Spectral Methods for Multimodal Data Analysis

Doctoral Dissertation submitted to the

Faculty of Informatics of the Università della Svizzera Italiana

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

presented by

Artiom Kovnatsky

under the supervision of

Prof. Michael Bronstein

November 2016

Dissertation Committee

Prof. Kai Hormann	Università della Svizzera Italiana, Switzerland
Prof. Rolf Krause	Università della Svizzera Italiana, Switzerland
Prof. Bruno Lévy	INRIA-Nancy Grand-Est, France
Prof. Maks Ovsjanikov	École Polytechnique, France

Dissertation accepted on 1 November 2016

Research Advisor	PhD Program Director
Prof. Michael Bronstein	**Prof. Walter Binder**

I certify that except where due acknowledgement has been given, the work presented in this thesis is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; and the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program.

Artiom Kovnatsky
Lugano, 1 November 2016

Ignorantia non est argumentum.

Baruch de Spinoza

Abstract

Spectral methods have proven themselves as an important and versatile tool in a wide range of problems in the fields of computer graphics, machine learning, pattern recognition, and computer vision, where many important problems boil down to constructing a Laplacian operator and finding a few of its eigenvalues and eigenfunctions. Classical examples include the computation of diffusion distances on manifolds in computer graphics, Laplacian eigenmaps, and spectral clustering in machine learning.

In many cases, one has to deal with multiple data spaces simultaneously. For example, clustering multimedia data in machine learning applications involves various modalities or "views" (*e.g.*, text and images), and finding correspondence between shapes in computer graphics problems is an operation performed between two or more modalities.

In this thesis, we develop a generalization of spectral methods to deal with multiple data spaces and apply them to problems from the domains of computer graphics, machine learning, and image processing. Our main construction is based on simultaneous diagonalization of Laplacian operators. We present an efficient numerical technique for computing joint approximate eigenvectors of two or more Laplacians in challenging noisy scenarios, which also appears to be the first general non-smooth manifold optimization method. Finally, we use the relation between joint approximate diagonalizability and approximate commutativity of operators to define a structural similarity measure for images. We use this measure to perform structure-preserving color manipulations of a given image.

To the best of our knowledge, the original contributions of this work are the following:

1 Introduction of joint diagonalization methods to the fields of machine learning, computer vision, pattern recognition, image processing, and graphics;

2 Formulation of the coupled approximate diagonalization problem that extends the joint diagonalization to cases with no bijective correspondence between the domains, and its application in a wide range of problems in the above fields;

3 Introduction of a new structural similarity measure of images based on the approximate commutativity of their respective Laplacians, and its application in image processing problems such as color-to-gray conversion, colors adaptation for color-blind viewers, gamut mapping, and multispectral image fusion;

4 Development of Manifold Alternating Direction Method of Multipliers (MADMM), the first general method for non-smooth optimization with manifold constraints, and its applications to several problems.

Acknowledgements

I am grateful to my advisor, Prof. Michael M. Bronstein, for his guidance, support, and patience. I would like to thank all the current and ex-members of the research group: Prof. Klaus Glashoff, Dr. Davide Eynard, Davide Boscaini, Dr. Ramunas Girdziusas, Dr. Jonathan Masci, Jan Svoboda, Dr. Emanule Rodolà and others, for making my stay here comfortable and pleasant.

I would also like to thank the Dean's office team, Janine Caggiano, Daniela Milicevic, and Elisa Larghi for their help and for making all the formal procedures as smooth as they possibly can be.

Last but not least, I would like to thank my family: Alexander Kovnatsky, Ludmila Kovnatsky, Alina Kovnatsky, and Eva Kovnatsky for their support and love.

Finally, the generous funding by USI – Università della Svizzera Italiana and the ERC grant COMET is gratefully acknowledged.

Contents

Figures

Tables

Chapter 1

Introduction

1.1 Motivation and Problem Formulation

Spectral methods proved to be a powerful and versatile tool, widely used in many applications in manifold learning, pattern recognition, computer vision, computer graphics, and image processing. In many applications in these fields, one can model the data as a non-Euclidean domain (*e.g.*, manifold or graph) and analyse its intrinsic properties captured by the Laplacian operator. The first few eigenvectors[1] and eigenvalues of the Laplace operator allow to express many of such intrinsic properties in a simple way.

In machine learning, one of the most known examples is spectral clustering [Ng et al., 2001] (over 4,700 citations as of the date of writing this thesis), in which the data is modelled as a graph capturing local proximity information of data points. Using the first few Laplacian eigenfunctions of this graph, one obtains a neighbourhood preserving embedding of the data: connected points are mapped close to each other, while those weakly connected are mapped far apart. This construction is closely related to Laplacian eigenmaps [Belkin and Niyogi, 2002] and diffusion maps [Coifman et al., 2005], used for non-linear dimensionality reduction and data visualization. In computer graphics, the influential paper of Lévy [2006] and numerous follow-up works have established spectral

[1] In the following, we will use the terms *eigenfunctions* and *eigenvectors* interchangeably, typically referring to the discrete setting by the latter.

methods as a standard tool, used in applications such as shape segmentation [Shi and Malik, 2000; Reuter, 2010], filtering [Vallet and Lévy, 2008], and computing intrinsic metrics [Coifman et al., 2005]. The notable advantage of such methods is that they can be applied to data in a variety of different representations (see Figure 1.1), provided a suitable discretization of the Laplacian operator is given [Meyer et al., 2003; von Luxburg, 2007; Belkin et al., 2008; Botsch et al., 2010]. In their seminal work Ovsjanikov et al. [2012] proposed a novel representation of correspondences between shapes as linear maps between functional spaces. These linear maps can be compactly represented in Laplacian eigenbases that act as a generalization of Fourier bases to non-Euclidean domains. Finally, in image processing, a spectral-based image segmentation of Shi and Malik [2000] is one of the most influential papers in the domain (over 10,000 citations as of the date of writing).

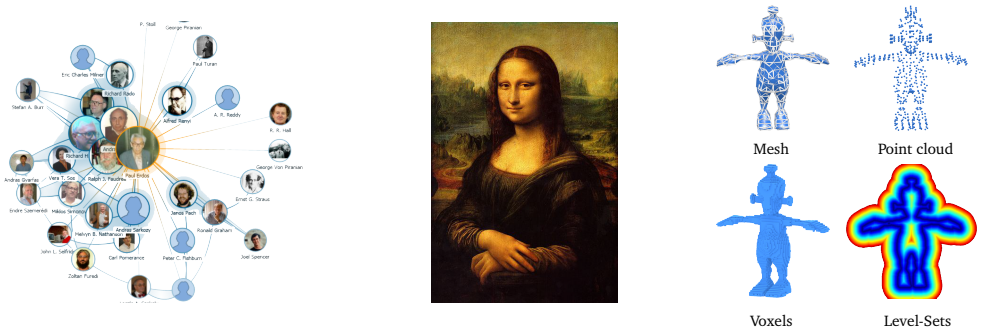


Figure 1.1. Examples of different data to which spectral methods can be applied. Shown left-to-right: a social network graph, an image, and several different representations of a 3D shape (mesh, point-clouds, volumetric, and implicit surface).

One of the main limitations of classical spectral constructions is that they typically work with a *single* domain (manifold, graph, etc.) at hand. Yet, in many applications we are interested in *simultaneously* analysing data from *multiple* domains. For example, in various applications, different manifolds can model observations and data measurements coming from different 'modalities' or 'views'.

Typical examples include multimedia documents such as annotated images [Weston et al., 2010; Rasiwasia et al., 2010; McFee and Lanckriet, 2011], audio and

video [Kidron et al., 2005; Alameda-Pineda et al., 2011], PET and CT modalities in medical imaging [Bronstein et al., 2010b]. In computer graphics, problems of shape correspondence, pose transfer, etc. involve multiple manifolds. In image processing, one often has to deal with images taken at different spectral bands or different illumination.

Extending spectral methods to multiple data spaces is difficult because Laplacian eigenbases do not behave consistently on different domains and thus transferring information from one domain to another is problematic. Even for isometric domains with simple spectrum (*i.e.*, multiplicity of eigenvalues equal to one), eigenfunctions are only defined up to sign flips. In the more general case, the eigenfunctions corresponding to an eigenvalue with non-trivial multiplicity span an eigen-subspace in which one can obtain an arbitrary orthonormal basis. Furthermore, due to numerical inaccuracies, the ordering of the eigenfunctions, especially those representing higher frequencies, can change (see Figure 1.2).

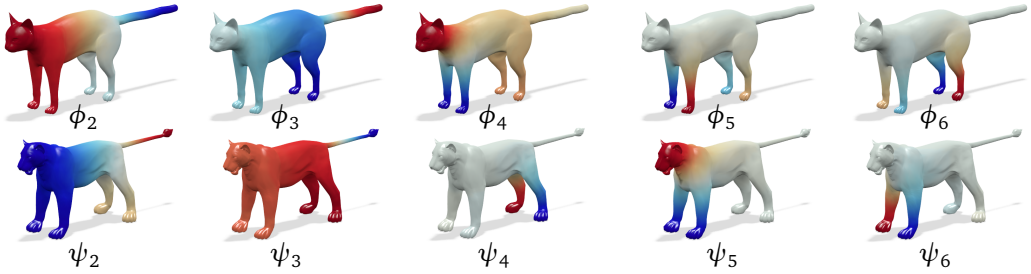
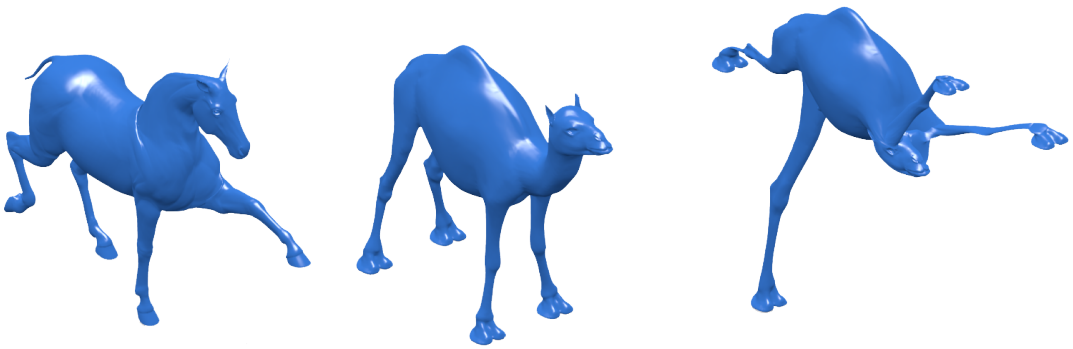


Figure 1.2. First Laplacian eigenfunctions of the cat (top) and lion (bottom) shapes. While the first two eigenfunctions differ only up to sign ($\phi_2 \approx -\psi_2, \phi_3 \approx -\psi_3$), at higher frequencies, the order of the eigenfunctions changes (e.g., $\phi_5 \approx -\psi_6$, $\phi_6 \approx -\psi_4$).

To illustrate the problems these inconsistencies may cause, we refer to the pose transfer application from [Lévy, 2006]. In this example, the embedding coordinates of two shapes (horse and camel, see Figure 1.3) are represented in the respective Laplacian eigenbases. Then, the first coefficients ("low frequency" capturing the pose) of the camel are replaced with those of the horse, while the rest of the coefficients ("high frequency" capturing the geometric details) are kept from the camel. This method relies on the assumption that the two bases behave consistently (at least for "low frequency"), or in Lévy's words, that the

corresponding Fourier coefficients “speak the same language”. Unfortunately, since usually this is not the case, as mentioned previously, such a pose transfer would typically fail.



Source Target Pose transfer result

Figure 1.3. Example of spectral pose transfer failure due to incompatibility of Laplacian eigenbases on different domains.

To visualise the same problem in the domain of machine learning, let us consider the classical example of concentric rings from [Ng et al., 2001] in Figure 1.4. Ng *et al.* use this example to illustrate the power of spectral clustering: standard clustering techniques (*e.g.*, k-means [Hochbaum and Shmoys, 1985]) fail to group points belonging to different circles into separate clusters. However, by constructing the Laplacian that captures the proximity relations of the points and embedding them with the first few Laplacian eigenfunctions, one can separate and cluster easily points belonging to each circle. We modify this example to mimic applications of spectral methods to real-world data that is often noisy. By moving the circles, we introduce topological noise to the proximity relations between points and, as a result, the performance of spectral clustering deteriorates dramatically (parts of two touching rings are clustered together). By taking two such sets of concentric circles, each with a different topological noise, we can build a toy model for multi-modal spectral clustering where each set represents a different modality (for a more realistic example see Figure 1.5).

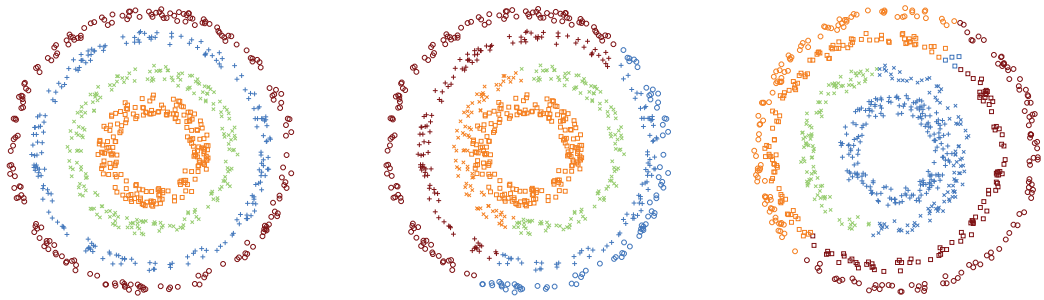


Figure 1.4. Spectral clustering example from [Ng et al., 2001] (left). We introduce a topological noise in proximity relations between points by moving the circles (center and right). Two sets of eccentric rings represent different modalities; in each modality, the rings touch at different points. Spectral clustering applied to each of the modalities separately produces poor results. In particular, parts of touching rings are clustered together. The idea is to analyse *simultaneously* two modalities to cluster the points correctly.

