transformations. We have introduced how to calculate the spectrum directly on triangle meshes. The results showed the spectrum relies only on the geometry of the manifold. It is very stable under Euclidean and isometric transformations and is independent of different triangulations. The spectrum energy domain is obtained by projecting the geometry onto the eigenfunctions. The salient features are the energy maxima in the geometry energy domain. The salient features share the nice properties of the Laplace-Beltrami spectrum. The maxima provide not only where the features are on the manifold but also the frequency where the features lie in. A scale of a local region can be determined with the frequency to support a feature point. With the IQP algorithm, correspondences can be built among variant shapes in very detailed levels. Salient spectral feature point representation is ideal for fundamental shape matching. Our experiments show its great power in shape retrieval and searching. Besides global matching, partial matching is also supported in our framework.

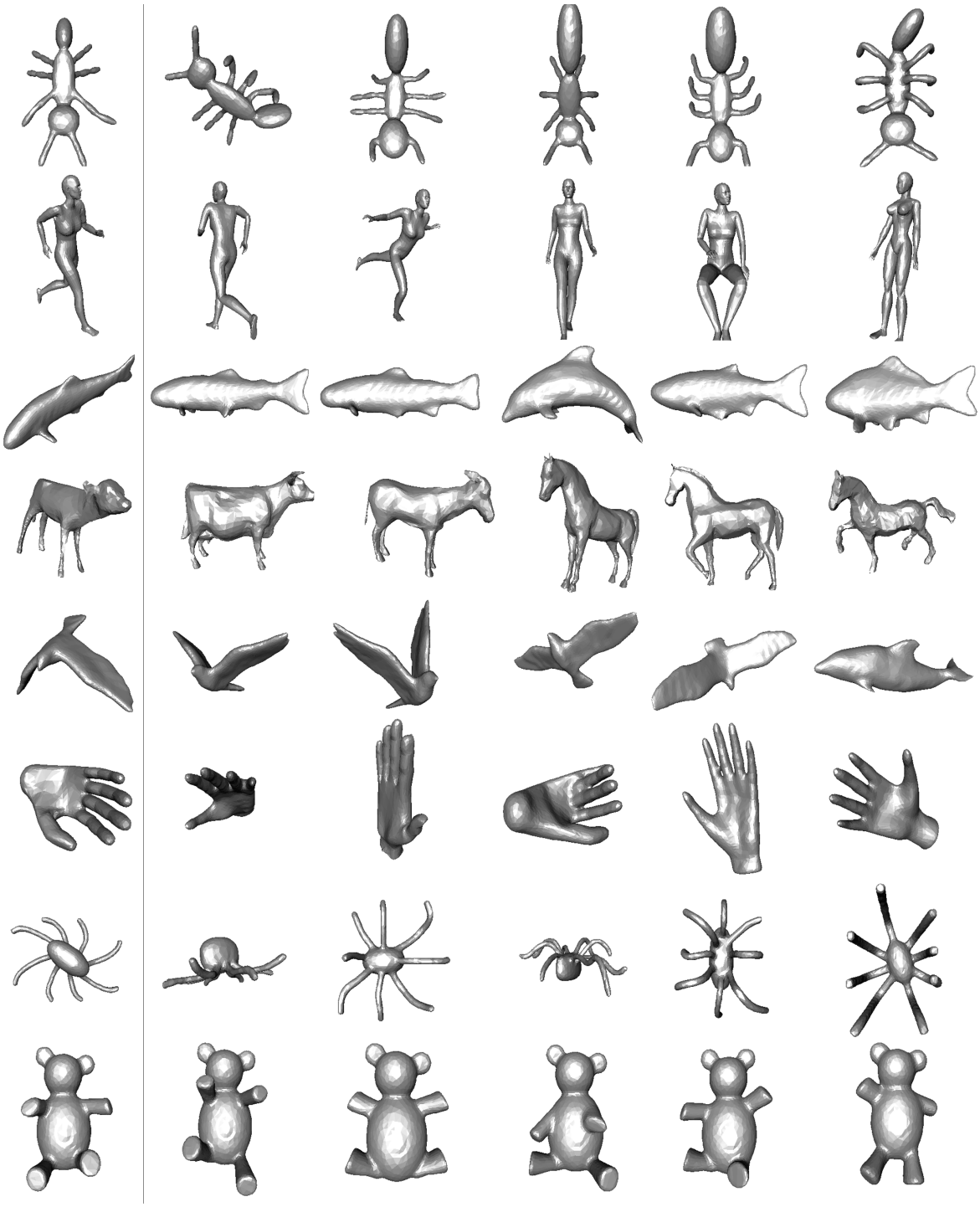


Figure 3.9: Examples of shape retrieval with salient spectral geometric features. The 3D shapes at the most left column are input queries, and those on the right are the first five retrieved results from the database.

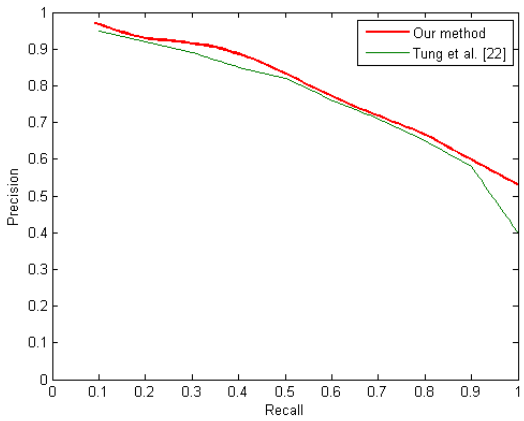


Figure 3.10: The overall averaged precision/recall graph of our method on the SHREC datasets and its comparison to the method by Tung et al. [70].

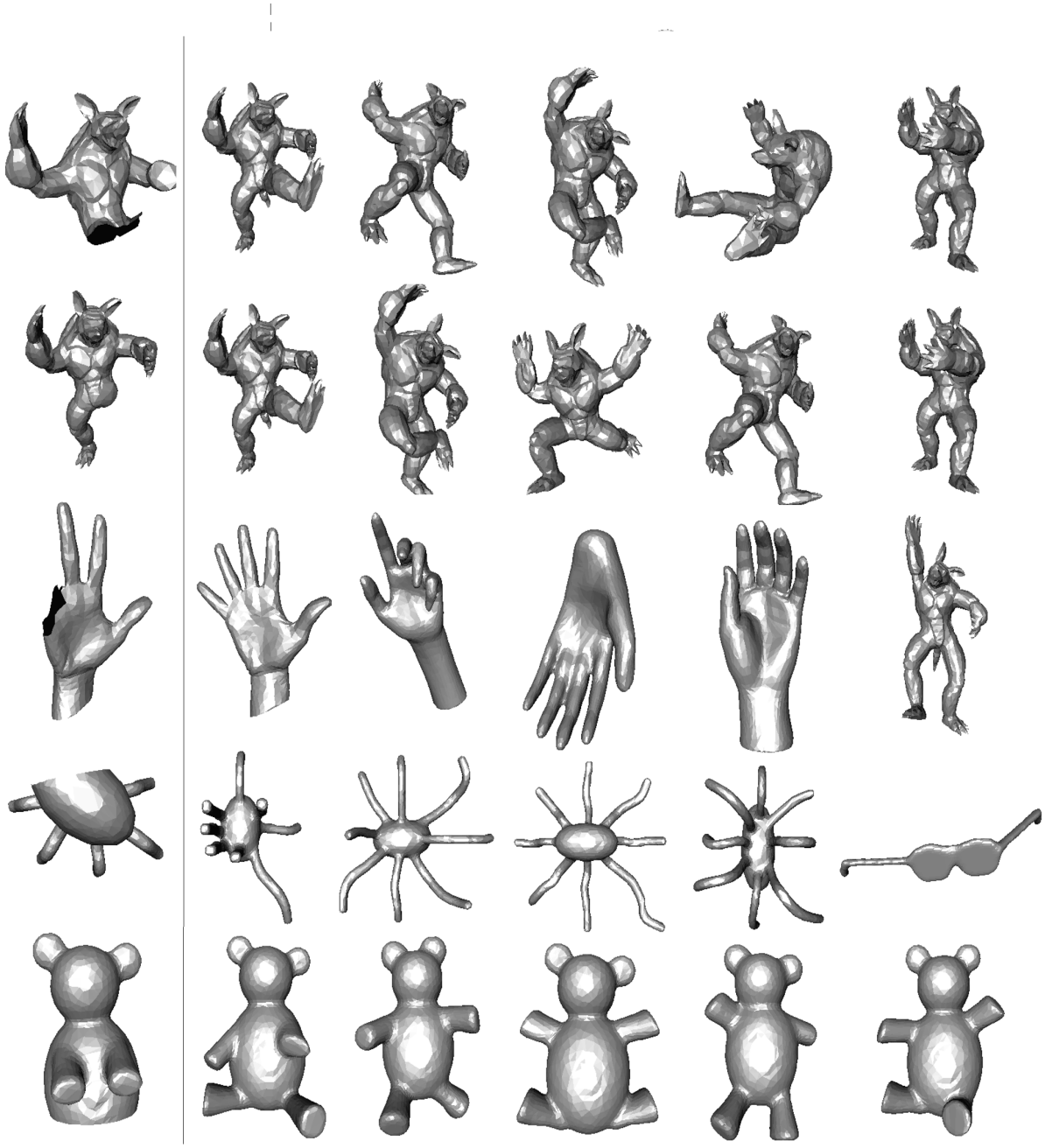


Figure 3.11: Examples of partial shape retrieval with salient spectral geometric features. The shapes at the most left column are input queries, and those on the right are the first five retrieved results from the database.

CHAPTER 4

NEAR ISOMETRIC MOTION ANALYSIS USING SPECTRAL GEOMETRY

The previous chapter solves the problem of measuring the similarities among different static shapes. Poses casted by the same original object are considered as the same shape in terms of spectral features. However, there is still isometric deformations which carry rich geometric information. In this chapter we propose a novel method to analyze a set of poses of 3D models that are represented with triangle meshes and unregistered. Different shapes of poses are transformed from the 3D spatial domain to a geometry spectral domain that is defined by Laplace-Beltrami operator. During this space-spectrum transform, all near isometric deformations, mesh triangulations and Euclidean transformations are filtered away. The different spatial poses from a 3D model are represented with near isometric deformations, therefore, they have similar behaviors in the spectral domain. Semantic parts of that model are then determined based on the computed geometric properties of all the mapped vertices in the geometry spectral domain. Semantic skeleton can be automatically built with joints detected as well. The method turns a rather difficult spatial problem into a spectral problem that is much easier to solve.

4.1 Related Work

Shape animation and deformation often relies on shape interpolation. Given two or more key frames of a shape, the intermediate deformations are interpolated or blended. Extrapolation can also be applied, which decides what the shape is going to be following the deforming direction. The interpolation is applied on locations of the corresponding vertices or faces. Certain constrains are considered to make the interpolation as natural as possible while avoiding some artifacts such as local shrinking or collapse. Kilian et al. [36] treated each pose of shapes

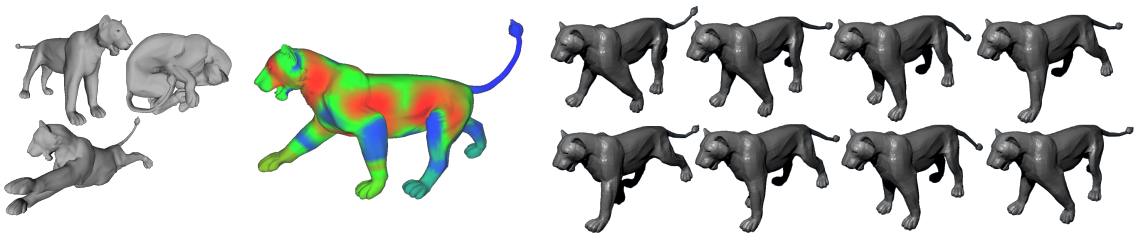


Figure 4.1: The procedure of our pose analysis method. Given several unregistered poses of a model which are unregistered and have different triangulations (shown in the left pane), an re-embedding from the spatial domain to a geometry spectral domain is built as shown in the middle. The poses are analyzed in the geometry spectral domain. The geometric behavior of each point on the pose surface is classified. Then, semantic parts on any poses from the same model can be determined. Colder color in the middle figure indicates rigid part on the surface, while warmer color denotes articulated part. With the graph and skeleton driven algorithms, the static 3D surface turns into a semantically articulated model which can cast animation.

as a point in a shape space. A best interpolation is a geodesic path between two poses, which can preserve original length on the surfaces as much as possible. James and Twigg [29] and Chu et al. [9] employed mean shift clustering to learn the near rigid parts of surface from a sequence of poses to guide the interpolation. This kind of methods usually requires one-to-one vertex-face correspondence, either pre-given or obtained by other registration algorithms. The correspondence requirement limits the capabilities of these methods since registration itself is another challenging problem. The shape interpolation focuses on global smoothness and influence in order to provide smooth and fluent shape sequences. It is based on signal interpolations with constrains.

Skeleton driven mesh deformation is another popular kind of shape approaches. A shape is deformed under the control of an articulated structure, which is more natural to human understanding [11]. It can provide local control and free deformation. Yan et al. [74] employed simplex transformations to make the skeletons drive the surfaces instead of vertices. Weber et al. [72] used geometric information to guild the skeleton to preserve local details. This kind of approaches usually requires the skeletons to be manually designed to reach a better result. He et al. [21] introduced harmonic function on the surface to build Reeb graph [5, 6, 10, 54, 55, 65]

as the skeleton. They reduced the manual operation to picking only one or a few reference points on the surface.

All work aforementioned is from a perspective of spatial analysis. They have to overcome many Euclidean factors such as translation, rotation, and scaling before they analyze the pure geometry properties. Recent research shows that 3D surfaces can also have spectral properties, to which the Euclidean factors are not significant. Karni and Gotsman [30] defined mesh Laplacian on polygon meshes based on the adjacent matrix. The eigenvectors of the Laplacian matrix form an orthogonal basis on the mesh surface. The eigenvalues denote different frequencies. Then signal processing algorithms can be applied to mesh surface, such as filtering, denoising, mesh compression. This mesh Laplacian relies on the triangulations of meshes. Reuter et al. [59] introduced Laplace-Beltrami operator to Riemann manifolds represented with surfaces in 3D Euclidean space. The operator is invariant to Euclidean transformations and isometric deformations. The eigenvalues can be used as shape descriptors which is not only invariant but also distinctive. The eigenvalues also contain much information such as the area of the surface, topology, and boundary length. Lévy [42] focused more on the eigenfunctions of the Laplacian equation. The eigenfunctions form a orthogonal basis for the functions defined on the Riemann manifold and can "understand the geometry". A lot of applications can be achieved, such as signal processing on surfaces, geometry processing, pose transfer and parametrization. Rustamov [61] defined a Global Shape Descriptor (GPS) embedding based both on eigenvalues and eigenfunctions and gave a G2 distribution based on the GPS, which can be used as a global shape descriptor stable to topology changes. Hu and Hua [24] analyzed shapes with salient features extracted from the shape spectra.

Our work starts from the perspective of spectral geometry. Therefore, it may extract the pure geometric information behind variant Euclidean factors. The discrete setting makes the Laplace-Beltrami operator can be applied on triangle mesh directly. This saves preprocessing and handles more types of data.

4.2 Shape Spectrum on Triangle Meshes

Considerig equation 2.3, the solution $\{0, \lambda_1, \lambda_2, \lambda_3, \cdots, \lambda_{n-1}\}$, is the family of eigenvalues of the generalized eigenvalue problem defined above. The eigenvectors $\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_{n-1}$ represent the eigenfunctions on the mesh. They define the spectrum of a shape. As it can be seen, the number of eigenvalues and eigenfunctions is reduced from infinite to n, because a triangle mesh is a finite discrete sampling of a continuous surface. It is similar to the discrete Fourier transform and the continuous one. In practice, infinite eigenvalues and eigenfunction is not necessary. Only first a few of eigenvalues and eigenfunction are employed to build the geometry spectral domain.

Figure 4.2 illuminates the 3rd, 5th and 10th eigenfunctions on different poses. Note that, the Laplace-Beltrami operator is define on continuous manifold, so the triangle meshes are required to be manifolds. They could be either closed manifolds or those with open boundaries, with the same topology. The color turns from cold to warm while the function value grows from a small one to a big one. The eigenfunctions always change along the surface geometry. The 3 poses are quite different from the spatial view, but the eigenfunctions stay stable on the surfaces. The eigenfunctions rely only on the surface geometry. The shapes are not only different from each other with poses, but also the triangulations. The pose in the first column has about 2,000 vertices; the one in the second column has about 10,000; the one in the last column has about 20,000. As it is discussed above, the eigenfunction is also invariant to triangulations. These properties guarantee the geometry spectrum embedding is invariant to pose deformations and mesh triangulations. In other words, vertices from different poses but at the same position in terms of surface geometry will be embedded together in the geometry spectral domain, no matter how the poses are deformed or how different the samplings and triangulations are.

The spectrum can describe the intrinsic geometry within the original shape. Theoretically, the shape spectrum is invariant to isometric deformations. However, problems arise when dealing with the real data. Different poses casted by an object are usually near isometric to each

other. The deformations near the joints break the isometric constraint. The computations also bring numerical errors. Dey et al. [12] studied the spectral stabilities under near isometric deformation. Their results show the spectra achieved with the cotangent scheme, including the discrete operator in our method, is stable in terms of eigenvalues. Our method produces the similar results. Figure 4.3 lists five shapes represented with triangle meshes while Table 4.1 lists their first a few eigenvalues. The eigenvalues are normalized with the first non-zero eigenvalue to filter away the global scaling according to [59]. Because the first 3 poses are casted by a same armadillo model, they are considered to be near isometric to each other. This fact is demonstrated by the similar eigenvalues. When the models different, eigenvalues are dramatically different too, as shown in Table 4.1. The eigenvalues have enough power to distinguish models and shapes globally. In addition, there are some other potential problems of eigenvectors/eigenfunctions as discussed below which may affect our algorithm. Rueter et al. also discussed these problems in [57].

- Sign flips occurs. If $\mathbf{v}$ is an eigenvector, $-\mathbf{v}$ is also one according to the definition. Rueter [57] admitted that, sign flips cannot be detected intrinsically on an almost perfect intrinsic symmetric shape. We employ the absolute value to avoid the sign flip problem.

- Eigenvectors switch. The neighbor eigenvalues may switch due to the perturbations of the deformations and numerical computations. So are the corresponding eigenvectors. It happens nearly on every mesh. Rueter [57] gave a solution to reorder the eigenvectors based on the Morse-Smale graph. We use the same scheme. Without further notations, all the eigenvectors, in the rest of the chapter, refer to the reordered ones.

- Higher dimensional eigenspaces can theoretically occur. However, they rarely happens in practical data. We have not found any example in our results so far.

- Duplicated eigenvalues may exist. A highly symmetric shape, e.g. sphere or cube, has duplicated eigenvalues. The linear combinations of the corresponding eigenvectors are

also eigenvectors. Nevertheless, practically used animation models do not have such high symmetry. That is to say that duplicated eigenvalues rarely happen practically in our application.

- Low frequency eigenvectors are stable under near isometric deformation. Rueter [57] had a detailed discussion about the stabilities of the shape spectrum with respect to near isometric deformations and noises, and used direct spectral embedding for the semantic shape segmentation. Our experiments also show that the low frequency eigenvectors are quite stable. The third row of Figure 4.2 demonstrates the stability of the 10th eigenvectors for different poses. Although the spectra are stable globally, the local values of eigenvector may shift. This usually happens when there is a twisting deformation. The shifting will affect the registration accuracy under this single frequency. However, with multi-frequency embedding and multiple shape data, the accuracy will be corrected by other values that are stable.

The above discussion shows the near isometric shapes will have similar behavior in the spectral spaces.

Table 4.1: Normalized eigenvalues of different shapes in Figure 4.3

Shape	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7
Armadillo(a)	0	1	1.23	1.64	2.90	4.37	6.32	8.83
Armadillo(b)	0	1	1.36	1.81	3.20	4.52	6.48	8.51
Armadillo(c)	0	1	1.25	1.33	2.28	4.83	6.76	8.68
Elephant(d)	0	1	2.44	3.07	3.51	3.98	4.24	4.70
Lion(e)	0	1	1.51	2.57	2.66	2.71	4.69	7.92

4.3 Geometry Spectral Domain Embedding

The Laplace-Beltrami operator define a family of eigenvalues and a family of eigenfunctions. The eigenvalues can be used as shape descriptors which are stable and distinctive. It also contains "frequency" information. The smaller eigenvalues denote lower frequencies. The eigenvectors form an orthogonal basis on the manifold. Any functions can be projected to the

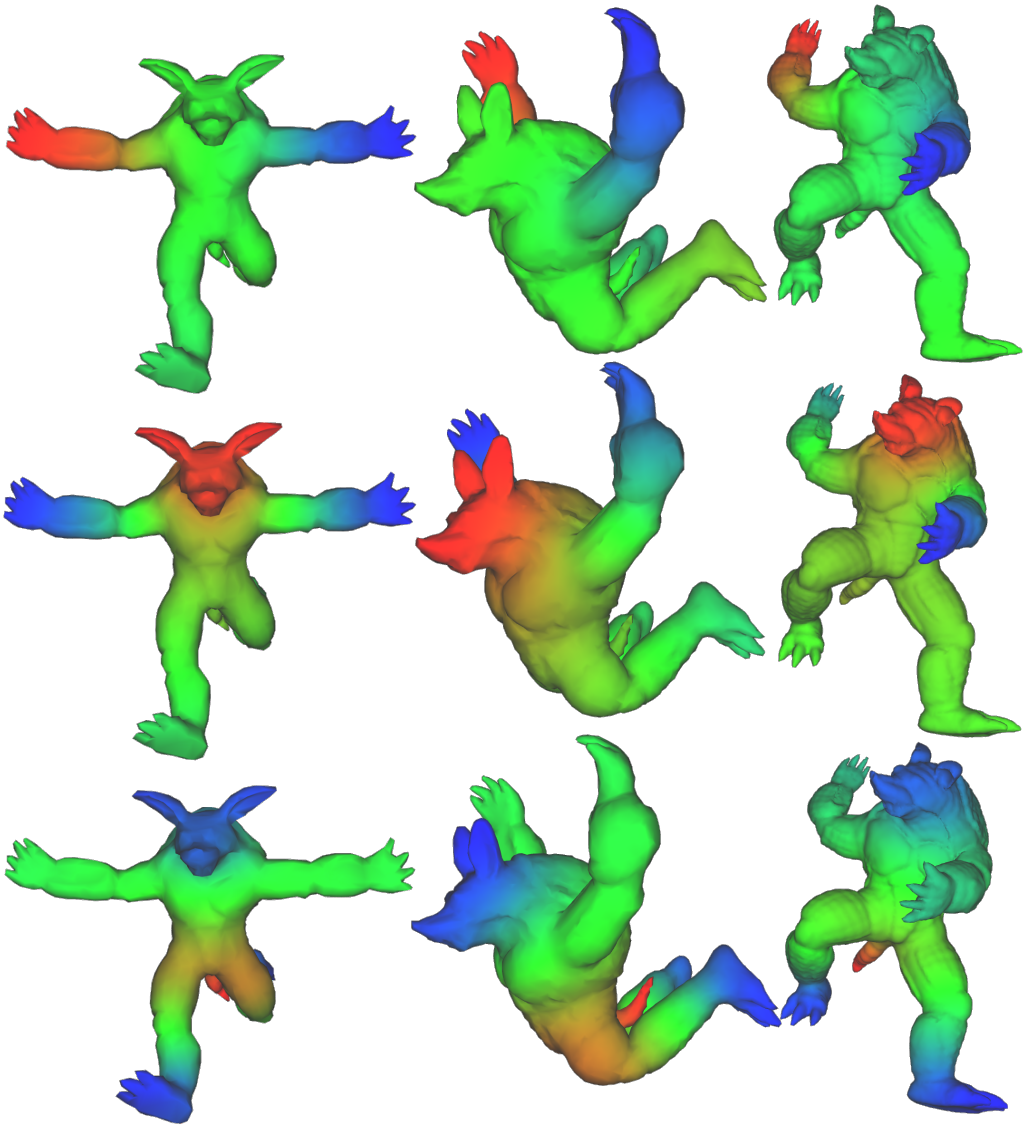
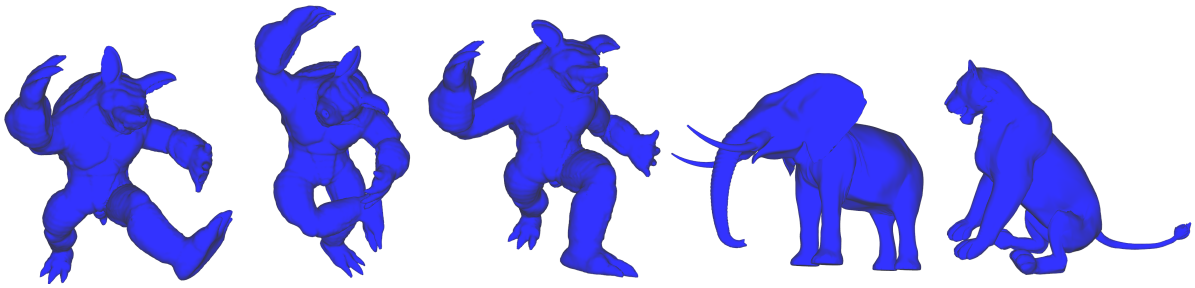


Figure 4.2: The 3rd, 5th, and 10th eigenvectors of discrete Laplace matrices on 3 different poses. Each column demonstrates a pose while each row shows the 3rd, 5th, and 10th eigenvectors from the top to the bottom. The color from blue to green and then to red demonstrates the value changes from small to large. Each eigenvector shows some meaning of the surface. Within a pose, higher order eigenvector shows higher frequency. Note that, the pose surfaces in the first column have about 2,000 vertices; the ones in the second column have about 10,000 vertices; the ones in the last column have about 20,000 vertices. The eigenvectors are not only meaningful but also stable to poses and triangulations.



(a) (b) (c) (d) (e)

Figure 4.3: 5 different shapes in the database. The first 3 shape are different poses from a same armadillo model. According to Table 4.1 the three armadillo poses have similar eigenvalues, while the eigenvalues of the elephant and the lion are quit different.

basis and reconstructed with the linear combination of these eigenfunctions. All of these are global analysis in the spectral domain. However, the main goal here is to study the local behaviors of the surfaces. It is obvious that each eigenfunction f_k is assigned a real value at every surface point p as $f_k(p)$. With respect to each point, there exists a mapping from a point on the surface in 3D spatial space to an infinite geometry spectrum space as:

$$G_S(p) = \left(\frac{f_1(p)}{\sqrt{\lambda_1}}, \frac{f_2(p)}{\sqrt{\lambda_2}}, \frac{f_3(p)}{\sqrt{\lambda_3}}, ... \right), \tag{4.1}$$

where p is a point on a surface S and f_k is the kth eigenfunction corresponding to the kth eigenvalue λ_k of S. Each eigenfunction is normalized by

$$< f_i, f_i >= 1, i = 1, 2, 3, ..., \tag{4.2}$$

on surface S. As we work on the poses casted by the same object, the scales of each surface can be normalized. Thus, the scales of the values in eigenfunctions represent the geometries of the shapes. We summarize some major advantages of this geometry spectral domain embedding as follows:

- If the surface in 3D space has no self-intersection, the embedding has no self-intersection

in the infinite domain either. That means $G_S(p_i) = G_S(p_j)$ if and only if $p_i = p_j$ on S.

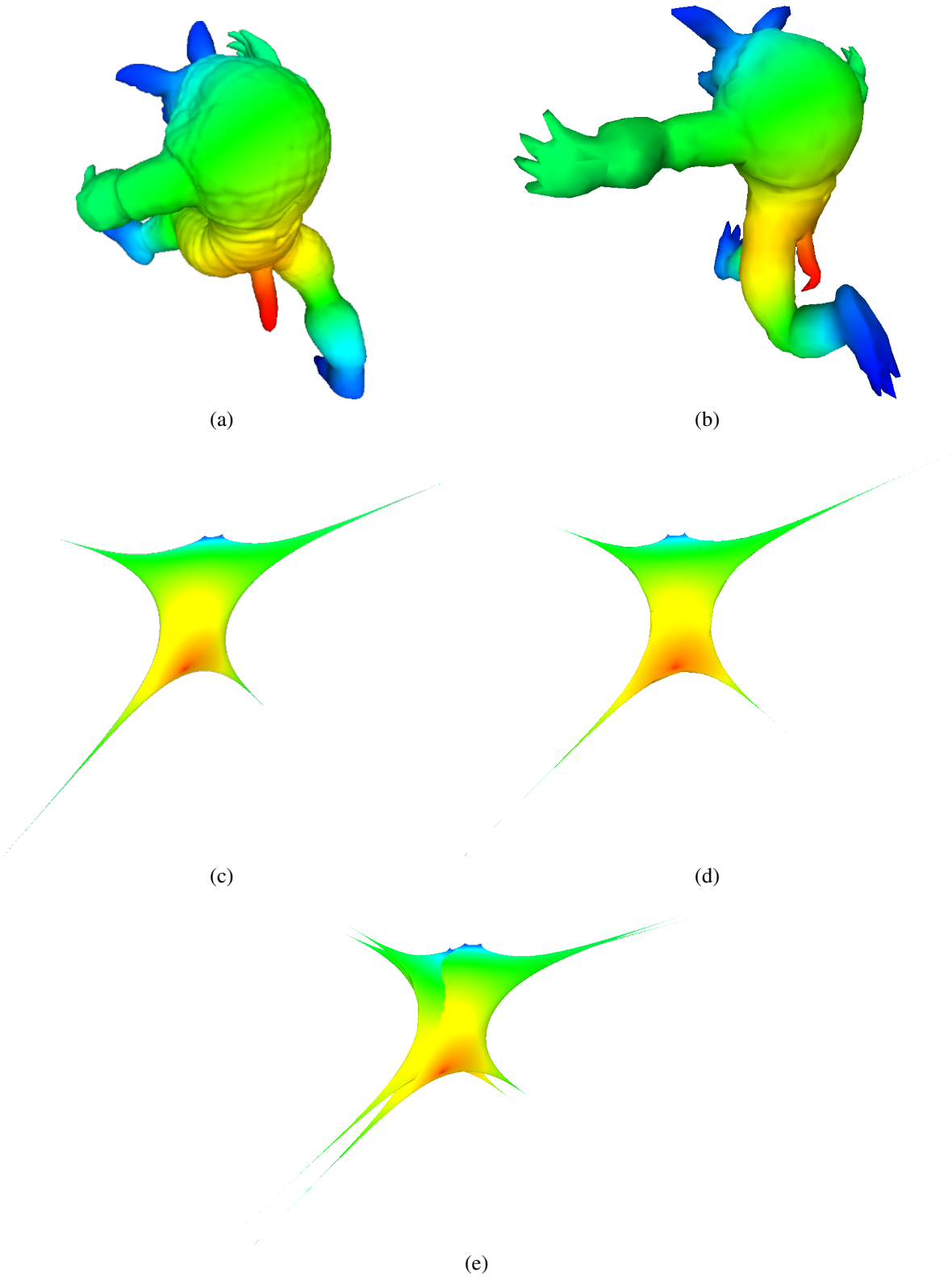
- The embedding is based only on eigenfunctions on the manifold. It only relies on the manifold metric and is invariant to the Euclidean embedding in 3D space of S. The embedding mapping filters away the Euclidean transformations and near isometric deformations.

- The embedding is invariant to different triangulations because of the implementations of discrete Laplacian.

With this embedding, our method does not require any preprocessing such as normalization, remeshing or registration. All these spatial factors do not mater in the geometry spectral domain. With a same surface, there is only one basis set for the embedding. Among different poses, the basis will keep stable. Figure 4.4 demonstrates different poses are registered in the spectral domain neutrally. The 3rd, 4th, and 5th eigenfunctions are picked for form a 3D subspace in the spectral domain. The Euclidean differences, including triangulations, transforms, and isometric deformations, are filtered away. The shapes are colored with the 3rd eigenfunction for rendering. The eigenfunctions of the manifold satisfies equation 2.3. If certain function f_k is a normalized eigenfunction corresponding to some eigenvalue λ_k, according to this equation, $-f_k$ is also a normalized eigenfunction. The experiments also show that eigenfunctions from different pose can flip with sign corresponding to the same eigenvalue. In order to overcome this flipping problem, the mapping is restricted as an absolute one as:

$$AG_S(p) = \left(\| \frac{f_1(p)}{\sqrt{\lambda_1}} \|, \| \frac{f_2(p)}{\sqrt{\lambda_2}} \|, \| \frac{f_3(p)}{\sqrt{\lambda_3}} \|, ... \right), \tag{4.3}$$

The absolute mapping will break the first property about self-intersection above, the symmetric points on the surface will be mapped together. In our framework, it is natural to assume that parts have similar physical behaviors when they are symmetric on the surface. Thus, the absolute mapping does not affect the accuracy of mapping in terms of symmetry. For example,



(a) (b) (c) (d) (e)

Figure 4.4: Transfer spatial manifolds to spectral domain. In the high dimensional space the near isometric deformation with be filtered out and registered uniformly. For rendering purpose, only the 3-5 eigenfunctions are chosen as a sub 3D space of the spectral domain. Despite the spatial deformations and different triangulations, the shapes are registered in the spectral domain neutrally. The color illustrates the 3rd eigenfunction distribution on shapes.

the left and right elbows have the symmetric geometry properties, the registration across each is acceptable, as we do not require the dense mapping and registration.

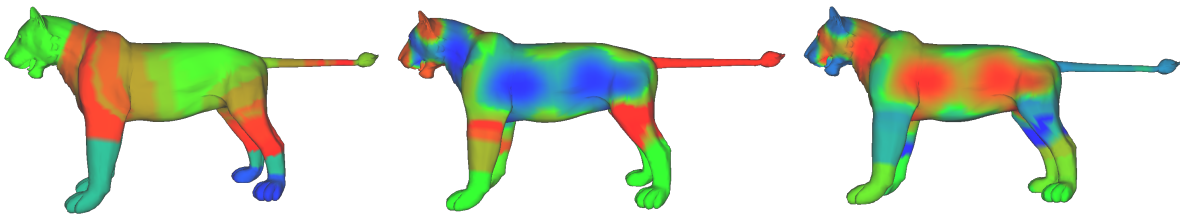
4.4 Semantic Shape Analysis

Semantic Point Classification

The geometry spectrum embedding transforms each point on the surface from the Euclidean space to an infinite geometry spectrum space. Suppose there exists a spatial surface which is near isometrically deformed along time, denoted by $S(t)$. The points with same positions relative to the surfaces S at different time will be mapping into the same coordinates in the spectrum space, despite the different locations, orientations and poses of the original surfaces. Although the point is fixed in the spectral domain, it can carry varying geometric properties on S. That is to say, in the spectral domain, the properties at a point varies while the pose changes. For each point p in the spectral domain, we can define a property function $f_p(t)$. which depends only on time t. Imagine that, if the properties are chosen to be invariant to Euclidean transformations but only sensitive to pose changes, what can be observed in the spectral domain is that properties vary on certain regions along with pose changes while do not on the other regions. The former situation indicates articulations while the latter one indicates rigid parts of the original shape. There are some well studied features on surfaces, such as curvatures, normals, geodesic fans, and so on. In our frame work, mean curvatures are a straightforward choice, as the Laplacian operator is also the mean curvature normal operator on the surface. When the shape deformation $S(t)$ is given, the pose behaviors of all the points can be classified into articulate or rigid.

The data in our frame work is not a continuous surface changing with time but N frames of meshes. N could range from 2 to 10 or even more. The property functions are reduced to a discrete set. A triangle mesh is a discrete sampling of a surface, therefore, an exact correspondence of a vertex may not exist on another near isometric mesh. Thus, the property set on

a vertex is built based on an approximation. Suppose f_p is a feature set which is going to be built at a vertex p on the surface S. First, the embedding of p is calculated, the mean curvature of p is put into f_p as an element. Then, for each following frame of meshes S_i, a point p_i is found as a nearest one to p in the geometry spectral domain base on the Euclidean distance, and the mean curvature of p_i is put into f_p as another element. Therefore, the element in f_p can classify the geometry behaviors of p through different poses. Figure 4.5 illustrates the maxima, minima, and range distributions on the surface among different poses.



(a) (b) (c)

Figure 4.5: Mean curvatures values in spectral pose analysis. (a) maximum mean curvature distribution on each vertex during pose transformations; (b) minimum mean curvature distributions; (c) mean curvature range distribution. The values are histogram equalized for visualization.

Property Smoothing

After the equation 2.27 is solved, vertices can be mapped from spatial space into to geometry spectral domain directly with the indices. The chosen properties can be assigned in the spectral domain. As aforementioned, mean curvature is chosen because the mean curvature normal vector can be obtained by multiplying the Laplacian matrix with the vertex position matrix. Mean curvature is also invariant to Euclidean transformation. However, directly assigning the mean curvature will cause stability problems. The embedding is applied on each discrete vertex. A particular vertex usually can not find the exact matching with other vertices from other surfaces, but has to use the neighbor information in the geometry spectral domain. Based on the definition, mean curvatures obtain by the mean curvature normal operator use only one

ring neighborhood on the mesh. When the mesh is constructed, noise could be involved during the modeling or reconstruction procedures. Thus, the direct mean curvatures will contain a lot of local variance, which will affect the accuracy and stability of the pose analysis in the spectral domain. Therefore, they have to be smoothed first.

The smoothing process is done with Laplacian eigenfunctions. As it is discussed in previous charpters and sections, any function f defined on the surface can be transformed into frequency domain by projecting it onto the eigenfunctions. The coefficient family $\{c_i\}$ forms the frequency spectrum of f as its counterpart in 1D which is well known as Fourier transform. The smoothing is done by applying a low pass filler in the frequency domain then transforming filtered coefficients back to the surface function. Figure 4.6 illustrates the mean curvature reconstruction procedure with different numbers of eigenfunctions. As it is shown, the reconstruction with first 130 eigenfunctions is usually sufficient.