1.2 Related Work

The Laplace operator and its spectrum play a pivotal role in many applications of various disciplines. Orthogonal eigenfunctions of the Laplacian, often referred to as *manifold harmonics*, play the role of Fourier bases, whereas eigenvalues are analogous to frequencies [Taubin, 1995]. By this analogy, standard operations in signal processing such as analysis and synthesis of signals can be carried out on manifolds [Taubin, 1995]. In computer graphics community, this analogy has been exploited in [Kim and Rossignac, 2005; Lévy, 2006; Vallet and Lévy, 2008; Lévy and Zhang, 2009; Kovnatsky et al., 2012] for shape filtering and editing in the frequency domain.

In image processing, modelling the image as a graph is considered somewhat exotic. One of the most notable example of such a model is the spectral relaxation of the normalized cut criterion, which is widely used for image segmentation [Shi and Malik, 2000]. More recently, Bansal and Daniilidis [2013] used the eigenvectors of image Laplacians to perform matching of images taken in different illumination conditions, arguing that the Laplacian acts as a self-similarity descriptor [Shechtman and Irani, 2007] of the image.

In manifold learning, important applications of spectral methods include *spec-*

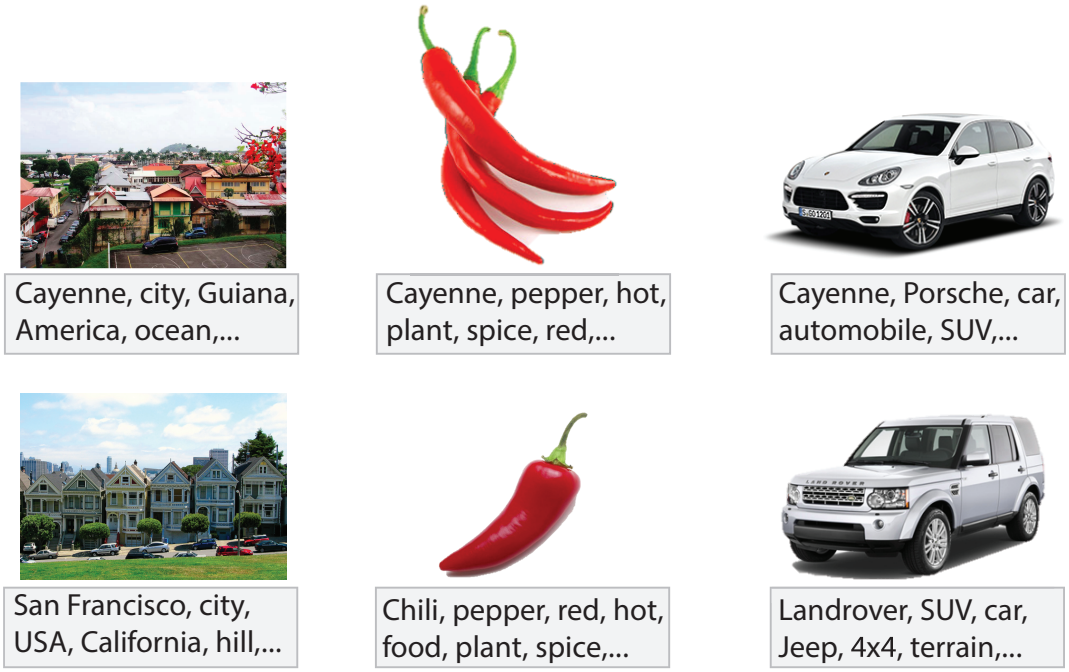


Figure 1.5. Example of ambiguities present in multimodal (image and text) data. Visually, the images (shown in the top and bottom rows) of Cayenne and San Francisco cities, Cayenne and chili pepper, and Porsche Cayenne and LandRover cars are similar. At the same time, the text annotations of the Porsche Cayenne, Cayenne city, and Cayenne pepper are similar (they all contain the word 'Cayenne').

tral clustering [Ng et al., 2001] where clusters are determined by the first eigenvectors of the Laplacian; *eigenmaps* [Belkin and Niyogi, 2002] and more generally *diffusion maps* [Nadler et al., 2005], where one tries to find a low-dimensional manifold structure using the first smallest eigenvectors of the Laplacian; and *diffusion metrics* [Coifman et al., 2005] measuring the "connectivity" of points on a manifold and expressed through the eigenvalues and eigenvectors of the Laplacian. Other applications heavily relying on the properties of the Laplacian include *spectral graph partitioning* [Ding et al., 2001], *spectral hashing* [Weiss et al., 2008], spectral graph correspondence [Clements and Zhang, 2006]. Because of the intimate relation between the Laplacian operator, Riemannian geometry, and diffusion processes [Nadler et al., 2005], it is common to encounter the umbrella term *spectral* or *diffusion geometry* in relation to the problems involving Laplace

eigendecomposition.

In computer graphics and geometry processing, spectral methods have been applied to a wide range of problems such as remeshing [Kobbelt, 1997; Nealen et al., 2006], parametrization [Floater and Hormann, 2005; Jones et al., 2008], compression [Karni and Gotsman, 2000], recognition [Reuter et al., 2006; Rustamov, 2007], segmentation [Reuter, 2010; Litman et al., 2011], symmetry detection [Ovsjanikov et al., 2008], editing [Lévy, 2006], shape similarity [Hu and Hua, 2009; Reuter et al., 2006; Rustamov, 2007], pose transfer and deformation [Rong et al., 2008], and functional correspondence [Ovsjanikov et al., 2012].

Applications dealing with multiple shapes often either assume compatible behaviour of manifold harmonics across the shapes [Vallet and Lévy, 2008], use manual processing, or employ heuristics to remove the inconsistencies due to sign flips, numerical instabilities, etc. [Reuter, 2010; Shtern and Kimmel, 2014; Ovsjanikov et al., 2008]. Such heuristics are designed with a particular application in mind and lack a rigorous theoretical foundation.

In the machine learning, computer vision, and pattern recognition community, there has been a recently increasing interest in dealing with multimodal data, such as multimedia documents [Weston et al., 2010; Rasiwasia et al., 2010; McFee and Lanckriet, 2011], audio and video [Kidron et al., 2005; Alameda-Pineda et al., 2011], or medical imaging modalities like PET and CT [Bronstein et al., 2010b]. Unfortunately, there have been only few attempts of a systematic extension of spectral methods to multimodal settings, most of them in the field of machine learning in application to multi-view clustering [Cai et al., 2011; Kumar et al., 2011; Dong et al., 2014]. A major drawback of these approaches is the assumption of bijective correspondence between the manifolds representing different modalities, which is too restrictive in many cases.

1.3 Main Contributions

The main contribution of our work is a principled framework allowing to extend classical spectral techniques (such as diffusion maps and spectral clustering) to multiple modalities. Our approach is based on simultaneous diagonalization of

Laplacians, which allows constructing orthogonal approximate eigenbases behaving consistently across multiple domains. While simultaneous diagonalization methods are known in numerical algebra [Bunse-Gerstner et al., 1993] and have been used before in some applications of signal processing such as blind source separation [Cardoso and Souloumiac, 1993, 1996; Yeredor, 2002; Ziehe, 2005], to the best of our knowledge, we were the first to introduce such methods in spectral data analysis.

First, we start from the joint diagonalization problem, in which one tries to construct a common eigenbases for several Laplacian matrices (Chapter 3). Such a formulation assumes that matrices have equal dimensionality and a bijective known correspondence between the underlying multi-modal data (*i.e.*, a known correspondence between rows and columns). A generalization referred to as a coupled diagonalization is presented in Chapter 4. This formulation resolves the main drawbacks of the joint diagonalization formulation.

We show that the coupled diagonalization problem boils down to optimization where variables are constrained to be orthogonal matrices, *i.e.* instances of the Stiefel matrix manifold. This type of problems are referred to as *manifold-constrained optimization*. There are several efficient numerical techniques designed to perform the optimization directly on the manifold. To deal with noisy data, we propose a robust formulation of the coupled diagonalization (using matrix norms such as $\|\cdot\|_1$ or $\|\cdot\|_{2,1}$), which results in a non-smooth manifold optimization. Chapter 5 is dedicated to developing a simple yet powerful technique referred to as Manifold Alternating Direction Method of Multipliers (MADMM) allowing to address such problems. To the best of our knowledge, this technique is the first general method for the optimization of non-smooth functions with manifold constraints, and has applications extending way beyond the specific problems considered in this thesis (we show a few such applications).

In Chapter 6, we study the use of simultaneous diagonalization techniques in pattern recognition and computer vision problems. In particular, we show that multi-modal spectral clustering can be performed using the joint approximate eigenvectors of the Laplacians built on different modalities. We also show that several previous approaches for multi-modal clustering can be considered as particular settings of our framework.

In Chapter 7, we apply simultaneous diagonalization methods to problems in computer graphics and geometric processing. Using the relation between our method and the functional correspondence (Section 4.4), we propose an efficient method for finding correspondence between non-rigid shapes. We also show examples of applications such as pose transfer and shape editing.

In Chapter 8, we study the connection between joint diagonalizability and commutativity of matrices, and use it to formulate a new structural similarity measure for images. This measure is employed to perform structure preserving color image manipulations, such as decolorization, gamut mapping, and color optimization for color-blind viewers. Our study in this chapter is supported by an extensive visual perception experiment.

Finally, in Chapter 9 rather than simply concluding the work and outlining a few promising directions, as one would usually expect in a PhD thesis, we refer to several follow-up works that emerged from our research and describe the works of other authors who used our code and the techniques we have developed.

1.4 Published works

This thesis is mainly based on the publications listed below:

- **A. Kovnatsky**, M. M. Bronstein, A. M. Bronstein, K. Glashoff, R. Kimmel, "Coupled quasi-harmonic bases", *Computer Graphics Forum* 32:439–448, 2013.

- D. Eynard, **A. Kovnatsky**, M. M. Bronstein, K. Glashoff, A. M. Bronstein, "Multimodal manifold analysis using simultaneous diagonalization of Laplacians", *Trans. Pattern Analysis and Machine Intelligence* 37(12):2505–2517, 2015.

- **A. Kovnatsky**, M. M. Bronstein, K. Glashoff, "MADMM: a generic algorithm for non-smooth optimization on manifolds", In *Proc. ECCV*, 2016.

- D. Eynard*, **A. Kovnatsky***[2], M. M. Bronstein, "Laplacian colormaps: struc-

[2] *Equal contribution

ture preserving color transformation", *Computer Graphics Forum* 33(2):215–224, 2014.

For the sake of the presentation coherence, several additional papers published by the author during his PhD studies are not included in this dissertation:

- **A. Kovnatsky**, M. M. Bronstein, X. Bresson, and P. Vandergheynst, "Functional correspondence by matrix completion", In *Proc. CVPR*, 2015.
- **A. Kovnatsky**, D. Eynard, M. M. Bronstein, "Gamut mapping with image Laplacian commutators", In *Proc. ICIP*, 2014.
- **A. Kovnatsky**, A. M. Bronstein, M. M. Bronstein, "Stable spectral mesh filtering", In *Proc. NORDIA*, 2012.

Chapter 2

Background

In this chapter, we cover the mathematical background focusing only on the definitions used afterwards. For a comprehensive treatment of the related topics, we refer the reader to [do Carmo, 1976, 1992; Burago et al., 2001].

2.1 Differential Geometry

A key concept in our work in the notion of *manifolds*. In computer graphics applications, two-dimensional manifolds representing boundary surfaces of 3D objects are a common way of modeling 3D shapes. In machine learning applications, high-dimensional data are usually assumed to have a low-dimensional intrinsic structure, also modeled as a manifold. Finally, some of the optimization problems we will encounter in the next chapters use functions of matrix arguments, where the variables are restricted to a sub-manifold of the space of matrices. In this section, we briefly overview basic notions from differential geometry, a branch of mathematics dealing with manifolds.

2.1.1 Manifolds

A p-dimensional *manifold*[1] $\mathscr{M}$ is a topological (Hausdorff) space locally homeomorphic to a p-dimensional Euclidean space. Such a homeomorphism $\alpha : U_m \rightarrow$

[1] In the following, we will tacitly assume that $\mathscr{M}$ is compact.

$\mathbb{R}^p$ from a neighborhood U_m of a point $m \in \mathscr{M}$ to $\mathbb{R}^p$ is called a *chart*. If transition between charts from overlapping neighborhoods is smooth (r-times continuously differentiable, where r is assumed to be some sufficiently high integer), $\mathscr{M}$ is said to be a *smooth manifold*.

Assume two smooth curves $\gamma_1, \gamma_2 : (-1,1) \to \mathscr{M}$ are given such that $\gamma_1(0) = \gamma_2(0) = m$. The curves are called equivalent if the derivatives $(\alpha \circ \gamma_1)'(0)$ and $(\alpha \circ \gamma_2)'(0)$ coincide. The equivalence classes of all curves with $\gamma(0) = m$ are called *tangent vectors*. The *tangent space* $T_m\mathscr{M}$ of $\mathscr{M}$ at point m is the set of all tangent vectors and does not depend on the choice of the chart α. In the following, we will simply write γ' to refer to a tangent vector to the curve.

A particular case of interest are *embedded manifolds* $\mathscr{M} \subset \mathbb{R}^d$. Such manifolds can be realized by a smooth mapping $\xi : U \subseteq \mathbb{R}^p \to \mathbb{R}^d$. $\xi(U)$ is called the *embedding* of $\mathscr{M}$ in $\mathbb{R}^d$; the latter is called the *ambient space* and $d - p > 0$ the *codimension*. For example, what is colloquially referred to as a '3D sphere' is a two-dimensional surface embedded in a three-dimensional Euclidean space. The derivatives $\frac{\partial}{\partial u_i}\xi(u)$ are tangent vectors to $\mathscr{M}$ at point $m = \xi(u)$; ξ is said to be *regular* if $\frac{\partial}{\partial u_1}\xi(u), \ldots, \frac{\partial}{\partial u_p}\xi(u)$ are linearly independent, in which case they span the tangent space $T_{\xi(u)}\mathscr{M}$. At each point m of the manifold, the vector $\mathbf{n}(m)$ orthogonal to $T_m\mathscr{M}$ is called the *normal*.

2.1.2 Riemannian metric

A *Riemannian metric* or the *first fundamental form* is an inner product $\langle \cdot, \cdot \rangle_{T_m\mathscr{M}} : T_m\mathscr{M} \times T_m\mathscr{M} \to \mathbb{R}$ defined on the tangent space $T_m\mathscr{M}$ and depending smoothly on m. On a parametric manifold, the embedding ξ pulls back a Riemannian metric from the ambient space, which can be expressed in the basis $\frac{\partial}{\partial u_1}\xi(u), \ldots, \frac{\partial}{\partial u_p}\xi(u)$ as a $p \times p$ symmetric positive-definite matrix with elements

$$g_{ij}(u) = \left\langle \tfrac{\partial}{\partial u_i}\xi(u), \tfrac{\partial}{\partial u_j}\xi(u) \right\rangle,$$

such that for tangent vectors $\mathbf{v}, \mathbf{w} \in T_{\xi(u)}\mathscr{M}$ expressed in this basis,

$$\langle \mathbf{v}, \mathbf{w} \rangle_{T_{\xi(u)}\mathscr{M}} = \mathbf{v}^\top \mathbf{G}\mathbf{w}.$$

Riemannian metrics allows to define notions such as angles, lengths, and volumes on the manifold. The length of a curve $\gamma : [0,1] \to \mathscr{M}$ is given by

$$\ell(\gamma) = \int_0^1 \sqrt{\langle \gamma'(t), \gamma'(t) \rangle_{T_{\gamma(t)}\mathscr{M}}} dt.$$

A p-dimensional volume element is given by

$$da = |\mathbf{G}|^{1/2} du_1 \dots du_p.$$

Two different embeddings of the manifold resulting in the same Riemannian metric are called *isometric*. Properties and structures that can be expressed solely in terms of the Riemannian metric are called *intrinsic*. Intrinsic structures would play a crucial role in what follows; in particular, in computer graphics applications they would allow to define invariants to isometric deformations of shapes.

2.1.3 Geodesics

Let $\mathbf{x}$ be a smooth tangent vector field on $\mathscr{M}$. Consider a point $m \in \mathscr{M}$, a tangent vector $\mathbf{v} \in T_m\mathscr{M}$ and let $\gamma : (-1,1) \to \mathscr{M}$ be a smooth curve such that $\gamma(0) = m$ and $\gamma'(0) = \mathbf{v}$. The *directional derivative* of the field $\mathbf{x}$ in the direction $\mathbf{v}$ at point m is defined as

$$D_{\mathbf{v}}\mathbf{x}(m) = \lim_{t \to 0} \tfrac{1}{t}(\mathbf{x}(\gamma(t)) - \mathbf{x}(\gamma(0))).$$

Note that the directional derivative $D_{\mathbf{v}}\mathbf{x}$ is not necessarily a tangent vector. Its projection on the tangent space,

$$\nabla_{\mathbf{v}}\mathbf{x}(m) = D_{\mathbf{v}}\mathbf{x}(m) - \langle D_{\mathbf{v}}\mathbf{x}(m), \mathbf{n}(m) \rangle \mathbf{n}(m)$$

is called the *covariant derivative*, and the normal component

$$II_m(\mathbf{v}, \mathbf{w}) = \langle D_{\mathbf{v}}\mathbf{w}(m), \mathbf{n}(m) \rangle$$

the *second fundamental form*. The covariant derivative is intrinsic and can be expressed entirely in terms of the Riemannian metric. $\mathbf{x}$ is said to be *parallel* along a curve γ if $\nabla_{\gamma'(t)}\mathbf{x} = 0$.

A curve γ whose tangent vector is parallel to the curve (*i.e.*, $\nabla_{\gamma'}\gamma' = 0$ along the curve) is called *geodesic*. In physical term, this implies that the acceleration vector γ'' of the curve is always normal to $\mathscr{M}$. For any $m \in \mathscr{M}$ and tangent vector $\mathbf{v} \in T_m\mathscr{M}$, there exists a unique geodesic $\gamma_{m,\mathbf{v}} : (-\epsilon, \epsilon) \to \mathscr{M}$ such that $\gamma_{m,\mathbf{v}}(0) = m$ and $\gamma'_{m,\mathbf{v}}(0) = \mathbf{v}$. It is tacitly assumed to be given in arclength parametrization, such that $\|\gamma'\| = 1$.

The *exponential map* $\exp_m : T_m\mathscr{M} \to \mathscr{M}$ assigns to every tangent vector $\mathbf{v} \in T_m\mathscr{M}$ the point $\exp_m(\mathbf{v}) = \gamma_{m,\mathbf{v}}(1)$. Intuitively, the tangent space can be thought of as a "linearization" of the manifold allowing to locally work with it as with a vectors space, *e.g.*, add or subtract two tangent vectors. The results of such operations can be then mapped to a point on the manifold by means of the exponential map.

2.2 Calculus on Manifolds

We are now interesting in studying functions defined on manifolds. Let $f, h : \mathscr{M} \to \mathbb{R}$ be smooth functions (also referred to as *scalar fields*) on $\mathscr{M}$. We use $\mathscr{F}(\mathscr{M})$ to denote the space of smooth scalar functions on $\mathscr{M}$. We define an inner product between f and h as

$$\langle\!\langle f, h \rangle\!\rangle_{\mathscr{M}} = \int_{\mathscr{M}} f(m)h(m)da(m), \tag{2.1}$$

and denote by $\|f\|_{\mathscr{M}} = |\langle\!\langle f, f \rangle\!\rangle_{\mathscr{M}}|^{1/2}$ the norm induced by the inner product 2.1, and by $L^2(\mathscr{M})$ the space of square-integrable scalar functions (*i.e.*, functions with $\|f\|_{\mathscr{M}} < \infty$).

2.2.1 Intrinsic gradient

Given $f \in \mathscr{F}(\mathscr{M})$, consider the composition $f \circ \exp_m : T_m\mathscr{M} \to \mathbb{R}$. The *intrinsic* or *Riemannian gradient* of f at m is a tangent vector given by

$$\nabla_{\mathscr{M}} f(m) = \nabla(f \circ \exp_m)(\mathbf{0}),$$

where ∇ denotes the standard Euclidean Laplacian operator acting in the tangent space, which is evaluated at the zero tangent vector (origin of $T_m\mathscr{M}$, or in other words, the point m). The intrinsic gradient can intuitively be interpreted as the direction at point m (tangent vector in the tangent space $T_m\mathscr{M}$) in which f changes the most. We can approximate f by first-order Taylor expansion as

$$\nabla(f \circ \exp_m)(\mathbf{v}) \approx f(x) + \langle \nabla_{\mathscr{M}} f(m), \mathbf{v} \rangle_{T_m\mathscr{M}},$$

where the last term is the directional derivative of the scalar field f along $\mathbf{v}$ at point m.

For an embedded manifold $\mathscr{M} \subset \mathbb{R}^d$, we can consider f as a function $f : \mathbb{R}^d \to \mathbb{R}$ restricted to $\mathscr{M}$ and compute the standard (*extrinsic* or *Euclidean*) gradient $\nabla f(m)$, which is a d-dimensional vector. The orthogonal projection of the extrinsic gradient on the tangent space $T_m\mathscr{M}$ coincides with the intrinsic gradient,

$$\nabla_{\mathscr{M}} f(m) = \nabla f(m) - \langle \nabla f(m), \mathbf{n}(m) \rangle \mathbf{n}(m).$$

2.2.2 Laplacian operator

Similarly to the definition of the intrinsic gradient, we can define the *Laplacian* or *Laplace-Beltrami* operator as

$$\Delta_{\mathscr{M}} f(m) = \Delta(f \circ \exp_m)(\mathbf{0}),$$

where, as previously, Δ denotes the usual Euclidean Laplacian applied on $T_m\mathscr{M}$. The Laplacian is related to the intrinsic gradient through the identity

$$\int_{\mathscr{M}} f(m) \Delta_{\mathscr{M}} h(m) da(m) = \int_{\mathscr{M}} \langle \nabla_{\mathscr{M}} f(m), \nabla_{\mathscr{M}} h(m) \rangle_{T_m\mathscr{M}} da(m), \tag{2.2}$$

known as the *Stokes formula*. It follows immediately that the quadratic form $\langle\!\langle f, \Delta_{\mathcal{M}} h \rangle\!\rangle_{\mathcal{M}}$ is symmetric

$$\langle\!\langle f, \Delta_{\mathcal{M}} h \rangle\!\rangle_{\mathcal{M}} = \int_{\mathcal{M}} f(m) \Delta_{\mathcal{M}} h(m) da(m) = \int_{\mathcal{M}} \langle \nabla_{\mathcal{M}} f(m), \nabla_{\mathcal{M}} h(m) \rangle_{T_m \mathcal{M}} da =$$
$$\int_{\mathcal{M}} \langle \nabla_{\mathcal{M}} h(m), \nabla_{\mathcal{M}} f(m) \rangle_{T_m \mathcal{M}} da(m) = \int_{\mathcal{M}} h(m) \Delta_{\mathcal{M}} f(m) da(m) = \langle\!\langle \Delta_{\mathcal{M}} f, h \rangle\!\rangle_{\mathcal{M}},$$

and positive semi-definite,

$$\langle\!\langle f, \Delta_{\mathcal{M}} f \rangle\!\rangle_{\mathcal{M}} = \int_{\mathcal{M}} f(m) \Delta_{\mathcal{M}} f(m) da(m) = \int_{\mathcal{M}} \langle \nabla_{\mathcal{M}} f(m), \nabla_{\mathcal{M}} f(m) \rangle_{T_m \mathcal{M}} da(m) \geqslant 0.$$

The Laplacian is a fundamental construction that will appear prominently in this thesis, so it is worth to devote a few words to its intuitive explanation. Geometrically, the Laplacian measures how a function value at point m differs from its average in an infinitesimal neighborhood of m. In physics, this intuition is intimately related to *Newton's law of cooling*, stating that an object cools at a rate proportional to the difference between its own temperature and the temperature of its surrounding. This is encoded in the *heat equation*, written in the simplest setting as

$$f_t(m,t) = c \Delta_{\mathcal{M}} f(m,t),$$

where $f(m,t)$ denotes the temperature at point m and time t; the lhs is the rate of change (temporal derivative) and the rhs is the local difference between the temperature at a point and and its surrounding (Laplacian). The proportion constant c is called the *thermal diffusivity coefficient*.

2.2.3 Fourier Analysis on Manifolds

On a compact manifold $\mathcal{M}$, the Laplacian operator has a countable set of eigenfunctions and eigenvalues,

$$\Delta_{\mathcal{M}} \phi_i = \lambda_i \phi_i, \tag{2.3}$$

where $0 = \lambda_1 \leqslant \lambda_2 \leqslant \cdots$ are eigenvalues and $\{\phi_i\}_{i\geqslant 1} \subseteq L^2(\mathscr{M})$ are the corresponding eigenfunctions. These eigenfunctions are orthogonal

$$\langle\!\langle \phi_i, \phi_j \rangle\!\rangle_{\mathscr{M}} = \delta_{ij}, \tag{2.4}$$

and form a basis that spans $L^2(\mathscr{M})$. Hence, any function $f \in L^2(\mathscr{M})$ can be expressed as their linear combination referred to as the *Fourier series*,

$$f(m) = \sum_{i\geqslant 1} \langle\!\langle f, \phi_i \rangle\!\rangle_{\mathscr{M}} \phi_i(m), \tag{2.5}$$

where $\langle\!\langle f, \phi_i \rangle\!\rangle_{\mathscr{M}}$ are the Fourier coefficients of f in the basis $\{\phi_i\}_{i\geqslant 1}$.

In computer graphics literature, Laplacian eigenfunctions are often referred to as *manifold harmonics* and are a generalization of Fourier bases to manifolds [Taubin, 1995; Lévy and Zhang, 2009]. The usual Fourier basis e^{inx}, is in fact formed by the eigenfunctions of the 1D Euclidean Laplacian operator: $-\frac{d^2}{dx^2}e^{inx} = -n^2 e^{inx}$.

2.3 Discretization

2.3.1 Discrete manifolds

In practical application, we work with discrete manifolds. The manifold $\mathscr{M}$ is sampled at n points, $\{m_1, \ldots, m_n\}$, upon which an additional structure is build. In this thesis, we will consider mostly two discrete structures, to which we will refer by a somewhat loose term *discrete manifolds*.