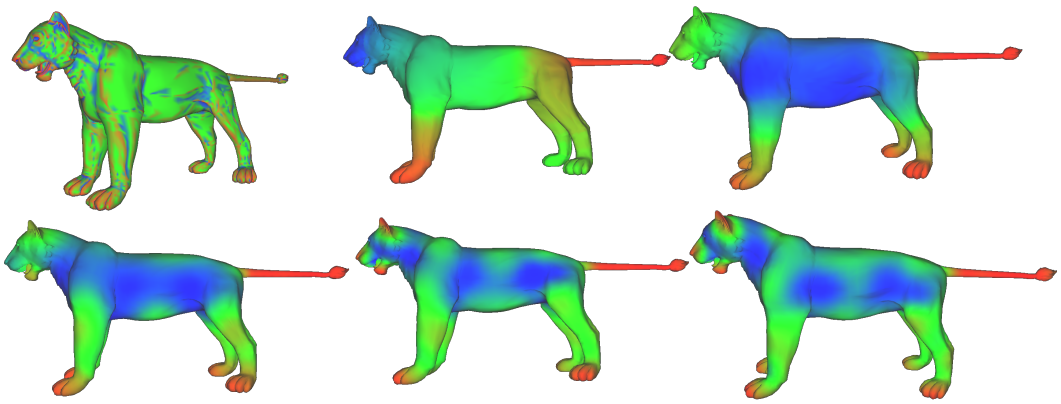


Figure 4.6: Mean curvature reconstruction on eigenfunctions of the manifold. Form left to right, top to buttom, the first figure is the discrete mean curvatures on the surface, obtained by applying Laplace-Beltrami operator on the Euclidean embeddings; the rest ones are reconstructions with the first 6, 20, 50, 100, and 130 eigenfunctions.

4.5 Automatic Skeleton and Joint Extraction

When all the points on the surface shape are classified and clustered into semantic parts, it enables an automatic skeleton construction with joint identification. Here we adopt Reeb graph to achieve this goal.

Skeleton Extraction

Reuter [57, 60] discussed the skeletal representation based on the eigenfunctions. We found that, in the practical data, the intersections of different parts are not stable if the centers of Reeb graph are employed directly. They may shift away from the semantic locations where they should be. The experiments show that iteratively shrinking the mesh to the center producing smoother results. Our skeleton construction is automatic with two simple steps as demonstrated in Figure 4.8.

Iso-contour shrinking. For each vertex on the mesh, the contour with the same function value of the vertex is traversed and found. Then, the vertex is moved the geometric center of the iso-contour. This results in a skeleton-like mesh. Figure 4.7 illustrates the iso-contours of the eigenfunction of the first nonzero eigenvalue.

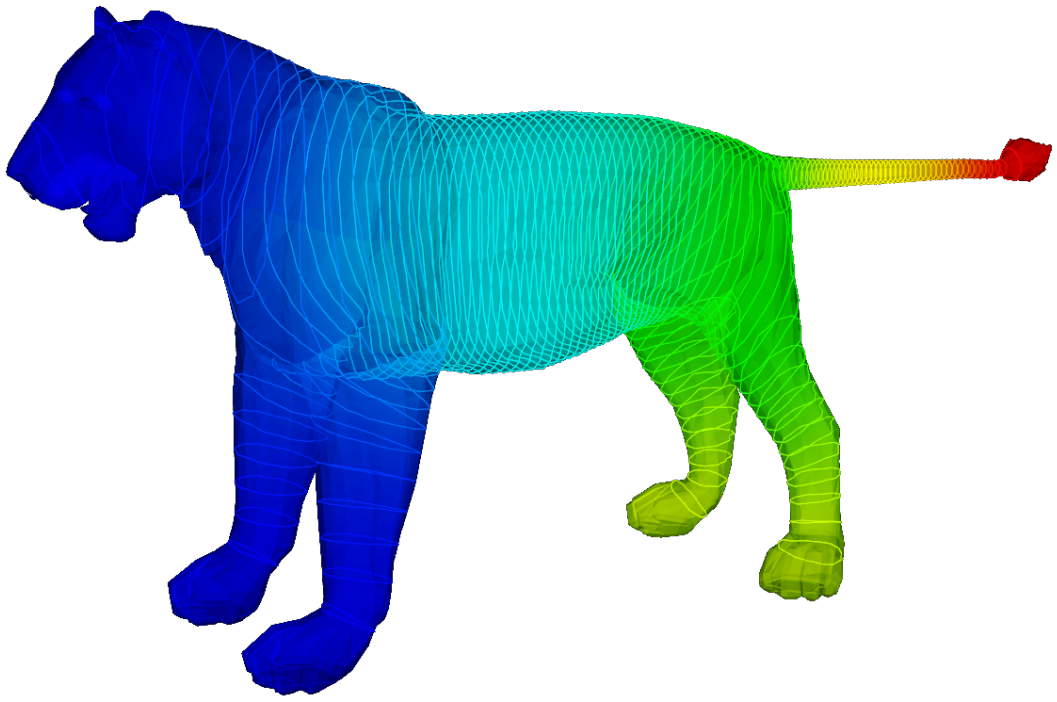


Figure 4.7: The iso-contours of the eigenfunction of the first nonzero eigenvalue.

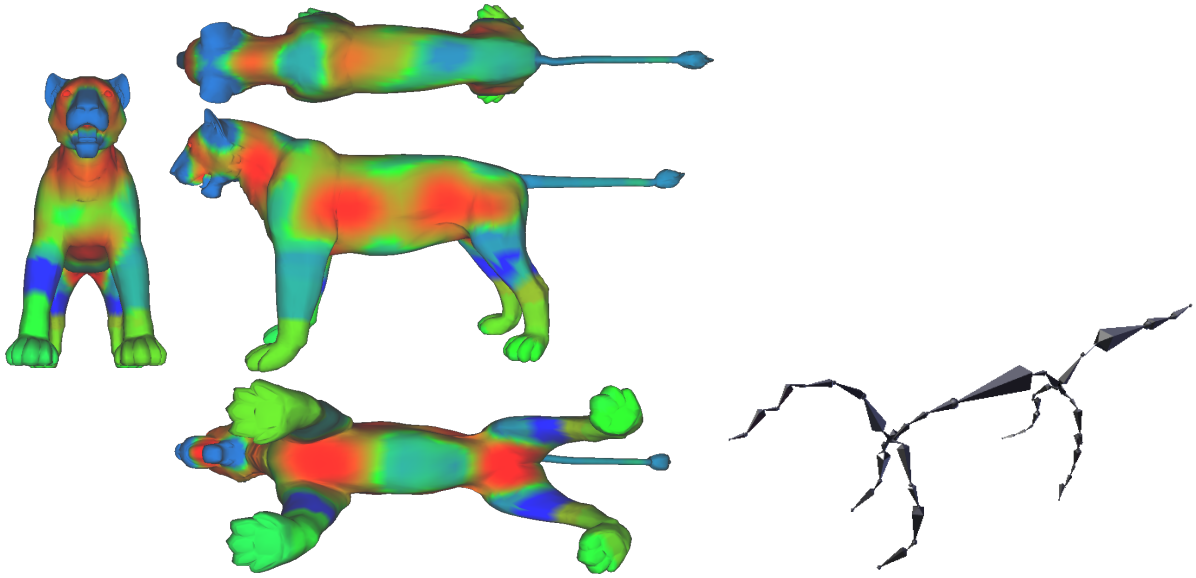
Skeleton construction. Applying the algorithm in [54] on the shrunk mesh with the original eigenfunction. Because the mesh is shrunk to the skeleton shape, the spatial embedding of

the Reeb graph is accurate to become skeleton.

Figure 4.8: Automatic skeleton generation. From left to right: the first non-trivial eigenfunction of the loin model; shrink mesh based on iso-contours; skeleton generated with Reeb graph algorithm; and the embedding of the skeleton within the original model.

Joint Detection

Based on the changing geometric behaviors of points in the geometry spectral domain, we are able to automatically spot out the joints as long as the deformation around the joints are presented in the given deformation sequence. Figure 4.5 demonstrates the basic ideal of the pose analysis in the geometry spectral domain. Figure 4.5(a) is the distribution of the mean curvature maxima on the surface. The larger the value is, the more the surface on that point can bend along relative to the negative direction of the normal at that point. Figure 4.5(b) is the distribution of mean curvature minima. It predicts the behavior that surface bends along the positive normal direction. Note that, the values on the surface are histogram equalized. The same color does not mean the same value across different surfaces. Ideally, if a part is always rigid during pose transformation, the geometry shape will never change. A point on that part has the exact constant mean curvature all the time. Thus, the minimum and maximum of mean curvature are equal to each other. On the contrary, if a part varies, the minima and maxima will fall away from each other. This mean curvature change range is a measurement describing how "rigid" the point and its neighborhood is, which is shown in Figure 4.5(c). The result is very natural. The articulations like neck have different forms under different poses. The parts like nose will not change too much during different poses. Figure 4.9 shows the complete example.



(a) (b)

Figure 4.9: Mean curvature range distributions on a lion model and the extracted semantic skeleton with joints identified based on the distribution.

4.6 Experiments and Applications

In this section, we show some experiment results of skeleton and joint extraction as well as some further applications based on the semantic skeletons. Note that, the pose shapes are represented with triangle meshes. In our experiments, we use mesh data sets from SHREC07 and the one Sumner and Popović [67] used.

Skeleton and Joint Extraction

Figure 4.10 gives an example of the armadillo shape. The main body, especially the chest and the back shell, will not have much variance when it casts different poses. Instead, when the armadillo often changes its postures of head, arms or legs, the neck, shoulder, and waist follow the pose changes. The mean curvature ranges on the surface leads to a segmentation directly, which segment the rigid parts and articulations apart. With the help of the mean curvature ranges, hierarchy graphs can be build as it is described in [9]. Figure 4.11 shows another example.

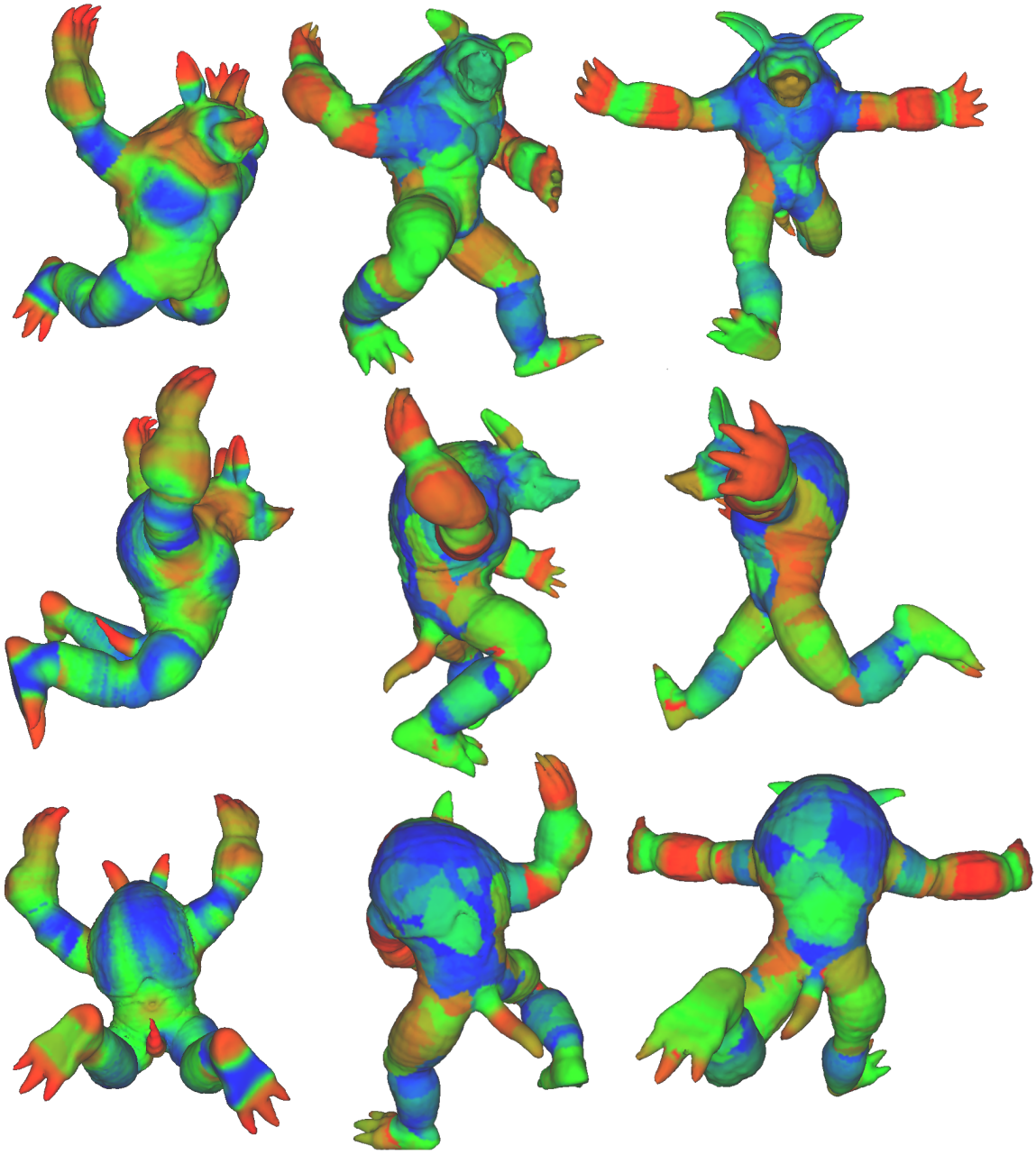
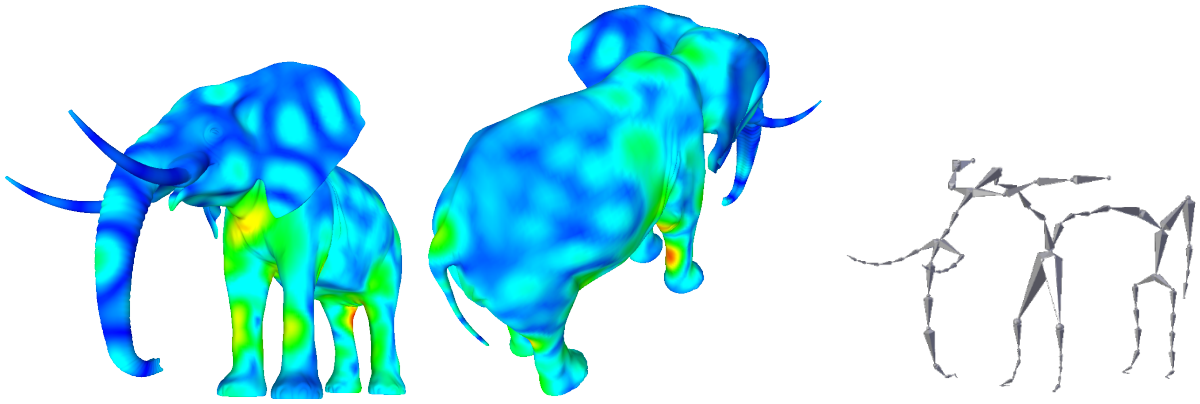


Figure 4.10: Mean curvature range distributions on armadillo models. The chest and back shell usually stay rigid while the neck, elbows, and waist vary during pose changes.



(a) (b) (c)

Figure 4.11: Mean curvature range distributions on an elephant model and the extracted semantic skeleton with joints identified based on the distribution.

Animation

Skeleton driven deformation has extensively studied. It is intuitive to human understanding. Most of poses of creatures are controlled by bones and muscles and then represented by the skin surfaces. The technique of the skeleton driven deformation and animation is widely used in the animation and gaming industry. The classical pipeline is as following: first, manually design a skeleton of a mesh surface; second, assign the vertices of that surface to semantic skeletal parts; then, deform the mesh along the skeleton. Fortunately, our method automatically classifies semantic parts of surfaces during pose changes, and then produces graphs that can be treated as skeletons of meshes. The vertices of the semantically classified surface are automatically associated with skeletal parts with joints identified. There are many existing algorithm can be employed to deform and control such a shape with skeletons. Figure 4.1 has already given an example. The skeletons are learnt from several key frames, but can control the shape to cast much more poses than that. Figure 4.12 also shows some other deformation sequences. These new poses are not any one in the reference frames, but some potential possibilities the models can cast based on the knowledge from existing frames.

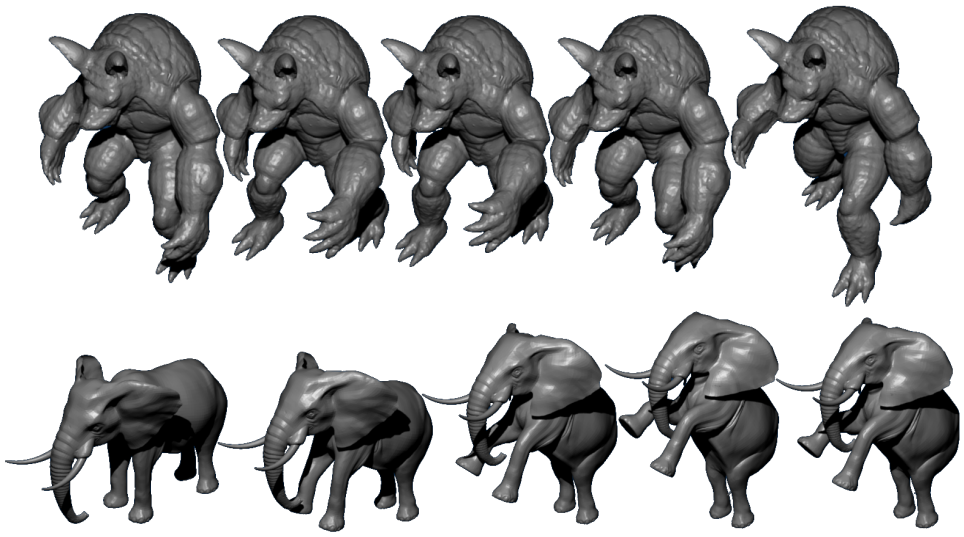


Figure 4.12: Animation sequence. With the automatically extracted semantic skeleton, user can edit the pose freely. The animation sequence can be generated among the edited key frames.

Pose Transformation

Pose transformation is another popular graphics application. The motivation is obvious. If a pose can be transformed form one shape to another similar shape automatically, a lot time can be saved by modeling one key shape instead of modeling lots of different shapes, respectively. In our frame work, the poses are represented with semantic parts. Two similar shapes will have similar semantic parts and skeletons. Graph or skeleton matching algorithm, such as [68] can find the correspondence between two similar skeleton. After that, a pose driven by a skeleton can be transformed to a similar pose with a corresponded skeleton. Figure 4.13 demonstrates how running poses are transformed from a lion model to a cat model.