The first one is an *undirected weighted graph* $G = (V = \{1, \ldots, n\}, E, \ell)$ with vertices V and edges (unordered pairs of vertices) $E = \{(i, j) : i, j \in V\}$. Note that the graph is a purely topological structure, in which the edges represent the local connectivity of the discrete manifold. Its intrinsic geometry is specified by the weighing function ℓ assigning to each edge some non-negative weight $\ell_{ij} \geqslant 0$, which can be interpreted as the discrete equivalent of the metric. Note that for constructing the intrinsic structures such as the Laplacian we consider in this thesis, this information is sufficient.

Graph-based discretization of manifolds is often encountered in machine learning applications. Typically, what is given is a set of points $\mathbf{x}_1, \dots, \mathbf{x}_n$ in the Euclidean space $\mathbb{R}^d$, which are assumed to be sampled from an embedded manifold $\mathscr{M} \subset \mathbb{R}^d$. The graph is constructed on these points by defining edges between K-nearest neighbors for each point. The discrete metric is simply $\ell_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$.

A discretization typically used in computer graphics is a *triangular mesh* $T = (V = \{1, \dots, n\}, E, F)$, where we have an additional set of triangular faces $F = \{(i,j,k) : i,j,k \in V,\ (i,j),(j,k),(i,k) \in E\}$. We tacitly assume *manifold meshes*, in which each edge is shared by exactly two triangles (one triangle in case of boundary edges). T is again a topological structure; the discrete metric is provided in the form of edge weights ℓ_{ij}, which must additionally satisfy the triangle inequality in each of the mesh faces.

Finally, a scalar function $f : \mathscr{M} \to \mathbb{R}$ is discretized and represented by the n-dimensional vector $\mathbf{f} = [f(m_1), \dots, f(m_n)]^\top$. The space of functions $L^2(\mathscr{M})$ can be thus identified with $\mathbb{R}^n$. An inner product on $L^2(\mathscr{M})$ is discretized as $\mathbf{f}^\top \mathbf{A} \mathbf{g}$, where $\mathbf{A} = \operatorname{diag}[a_1, \dots, a_n]^\top$ and $a_1, \dots, a_n$ are discrete area elements associated with each vertex (typically, assumed $a_i = 1$ on graphs and $a_i = \frac{1}{3} \sum_{i,k:(ijk) \in F} A_{ijk}$ on meshes, where A_{ijk} is the area of the triangular face ijk).

2.3.2 Discrete Laplacians

In the discrete setting, the Laplacian is represented as an $n \times n$ matrix, acting on n-dimensional vectors discretizing function as

$$(\mathbf{L}\mathbf{f})_i = \frac{1}{a_i} \sum_{j:(ij) \in E} \hat{w}_{ij} (f_i - f_j), \tag{2.6}$$

or in matrix notation,

$$\mathbf{L} = \mathbf{A}^{-1}(\mathbf{D} - \hat{\mathbf{W}}) = \mathbf{A}^{-1}\mathbf{W}, \tag{2.7}$$

where $\mathbf{D} = \operatorname{diag}(\sum_{i \neq j} \hat{w}_{ij})$[2]. The particular discretization of the Laplacian amounts to different selection of the weights $\hat{w}_{ij}$. Remarkably, there exists no discretization satisfying all the continuous properties of the Laplacian [Wardetzky et al.,

[2]From now on with w_{ij} we will refer to elements of matrix $\mathbf{W}$, the stiffness matrix.

2008]; therefore, different discretizations that suit a particular application at hand are used. In our case, we are interested in having symmetric weight ($\mathbf{W} = \mathbf{W}^\top$).

In machine learning applications, the weight is usually defined with a Gaussian kernel $w_{ij} = \exp^{-\ell_{ij}^2/2\sigma^2}$ (Figure 2.1, left). In case $\mathbf{A} = \mathbf{I}$, the discretization 2.6 of the Laplacian on a graph with Gaussian edge weights is referred to as *unnormalized graph Laplacian*. The *normalized graph Laplacian* is defined as $\mathbf{L}_n = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$ [von Luxburg, 2007]. Both the normalized and unnormalized Laplacians are symmetric and positive-semidefinite, hence diagonalized by an orthonormal matrix of eigenvectors, $\mathbf{L} = \mathbf{\Phi}\mathbf{\Lambda}\mathbf{\Phi}^\top$, where $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_n]$ is the orthonormal matrix of eigenvectors ($\mathbf{\Phi}^\top\mathbf{\Phi} = \mathbf{I}$), and $\mathbf{\Lambda} = \mathrm{diag}([\lambda_1, \lambda_1, \ldots, \lambda_n])$ is the diagonal matrix of corresponding non-negative eigenvalues.

In computer graphics applications, a popular discretization uses the weight

$$w_{ij} = \frac{-\ell_{ij}^2 + \ell_{jk}^2 + \ell_{ki}^2}{8A_{ijk}} + \frac{-\ell_{ij}^2 + \ell_{jh}^2 + \ell_{hi}^2}{8A_{ijh}}, \tag{2.8}$$

where the triangle areas are expressed in terms of the discrete metric through Heron's formula

$$A_{ijk} = \sqrt{s(s-\ell_{ik})(s-\ell_{kj})(s-\ell_{ij})},$$

and $s = (\ell_{ik} + \ell_{kj} + \ell_{ij})/2$ is the triangle semi-perimeter [Jacobson and Sorkine, 2012]. Though not intuitive at a first glance, this expression is intrinsic (expressed entirely in terms of the metric), as the Laplacian operator is supposed to be. In the case when the metric is induced by the embedding, $\ell_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$, it boils down to the familiar *cotangent formula* [Pinkall and Polthier, 1993; Meyer et al., 2003]:

$$w_{ij} = (\cot(\alpha_{ij}) + \cot(\beta_{ij}))/2, \tag{2.9}$$

with α_{ij}, β_{ij} being two angles opposite the edge (ij) in the two triangles sharing the edge (see Figure 2.1, middle).

2.3.3 Discrete Laplacian eigenvectors

With Laplacian discretizations of form $\mathbf{L} = \mathbf{A}^{-1}\mathbf{W}$, finding eigenvectors boils down to a generalized eigenvalue problem

$$\mathbf{W}\mathbf{\Phi} = \mathbf{A}\mathbf{\Phi}\mathbf{\Lambda}. \tag{2.10}$$

Since $\mathbf{W}$ is symmetric and $\mathbf{A}$ is positive definite, there exist non-negative eigenvalues $\mathbf{\Lambda}$ and $\mathbf{A}$-orthonormal eigenvectors ($\mathbf{\Phi}^\top\mathbf{A}\mathbf{\Phi} = \mathbf{I}$) such that [Parlett, 1998]

$$\mathbf{\Phi}^\top\mathbf{W}\mathbf{\Phi} = \mathbf{\Lambda}. \tag{2.11}$$

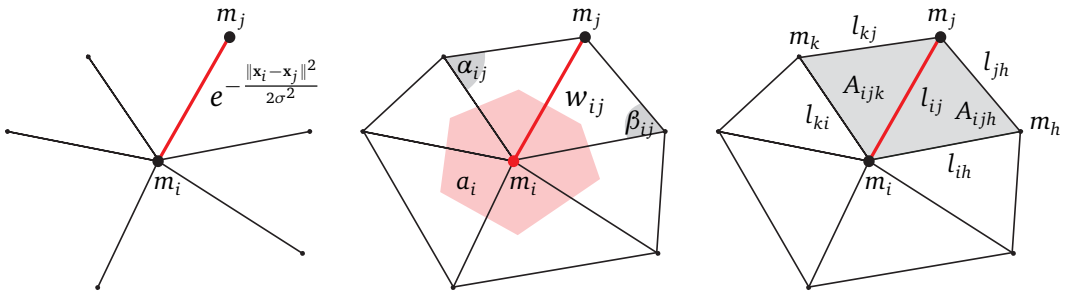


Figure 2.1. Discretization of the Laplace operator. Left: Laplacian on graphs. Middle: cotangent scheme on meshes. Right: Laplacian expressed solely through discrete metric, *e.g.*, edge lengths.

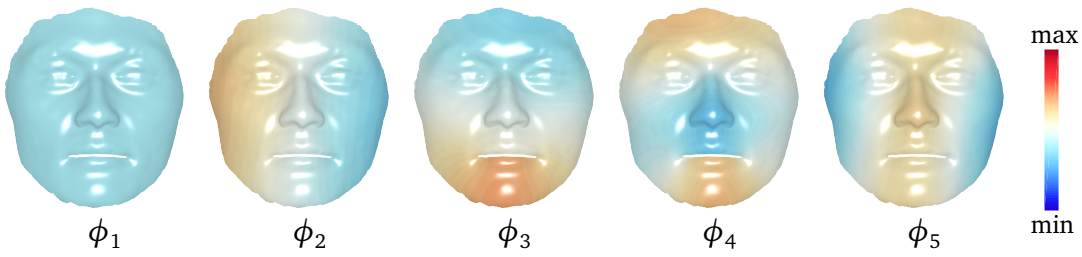


Figure 2.2. Example of Laplace eigenvectors calculated on a two-dimensional manifold (surface). Note, that the first eigenvector is constant.

2.4 Laplacian embeddings

2.4.1 Eigenmaps

Laplacian eigenfunctions are frequently used in machine learning applications as a tool for dimensionality reduction. A common model for complex datasets is that they have low intrinsic dimensionality; in other words, given a set of data points $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ they are assumed to be sampled from a p-dimensional ($p \ll d$) manifold $\mathscr{M} \subset \mathbb{R}^d$; the local structure of the manifold is represented by a graph as described above. The problem of *non-linear dimensionality reduction* is obtaining a p-dimensional parametrization of the dataset representing the structure of the underlying manifold.

Belkin and Niyogi [2002] showed that such a parametrization can be recovered by solving the optimization problem

$$\min_{\boldsymbol{\Phi}} \ \mathrm{tr}(\boldsymbol{\Phi}^\top \mathbf{L} \boldsymbol{\Phi}) \ \ s.t. \ \ \boldsymbol{\Phi}^\top \boldsymbol{\Phi} = \mathbf{I}, \tag{2.12}$$

which provides a neighborhood-preserving k-dimensional embedding of the discrete manifold. Typically, k is chosen to be as large as the intrinsic dimensionality of $\mathscr{M}$. The cost function is the discrete version of the *Dirichlet energy* $\|\nabla_{\mathscr{M}} \boldsymbol{\Phi}\|_{\mathscr{M}}$ measuring the smoothness of a function, and the set of k orthonormal most smooth functions are in fact the first ('low frequency') eigenfunctions of the Laplacian, $\bar{\boldsymbol{\Phi}} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_k]$. Therefore, we can embed efficiently the data using the first k-eigenvectors of the Laplace matrix $\mathbf{L}$. Such an embedding is referred to as *Laplacian eigenmap* [Belkin and Niyogi, 2002]. Note that the constant eigenvectors are usually discarded.

More generally, a *diffusion map* [Coifman et al., 2005; Nadler et al., 2005] is a mapping

$$m \in \mathscr{M} \mapsto \{K_s(\lambda_i)\phi_i(m)\}_{i \geqslant 1}, \tag{2.13}$$

where $K_s(\lambda)$ is some transfer function acting as a "low-pass filter" on eigenvalues λ. Different choices of $K_s(\lambda)$ considered in the literature are summarized in Table 2.1.

$K_s(\lambda)$	Name of the map
1	Laplacian eigenmap [Belkin and Niyogi, 2002]
$e^{-s\lambda}$	Diffusion map [Coifman et al., 2005; Bèrard et al., 1994]
$\frac{1}{\sqrt{\lambda}}$	Global point signature [Rustamov, 2007]

Table 2.1. Different choice of $K_s(\lambda)$ leads to different dimensionality reduction maps.

Figure 2.3. Examples of eigenmaps. Left: original shape, middle: 3D eigenmap. right: 2D eigenmap. Note, the first constant eigenvector was neglected.

2.4.2 Diffusion distances

Coifman et al. [2005] and Nadler et al. [2005] defined a family of *diffusion metric*,

$$d_{\mathrm{d}}^{s}(m,m') = \left(\sum_{i \geqslant 1} K_s^2(\lambda_i)(\phi_i(m) - \phi_i(m'))^2 \right)^{1/2}. \tag{2.14}$$

For $K_s(\lambda) = e^{-s\lambda}$ the metric is called the *heat diffusion distance* and is related to the connectivity of points m and m' on the manifold by means of diffusion process of length s. Such distances are intrinsic, since they are expressed in terms of the Laplacian eigenvectors and eigenvalues, and robust to topological noise. Note that the heat diffusion distance can be well approximated with few first eigenvectors, since the transfer function $K_s(\lambda) \to 0$ for $\lambda \to \infty$.

2.4.3 Spectral clustering

Ng et al. [2001] proposed an efficient and robust clustering algorithm relying on the fact that the number of disconnected components of a graph equals to the multiplicity of the null eigenvalue of its Laplacian. The corresponding eigenvectors act as indicator functions of these components. Basing on this, Ng et al. [2001] proposed to embed the data with these null eigenvectors and apply some standard clustering algorithm such as k-means. Clustering the data in this embedding space produces much better results than clustering the high-dimensional data directly.

2.5 Functional Correspondence

Finding correspondence between two or more shapes is one of the fundamental problems in computer graphics. Traditional approaches consider *point-wise* correspondence, relating a set of points on one manifold (or a discretization thereof) to a set of poinst on another one. Ovsjanikov et al. [2012] proposed considering instead correspondences between the *spaces of functions* defined on the manifolds.