4.7 Summary

Differing from the existing spatial approaches, our method enables to understand the poses in the geometry spectral domain. The geometry spectrum is based the eigenvalues and eigenfunctions that are defined by the Laplace-Beltrami operator on the surface. The Laplace-Beltrami operator relies only on the metric on the surface, therefore, it is variant to Euclidean translation, rotation and scaling. It is also invariant to isometric deformations. Thus, the eigen-

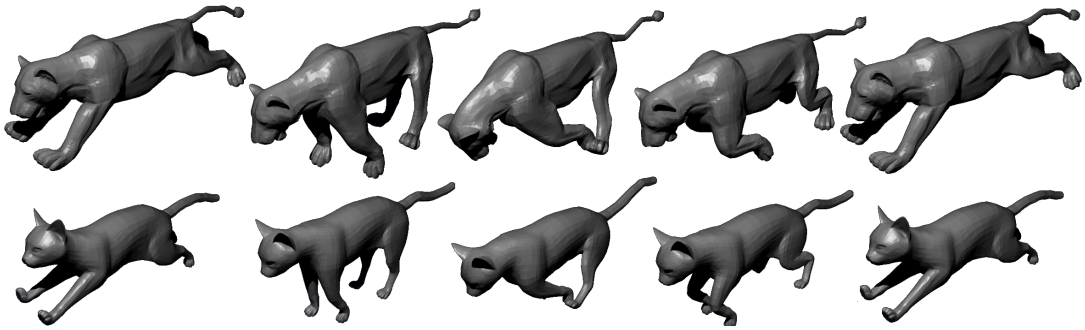


Figure 4.13: Motion transform from a lion model to a cat model.

values, eigenfunctions, and the geometry spectral domain share the invariance. Ideally, every point on a spatial surface should be embedded to the geometry spectral domain only by its geometric meaning. As long as the poses casted by one model are near isometric to each other, they will be re-embedded to a uniform surface in the infinite geometry spectrum. In practice, the shape spectrum is stable under the near isometric deformations. For example, the points on the elbow of the model will always be embedded around a common location in the spectral domain, no matter how the model's pose changes. The spectrum reflects the intrinsic characteristics of a surface despite varying Euclidean space embeddings.

The discrete setting makes it is possible and easy to apply the Laplace-Beltrami operator directly on the surfaces represented by triangle meshes. The continuous Laplacian equation turns into a symmetric generalized sparse matrix eigen problem. The eigenvalues are kept the same within a finite number, and eigenfunctions are represented with eigenvectors as area weighted samplings. This also makes the spectral domain invariant to different sampling rates and triangulations.

Our method analyzes data without preproccess like remeshing or registration. It first transforms spatial surfaces into geometry spectral domain. Each point is mapped along with its spatial geometry properties. The properties are smoothed with a low pass filter defined on the basis of eigenfunctions. In the spectral domain, each point carries a set of properties during the pose variations. It is efficient to classify points on the surface into rigid parts and articulated

parts by analyzing the geometric property changes on those points mapped in the geometry spectral domain. The eigenfunction can also provide rich geometric meaning, which leads to an automatic semantic skeleton with joints identified. The experimental results show that the filtered mean curvature range can predict different semantics of parts on the original surface. It may be very useful in motion analysis in computer vision and pattern recognition tasks as well.

CHAPTER 5

NON-ISOMETRIC MOTION ANALYSIS BY VARIATION OF SHAPE SPECTRUM

In previous chapters, we introduced shape spectrum is invariant to different triangulations and isometric deformations. By definition, shape spectrum represents the information of the intrinsic local geometry. A lot of existing approaches and experiments also show shape spectrum is stable with noises. Analysis on the variation of shape spectrum is less studied. In the chapter, we prove that the variation of shape spectrum can be analytically expressed and it can be used to analyze non-isometric deformations.

5.1 Related Work

The spectrum approach started on graphs [48, 49, 46]. Considering discrete meshes are also graphs, Laplacian matrix is defined on vertices and connections, weights may also applied. The eigenvalues are defined as the spectra of graphs, and the eigenfunctions are the orthogonal bases. This spectrum has a lot similarities with Fourier transform. The graphs are then projected onto those bases and analyzed in the spectral domain. Karni and Gotsman [31] used the projections of geometry on the eigenfunctions for mesh compression. and smoothing. Jain and Zhang [28] extended it for shape registration in the spectral domain. The Laplace spectrum focuses on the connection of graph, instead of the intrinsic geometry of the manifolds. Only using the connectivity of the graph may lead to highly distorted mappings [75].

The Laplace-Beltrami spectrum, also refers to shape spectrum in this work, was brought to computer graphics to describe shapes [61, 42, 59, 60]. There are a lot advantages of this spectral approach compared with transitional spacial ones. The spectrum depends on only the intrinsic geometry of a manifold. It is invariant to spacial translations, rotations, and scaling. It is also invariant to isometric deformations. On discrete domain, it is fine defined on digital

manifold, e.g., triangle meshes. The computing time is affordable. It is invariant to different triangulations. Due to those properties, A lot of shape analysis approaches are base on the Laplace-Beltrami spectrum, including our previous work.

Isometry is a fundamental condition of the shape spectrum. Non-isometry is less studied in this area. By definition, the spectrum is not invariant any more. Reuter et al. [58] discussed that the spectrum is stable to small amount of noise. High level noise on free form deformations will change the spectrum dramatically. Recent approaches showed the shape spectrum can be controlled with a scale function on the Riemann metric. It is the clue to have shape spectrum work on general different shapes. Shi et al. [64] discussed that the eigenvalues and eigenfunctions change according to the Riemann metric of the manifold. The derivate of the eigenvalues can be represented with that of the scale of the Riemann metric. In [64], eigenfunctions are registered across objects by calculating the Riemann metric scaling on both shapes. A dense registration is more focused there and the eigenvalue variation is no studied. Also the algorithm takes 15 minutes to register two shapes with around thousands vertices, which is not efficient.

In this chapter, we focus on a spectrum alignment for general shapes, and also a computationally affordable discrete algorithm, which can support non-isometric analysis.

5.2 Variation of the Eigenvalues and Eigenfunctions

In the real world motion cases, isometry is not usually preserved. The non-isometric deformations result in instability of eigenvalues and dramatic changes of eigenfunctions. In this section, we prove the eigenvalues are analytic function of motions.

On a compact closed manifold M with Riemann metric g, we define motion as a time variant positive scale function $\omega(t) : M \longmapsto R^+$ such that $g^{\omega}_{ij} = \omega g_{ij}$ and $d\sigma^{\omega} = \omega d\sigma$, where $\omega(t)$ is nonnegative and continuously differentiable. By definition, the weighted Laplace-Beltrami operator becomes

$$\Delta^{g^{\omega}} = \frac{1}{\omega}\Delta^{g}.$$

Consider the ith solution of the weighted eigen problem

$$\Delta^{g^\omega} f_i = -\lambda_i f_i, \tag{5.1}$$

or rewritten as

$$\Delta^g f_i = -\lambda_i \omega f_i, \tag{5.2}$$

where the eigenfunction f_i is normalized as

$$\int_M f_i^2 d\sigma^\omega = 1 \textbf{ for } i = 0, 1, 2, \ldots, \tag{5.3}$$

and orthogonal to other eigenfunctions

$$\int_M f_i f_j d\sigma^\omega = 0, j \neq i. \tag{5.4}$$

Theorem 1. *$\lambda_i(t)$ is piecewise analytic and, at any regular point, the t-derivative of $\lambda_i(t)$ is given by:*

$$\dot{\lambda_i} = -\lambda_i \int_M \dot{\omega} {f_i}^2 d\sigma \tag{5.5}$$

Proof. ω is a nonnegative and continuously differentiable function, and Δ^g is analytic. We can compute the derivative of the eigenvalue equation, equation 5.2, and get

$$\Delta^g \dot{f_i} = -\dot{\lambda_i} \omega f_i - \lambda_i \dot{\omega} f_i - \lambda_i \omega \dot{f_i}.$$

Then, we multiply both sides by f_i and take the integral on M and get

$$\int_M f_i \Delta^g \dot{f_i} d\sigma = -\dot{\lambda_i} \int_M \omega {f_i}^2 d\sigma - \lambda_i \int_M \dot{\omega} {f_i}^2 d\sigma - \int_M \dot{f_i} \lambda_i \omega f_i d\sigma,$$

which can be simplified with equation 5.2 and 5.3 as

$$\int_M f_i \Delta^g \dot{f}_i d\sigma = -\dot{\lambda}_i - \lambda_i \int_M \dot{\omega} {f_i}^2 d\sigma + \int_M \dot{f}_i \Delta^g f_i d\sigma,$$

Note that, M is a closed manifold. According to divergence theorem, we can have

$$\int_M f_i \Delta^g \dot{f}_i d\sigma = -\int_M \nabla \dot{f}_i \cdot \nabla f_i d\sigma = \int_M \dot{f}_i \Delta^g f_i d\sigma,$$

so we get equation 5.5. □

In discrete matrix form, we can get the similar result. Assume Ω is a nonnegative, continuously differentiable, diagonal matrix, and consider a weighted generalized eigenvalue problem

$$W\mathbf{v}_i = \lambda_i \Omega S \mathbf{v}_i, \tag{5.6}$$

where λ_i and $\mathbf{v}_i$ are ith corresponding solution. The eigenvectors can be normalized as

$$< \mathbf{v}_i, \mathbf{v}_i >_{\Omega S} = 1 \textbf{ for } i = 0, 1, 2, \ldots, \tag{5.7}$$

and orthogonal to each other

$$< \mathbf{v}_i, \mathbf{v}_j >_{\Omega S} = 0, i \neq j. \tag{5.8}$$

Theorem 2. *$\lambda_i(t)$ is piecewise analytic and, at any regular point, the t-derivative of $\lambda_i(t)$ is given by:*

$$\dot{\lambda}_i = -\lambda_i {\mathbf{v}_i}^T \dot{\Omega} S \mathbf{v}_i \tag{5.9}$$

Proof. We can compute the derivative of the eigenvalue equation, equation 5.6, and get

$$W\dot{\mathbf{v}}_i = \dot{\lambda}_i \Omega S \mathbf{v}_i + \lambda_i \dot{\Omega} S \mathbf{v}_i + \lambda_i \Omega S \dot{\mathbf{v}}_i,$$

multiply $\mathbf{v}_i^T$ from the left

$$\mathbf{v}_i^T W \dot{\mathbf{v}}_i = \dot{\lambda}_i \mathbf{v}_i^T \Omega S \mathbf{v}_i + \lambda_i \mathbf{v}_i^T \dot{\Omega} S \mathbf{v}_i + \mathbf{v}_i^T \lambda_i \Omega S \dot{\mathbf{v}}_i,$$

and simplify it with equation 5.6 and 5.7 as

$$\mathbf{v}_i^T W \dot{\mathbf{v}}_i = \dot{\lambda}_i + \lambda_i {\mathbf{v}_i}^T \dot{\Omega} S \mathbf{v}_i + {\mathbf{v}_i}^T W^T \dot{\mathbf{v}}_i.$$

Then we get equation 5.9 as W is symmetric. □

5.3 Algorithm

In previous chapters and sections, we already discussed the properties of shape spectrum and proved that the eigenvalues can be controlled with a time variant continuous function. By definition, the shape spectrum is invariant to isometric deformations. However, in real cases, isometry is not guaranteed. For example, in pose deformations, the surfaces at the joints are locally scaled, and in heart motions, the surface will contract and expand globally. These deformations break isometry. The former experiments showed that the shape spectrum is stable to these non-isometric deformations and noises. Our eigenvalue variation theorems prove that the spectrum is smooth and analytic to a non-isometric local scale deformation. They support, analytically, aligning the shape spectrum among non-isometric deformations, hence facilitating a registration-free solution for motion analysis.

In this section, we focus on the discrete algorithm to align the shape spectrum among non-isometric deformations. Consider two closed manifolds, M and N, represented with discrete triangle meshes. Their first k nonzero eigenvalues and eigenvectors are

$$\lambda_{Mi}, \mathbf{v}_{Mi}, \lambda_{Ni}, \text{ and } \mathbf{v}_{Ni}, \text{ for } i = 1, 2, \ldots, k.$$

Due to the non-isometry, the first k eigenvalues are not necessary to be aligned. In order to

align the first k eigenvalues of N to those of M, a continuous scale diagonal matrix $\Omega(t)$ is applied on N. Ω is an n by n matrix, where n is number of vertices on N. The element Ω_{ii} on the diagonal is a scale factor defined on each vertex on N. According to Theorem 2, the derivative of each eigenvalue is expressed by those of Ω_{ii} analytically. Thus, the scale matrix Ω will introduce an alignment from N to M on eigenvalues. The rest of this section will describe the details to obtain the diagonal matrix Ω numerically.

5.3.1 Linear Interpolation

Assume that the eigenvalues of N vary linearly towards those of M. This linear interpolation is represented as

$$\lambda_i(t) = (1-t)\lambda_{Ni} + t\lambda_{Mi}, t \in [0,1]. \tag{5.10}$$

At the beginning, $t = 0$, and $\lambda_i(0)$ starts as λ_{Ni}, while t reaches 1, $\lambda_i(1)$ aligned to λ_{Mi}. At any regular time $t \in [0,1]$, the derivative is calculated on both sides as

$$\dot{\lambda_i}(t) = \lambda_{Mi} - \lambda_{Ni}, t \in [0,1]. \tag{5.11}$$

The derivative of $\lambda_i(t)$ is a constant all the time, and it can be expressed by the derivative of the scale matrix Ω.

5.3.2 Matrix Eigenvalue Variation

Each diagonal element Ω_{ii} represents a scale factor at vertex i on manifold N. $\Omega(0)$ is a identity matrix on N, and $\Omega(1)$ aligns the first k nonzero eigenvalues of N to those of M. Combining equation 5.9 and 5.11, the derivative of each $\lambda_i(t)$ leads to a equation of Ω as

$$-\lambda_i(t)\mathbf{v}_i(t)^T\dot{\Omega}S\mathbf{v}_i(t) = \lambda_{Mi} - \lambda_{Ni}, t \in [0,1], \tag{5.12}$$

where S is also diagonal Voronoi area matrices and $\mathbf{v}_i(t)$ is the corresponding eigenvector as described in previous chapters. The diagonal elements of S are defined as Voronoi area of

vertices. Although we have the equation of the time derivative of Ω, it is hidden in the discrete integration and not straight forward to solve. We have to reform the individual integration equation into a linear system. If we extract the diagonals as vectors $\mathbf{v}_\Omega$ and $\mathbf{v}_S$, and employ Hadamard production, which is an element wise matrix production as

$$A \circ B = C \text{ such that } A_{ij} \cdot B_{ij} = C_{ij}, \tag{5.13}$$

the equation 5.11 can be rewritten with a linear form as

$$(\mathbf{v}_S \circ \mathbf{v}_i \circ \mathbf{v}_i)^T \cdot \mathbf{v}_{\dot{\Omega}} = \frac{\lambda_{Ni} - \lambda_{Mi}}{\lambda_i(t)}, t \in [0, 1]. \tag{5.14}$$

Note that, as the first k eigenvalues are to be aligned, we got k independent equations, which leads to an under-determined linear system as

$$A \cdot \mathbf{v}_{\dot{\Omega}} = \mathbf{b} \tag{5.15}$$

where A is a row stack of $(\mathbf{v}_S \circ \mathbf{v}_i \circ \mathbf{v}_i)^T$ with k rows

$$A_{k \times n} = \begin{pmatrix} (\mathbf{v}_S \circ \mathbf{v}_1 \circ \mathbf{v}_1)^T \\ (\mathbf{v}_S \circ \mathbf{v}_2 \circ \mathbf{v}_2)^T \\ \vdots \\ (\mathbf{v}_S \circ \mathbf{v}_k \circ \mathbf{v}_k)^T \end{pmatrix}$$

and $\mathbf{b}$ is a k dimensional vector with

$$b_i = \frac{\lambda_{Ni} - \lambda_{Mi}}{\lambda_i(t)}, t \in [0, 1].$$

Note that, practically, k is much less than n. For example, on a triangle mesh with 20000 vertices, and only first 50 eigenvalues are aligned. This means the linear the system is unde-

termined and has no unique solution. More constraints are necessary to provide an optimized solution for the linear system.

5.3.3 Smoothness Constraints

In our case, we focus on global smoothness of the scale factors distributed on N. Consider a scalar function $f \in C^2$ is define on the continuous manifold $< N_c, g >$. The gradient of f, ∇f, describes the local change of f. For example, if f is a constant function, which is considered as the smoothest distribution, the gradient ∇f is zero everywhere. A smoothness energy of f is defined with the total square magnitude of the gradient ∇f on N_c

$$E = \int_{N_c} \|\nabla f\|^2 d\sigma. \tag{5.16}$$

Note that ∇f is a vector, and the square magnitude is calculated as a dot product

$$\|\nabla f\|^2 = \nabla f \cdot \nabla f. \tag{5.17}$$

Then the integral on N_c becomes

$$E = -\int_{N_c} f \Delta^g f d\sigma. \tag{5.18}$$

At time t, we investigate the scale function $\omega(t)$ and $d\omega|_t$. Then we obtain the following smoothness energy

$$\begin{aligned} E &= -\int_{N_c} (\omega + d\omega)\Delta^g(\omega + d\omega)d\sigma \\ &= -\int_{N_c} d\omega\Delta^g d\omega d\sigma - 2\int_{N_c} \omega\Delta^g d\omega d\sigma - \int_{N_c} \omega\Delta^g\omega d\sigma. \end{aligned} \tag{5.19}$$

On the discrete triangle mesh N, the scale function is a vector $\mathbf{v}_\Omega$, which is the diagonal of matrix Ω. The integral is a matrix product as

$$\begin{aligned} E &=< \mathbf{v}_\Omega + \mathbf{v}_{\dot{\Omega}}, L \cdot (\mathbf{v}_\Omega + \mathbf{v}_{\dot{\Omega}}) >_S \\ &= (\mathbf{v}_\Omega + \mathbf{v}_{\dot{\Omega}})^T \cdot S \cdot L \cdot (\mathbf{v}_\Omega + \mathbf{v}_{\dot{\Omega}}) \\ &= (\mathbf{v}_\Omega + \mathbf{v}_{\dot{\Omega}})^T \cdot W \cdot (\mathbf{v}_\Omega + \mathbf{v}_{\dot{\Omega}}) \\ &= \mathbf{v}_{\dot{\Omega}}^T \cdot W \cdot \mathbf{v}_{\dot{\Omega}} + 2\mathbf{v}_\Omega^T \cdot W \cdot \mathbf{v}_{\dot{\Omega}} + \mathbf{v}_\Omega^T \cdot W \cdot \mathbf{v}_\Omega. \end{aligned} \tag{5.20}$$

Assume that $\mathbf{v}_\Omega$ is known at each time t and $\mathbf{v}_{\dot{\Omega}}$ is to be solved in equation 5.15. $\mathbf{v}_\Omega$ is constant to $\mathbf{v}_{\dot{\Omega}}$. $\mathbf{v}_{\dot{\Omega}}$ is going to minimize the quadratic smooth energy E_q at any time

$$E_q = \mathbf{v}_{\dot{\Omega}}^T \cdot W \cdot \mathbf{v}_{\dot{\Omega}} + 2\mathbf{c}^T \cdot \mathbf{v}_{\dot{\Omega}}, \tag{5.21}$$

where $\mathbf{c} = W \cdot \mathbf{v}_\Omega$. In order to preserve the physical availability, $\mathbf{v}_\Omega$ must be bounded. The scale factor cannot be zero or negative. Further more, any point cannot be infinity either. We denote a lower bound and an upper bound with $\mathbf{h}_l, \mathbf{h}_u > 0$, where $\mathbf{h}_l$ and $\mathbf{h}_u$ are n dimensional constant vector. $\mathbf{v}_{\dot{\Omega}}$ must satisfy

$$\mathbf{h}_l \leq \mathbf{v}_\Omega + \mathbf{v}_{\dot{\Omega}} \leq \mathbf{h}_u. \tag{5.22}$$

This inequality bound can be written into a matrix form

$$G \cdot \mathbf{v}_{\dot{\Omega}} \leq \mathbf{h}, \tag{5.23}$$

where G is stack of identity matrices as

$$G_{2n\times n} = \begin{pmatrix} -I_{n\times n} \\ I_{n\times n} \end{pmatrix}, \tag{5.24}$$

and $\mathbf{h}$ is a $2n$ dimensional vector as

$$\mathbf{h}_{2n\times 1} = \begin{pmatrix} \mathbf{v}_\Omega - \mathbf{h}_l \\ \mathbf{h}_u - \mathbf{v}_\Omega \end{pmatrix}, \tag{5.25}$$

The linear system, equation 5.15, smoothness constraint, equation 5.20, and constant bound, equation 5.23, introduce a quadratic programming problem at each time t. Assume the eigenvalues and eigenvectors are known at each time t, the derivative of the scale matrix $\dot{\Omega}$ is the solution of such quadratic programming.

5.3.4 Linear Integration

The discussions above prove that at each time t, the derivative of the scale matrix $\dot{\Omega}$ is the solution of a quadratic programming. As an initial state, Ω is an identity matrix as it starts from N itself. The final scale matrix is achieved by an integral

$$\Omega(1) = I + \int_0^1 \dot{\Omega} dt, \tag{5.26}$$

which aligns the first k nonzero eigenvalues from N to M.

This integration is discretely approximated with an iteration. The time interval $[0, 1]$ is divided into K steps. The index of each steps is j. Initially, $j = 0$, $\Omega(0) = I$, $\lambda_i(0) = \lambda_{Ni}$, and $\mathbf{v}_i(0) = \mathbf{v}_{Ni}$. In order to reduce the numerical error, we reinitialize the problem at the beginning of each step $j = 0, 1, \cdots, K$. In stead of aligning λ_{Ni} to λ_{Mi}, we are aligning $\lambda_i(j)$. $\lambda_i(j)$ and $\mathbf{v}_i(j)$ are re-calculated with equation 5.6 and current $\Omega(j)$ on N. The diagonal of $\Omega(j)$ is vector $\mathbf{v}_\Omega(j)$ while the diagonal of S is $\mathbf{v}_S$. And then, the current quadratic programming

problem is constructed with equation 5.15, 5.20, and 5.23as

$$\begin{aligned}
\underset{\mathbf{v}_{\dot{\Omega}}(j)}{\text{minimize}} \quad & E_q = \mathbf{v}_{\dot{\Omega}}(j)^T \cdot W \cdot \mathbf{v}_{\dot{\Omega}}(j) + 2\mathbf{c}^T \cdot \mathbf{v}_{\dot{\Omega}}(j) \\
\text{subject to} \quad & G \cdot \mathbf{v}_{\dot{\Omega}}(j) \leq \mathbf{h} \text{ (inequality constraint)} \\
& A \cdot \mathbf{v}_{\dot{\Omega}}(j) = \mathbf{b} \text{ (equality constraint)}
\end{aligned}$$

where

$$\begin{aligned}
\mathbf{c} &= W \cdot \mathbf{v}_{\Omega}(j) \\
G_{2n\times n} &= \begin{pmatrix} -I_{n\times n} \\ I_{n\times n} \end{pmatrix} \\
\mathbf{h}_{2n\times 1} &= \begin{pmatrix} \mathbf{v}_{\Omega}(j) - \mathbf{h}_l \\ \mathbf{h}_u - \mathbf{v}_{\Omega}(j) \end{pmatrix} \\
A_{k\times n} &= \begin{pmatrix} (\mathbf{v}_S \circ \mathbf{v}_1(j) \circ \mathbf{v}_1(j))^T \\ (\mathbf{v}_S \circ \mathbf{v}_2(j) \circ \mathbf{v}_2(j))^T \\ \vdots \\ (\mathbf{v}_S \circ \mathbf{v}_k(j) \circ \mathbf{v}_k(j))^T \end{pmatrix} \\
\mathbf{b}_{k\times 1} &= \begin{pmatrix} \frac{\lambda_1(j)-\lambda_{M1}}{\lambda_1(j)} \\ \frac{\lambda_2(j)-\lambda_{M2}}{\lambda_2(j)} \\ \vdots \\ \frac{\lambda_k(j)-\lambda_{Mk}}{\lambda_k(j)} \end{pmatrix}.
\end{aligned} \tag{5.27}$$

Note that, as we reinitialize the problem, λ_{Ni} in equation 5.15 is replaced with λ_k at the beginning of each step. $\dot{\Omega}(j)$ is the solution of the quadratic programming above. Then $\Omega(j+1)$ is approximated with

$$\Omega(j+1) = \Omega(j) + \frac{1}{K-j}\dot{\Omega}(j). \tag{5.28}$$

After K steps, the desired $\Omega(K)$ is achieved.

5.3.5 Algorithm Summary

The algorithm is summarized as follows:

Algorithm 1 Eigenvalue Alignment

Input: Closed 2D manifolds N and M, represented triangle meshes and constant k

Output: Diagonal weight matrix $\Omega(j)$ on N, aligning first k nonzero eigenvalues from N to M

Initialize $\Omega(0) \leftarrow I$, calculate matrices W and S on N, and $\lambda_{Mi}, \mathbf{v}_{Mi}, \lambda_{Ni},$ and $\mathbf{v}_{Ni},$ for $i = 1, 2, \ldots, k$

while $j < K$ **do**

Calculate $\lambda_i(j), \mathbf{v}_i(j),$ for $i = 1, 2, \ldots, k$ using equation 5.6 with $\Omega(j)$

Construct the quadratic programming problem (5.27)

Solve the quadratic programming problem to get $\dot{\Omega}(j)$

$\Omega(j+1) \leftarrow \Omega(j) + \frac{1}{K-j}\dot{\Omega}(j)$

$j \leftarrow j + 1$

end while

5.4 Results

Our algorithm is implemented with Python and C++ on a 64-bit Linux platform. The Python libraries, Numpy, Scipy and cvxopt, are employed for algebra calculations, OpenGL, VTK, and Blender 3D for rendering and visualization. The experiments are conducted on an Intel Celeron 2955U 1.4GHz laptop with 4GB RAM. We apply our algorithm on 2D manifold, represented with triangle meshes.

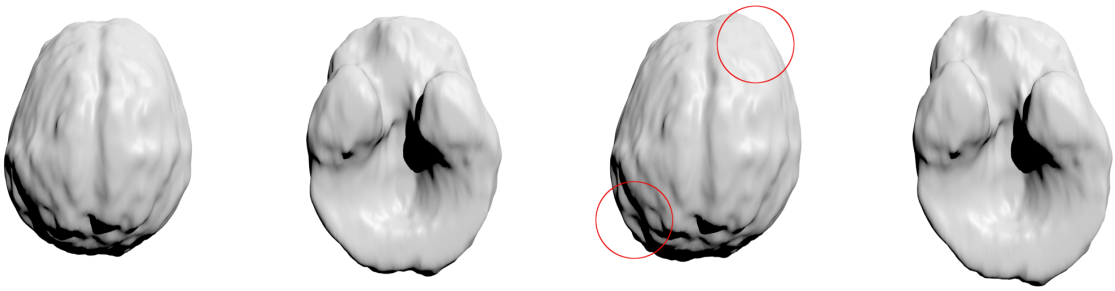
There are typically two kinds of data in our experiments, brains surfaces and heart left ventricle (LV) motion sequences. They are all extracted from 3D medical images. Each brain and LV surface contains 20000 and 10000 vertices respectively. We first evaluate the computation performance of our algorithm. Besides the vertex number, there are two constants, K iterations and the first k nonzero eigenvalues to be align. According to the algorithm described in previous sections, each iteration is an independent quadratic programming problem. Thus the complexity is linear to the step number K. k determines how many eigenvalues to be re-initialized at the beginning of each step. The Scipy libraries we employed calculates the eigenvalues by iterations. The complexity is $O(n^2)$ to the number of vertices and linear to k.

The average computing time is shown in Table 5.1, which matches the analysis above. Note that, the larger the K is, the more accurate the approximation is, in terms of the linear interpolation. In practical, we found $K = 5$ is sufficient to get the accurate result and save computing time. Ideally, including more eigenvalues for alignment can be more accurate. However, the numeric eigenvalue calculation is not reliable on higher indexed eigenvalues, which will bring more unsuitability. We usually choose $k = 50$ in our experiments without further notations. The computation is quite affordable on a low profile laptop.

Manifold	$k = 30, K = 5$	$k = 50, K = 5$	$k = 30, K = 10$	$k = 50, K = 10$
Brain (20000)	21.2s	30.6s	42.6s	59.1s
LV (10000)	8.91s	14.0s	18.0s	27.4s

Table 5.1: Performance evaluation.

In order to evaluate our eigenvalues variation algorithm, we synthetically generate some non-isometric deformations. In this case, the dense vertex to vertex correspondence is known. A reference is chosen as a brain surface, which is extracted from MRI scans. The surface is then deformed manually with local controls, which is non-isometric. The deformation and local controls are shown in Figure 5.1. Due to the non-isometry, the spectrum is expected to



(a) (b) (c) (d)

Figure 5.1: Synthetic deformation. (a) and (b) are original triangle mesh, which is generate from 3D medical image. (c) and (d) are obtained by manually editing the original surface. The synthetic deformation is local and non-linear. The locally deformed area are marked with red circle in (c).

vary, both eigenvalues and eigenfunctions. Table 5.2 shows the eigenvalue variations under the

non-isometric deformation. The eigenvalues in the table are normalized by the first nonzero

Manifold	λ 2-6	λ 51-55
Synthetic Deformation	1.31, 1.36, 2.98, 3.38, 3.69	30.29, 30.69, 31.57, 32.59, 32.94
Brain	1.23, 1.29, 2.93, 3.29, 3.61	29.65, 29.84, 30.96, 31.37, 31.71
Aligned	1.31, 1.36, 2.98, 3.38, 3.69	30.25, 30.66, 31.61, 32.62, 32.87

Table 5.2: Eigenvalues alignment on synthetic deformation.

one to remove the scale factor. One can compare how the non-isometry breaks eigenvalue invariabilities from the original brain shape to the synthetic deformation in Table 5.2. Not only eigenvalues, eigenfunctions vary even more dramatically, which is illustrated in Figure 5.2. We randomly pick the 12th, 14th, and 16th eigenfunction distributions for comparison, where eigenfunction shifting is more noticeable in the middle range. The eigenfunctions are normalized between -1 and 1. Their values are expressed with color maps, where red means larger value, blue means smaller ones, and green means zero. The patterns of the eigenfunctions shift around. They may not represent the corresponding geometry across non-isometric manifolds. In the worst case, the topology of the eigenfunction distribution may also change. With the synthetic deformation, our spectrum alignment is applied on the first 50 nonzero eigenvalues and a scale function is obtained. We compare the eigenvalues, on the original brain and after alignment, in Table 5.2. After applying the spectrum alignment algorithm, the eigenvalue are aligned perfectly. Furthermore, the higher indexed eigenvalues are also aligned even they are not in the linear system constraint. We define a relative error of eigenvalues as

$$er = \frac{|\lambda_i^{\text{reference}} - \lambda_i|}{\lambda_i^{\text{reference}}}.$$

Table 5.3 records the accuracy of the spectrum alignment algorithm in percentage on the eigenvalues. Both ranges, within the first 50 constraint and first 100, are measured. The alignment reduces the eigenvalue error by two orders of magnitude. Figure 5.3 demonstrates the scale function distributions on the original surface. The color represents the log of the scales. Red

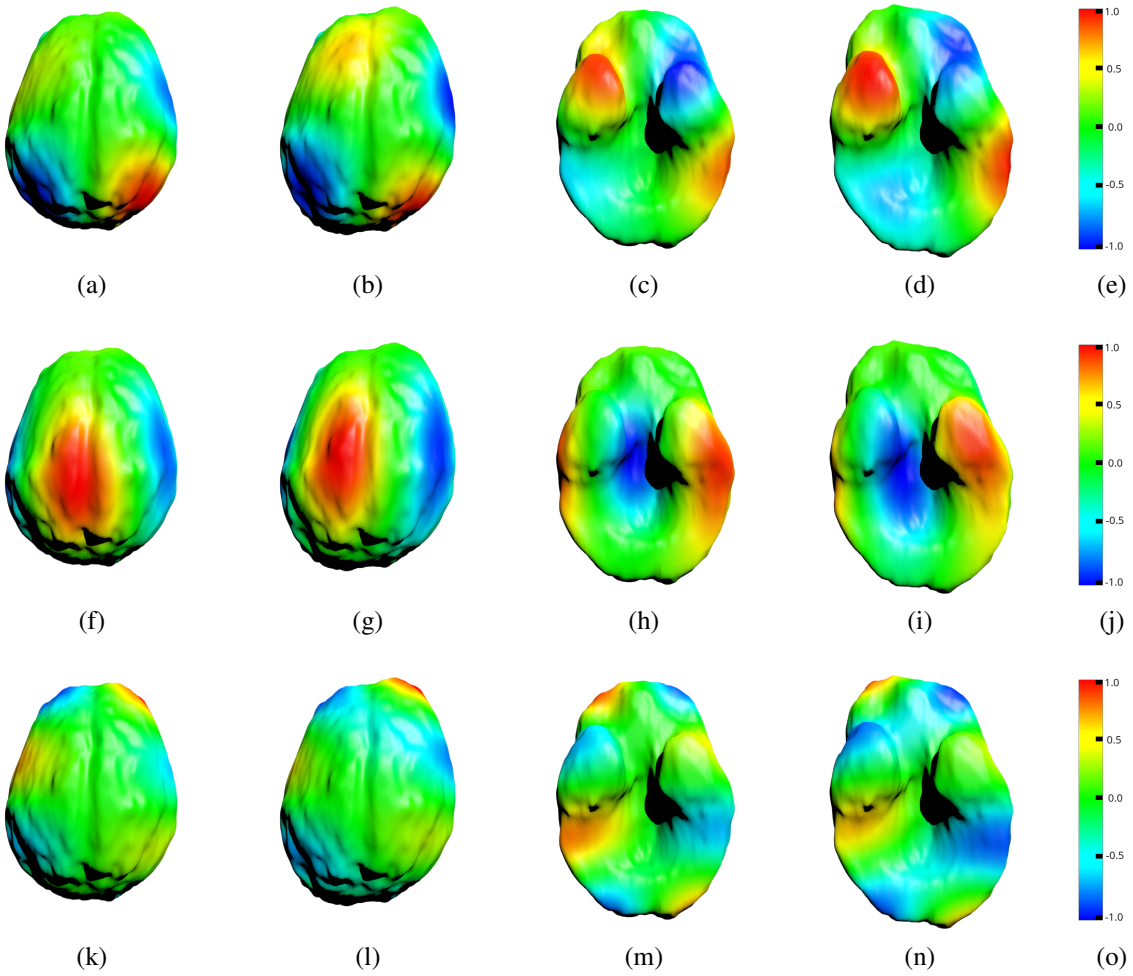
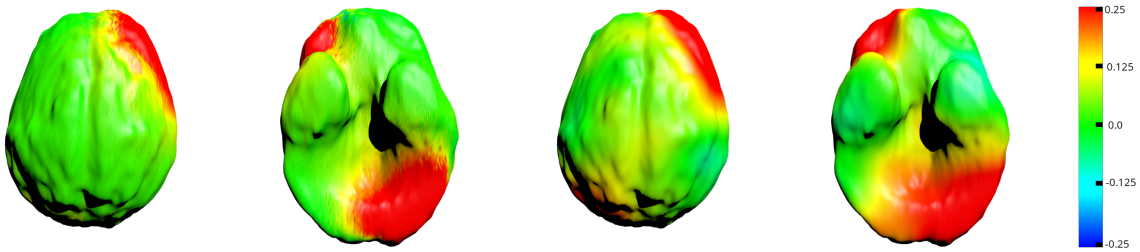


Figure 5.2: Synthetic spectrum shifting. The shape spectrum is invariant to isometric deformations. The non-isometric ones breaks the invariability. We randomly pick the 12th, 14th, and 16th eigenfunctions to show the shifting, represented by each row respectively. The rows of (a) and (c) are the original shape, while (b) and (d) are the synthetic deformation. Even small non-isometric deformation introduces noticeable eigenfunction shifting.

	Synthetic		Brain		LV	
	first 50	first 100	first 50	first 100	first 50	first 100
before	8.93%	9.00%	6.55%	6.61%	25.9%	24.5%
after	0.0186%	0.0187%	0.202%	0.203%	0.290%	0.286%

Table 5.3: Average normalized eigenvalue errors before and after alignment.

means dilating, blue contraction, and green no distortion. (a) and (b) are ground truth by calculating the vertex to vertex Voronoi area distortions on the synthetic surface. (c) and (d) are the result of the spectrum alignment. It is clear that the spectrum alignment predicts the local de-



(a) (b) (c) (d) (e)

Figure 5.3: Synthetic deformation log ratio. The scale function from the eigenvalue alignment algorithm is evaluated with the synthetic ground truth. With the synthetic deformation, the dense vertex to vertex correspondence is known. The vertex-wise scale function is calculated with the Voronoi area ratio after and before the deformation, demonstrated in (a) and (b). In order to make it linear to compare, log operation is employed. (c) and (d) represent the scale function from the eigenvalue alignment. Our algorithm accurately recovers the local deformation without any pre-information but shape spectra.

formation precisely, with spectrum information only. Please note that, the result scale function is much smoother than the one of ground truth and they have slight difference, because we are using smoothness constraints to solve the linear system. The next step is to investigate how the eigenvalue distributions change under the spectrum alignment. We pick the same eigenfunctions, 12th, 14th, and 16th, after alignment for comparison in Figure 5.4. Compared with the unaligned eigenfunctions in Figure 5.1, those ones after the spectrum alignment are more responsible. Especially, the 16th eigenfunction on the top has not only shifting but also topology change on the top. The blue part is a connected strip shape on the deformed shape, but separate on the original brain, which is also corrected with the spectrum alignment in Figure 5.4. The synthetic deformation is aligned in shape spectrum.