Let $\mathscr{X}$ and $\mathscr{Y}$ be two manifolds, and $t : \mathscr{X} \to \mathscr{Y}$ be a bijective point-wise correspondence between them. The *functional map* $\mathrm{T} : L^2(\mathscr{X}) \to L^2(\mathscr{Y})$ is defined as

$$\mathrm{T}(f) \triangleq f \circ t^{-1}, \tag{2.15}$$

where $f \in L^2(\mathscr{X})$ and $f \circ t^{-1} \in L^2(\mathscr{Y})$. Note that a particular functional map T mapping a delta function at x to a delta function at $t(x)$ gives rise to the standard point-wise correspondence.

One can easily verify that the functional map T is linear:

$$\begin{aligned} \mathrm{T}(\alpha_1 f_1 + \alpha_2 f_2) &= (\alpha_1 f_1 + \alpha_2 f_2) \circ t^{-1} = \\ \alpha_1 f_1 \circ t^{-1} + \alpha_2 f_2 \circ t^{-1} &= \alpha_1 \mathrm{T}(f_1) + \alpha_2 \mathrm{T}(f_2), \end{aligned}$$

for some scalars α_1, α_2 and functions $f_1, f_2 \in L^2(\mathscr{X})$. Thus, given two orthonor-

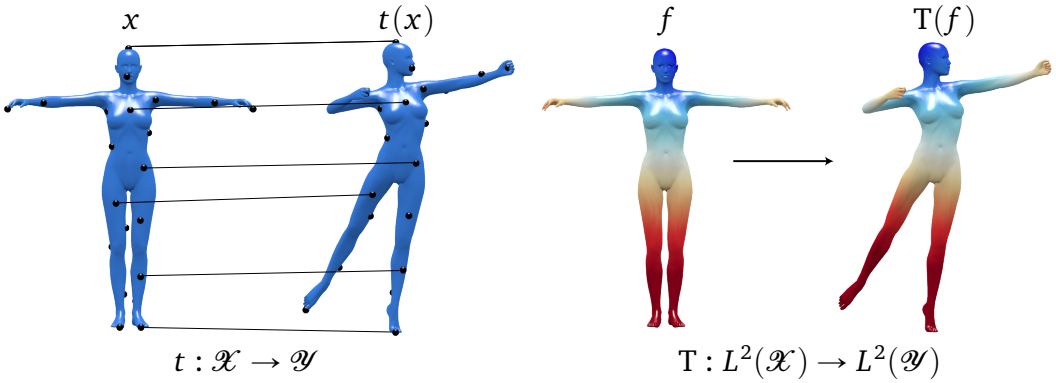


Figure 2.4. Illustration of different correspondence paradigms. Left: the pointwise correspondences t between two surfaces $\mathscr{X}$ and $\mathscr{Y}$; right: the functional correspondence T between two functional spaces $L^2(\mathscr{X})$ and $L^2(\mathscr{Y})$.

mal bases $\{\phi_i\}_{i\geqslant 1}$, $\{\psi_j\}_{j\geqslant 1}$ on $L^2(\mathscr{X})$ and $L^2(\mathscr{Y})$, respectively, we can express T using the Fourier expansion of functions as

$$\begin{aligned}\mathrm{T}(f) &= \mathrm{T}\left(\sum_{i\geqslant 1}\langle\!\langle f,\phi_i\rangle\!\rangle_{\mathscr{X}}\phi_i\right) = \sum_{i\geqslant 1}\langle\!\langle f,\phi_i\rangle\!\rangle_{\mathscr{X}}\mathrm{T}(\phi_i) = \\ &= \sum_{i\geqslant 1}\langle\!\langle f,\phi_i\rangle\!\rangle_{\mathscr{X}}\sum_{j\geqslant 1}\underbrace{\langle\!\langle \mathrm{T}(\phi_i),\psi_j\rangle\!\rangle_{\mathscr{Y}}}_{c_{ij}}\psi_j = \\ &= \sum_{i,j\geqslant 1}\langle\!\langle f,\phi_i\rangle\!\rangle_{\mathscr{X}}c_{ij}\psi_j \approx \sum_{i,j=1}^{k}\langle\!\langle f,\phi_i\rangle\!\rangle_{\mathscr{X}}c_{ij}\psi_j; \end{aligned} \tag{2.16}$$

truncating the series at k first elements allows to represent the correspondence as a $k\times k$ matrix $\mathbf{C} = (c_{ij})$.

In the discrete setting, the manifolds $\mathscr{X}$, $\mathscr{Y}$ are discretized by sampling them at n_X and n_Y points, respectively. The orthonormal bases are represented as $n_X\times n_X$ and $n_Y\times n_Y$ matrices $\mathbf{\Phi} = \left[\boldsymbol{\phi}_1,\dots,\boldsymbol{\phi}_{n_X}\right]$ and $\mathbf{\Psi} = \left[\boldsymbol{\psi}_1,\dots,\boldsymbol{\psi}_{n_Y}\right]$, and functions on $\mathscr{X}$ and $\mathscr{Y}$ are discretized as n_X- and n_Y-dimensional vectors, respectively. We assume inner products are weighted by the diagonal matrices $\mathbf{A}_X$ and $\mathbf{A}_Y$. Formula 2.16 is thus expressed in matrix-vector form as

$$\mathbf{Tf} = \mathbf{\Psi}\mathbf{C}^\top\mathbf{\Phi}\mathbf{A}_X\mathbf{f} \approx \bar{\mathbf{\Psi}}\mathbf{C}_k^\top\bar{\mathbf{\Phi}}\mathbf{A}_X\mathbf{f} = \mathbf{T}_k\mathbf{f},$$

where we denote by $\bar{\mathbf{\Phi}} = [\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_k]$ and $\bar{\mathbf{\Psi}} = [\boldsymbol{\psi}_1, \dots, \boldsymbol{\psi}_k]$ the truncated bases. $\mathbf{T}_k$ can be considered as a rank-k approximation of the matrix $\mathbf{T}$.

Now assume that one is able, given two shapes $\mathscr{X}$ and $\mathscr{Y}$, to provide a set of q corresponding functions, represented as $n_X \times q$ and $n_Y \times q$ matrices $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_q]$ and $\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_q]$ satisfying $\mathbf{TF} \approx \mathbf{G}$ (the question of how to find such functions automatically is a difficult problem on its own and is beyond the scope of this thesis; in the following chapters, we refer to several standard ways of finding such correspondences). Then, one has a linear system w.r.t. $\mathbf{C}$ of kq equations in k^2 variables,

$$\bar{\mathbf{\Psi}}\mathbf{C}_k^\top \bar{\mathbf{\Phi}}\mathbf{A}_X\mathbf{F} = \mathbf{G}, \tag{2.17}$$

or alternatively,

$$\mathbf{G}^\top\mathbf{A}_Y\bar{\mathbf{\Psi}} = \mathbf{F}^\top\mathbf{A}_X\bar{\mathbf{\Phi}}\mathbf{C}. \tag{2.18}$$

Assuming $q \geqslant k$, this system is (over)determined and can be solved in the least-squares sense.

Note that $\mathbf{C}$ depends on the choice of the bases. In particular, when Laplacian eigenfunctions are used, $\mathbf{C}$ has a diagonal structure only if the eigenfunctions behave consistently across $\mathscr{X}$ and $\mathscr{Y}$ (ideally, $\mathrm{T}(\phi_i) = \pm\psi_i$, implying $c_{ij} = \pm\delta_{ij}$). Moreover, if we were able to find some bases $\bar{\mathbf{\Phi}}$, $\bar{\mathbf{\Psi}}$ in which the matrix $\mathbf{C}$ has a diagonal structure, the system of equations 2.18 would have only k variables instead of k^2. Hence, in theory it is sufficient to have *only one* corresponding pair of functions ($q = 1$) to overdetermine 2.18. Therefore, it is important to develop methods of finding consistent bases across shapes, this allows to find a correspondence with less data. This will be one of the main motivations for our constructions in the next chapters.

2.5.1 From Functional to Point-wise Correspondence

There are several ways of deriving a point-wise map from its functional representation. We have already mentioned assigning corresponding points by mapping delta functions with the functional map. In the discrete setting, the mapping of

a delta function δ_{x_i} by T corresponds to taking the ith column $\mathbf{t}^i$ of the matrix $\mathbf{T} = [\mathbf{t}^1, \ldots, \mathbf{t}^{n_X}]$. The most straightforward way of recovering point-wise correspondence is by assigning point x_i to y_j, where

$$j(i) = \underset{j'=1,\ldots,n_Y}{\operatorname{argmax}} \; t_{j'i}.$$

Such a method can be computationally intensive when n_X, n_Y are large, and Ovsjanikov et al. [2012] proposed an efficient alternative. The idea is to map transformed delta functions to the Fourier domain and, employing the Parseval identity (stating the L_2 distance in the frequency domain equals to that in the spatial domain) search for the nearest Fourier coefficients of delta functions on the target shape. This may be done efficiently using advanced proximity search structures, such as kd-trees. With the calculated point-wise map they update the functional map representation $\mathbf{C}$, and repeat the correspondence refinement several times or until convergence, *i.e.* there is no changes in $\mathbf{C}$ (Algorithm 1).

Input a $k \times k$ functional map matrix $\mathbf{C}_0$ from $\mathscr{X}$ to $\mathscr{Y}$ obtained by solving, *e.g.*, Eq. 2.18.

Output point-wise mapping $t : \mathscr{X} \to \mathscr{Y}$, and a refined functional map matrix $\mathbf{C}$.

Initialize $k \leftarrow 1$, $\mathbf{C}^{(1)} \leftarrow \mathbf{C}_0$,
$t(i)^{(1)} = \operatorname{argmin}_j \|\mathbf{A}_X(i,:)\bar{\mathbf{\Phi}}\mathbf{C}^{(1)} - \mathbf{A}_Y(j,:)\bar{\mathbf{\Psi}}\|_F^2$.

repeat
$\mathbf{C}^{(k+1)} = \operatorname{argmin}_{\mathbf{C}} \sum_i \|\mathbf{A}_X(i,:)\bar{\mathbf{\Phi}}\mathbf{C}^{(k)} - \mathbf{A}_Y(t(i)^{(k)},:)\bar{\mathbf{\Psi}}\|_F^2$
$t(i)^{(k+1)} = \operatorname{argmin}_j \|\mathbf{A}_X(i,:)\bar{\mathbf{\Phi}}\mathbf{C}^{(k+1)} - \mathbf{A}_Y(j,:)\bar{\mathbf{\Psi}}\|_F^2$

until *convergence*;

Algorithm 1: ICP-like iterative refinement procedure for calculating a point-wise map [Ovsjanikov et al., 2012]. Note that if a point-wise map between $\mathscr{X}$ and $\mathscr{Y}$ is area-preserving, then $\mathbf{C}^\top\mathbf{C} = \mathbf{I}$, and therefore $\mathbf{C}^{(k+1)}$ is obtained by solving orthogonal Procrustes problem. $\mathbf{M}(i,:)$ denotes the ith row of matrix $\mathbf{M}$.

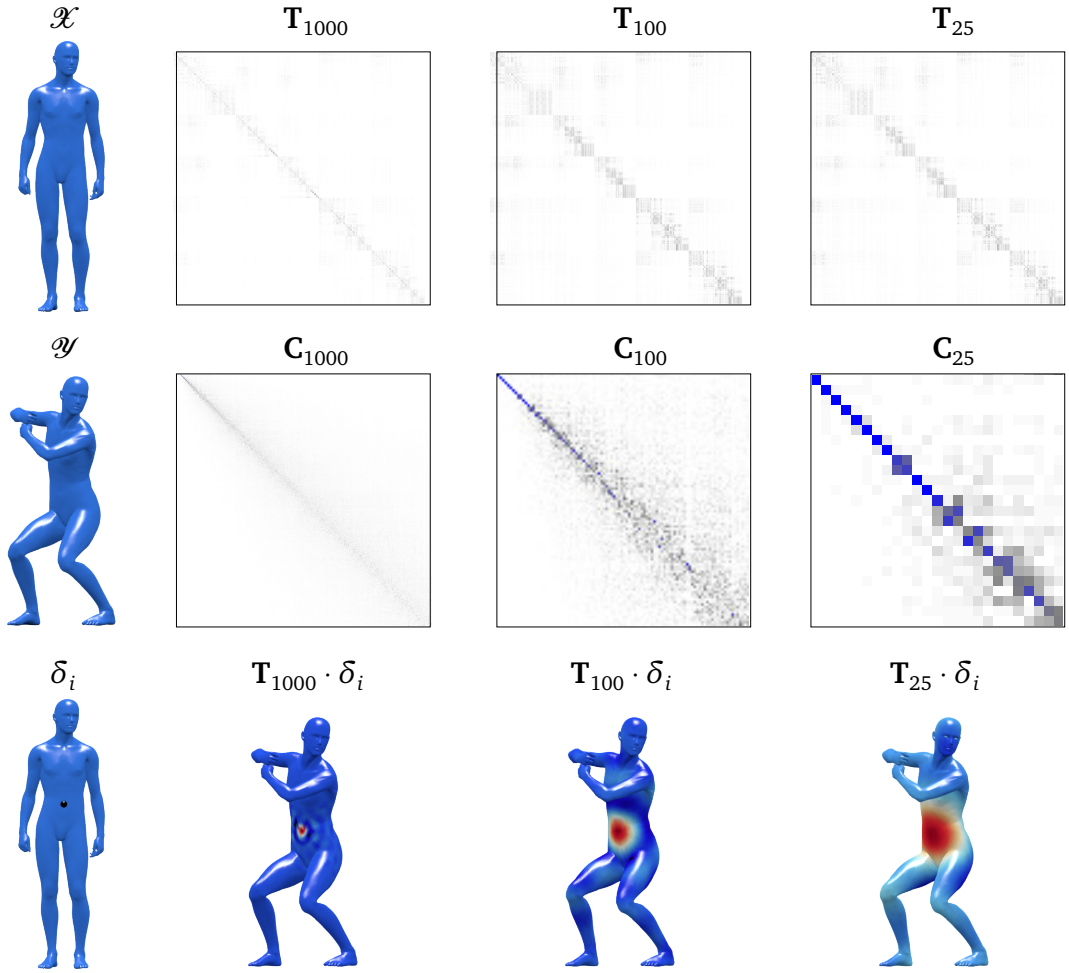


Figure 2.5. Example of rank-k functional map approximations for different k with Laplacian eigenbases between two near-isometric shapes $\mathscr{X}$ and $\mathscr{Y}$. Note that theoretically matrix $\mathbf{C}$ should be diagonal, since Laplacian spectrum is invariant to isometric transformations, however off-diagonal elements become more prominent with increasing rank k. The last row shows the mappings of a delta function at point i on $\mathscr{X}$ by means of different rank-k functional map approximations; note that mappings result in a "blob" function on $\mathscr{Y}$ centered approximately around the ground-truth corresponding point on $\mathscr{Y}$. The blob becomes more localized for larger values of k.