We move next step to real clinical data. Brain surface registration is a fundamental research in image analysis and vision. The brain data in our experiment is extracted from MRI scans with marching cube and re-sampled with 20000 vertices for each shape. The shapes from

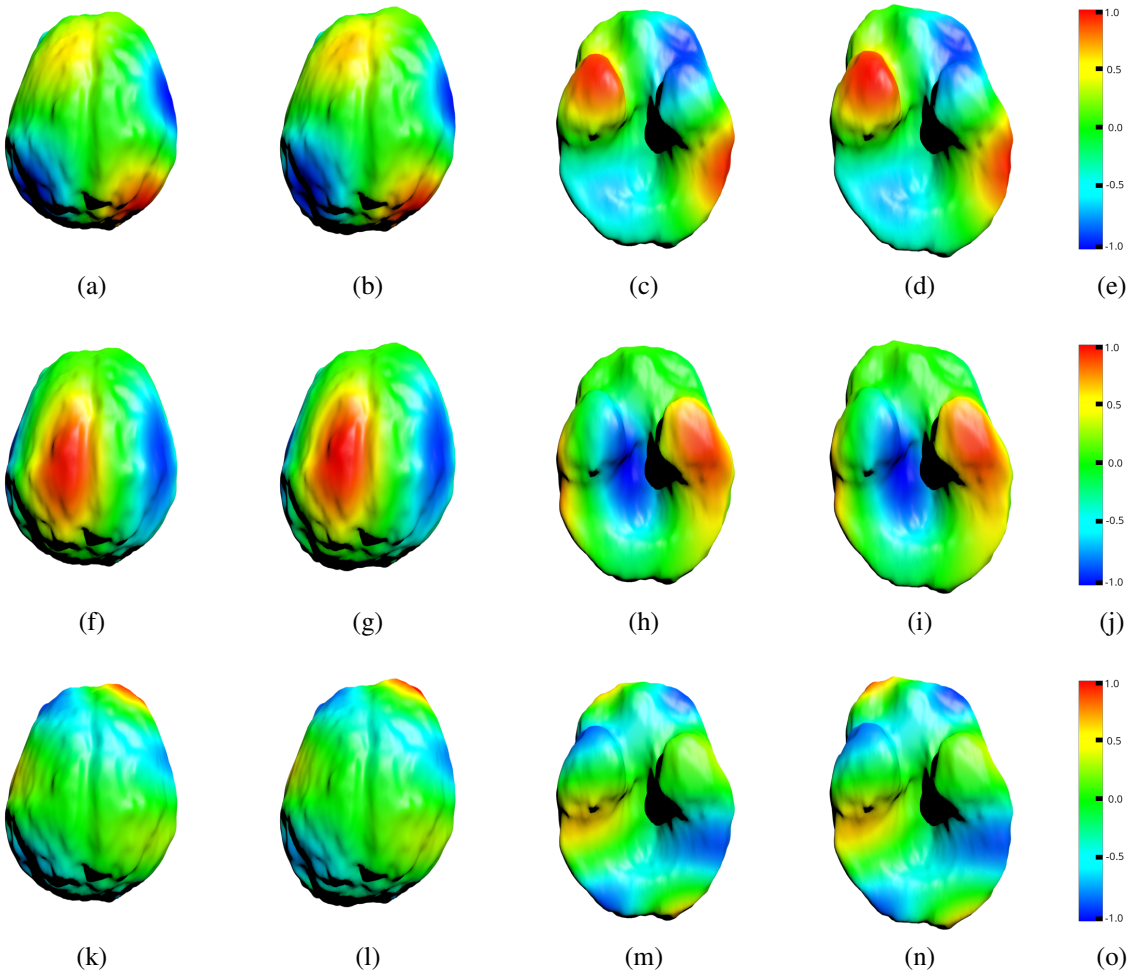


Figure 5.4: Synthetic spectrum alignment. The eigenvalues are aligned form the original shape to the synthetic deformation. The eigenfunctions are aligned as well. The 12th, 14th, and 16th eigenfunctions are represented by each row respectively. Those eigenfunctions are more consistent after eigenvalue alignment over deformations.

different persons are usually non-isometric to each other. The original shape spectrum fails in this case. In Table 5.3, the average eigenvalue error is 6.55% among different brain surfaces. Two brains samples are demonstrated side by side in Figure 5.5. It is obvious that they are not isometric even up to a global scale factor. Figure 5.6 shows how the eigenfunction distributions change among different brains. Here, the 12th, 14th, and 16th eigenfunctions are chosen. The patterns, e.g., maxima and minima, are shifting around, and some of them are even missing.

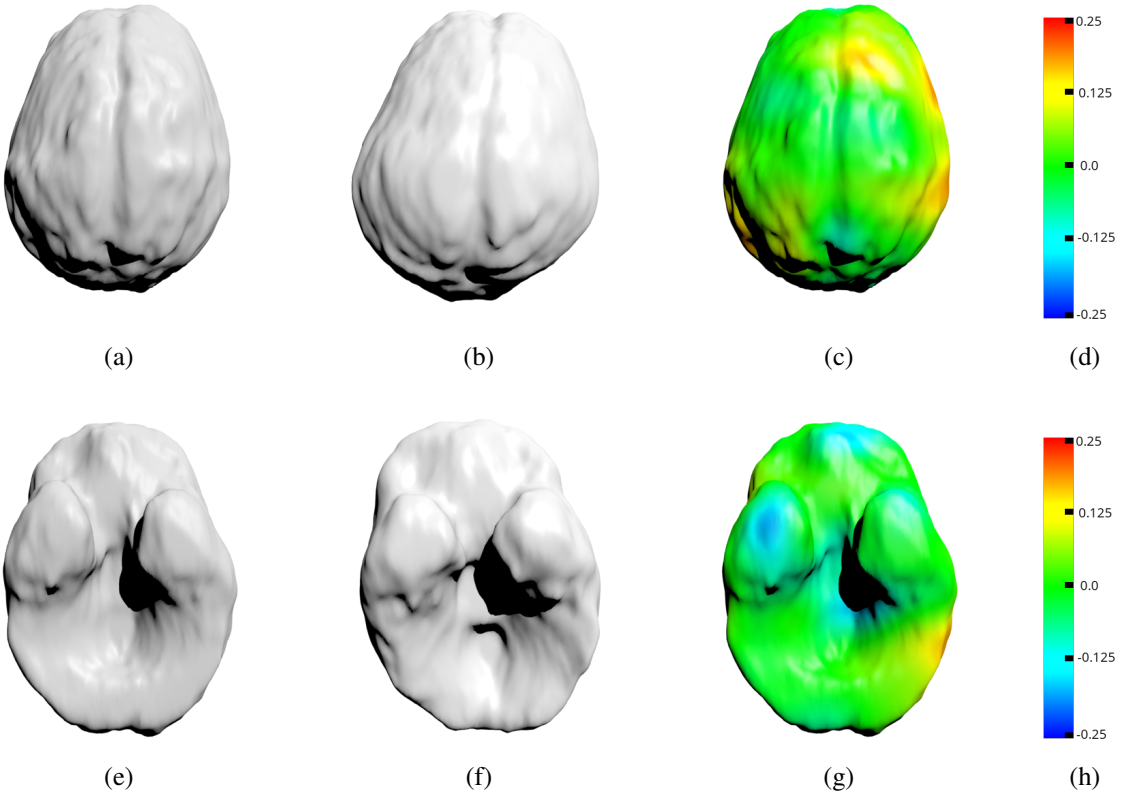


Figure 5.5: Different shapes are aligned with a scale function. The column of (a) is the reference shape, (b) the target one to be aligned, and (c) the scale function distribution on the reference shape. The color the represents the log values of the scale factors.

The spectra of these two brains can be aligned, with a scale function. Such scale function is illustrated in Figure 5.5, whose log values are mapped with color. Intuitively, the red and yellow areas expand themselves, while the blue ones contracts. Then the geometry on the left brain deformed to the right one. Table 5.3 shows the spectrum alignment reduces the eigenvalue

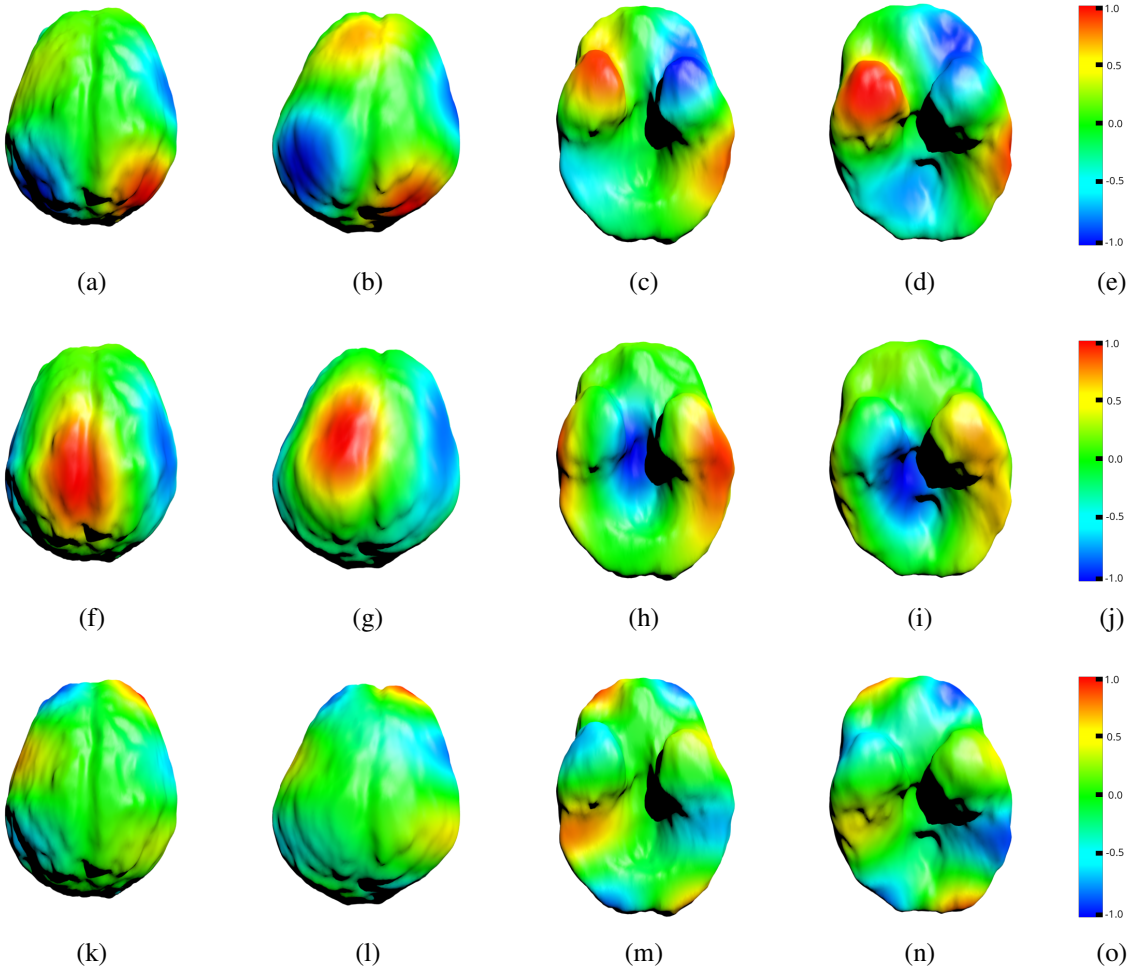


Figure 5.6: Brain spectrum shifting. The two brains shapes are from different persons, which are not isometric to each other. The rows show the 12th, 14th, and 16th eigenfunction distributions respectively on the two brains. The columns of (a) and (c) represent one brain and (b) and (d) the other. The eigenfunctions shift due to the non-isometry.

error from 6.55% to 0.202%. The eigenfunctions are also aligned as they are geometrically more similar than the original case, shown in Figure 5.7. Comparing Figure 5.7 and 5.6, the great improvement is obtained after spectrum alignment, in terms of those maxima, minima, and transition edges on the geometry. Another interesting case to study is LV motion in heart.

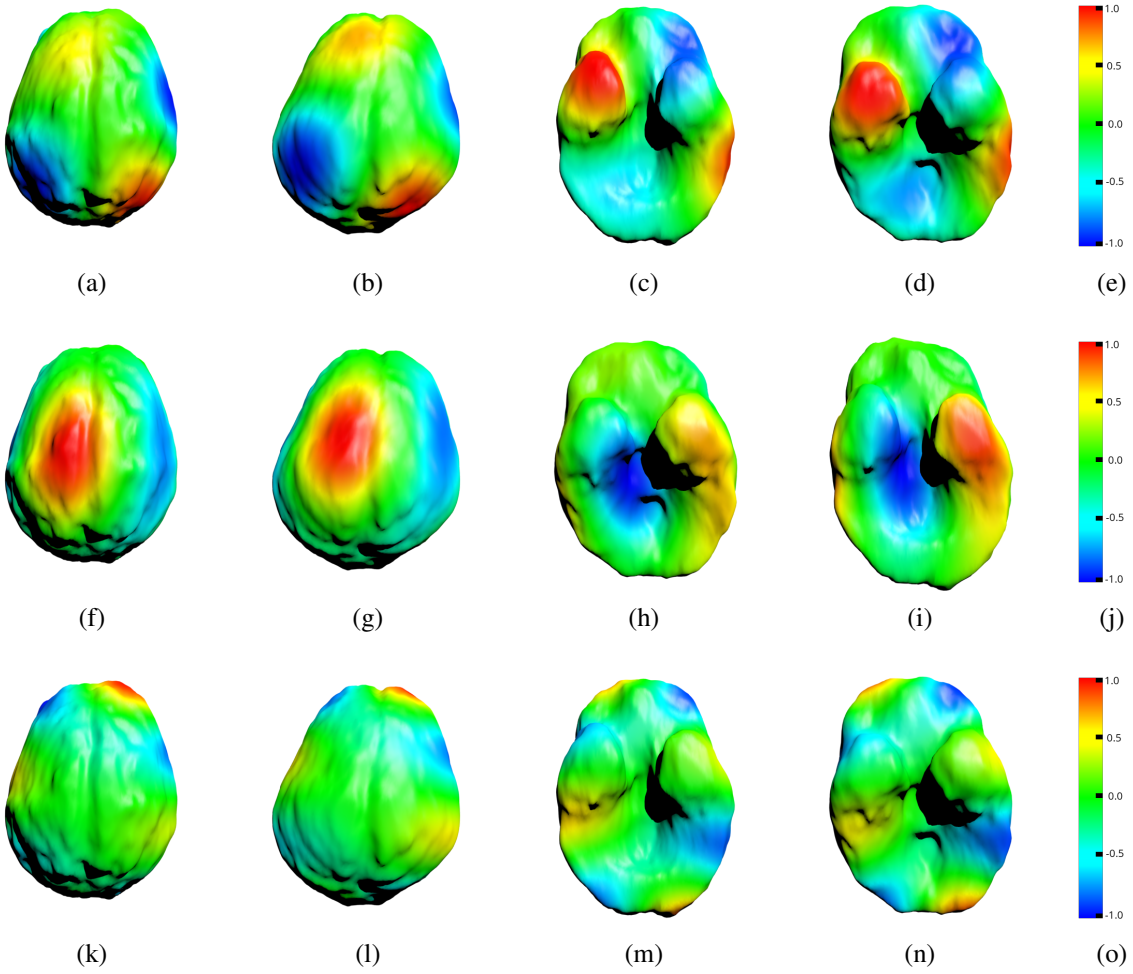


Figure 5.7: Brain spectrum alignment. The spectra of the two brains are aligned with a scale function. The rows show the 12th, 14th, and 16th eigenfunction distributions respectively on the two brains. The columns of (a) and (c) represent one brain and (b) and (d) the other. The eigenfunctions are aligned as well.

LV changes its shape within cycles by contracting and dilating the muscles. This motion results in changes of the surface geometry. Figure 5.8 shows a sequence of 8 samples during a LV motion. It can been seen the geometry changes dramatically. The middle sample, first one in

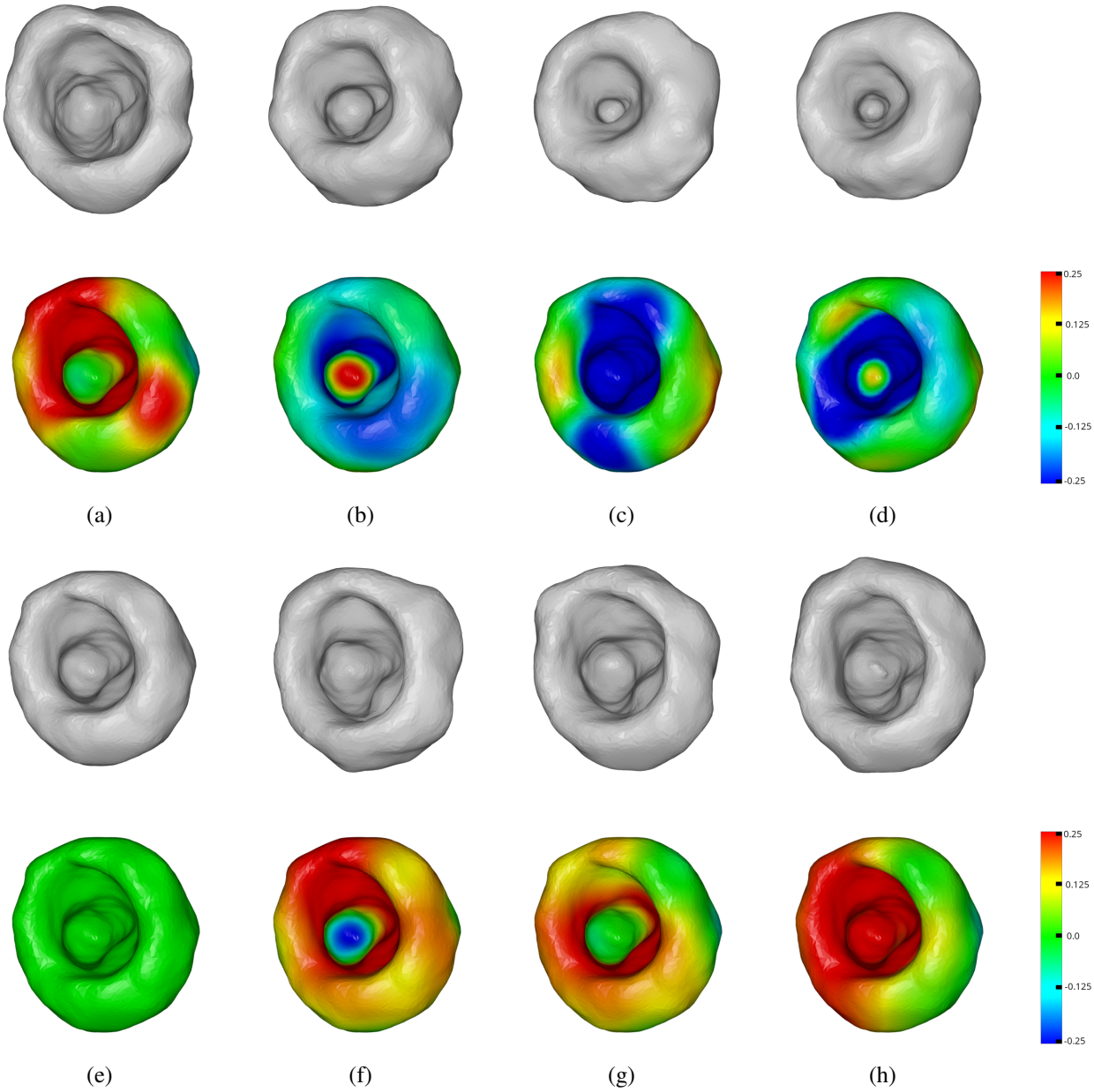


Figure 5.8: LV motion represented with scale functions. Although the LV motion introduces a sequence of non-isometric deformations, the spectra of those deformations can still be aligned with scale functions. Each deformation is then represented with a scale function on the reference frame.

row 3, is chosen as a reference shape, and its spectrum is aligned to all the other shapes. The eigenvalue errors shown in Table 5.3 verify the large geometry changes as the average errors is 25.9% and the spectrum alignment brings it down to 0.290%. Figure 5.9 and 5.10 demonstrate the eigenfunction distributions before and after the spectrum alignment. The results are similar to the previous synthetic shape and brain surfaces. While the eigenvalues are aligned among the shapes, the eigenfunctions are also aligned according to the geometry. The spectrum alignment result a scale function to each time frame, illustrated in Figure 5.8. Red color indicates dilation, blue contraction, and green no scaling. For example, from the reference shape to the first frame, the interior has to expand, and to the third frame, it is necessary to contract instead. In this way, the spacial geometry deformations turn into scale function distributions. Note that the scale functions are globally smooth and predict local deformations. Another potential application would be abnormality visualizations and diagnosis. For example, Cardiomyopathy is the main cardiac disease which affects the wall thickness and its functionality. This disease can be detected from the abnormal motion of LV [69]. Usually, the variations are weaker on the myopathy parts than those on the normal parts. In this work, the scale function illustrates the contractions and expansions of local part. For rendering, we cut interior wall of the LV, shown in Figure 5.11. It is in the during a contraction phase. Most of the surface is contracting, which is indicated by blue color. There is a abnormal patch receives less or no deformations.