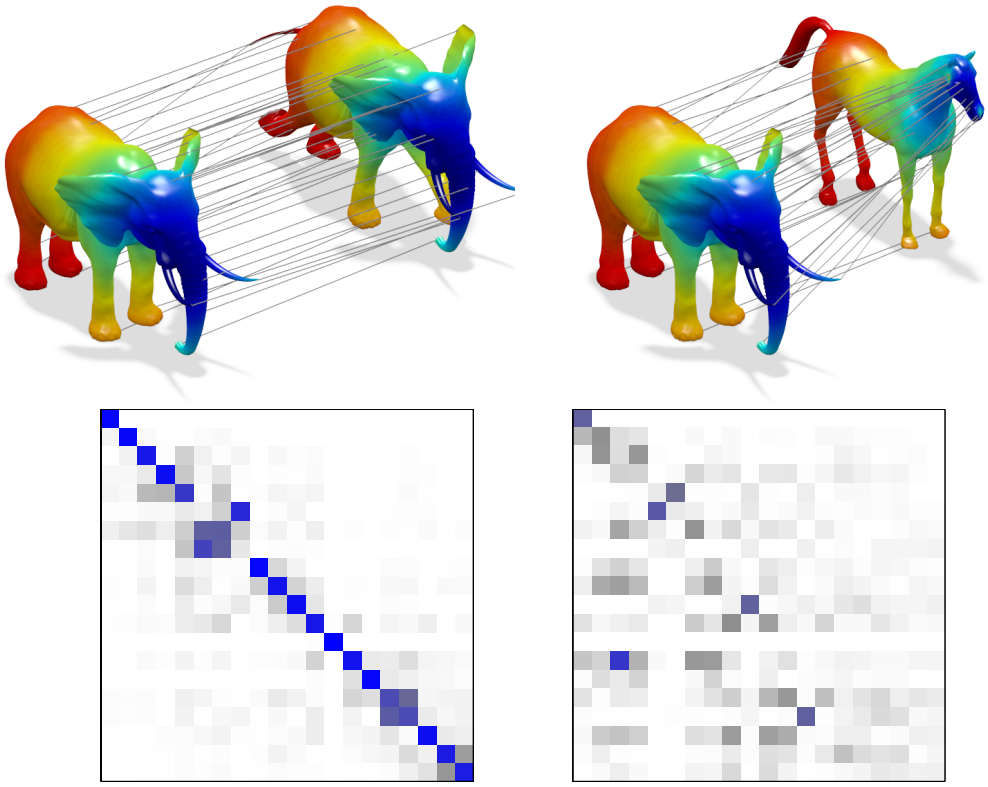


Figure 2.6. Matrix **C** of coefficients expressing a given correspondence between two poses of an elephant (left) and elephant and horse (right) in the Laplacian eigenbases (second row). First row: correspondence between shapes shown with similar colors.

Chapter 3

Joint Diagonalization

In this chapter, we consider the baseline setting of constructing common approximate Laplacian eigenbasis for several domains with known bijective correspondence between them. A typical application example is multi-view clustering, where each data point is observed in two or more modalities. In each modality, the data is modelled as a manifold (often discretized as a graph); our approach relies on the simultaneous diagonalization of the Laplacians built upon the data points in different modalities. This chapter is mainly based on our papers [Eynard et al., 2015; Kovnatsky et al., 2013].

3.1 Joint Approximate Diagonalization

Let us be given p different manifolds $\mathscr{M}_1, \dots, \mathscr{M}_p$. In machine learning applications, one can think of these manifolds as different 'views' of data in different modalities (*e.g.*, images could be characterised by their visual features or by the tags associated to them). The manifolds may have different intrinsic dimensionality and may be embedded into spaces of different dimension. We assume that the manifolds are sampled at n *corresponding points*, and assume that the $n \times n$ symmetric Laplacian matrices $\{\mathbf{L}_i\}_{i=1}^p$ are given.[1] Under the assumption of known correspondence, the rows/columns of the Laplacians are ordered consistently.

[1] In typical applications, data points in each modality are given as feature-vectors, and one constructs the affinity matrix and the Laplacian considering the similarity between these vectors, as described in Section 2.3.

Spectral constructions use the eigenvectors $\mathbf{\Phi}_i = [\boldsymbol{\phi}_{i1}, \ldots, \boldsymbol{\phi}_{in}]$ and eigenvalues $\mathbf{\Lambda}_i = \mathrm{diag}([\lambda_{i1}, \ldots, \lambda_{in}])$ of the Laplacian matrices $\mathbf{L}_i = \mathbf{\Phi}_i \mathbf{\Lambda}_i \mathbf{\Phi}_i^\top$. As mentioned earlier in Section 2.5, even for near-isometric manifolds the Laplacian eigenbases behave inconsistently, and these inconsistencies tend to increase with frequency. Even if the manifolds are exactly isometric (have the same intrinsic structure) and have simple spectrum (the Laplacian eigenvalues have no multiplicity), the corresponding eigenvectors may differ up to a sign. For the case where an eigenvalue has multiplicity greater than one, the ambiguity is a unitary transformation of the eigenvectors corresponding to that eigenvalue. More generally, for non-isometric manifolds (which is usually the case in real applications), the Laplacian eigenvectors can differ dramatically. We illustrate this phenomenon in Figure 3.1 (top), depicting two versions of the Swiss roll surface, discretized as graphs with slightly different connectivity. The Laplacian eigenfunctions are dramatically different.

The key idea of our approach in addressing this problem is to try to find a *joint eigenbasis* by diagonalizing the Laplacians *simultaneously*. If the Laplacians $\{\mathbf{L}_i\}_{i=1}^p$ are symmetric and commute (*i.e.*, $\left[\mathbf{L}_i, \mathbf{L}_j\right] = \mathbf{L}_i\mathbf{L}_j - \mathbf{L}_j\mathbf{L}_i = 0$ for all $i, j = 1, \ldots, p$), they are *jointly diagonalizable*, in the sense that there exist a single set of orthonormal vectors $\tilde{\mathbf{\Phi}} = \left[\tilde{\boldsymbol{\phi}}_1, \ldots, \tilde{\boldsymbol{\phi}}_n\right]$ (referred to as *joint eigenvectors*) diagonalizing all the Laplacians [Horn and Johnson, 1990], *i.e.*,

$$\tilde{\mathbf{\Phi}}^\top \mathbf{L}_i \tilde{\mathbf{\Phi}} = \tilde{\mathbf{\Lambda}}_i = \mathrm{diag}(\left[\tilde{\lambda}_{i1}, \ldots, \tilde{\lambda}_{in}\right]), \quad i, j = 1, \ldots, p.$$

Note that the corresponding eigenvalues $\tilde{\lambda}_{i1}, \ldots, \tilde{\lambda}_{in}$ may be different.

Joint diagonalization allows to remove the incompatibilities between different modalities. Returning to the previous Swiss rolls example, we show in Figure 3.1 (bottom) the joint eigenvectors of the two Laplacians. Joint diagonalization also allows to naturally extend the spectral geometric methods discussed in Chapter 2.2 (eigenmaps, diffusion distances, spectral clustering, etc.) to the multimodal setting by simply replacing the eigenvectors of individual Laplacians $\mathbf{\Phi}_i$ by the joint ones $\tilde{\mathbf{\Phi}}$ obtained from multiple Laplacians.

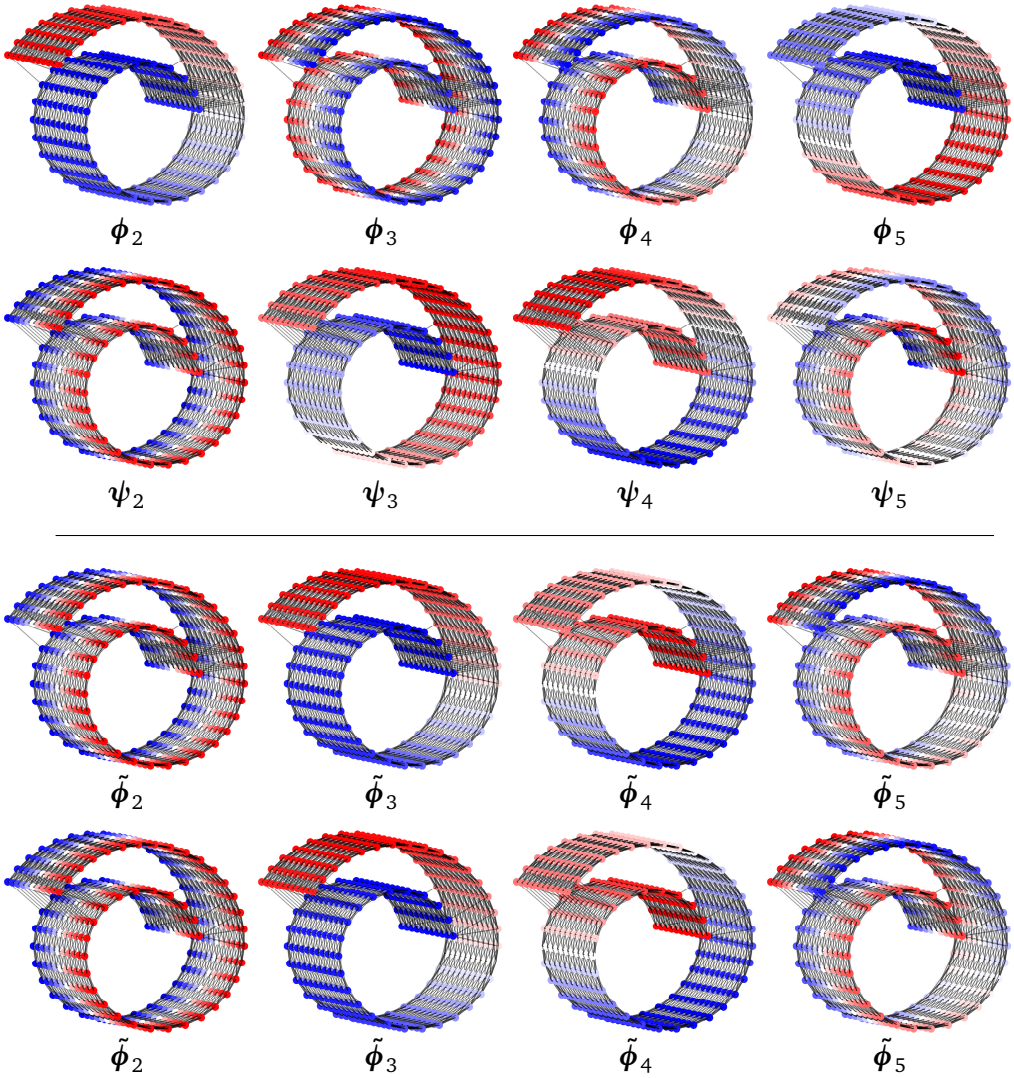


Figure 3.1. Top: the first few Laplacian eigenvectors (denoted by $\boldsymbol{\phi}_i$ and $\boldsymbol{\psi}_j$) of two Swiss rolls (first and second rows) with slightly different connectivity (shown with lines). The difference in the connectivity results in different behaviour of the eigenvectors (*e.g.*, the third and the second eigenvectors are swapped). Bottom: joint approximate eigenvectors $\tilde{\boldsymbol{\phi}}_i$ computed on the same datasets using JADE (3.3) behave in the same way. (Hot colors represent positive values; cold colors represent negative ones).

3.1.1 Jacobi method

Recall that the eigendecomposition of a symmetric $n \times n$ matrix $\mathbf{L}$ can be alternatively formulated as the minimization of the off-diagonal elements of $\boldsymbol{\Phi}^\top \mathbf{L} \boldsymbol{\Phi}$,

where $\mathbf{\Phi}$ is an orthonormal matrix,

$$J(\mathbf{L}) = \min_{\mathbf{\Phi}^\top \mathbf{\Phi} = \mathbf{I}} \operatorname{off}(\mathbf{\Phi}^\top \mathbf{L} \mathbf{\Phi}). \tag{3.1}$$

Here, $\operatorname{off}(\mathbf{M})$ is some off-diagonality criterion, *e.g.*, the sum of squared off-diagonal elements, $\operatorname{off}(\mathbf{M}) = \|\mathbf{M} - \operatorname{Diag}(\mathbf{M})\|_{\mathrm{F}}^2$. For a symmetric matrix $\mathbf{L}$, optimization (3.1) achieves the minimum value $J = 0$, with a minimizer $\mathbf{\Phi}$ being a set of eigenvectors of $\mathbf{L}$.