5.5 Summary

In this chapter, we have introduced spectrum variation theorems for general shapes. A shape is represented with a closed 2D manifold with Riemann metric. The Laplace-Beltrami spectrum is defined on the intrinsic geometry of the manifold, which is invariant to rigid operations and isometric deformations. In real cases, isometry is hard to preserve. Even small non-isometric deformation will cause spectrum variations. We prove that the eigenvalues of the spectrum is an analytic function of a scale function applied on the Riemann metric. The derivative of each eigenvalue is an integral of the derivative of the scale function. The theorem

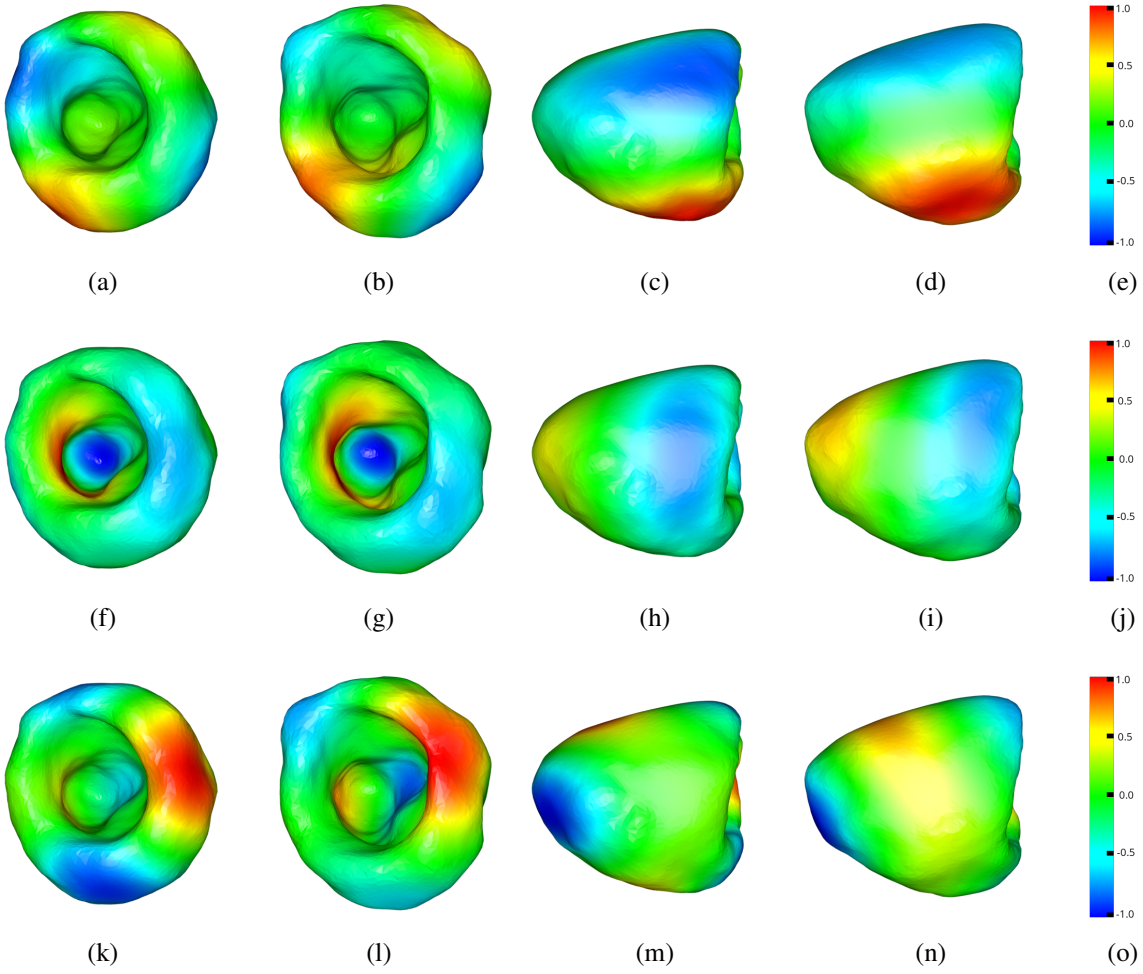


Figure 5.9: LV spectrum shifting during motion. The local parts of a LV contract and expand. Those deformations are usually not isometric. The rows show the 8th, 12th, and 14th eigenfunction shifting respectively. The columns of (a) and (c) represent one time frame in the LV motion and (b) and (d) another.

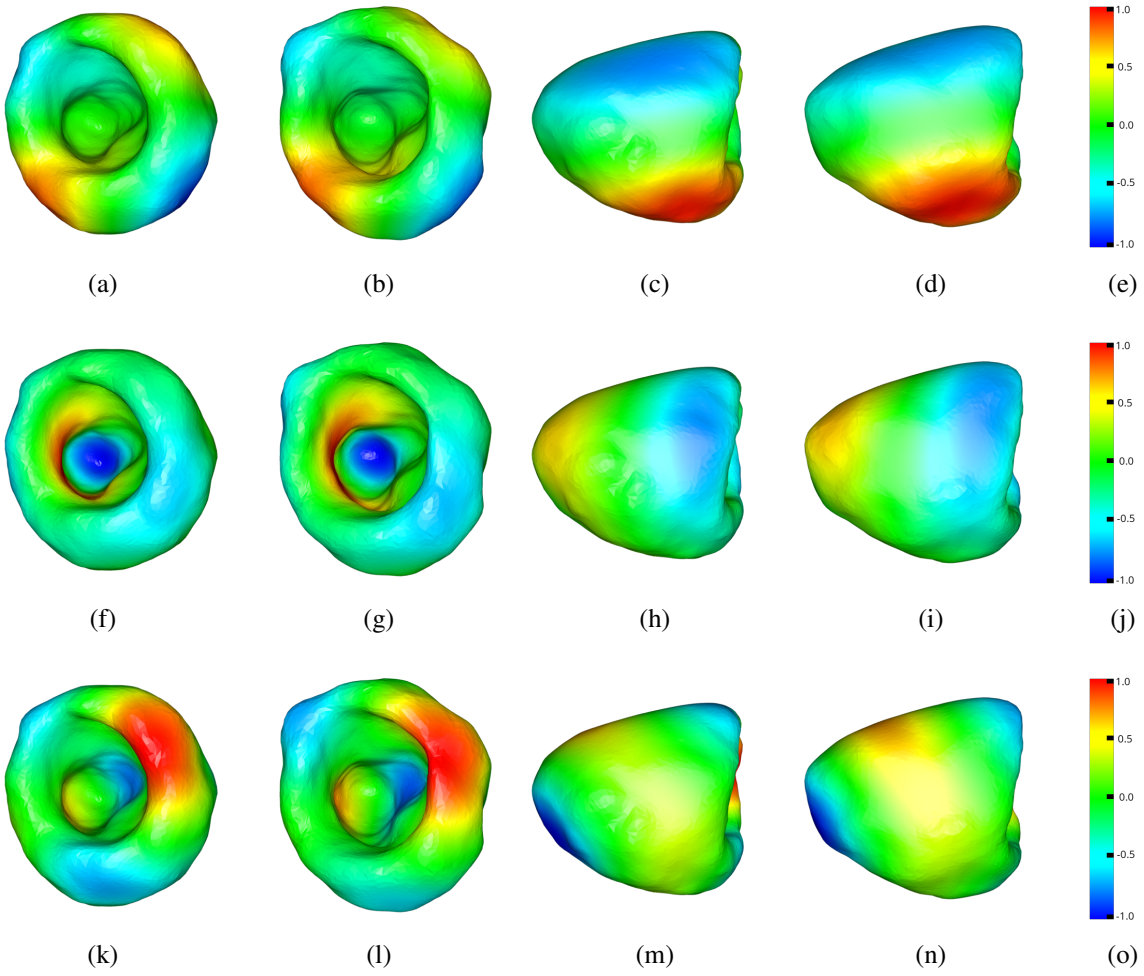
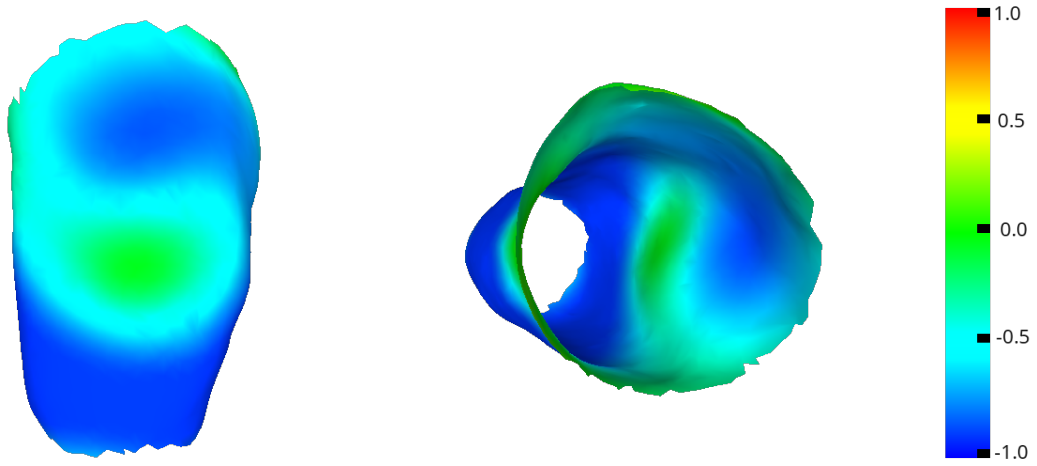


Figure 5.10: LV motion spectrum alignment. The LV motion introduces a sequence of non-isometric deformations. The spectra of each time frame can be aligned with a scale function. Both eigenvalues and eigenfunctions are aligned during the motion. The rows show the 8th, 12th, and 14th eigenfunction shifting respectively. The columns of (a) and (c) represent one time frame in the LV motion and (b) and (d) another.



(a) (b) (c)

Figure 5.11: LV abnormality on the interior wall. Blue color indicates contractions at this time frame. The major parts of the interior wall contract normally. Some myopathy ones have much less or no deformations, which are colored with green.

applies both continuous analytic and discrete cases.

We have also developed an algorithm to align the shape spectra on discrete shapes represented with triangle meshes. Given two closed triangle meshes, the spectra can be aligned from one to another with a scale function defined on each vertex. The alignment is expressed as a linear interpolation of eigenvalues. The interpolation is then decomposed into discrete iterations. In each step, a quadratic programming problem is constructed with the spectrum variation theorem and smoothness energy constraint. The derivate of the scale function is solution of such problem. The final scale function is approximated with integral of the derivate from each step.

Our experiments verify the spectrum variation theorem and illustrate the accuracy and efficiency of the alignment algorithm on general shapes and their motions to shape motion analysis.

CHAPTER 6

CONCLUSION

Shape analysis is a fundamental research topic in computer graphics and computer vision including matching, retrieval, mapping, etc. Inspired by the recent research, we focus on the shapes represented with differential geometry as the differential operators contain the intrinsic geometry information of the original shape. On one hand, the second order Laplace-Beltrami operator introduce a spectral domain where the Euclidean transformations and isometric deformations are filtered out. In such a spectral domain, only the intrinsic shape properties are left. Our research is the shape analysis based on such geometry behind the differential operators.

6.1 Contributions

The main contributions include:

- We have introduced a novel 3D shape representation with a set of salient feature points in Laplace-Beltrami spectrum. The Laplace-Beltrami operator is defined on the geometry of a Riemann manifold. The Laplace eigenvalue problem introduces spectra of shape geometry similar to Fourier transfer on time domain. They also have the similar properties. The shape spectra depend shape geometry, and are invariant to translation, rotation, scaling, and isometric deformations. The shape is then projected to the spectral bases and represented with a linear combination of them. The salient features are extracted from the "frequency" domain as local geometry energy maxima, which shared the invariability of the original spectrum. The maxima provide not only where the features are on the manifold but also the "frequency" where the features lie in, so the scales of the salient features are predicted. With spectra, shapes are represented with a set of invariant salient features with scales. IQP is employed to retrieve the correspondences among variant shapes in very detailed levels. The experiment results show the applications in shape

matching, retrieval, and searching. Partial matching is also supported in our framework.

- We present a novel method to understand the poses in the geometry spectral domain. Poses are defined as a set of near-isometric shapes casted by the same model. Their Laplace-Beltrami spectra stay stable under minor non-isometry deformations. All the poses from the same object can be re-embedded to a high dimensional spectral domain. By transferring spatial poses into the spectral domain, their geometry are aligned naturally there despite Euclidean transformations, triangulations, and, near-isometric deformations. In this case, the pose difference are represented with local geometry properties, e.g., mean curvatures. Investigating in spectral domain, the pose motions will vary the local properties on the fixed geometry. Large variations indicate joints and Small ones rigid parts. The eigenfunction also carry rich geometric meaning, which leads to an automatic skeleton extraction. Combining the part understanding and the skeleton, the semantic deformable model is obtained. The shape spectra help shape understandings and segmentations. It contributes to motion analysis in computer vision and pattern recognition tasks as well.

- We prove the shape spectrum is a piecewise analytic to a scale function on the conformal factor on the Riemann metric of the manifold. The derivatives of the eigenvalues are expressed with those of the scale function at each time. The property applies to both continuous domain and discrete triangle meshes. Further more, a spectrum alignment algorithm is developed on the triangle meshes. In the discrete domain, integration is represented with matrix product. The derivatives of the shape spectrum and the scale vector can be turned into a matrix form, which introduces a linear system. We apply the smoothness and local bound constraints to solve the linear system by reaching the minimum energy of a quadratic programming problem. Given two closed triangle meshes, representing manifolds, the eigenvalues can be aligned from one to the other. After the eigenvalues are registered, the eigenfunction distributions are aligned as well. This

means the shape spectrum can be controlled by the user analytically with a scale vector and non-isometric deformations analysis is available within shape spectra.

6.2 Future Work

This dissertation work introduces the great analysis power with shape spectrum. Along this direction, there exist topics for future work.

1. Spectrum alignment with local deformation. We have proved that the eigenvalues of the shape spectrum is a analytic function of a scale function on the Riemann metric. This linear constraint is under-determined on triangle meshes. The solution has not to be unique with this constraint only. In this dissertation work, we employed global smoothness as extra constraints to achieve an energy optimized solution. It is accurate and efficient on global deformations or smooth deformation. This global smoothness constraints may not precisely locate very sparse and local deformations on a shape. For example, the global optimization often fails in pose changes, where deformations occur only on sparse joints and other parts remain rigid during most of time. Adaptive constraints for specific types of deformations remain open for study in future.

2. Shape level interpolation based on spectra. This dissertation work closes the gap of the shape spectra on non-isometric deformations. We prove that the spectrum can be interpolated and aligned with a scale function on the Riemann metric. Given any general shapes, they can be registered in spectral domain. At current stage, the conversion from spacial to spectral domains is one-way only. The interior process in the spectrum interpolation does not have too much spacial meanings, except the two end points. Our work builds a two-way bridge between the spacial scale function and the spectrum. Following this clue, the spacial properties of the shape spectrum is expected to be future recovered. Also, a shape space may be constructed along this direction.

APPENDIX

List of Publications

Journals

[1] Jiaxi Hu, Guangyu Zou, and Jing Hua, "Volume-Preserving Mapping and Registration for Collective Data Visualization," IEEE Transactions on Visualization and Computer Graphics (also appeared in IEEE VIS '14 conference), Vol. 20, No. 6, pp. 2664-2673, 2014.

[2] Xuejiao Chen, Jiaxi Hu, Huiguang He, and Jing Hua, "Spherical Volume-Preserving Demons Registration," Computer-Aided Design (also appeared in ACM SPM '14 conference), 2014.

[3] Jiaxi Hu and Jing Hua, "Pose Analysis Using Spectral Geometry," The Visual Computer (also appeared in CGI '13 conference), Vol. 29, No. 9, pp. 949-958, 2013.

[4] Guangyu Zou, Jiaxi Hu, Xianfeng Gu, and Jing Hua, "Authalic Parameterization of General Surfaces Using Lie Advection," IEEE Transactions on Visualization and Computer Graphics (also appeared in IEEE VIS '11 conference), Vol. 17, No. 12, pp. 2005-2014, 2011.

[5] Jiaxi Hu and Jing Hua, "Salient Spectral Geometric Features for Shape Matching and Retrieval," The Visual Computer (also appeared in CGI '09 conference), Vol. 25, No. 5-7, pp. 667 - 675, 2009.

Conferences

[6] Guangyu Zou, Jiaxi Hu, Xianfeng Gu, and Jing Hua, "Area-preserving Surface Flattening Using Lie Advection," In Proceedings of the 14th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), pp. 335-342, 2011.

[7] Zhaoqiang Lai, Jiaxi Hu, Chang Liu, Vahid Taimouri, Darshan Pai, Jiong Zhu, Jianrong Xu, and Jing Hua, "Intra-patient Supine-Prone Colon Registration in CT Colonography

Using Shape Spectrum," In Proceedings of the 13th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), pp. 332-339, 2010.

[8] Chang Liu, Zhanoqiang Lai, Jiaxi Hu, and Jing Hua, "Detail Preserving 3D Motion Compression Based on Local Transformation," In Proceedings of the 4th Pacific Rim Symposium on Image and Video Technology, pp. 507 - 514, 2010.

[9] Chang Liu, Jiaxi Hu, Jing Hua, and Hong Qin. "Hierarchical Surface Abstraction Using Adaptive Mean Shift," In Proceedings of the 21st International Conference on Computer Animation and Social Agents (CASA), 2008.

Papers in Preparation

[1] "Non-Isometric Motion Analysis by Variation of Shape Spectrum"

BIBLIOGRAPHY

[1] ANKERST, M., KASTENMÜLLER, G., KRIEGEL, H.-P., AND SEIDL, T. 3D shape histograms for similarity search and classification in spatial databases. In *Proceedings of the 6th International Symposium on Advances in Spatial Databases* (London, UK, UK, 1999), SSD '99, Springer-Verlag, pp. 207–226.

[2] BEMPORAD, A., MIGNONE, D., AND MORARI, M. An Efficient Branch and Bound Algorithm for State Estimation and Control of Hybrid Systems. In *European Control Conference* (1999).

[3] BEN-CHEN, M., GOTSMAN, C., AND BUNIN, G. Conformal flattening by curvature prescription and metric scaling. *Comput. Graph. Forum 27*, 2 (2008), 449–458.

[4] BESPALOV, D., REGLI, W. C., AND SHOKOUFANDEH, A. Reeb graph based shape retrieval for CAD. In *ASME Design Engineering Technical Conferences* (2003), pp. 2–6.

[5] CARR, H., SNOEYINK, J., AND AXEN, U. Computing contour trees in all dimensions. In *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms* (Philadelphia, PA, USA, 2000), SODA '00, Society for Industrial and Applied Mathematics, pp. 918–926.

[6] CARR, H., SNOEYINK, J., AND VAN DE PANNE, M. Simplifying flexible isosurfaces using local geometric measures. In *Proceedings of the Conference on Visualization '04* (Washington, DC, USA, 2004), VIS '04, IEEE Computer Society, pp. 497–504.

[7] CHAVEL, I. *Eigenvalues in Riemannian Geometry*. Pure and applied mathematics. Academic Press, 1984.

[8] CHEN, D., AND OUHYOUNG, M. A 3D object retrieval system based on multi-resolution reeb graph. In *Computer Graphics Workshop* (2002), pp. 16–20.

[9] CHU, H., AND LEE, T. Multiresolution mean shift clustering algorithm for shape interpolation. *IEEE Transactions on Visualization and Computer Graphics 15*, 5 (Sept. 2009), 853–866.

[10] COLE-MCLAUGHLIN, K., EDELSBRUNNER, H., HARER, J., NATARAJAN, V., AND PASCUCCI, V. Loops in reeb graphs of 2-manifolds. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry* (New York, NY, USA, 2003), SCG '03, ACM, pp. 344–350.

[11] CORNEA, N. D., SILVER, D., AND MIN, P. Curve-skeleton properties, applications, and algorithms. *IEEE Transactions on Visualization and Computer Graphics 13*, 3 (May 2007), 530–548.

[12] DEY, T. K., RANJAN, P., AND WANG, Y. Convergence, stability, and discrete approximation of laplace spectra. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms* (Philadelphia, PA, USA, 2010), SODA '10, Society for Industrial and Applied Mathematics, pp. 650–663.

[13] DUDA, R. M., AND HART, P. E. *Pattern Classification and Scene Analysis*. Wiley, 1973.

[14] EL-MEHALAWI, M., AND MILLER, R. A database system of mechanical components based on geometric and topological similarity. part i: Representation. *Computer-Aided Design 35* (2003), 83–94.

[15] EL-MEHALAWI, M., AND MILLER, R. A database system of mechanical components based on geometric and topological similarity. part ii: Indexing, retrieval, matching, and similarity assessment. *Computer-Aided Design 35* (2003), 95–105.

[16] ELAD, M., TAL, A., AND AR, S. Content based retrieval of VRML objects: An iterative and interactive approach. In *Proceedings of the Sixth Eurographics Workshop on Multimedia 2001* (New York, NY, USA, 2002), Springer-Verlag New York, Inc., pp. 107–118.

[17] GAL, R., AND COHEN-OR, D. Salient geometric features for partial shape matching and similarity. *ACM Trans. Graph. 25*, 1 (Jan. 2006), 130–150.

[18] GU, X., GORTLER, S. J., AND HOPPE, H. Geometry images. *ACM Trans. Graph. 21*, 3 (July 2002), 355–361.

[19] GU, X., AND YAU, S. Global conformal surface parameterization. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2003), SGP '03, Eurographics Association, pp. 127–137.

[20] GUÉZIEC, A. P., PENNEC, X., AND AYACHE, N. Medical image registration using geometric hashing. *IEEE Comput. Sci. Eng. 4*, 4 (Oct. 1997), 29–41.

[21] HE, Y., XIAO, X., AND SEAH, H.-S. Harmonic 1-form based skeleton extraction from examples. *Graph. Models 71*, 2 (2009), 49–62.

[22] HILAGA, M., SHINAGAWA, Y., KOHMURA, T., AND KUNII, T. L. Topology matching for fully automatic similarity estimation of 3D shapes. In *The 28th Annual Conference on Computer Graphics and Interactive Techniques* (2001), pp. 203–212.

[23] HORN, B. K. P. Extended gaussian images. *Proceedings of the IEEE 72*, 2 (1984), 1671–1686.

[24] HU, J., AND HUA, J. Salient spectral geometric features for shape matching and retrieval. *Vis. Comput. 25*, 5-7 (Apr. 2009), 667–675.

[25] HUA, J., LAI, Z., DONG, M., GU, X., AND QIN, H. Geodesic distance-weighted shape vector image diffusion. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (Nov. 2008), 1643–1650.

[26] IP, C. Y., LAPADAT, D., SIEGER, L., AND REGLI, W. C. Using shape distributions to compare solid models. In *Proceedings of the Seventh ACM Symposium on Solid Modeling and Applications* (New York, NY, USA, 2002), SMA '02, ACM, pp. 273–280.

[27] IYER, N., KALYANARAMAN, Y., LOU, K., JAYANTI, S., AND RAMANI, K. A reconfigurable 3D engineering shape search system part i: Shape representation. In *ASME Computers and Information in Engineering Conference* (2003).

[28] JAIN, V., AND ZHANG, H. Robust 3D shape correspondence in the spectral domain. In *Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006* (Washington, DC, USA, 2006), SMI '06, IEEE Computer Society, p. 19.

[29] JAMES, D. L., AND TWIGG, C. D. Skinning mesh animations. *ACM Trans. Graph. 24*, 3 (July 2005), 399–407.

[30] KARNI, Z., AND GOTSMAN, C. Spectral compression of mesh geometry. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2000), SIGGRAPH '00, ACM Press/Addison-Wesley Publishing Co., pp. 279–286.

[31] KARNI, Z., AND GOTSMAN, C. Compression of soft-body animation sequences. *Computers & Graphics 28*, 1 (2004), 25–34.

[32] KAZHDAN, M., AND FUNKHOUSER, T. Harmonic 3D shape matching. In *ACM SIGGRAPH 2002 Conference Abstracts and Applications* (New York, NY, USA, 2002), SIGGRAPH '02, ACM, pp. 191–191.

[33] KAZHDAN, M., FUNKHOUSER, T., AND RUSINKIEWICZ, S. Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2003), SGP '03, Eurographics Association, pp. 156–164.

[34] KENDALL, D. G. The diffusion of shape. *Advances in Applied Probability 9*, 3 (1977), 428–430.

[35] KHAREVYCH, L., SPRINGBORN, B., AND SCHRÖDER, P. Discrete conformal mappings via circle patterns. *ACM Trans. Graph. 25*, 2 (Apr. 2006), 412–438.

[36] KILIAN, M., MITRA, N. J., AND POTTMANN, H. Geometric modeling in shape space. *ACM Trans. Graph. 26*, 3 (July 2007).

[37] LAI, Z., AND HUA, J. 3D surface matching and registration through shape images. In *Proceedings of the 11th International Conference on Medical Image Computing and Computer-Assisted Intervention, Part II* (Berlin, Heidelberg, 2008), MICCAI '08, Springer-Verlag, pp. 44–51.

[38] LAMDAN, Y., AND WOLFSON, H. J. Geometric hashing: a general and efficient model based recognition scheme. In *International Conference on Computer Vision* (1988), pp. 238–249.

[39] LEE, C. H., VARSHNEY, A., AND JACOBS, D. W. Mesh saliency. *ACM Trans. Graph. 24*, 3 (July 2005), 659–666.

[40] LEIBOWITZ, N., FLIGELMAN, Z. Y., NUSSINOV, R., AND WOLFSON, H. J. Multiple structural alignment and core detection by geometric hashing. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology* (1999), AAAI Press, pp. 169–177.

[41] LEVOY, M., PULLI, K., CURLESS, B., RUSINKIEWICZ, S., KOLLER, D., PEREIRA, L., GINZTON, M., ANDERSON, S., DAVIS, J., GINSBERG, J., SHADE, J., AND FULK, D. The digital michelangelo project: 3D scanning of large statues. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (New

York, NY, USA, 2000), SIGGRAPH '00, ACM Press/Addison-Wesley Publishing Co., pp. 131–144.

[42] LEVY, B. Laplace-beltrami eigenfunctions towards an algorithm that "understands" geometry. In *Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006* (Washington, DC, USA, 2006), SMI '06, IEEE Computer Society, pp. 13–.

[43] LÉVY, B., PETITJEAN, S., RAY, N., AND MAILLOT, J. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph. 21*, 3 (July 2002), 362–371.

[44] LOU, K., JAYANTI, S., IYER, N., KALYANARAMAN, Y., RAMANI, K., AND PRABHAKAR, S. A reconfigurable 3D engineering shape search system. part ii: Database indexing, retrieval and clustering. In *ASME Computers and Information in Engineering Conference* (2003).

[45] MARR, D. *Vision: a Computational Investigation into the Human Representation and Processing of Visual Information*. W.H. Freeman, 1982.

[46] MERRIS, R. Laplacian matrices of graphs: a survey. *Linear Algebra and its Applications 197-198* (1994), 143–176.

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

[48] MOHAR, B. *The Laplacian spectrum of graphs*. Wiley, 1991.

[49] MOHAR, B. Laplace eigenvalues of graphs—a survey. *Discrete Math. 109*, 1-3 (Nov. 1992), 171–183.

[50] NOVOTNI, M., AND KLEIN, R. 3D Zernike descriptors for content based shape retrieval. In *Proceedings of the Eighth ACM Symposium on Solid Modeling and Applications* (New York, NY, USA, 2003), SM '03, ACM, pp. 216–225.

[51] OHBUCHI, R., NAKAZAWA, M., AND TAKEI, T. Retrieving 3D shapes based on their appearance. In *Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval* (New York, NY, USA, 2003), MIR '03, ACM, pp. 39–45.

[52] OHBUCHI, R., AND TAKEI, T. Shape-similarity comparison of 3D models using alpha shapes. In *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications* (Washington, DC, USA, 2003), PG '03, IEEE Computer Society, pp. 293–.

[53] OSADA, R., FUNKHOUSER, T., CHAZELLE, B., AND DOBKIN, D. Shape distributions. *ACM Trans. Graph. 21*, 4 (Oct. 2002), 807–832.

[54] PASCUCCI, V., SCORZELLI, G., BREMER, P.-T., AND MASCARENHAS, A. Robust online computation of reeb graphs: Simplicity and speed. *ACM Trans. Graph. 26*, 3 (July 2007).

[55] PATANÈ, G., SPAGNUOLO, M., AND FALCIDIENO, B. A minimal contouring approach to the computation of the reeb graph. *IEEE Transactions on Visualization and Computer Graphics 15*, 4 (July 2009), 583–595.

[56] REGLI, W. C., AND CICIRELLO, V. A. Managing digital libraries for computer-aided design. *Computer-Aided Design 32*, 2 (2000), 119–132.

[57] REUTER, M. Hierarchical shape segmentation and registration via topological features of laplace-beltrami eigenfunctions. *Int. J. Comput. Vision 89*, 2-3 (Sept. 2010), 287–308.

[58] REUTER, M., NIETHAMMER, M., WOLTER, F.-E., BOUIX, S., AND SHENTON, M. Global medical shape analysis using the volumetric laplace spectrum. In *Proceedings of*

the 2007 International Conference on Cyberworlds (Washington, DC, USA, 2007), CW '07, IEEE Computer Society, pp. 417–426.

[59] REUTER, M., WOLTER, F.-E., AND PEINECKE, N. Laplace-beltrami spectra as 'shape-dna' of surfaces and solids. *Comput. Aided Des. 38*, 4 (Apr. 2006), 342–366.

[60] REUTER, M., WOLTER, F.-E., SHENTON, M., AND NIETHAMMER, M. Laplace-beltrami eigenvalues and topological features of eigenfunctions for statistical shape analysis. *Comput. Aided Des. 41*, 10 (Oct. 2009), 739–755.

[61] RUSTAMOV, R. M. Laplace-beltrami eigenfunctions for deformation invariant shape representation. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2007), SGP '07, Eurographics Association, pp. 225–233.

[62] SADJADI, F. A., AND HALL, E. L. Three-dimensional moment invariants. *IEEE Trans. Pattern Anal. Mach. Intell. 2*, 2 (Feb. 1980), 127–136.

[63] SAUPE, D., AND VRANIC, D. V. 3D model retrieval with spherical harmonics and moments. In *Proceedings of the 23rd DAGM-Symposium on Pattern Recognition* (London, UK, UK, 2001), Springer-Verlag, pp. 392–397.

[64] SHI, Y., LAI, R., GILL, R., PELLETIER, D., MOHR, D., SICOTTE, N., AND TOGA, A. W. Conformal metric optimization on surface (cmos) for deformation and mapping in laplace-beltrami embedding space. In *Proceedings of the 14th International Conference on Medical Image Computing and Computer-assisted Intervention - Volume Part II* (Berlin, Heidelberg, 2011), MICCAI'11, Springer-Verlag, pp. 327–334.

[65] SHINAGAWA, Y., AND KUNII, T. L. Constructing a reeb graph automatically from cross sections. *IEEE Comput. Graph. Appl. 11*, 6 (Nov. 1991), 44–51.

[66] SHUM, H., HEBERT, M., AND IKEUCHI, K. On 3D shape similarity. In *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)* (Washington, DC, USA, 1996), CVPR '96, IEEE Computer Society, pp. 526–.

[67] SUMNER, R. W., AND POPOVIĆ, J. Deformation transfer for triangle meshes, Aug. 2004.

[68] SUNDAR, H., SILVER, D., GAGVANI, N., AND DICKINSON, S. Skeleton based shape matching and retrieval. In *Proceedings of the Shape Modeling International 2003* (Washington, DC, USA, 2003), SMI '03, IEEE Computer Society, pp. 130–.

[69] TAIMOURI, V., AND HUA, J. Visualization of shape motions in shape space. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (Dec. 2013), 2644–2652.

[70] TUNG, T., AND SCHMITTT, F. Shape retrieval of noisy watertight models using aMRG. In *International Conference on Shape Modeling and Applications* (2008), pp. 229–230.

[71] VRANIC, D., SAUPE, D., AND RICHTER, J. Tools for 3D object retrieval: Karhunen-cloeve transform and spherical harmonics. In *The IEEE 2001 Workshop on Multimedia Signal Processing* (2001), pp. 293–298.

[72] WEBER, O., SORKINE, O., LIPMAN, Y., AND GOTSMAN, C. Context-aware skeletal shape deformation. *Computer Graphics Forum (Proceedings of Eurographics) 26*, 3 (2007), 265–274.

[73] WOLFSON, H. J., AND RIGOUTSOS, I. Geometric hashing: An overview. *IEEE Comput. Sci. Eng. 4*, 4 (Oct. 1997), 10–21.

[74] YAN, H., HU, S., MARTIN, R. R., AND YANG, Y. Shape deformation using a skeleton to drive simplex transformations. *IEEE Transactions on Visualization and Computer Graphics 14*, 3 (May 2008), 693–706.

[75] ZHOU, K., SYNDER, J., GUO, B., AND SHUM, H. Iso-charts: Stretch-driven mesh parameterization using spectral analysis. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing* (New York, NY, USA, 2004), SGP '04, ACM, pp. 45–54.

[76] ZOU, G., HU, J., GU, X., AND HUA, J. Authalic parameterization of general surfaces using lie advection. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (2011), 2005–2014.

[77] ZOU, G., HUA, J., DONG, M., AND QIN, H. Surface matching with salient keypoints in geodesic scale space. *Comput. Animat. Virtual Worlds 19*, 3-4 (Sept. 2008), 399–410.

ABSTRACT

SHAPE ANALYSIS USING SPECTRAL GEOMETRY

by

JIAXI HU

May 2015

Advisor: Dr. Jing Hua

Major: Computer Science

Degree: Doctor of Philosophy

Shape analysis is a fundamental research topic in computer graphics and computer vision. To date, more and more 3D data is produced by those advanced acquisition capture devices, e.g., laser scanners, depth cameras, and CT/MRI scanners. The increasing data demands advanced analysis tools including shape matching, retrieval, deformation, etc. Nevertheless, 3D Shapes are represented with Euclidean transformations such as translation, scaling, and rotation and digital mesh representations are irregularly sampled. The shape can also deform non-linearly and the sampling may vary. In order to address these challenging problems, we investigate Laplace-Beltrami shape spectra from the differential geometry perspective, focusing more on the intrinsic properties. In this dissertation, the shapes are represented with 2 manifolds, which are differentiable.

First, we discuss in detail about the salient geometric feature points in the Laplace-Beltrami spectral domain instead of traditional spatial domains. Simultaneously, the local shape descriptor of a feature point is the Laplace-Beltrami spectrum of the spatial region associated to the point, which are stable and distinctive. The salient spectral geometric features are invariant to spatial Euclidean transforms, isometric deformations and mesh triangulations. Both global and partial matching can be achieved with these salient feature points. Next, we introduce a novel method to analyze a set of poses, i.e., near-isometric deformations, of 3D models that are

unregistered. Different shapes of poses are transformed from the 3D spatial domain to a geometry spectral one where all near isometric deformations, mesh triangulations and Euclidean transformations are filtered away. Semantic parts of that model are then determined based on the computed geometric properties of all the mapped vertices in the geometry spectral domain while semantic skeleton can be automatically built with joints detected. Finally we prove the shape spectrum is a continuous function to a scale function on the conformal factor of the manifold. The derivatives of the eigenvalues are analytically expressed with those of the scale function. The property applies to both continuous domain and discrete triangle meshes. On the triangle meshes, a spectrum alignment algorithm is developed. Given two closed triangle meshes, the eigenvalues can be aligned from one to the other and the eigenfunction distributions are aligned as well. This extends the shape spectra across non-isometric deformations, supporting a registration-free analysis of general motion data.

AUTOBIOGRAPHICAL STATEMENT

JIAXI HU

Jiaxi Hu is a Ph.D. candidate in Computer Science at Wayne State University, where he is also a research assistant in the Graphics and Imaging Laboratory. He received his BS degree (2003) and MS degree (2005) in Electronic Engineering, from Huazhong University of Science and Technology (HUST), Wuhan, China. His research interests include computer graphics, visualization, and their applications to computer vision and medical imaging.