This type of optimization lies in the heart of a class of eigensolvers based on the *Jacobi iteration* [Jacobi, 1846]. One can observe that if $\mathbf{\Omega}_\theta$,

$$\mathbf{\Omega}_\theta = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & \cos\theta & \dots & -\sin\theta & & \\ & & \vdots & & \vdots & & \\ & & \sin\theta & \dots & \cos\theta & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix}, \tag{3.2}$$

is a rotation matrix by angle θ in the plane ij, then $\mathbf{L}' = \mathbf{\Omega}_\theta^\top \mathbf{L} \mathbf{\Omega}_\theta$ is similar to $\mathbf{L}$ (*i.e.*, both matrices have the same eigenvalues) and $\|\mathbf{L}'\|_{\mathrm{F}} = \|\mathbf{L}\|_{\mathrm{F}}$. However, we can choose $\theta = \tan^{-1}(\frac{2l_{ij}}{l_{jj} - l_{ii}})$ which results in $\mathbf{L}'_{ij} = 0$. This way, we find the angle θ minimizing $\operatorname{off}(\mathbf{\Omega}_\theta^\top \mathbf{L} \mathbf{\Omega}_\theta)$ which means that by applying such rotation $\mathbf{\Omega}_\theta$ we reduce the off-diagonal elements of $\mathbf{L}$. Also note that since $\mathbf{L}'$ has only the ith and jth rows and columns different from $\mathbf{L}$, the rotation can be applied "in-place" and does not require matrix multiplication.

The idea of the Jacobi method for eigendecomposition is to construct $\mathbf{\Phi}$ as a sequence of plane rotations $\mathbf{\Phi} = \cdots \mathbf{\Omega}_2 \mathbf{\Omega}_1$ in order to sequentially minimize the off-diagonal elements in (3.1). Being a product of rotations, the matrix $\mathbf{\Phi}$ is orthonormal by construction.

3.1.2 Generalized Jacobi method

The same idea can be extended to finding the approximate joint eigenvectors of several matrices [Bunse-Gerstner et al., 1993; Cardoso and Souloumiac, 1993, 1996]. We try to find a common orthonormal matrix $\tilde{\mathbf{\Phi}}$ by solving the optimization problem

$$J(\mathbf{L}_1, \dots, \mathbf{L}_p) = \min_{\tilde{\mathbf{\Phi}}^\top \tilde{\mathbf{\Phi}} = \mathbf{I}} \sum_{i=1}^{p} \operatorname{off}(\tilde{\mathbf{\Phi}}^\top \mathbf{L}_i \tilde{\mathbf{\Phi}}). \tag{3.3}$$

In this case, $J = 0$ is achieved only if the matrices $\mathbf{L}_1, \dots, \mathbf{L}_p$ commute. In general, we will have $J > 0$, *i.e.* $\tilde{\mathbf{\Phi}}$ are only *approximate* eigenvectors of $\mathbf{L}_1, \dots, \mathbf{L}_p$.

Cardoso and Souloumiac [1996] proposed the *generalized Jacobi method* (referred to as JADE), which follows the standard Jacobi method described above, with the difference that now in each step the rotations are applied to reduce the off-diagonality criterion $\sum_{i=1}^{p} \operatorname{off}(\mathbf{\Omega}_\theta^\top \mathbf{L}_i \mathbf{\Omega}_\theta)$ for multiple matrices rather than $\operatorname{off}(\mathbf{\Omega}_\theta^\top \mathbf{L} \mathbf{\Omega}_\theta)$.

Cardoso and Souloumiac [1996] show that for a rotation matrix $\mathbf{\Omega}_\theta$ the rotation angle θ minimizing $\sum_{i=1}^{p} \operatorname{off}(\mathbf{\Omega}_\theta^\top \mathbf{L}_i \mathbf{\Omega}_\theta)$ can be expressed in closed form, as follows: Let $\mathbf{M} = \sum_{i=1}^{p} h(\mathbf{L}_i) h(\mathbf{L}_i)^\top$, where $h(\mathbf{L}) = (l_{ii} - l_{jj}, l_{ij} + l_{ji})^\top$, and let $\alpha = m_{11} - m_{22}$ and $\beta = m_{12} + m_{22}$. Then, $\theta = \frac{1}{2} \tan^{-1}(\beta / (\alpha + \sqrt{\alpha^2 + \beta^2}))$. The complexity of this approach is akin to that of the standard Jacobi iteration.

We should note that there exist alternative algorithms for the approximate joint diagonalization problem, based on the idea of minimizing a suitable cost function on the Stiefel manifold of orthonormal matrices [Rahbar and Reilly, 2000; Ma and Lee, 2008; Cai et al., 2011; Dong et al., 2014] or by penalizing the deviation of the joint eigenvectors from the eigenvectors of the operators on each modality [Kumar et al., 2011; Yeredor, 2002]. The analysis of these methods is beyond the scope of this thesis.

3.2 Drawbacks

We should point to several drawbacks of the joint approximate diagonalization model, making it impractical to perform multimodal spectral analysis in many applications. These issues will be addressed in Chapter 4, where we propose a

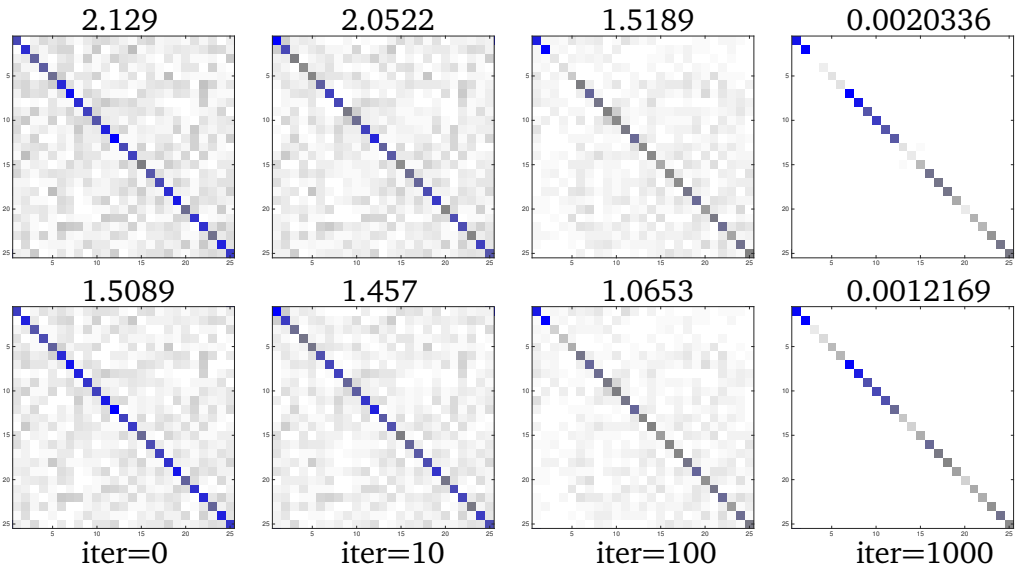


Figure 3.2. Illustration of generalized Jacobi iterations for approximate joint diagonalization of two matrices. The numbers on top show the sum of squared off-diagonal elements.

new approach for *coupled diagonalization* generalizing the joint diagonalization problem.

First, the problem assumes that we are given Laplacian matrices with ordered rows/columns, arising, in turn, from manifolds sampled at an equal number of points n with known point-wise bijective correspondence. While a reasonable assumption in many machine learning problems such as multimodal clustering discussed in Chapter 6 (wherein corresponding points on different manifolds represent the same object seen in different modalities, such as an image and its text annotation), the knowledge of bijective correspondence is way too restrictive in computer graphics applications treated in Chapter 7, where in fact we will use simultaneous diagonalization to *find* correspondence between shapes.

Second, the generalized Jacobi method calculates the *whole* approximate joint eigenbasis of the Laplacians. Since in most spectral analysis applications we are interested only in the *first* $k \ll n$ eigenvectors, such a computation can be wasteful.

Finally, the assumption of orthogonality of $\tilde{\mathbf{\Phi}}$ tacitly implies that the Laplacian matrices are symmetric, which may rule out several non-symmetric Lapla-

cian constructions. In particular, the cotangent formula mentioned in Section 2.3 and similar Laplacian discretization used in computer graphics applications are of the form $\mathbf{L} = \mathbf{A}^{-1}\mathbf{W}$, where $\mathbf{A}$ is the mass matrix and $\mathbf{W}$ is the stiffness matrix. Typically, $\mathbf{A}$ is diagonal and its elements are local area elements of the discretized manifold, which can be interpreted as an inner product, w.r.t. which the eigenvectors are orthonormal,

$$\boldsymbol{\Phi}^\top \mathbf{W} \boldsymbol{\Phi} = \boldsymbol{\Lambda}, \quad s.t. \;\; \boldsymbol{\Phi}^\top \mathbf{A}_i \boldsymbol{\Phi} = \mathbf{I}. \tag{3.4}$$

The problem can be renormalized by defining $\boldsymbol{\Psi} = \mathbf{A}^{\frac{1}{2}}\boldsymbol{\Phi}$,

$$\boldsymbol{\Psi}^\top \bar{\mathbf{W}} \boldsymbol{\Phi} = \boldsymbol{\Lambda}, \quad s.t. \;\; \boldsymbol{\Psi}^\top \boldsymbol{\Psi} = \mathbf{I}; \tag{3.5}$$

note than now $\bar{\mathbf{W}} = \mathbf{A}^{-\frac{1}{2}}\mathbf{W}\mathbf{A}^{-\frac{1}{2}}$ is symmetric, and $\boldsymbol{\Psi}$ are orthonormal.

3.3 Theoretical results on approximate joint diagonalization

We conclude this chapter by stating two theoretical results that will be used next. First, the joint approximate eigenvectors obtained as the solution of problem (3.3) are related to the eigenvectors of the Laplacian matrices by the following

Theorem 1 *Let $\mathbf{L}_1 = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^\top$ have a simple τ-separated spectrum ($|\lambda_i - \lambda_j| \geqslant \tau$ for all $i \neq j$), and let $\mathbf{L}_2 = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^\top + \epsilon\delta\mathbf{M}$ be a perturbation of $\mathbf{L}_1$. Then (ignoring permutation of eigenfunctions and sign flips), the joint approximate eigenbasis can be written as the first-order perturbation*

$$\tilde{\boldsymbol{\phi}}_i \;\; = \;\; \boldsymbol{\phi}_i + \epsilon \sum_{j \neq i} \alpha_{ij} \boldsymbol{\phi}_j + \mathcal{O}(\epsilon^2), \tag{3.6}$$

where $\alpha_{ij} = \boldsymbol{\phi}_i^\top \delta\mathbf{M}\boldsymbol{\phi}_j / 2(\lambda_j - \lambda_i)$.

For a proof, we refer the reader to [Cardoso, 1995]. In particular, this theorem implies that if two Laplacians are approximately jointly diagonalizable, the sub-

spaces spanned by the first few eigenvectors of one Laplacian and their first few joint approximate eigenvectors are close. This result will be used in Chapter 4 for the subspace formulation of the coupled diagonalization problem, allowing an efficient computation of a few joint approximate eigenvectors with complexity independent of the sampling size n.

Second, we notice that joint diagonalizability of matrices is intimately related to their commutativity. As mentioned previously, two symmetric matrices $\mathbf{M}_1$ and $\mathbf{M}_2$ are jointly diagonalizable iff they commute, *i.e.*, $[\mathbf{M}_1, \mathbf{M}_2] = \mathbf{M}_1\mathbf{M}_2 - \mathbf{M}_2\mathbf{M}_1 = 0$ [Horn and Johnson, 1990]. It appears that this relation also holds for almost commuting matrices with $\| [\mathbf{M}1, \mathbf{M}2] \|_{\mathrm{F}} \leqslant \epsilon$, in the sense that almost commuting matrices are almost jointly diagonalizable and vice versa [Glashoff and Bronstein, 2013]. This statement can be formalized at the following

Theorem 2 *There exist functions $\epsilon_1(s), \epsilon_2(s)$ satisfying $\lim_{s\to 0} \epsilon_i(s) = 0$, $i = 1, 2$, such that for any two $n \times n$ symmetric matrices $\mathbf{M}_1$, $\mathbf{M}_2$ with $\|\mathbf{M}_1\|_{\mathrm{F}} = \|\mathbf{M}_2\|_{\mathrm{F}} = 1$,*

$$\epsilon_1(\| [\mathbf{M}_1, \mathbf{M}_2] \|_{\mathrm{F}}) \leqslant J(\mathbf{M}_1, \mathbf{M}_2) \leqslant n\epsilon_2(\| [\mathbf{M}_1, \mathbf{M}_2] \|_{\mathrm{F}}).$$

The term $J(\mathbf{M}_1, \mathbf{M}_2)$ measures the smallest possible sum of squared off-diagonal elements, *i.e.*, how jointly diagonalizable the matrices $\mathbf{M}_1$ and $\mathbf{M}_2$ are. For proof, we refer the reader to [Glashoff and Bronstein, 2013].

By virtue of this result, the commutator norm $\| [\mathbf{M}_1, \mathbf{M}_2] \|_{\mathrm{F}}$ can be used as a measure of joint diagonalizability of matrices without the need to find explicitly the joint bases. Since joint diagonalizability is a similarity measure between manifolds (two manifolds/shapes are intrinsically similar if their Laplacians $\mathbf{L}_1, \mathbf{L}_2$ are jointly diagonalizable), this in turn implies that the commutator can be used as a measure of intrinsic similarity. We will exploit this fact in image processing applications discussed in Chapter 8.

Chapter 4

Coupled Diagonalization

A notable drawback of the joint diagonalization problem discussed in Chapter 3 is the full coupling assumption, requiring an equal number of data points in all the modalities and bijective correspondence between them. In many settings, this assumption could be too restrictive: for example, establishing the correspondence between images and text annotations requires some human intelligence (tagging the images), and establishing correspondence between shapes is one of the cornerstone problems in computer graphics. In this chapter, we introduce *approximate coupled diagonalization*, which is a generalization of approximate joint diagonalization. This chapter is mainly based on our EUROGRAPHICS [Kovnatsky et al., 2013] and PAMI [Eynard et al., 2015] papers. The robust formulation is presented according to our technical report [Kovnatsky et al., 2016].

4.1 Coupled approximate diagonalization

As in the previously discussed joint diagonalization case, we model the data-modalities as a set of p manifolds, $\mathscr{M}_1, \dots, \mathscr{M}_p$, which may have different structure and dimensionality. Unlike the previous setting, we assume that the manifolds are sampled at a possibly *different* number of points, denoted by $n_1, \dots, n_p$ (in joint diagonalization, we assumed $n_1 = \dots = n_p = n$). We denote the Laplacian matrices constructed on these discretized manifolds by $\{\mathbf{L}_i = \mathbf{A}_i^{-1}\mathbf{W}_i \in \mathbb{R}^{n_i \times n_i}\}_{i=1}^p$. We are looking for a set of *coupled bases* $\{\hat{\mathbf{\Phi}}_i \in \mathbb{R}^{n_i \times k}\}_{i=1}^p$, where by

coupling we intend that Fourier coefficients of corresponding functions in these bases are approximately equal (or figuratively, "speak the same language"), as formalized in the following.

Note that in this setting, we can allow for different dimensions ($n_i \neq n_j$) and also find only the first k eigenvectors ($k \leqslant n_i$). Furthermore, we replace the assumption of bijective point-wise correspondence by a weaker assumption of knowing a set of corresponding functions on $\mathscr{M}_1, \dots, \mathscr{M}_p$. Denoting by $\mathbf{F}_{ij} = \left[\mathbf{f}_{ij,1} \dots, \mathbf{f}_{ij,q_{ij}}\right]$ and $\mathbf{G}_{ij} = \left[\mathbf{g}_{ij,1} \dots, \mathbf{g}_{ij,q_{ij}}\right]$ the $n_i \times q_{ij}$ and $n_j \times q_{ij}$ matrices of corresponding discretized functions on $\mathscr{M}_i$ and $\mathscr{M}_j$, respectively, coupling is understood in the sense that

$$\hat{\mathbf{\Phi}}_i^\top \mathbf{A}_i \mathbf{F}_{ij} \approx \hat{\mathbf{\Phi}}_j^\top \mathbf{A}_j \mathbf{G}_{ij}.$$

Corresponding vectors can be delta functions (representing sparse point-wise correspondence between the manifolds), blobs or stable regions, distance functions, dense descriptors, *etc*. (see Figure 4.1). We will discuss specific choices in the next chapters. Note that this formulation allows both the number q_{ij}, q_{il} of corresponding vectors between $\mathscr{M}_i$, $\mathscr{M}_j$ and $\mathscr{M}_i$, $\mathscr{M}_l$ and the vectors themselves to be different.

Similarly to coupling the Fourier coefficients of some known corresponding functions, we can *decouple* the Fourier coefficients of some known non-corresponding functions, represented by matrices $\mathbf{E}_{ij} = \left[\mathbf{e}_{ij,1} \dots, \mathbf{e}_{ij,q'_{ij}}\right]$ and $\mathbf{H}_{ij} = \left[\mathbf{h}_{ij,1} \dots, \mathbf{h}_{ij,q'_{ij}}\right]$ of size $n_i \times q'_{ij}$ and $n_j \times q'_{ij}$, respectively. Decoupling is understood in the sense

$$\hat{\mathbf{\Phi}}_i^\top \mathbf{A}_i \mathbf{E}_{ij} \neq \hat{\mathbf{\Phi}}_j^\top \mathbf{A}_j \mathbf{H}_{ij}.$$

Our *coupled diagonalization problem* is formulated as follows,

$$\begin{aligned} \min_{\hat{\mathbf{\Phi}}_i^\top \mathbf{A}_i \hat{\mathbf{\Phi}}_i = \mathbf{I}} \quad & \sum_{i=1}^{p} \operatorname{off}(\hat{\mathbf{\Phi}}_i^\top \mathbf{W}_i \hat{\mathbf{\Phi}}_i) \quad + \quad \mu_{\mathrm{c}} \sum_{i,j=1}^{p} \|\mathbf{F}_{ij}^\top \mathbf{A}_i \hat{\mathbf{\Phi}}_i - \mathbf{G}_{ij}^\top \mathbf{A}_j \hat{\mathbf{\Phi}}_j\|_{\mathrm{F}}^2 \\ & \qquad - \quad \mu_{\mathrm{d}} \sum_{i,j=1}^{p} \|\mathbf{E}_{ij}^\top \mathbf{A}_i \hat{\mathbf{\Phi}}_i - \mathbf{H}_{ij}^\top \mathbf{A}_j \hat{\mathbf{\Phi}}_j\|_{\mathrm{F}}^2. \end{aligned} \tag{4.1}$$

The parameters μ_{c}, μ_{d} determine the coupling and decoupling strengths, respec-

tively. For $\mu_c = \mu_d = 0$, the problem becomes uncoupled and boils down to individual diagonalization of the respective Laplacians. We define the *coupled eigenvalues* as the diagonal of $\hat{\mathbf{\Lambda}}_i = \operatorname{diag}\left[\hat{\lambda}_{i,1}, \ldots, \hat{\lambda}_{i,k}\right] = \operatorname{Diag}(\hat{\mathbf{\Phi}}_i^\top \mathbf{W}_i \hat{\mathbf{\Phi}}_i)$.

Note that in the case $n_i = k = n = q$, $\mathbf{F}_{ij} = \mathbf{G}_{ij} = \mathbf{I}$, $\mu_c \to \infty$ and $\mu_d = 0$, the coupled diagonalization problem 4.1 boils down to the joint diagonalization problem 3.1. Therefore, our problem is a generalization of the joint diagonalization approach discussed before. As opposed to the joint diagonalization problem where we need to find a single common basis (giving rise to n^2 optimization variables problem 3.1), in coupled diagonalization we are looking for p different $n_i \times k$ matrices. The number of variables in problem 4.1 is $\sum_{i=1}^{p} n_i k$.

4.1.1 Subspace parametrization

In order to reduce the number of variables in our problem, we parametrize the k vectors in the ith coupled basis as a linear combination of $k' \geqslant k$ eigenvectors $\bar{\mathbf{\Phi}}_i = \left[\boldsymbol{\phi}_{i,1}, \ldots, \boldsymbol{\phi}_{i,k'}\right]$ of the Laplacian $\mathbf{L}_i$. Denoting by $\mathbf{R}_i$ the $k' \times k$ matrix of linear combination coefficients, we thus have

$$\hat{\mathbf{\Phi}}_i = \bar{\mathbf{\Phi}}_i \mathbf{R}_i.$$

We refer to this approximation as *subspace parametrization* of the coupled bases. When $k' = n$, it is exact and amounts just to a change of coordinates. From the $\mathbf{A}_i$-orthogonality of $\bar{\mathbf{\Phi}}_i$, it follows that $\mathbf{R}_i^\top \bar{\mathbf{\Phi}}_i^\top \mathbf{A}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i = \mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}$. Theorem 1 states that for approximately jointly diagonalizable Laplacians, $\operatorname{span}\{\hat{\boldsymbol{\phi}}_{i,1}, \ldots, \hat{\boldsymbol{\phi}}_{i,k}\} \approx \operatorname{span}\{\boldsymbol{\phi}_{i,1}, \ldots, \boldsymbol{\phi}_{i,k}\}$. Therefore, this approximation is good when the Laplacians are approximately jointly diagonalizable.

Plugging this subspace parameterization into 4.1 and observing that $\hat{\mathbf{\Phi}}_i^\top \mathbf{W}_i \hat{\mathbf{\Phi}}_i = \mathbf{R}_i^\top \bar{\mathbf{\Phi}}_i^\top \mathbf{W}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i = \mathbf{R}_i^\top \bar{\mathbf{\Lambda}}_i \mathbf{R}_i$, where $\bar{\mathbf{\Lambda}}_i = \operatorname{diag}(\left[\lambda_{i,1}, \ldots, \lambda_{i,k'}\right])$ is the diagonal matrix containing the first k' eigenvalues of $\mathbf{L}_i$, we get a problem with pkk' variables,

$$\begin{aligned} \min_{\mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}} \sum_{i=1}^{p} \operatorname{off}(\mathbf{R}_i^\top \bar{\mathbf{\Lambda}}_i \mathbf{R}_i) \quad &+ \quad \mu_c \sum_{i,j=1}^{p} \|\mathbf{F}_{ij}^\top \mathbf{A}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i - \mathbf{G}_{ij}^\top \mathbf{A}_j \bar{\mathbf{\Phi}}_j \mathbf{R}_j\|_F^2 \\ &- \quad \mu_d \sum_{i,j=1}^{p} \|\mathbf{E}_{ij}^\top \mathbf{A}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i - \mathbf{H}_{ij}^\top \mathbf{A}_j \bar{\mathbf{\Phi}}_j \mathbf{R}_j\|_F^2. \end{aligned} \tag{4.2}$$

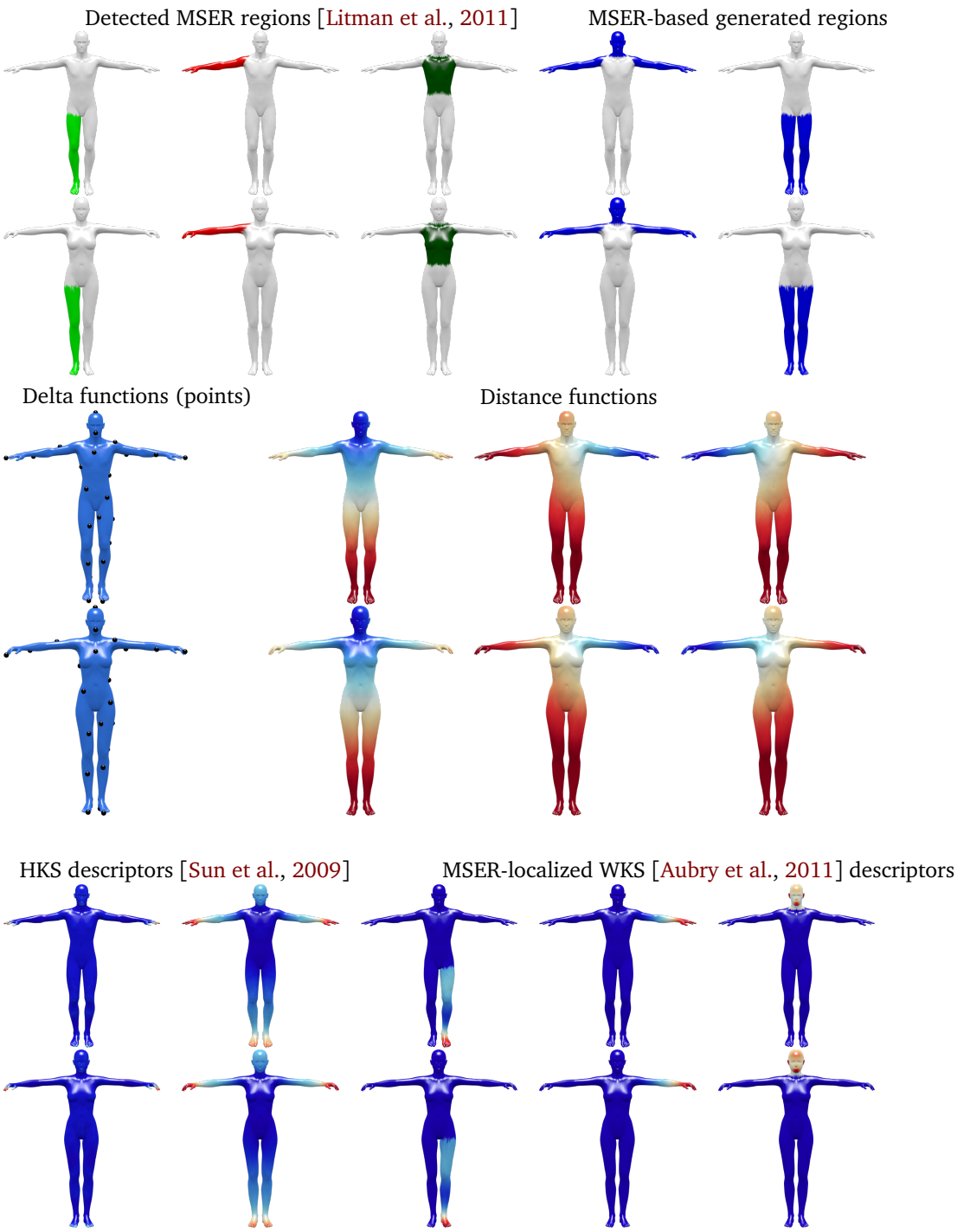


Figure 4.1. Illustration of possible corresponding functions used for coupled diagonalization in computer graphics applications.

The geometric interpretation of 4.2 is an alignment of the Laplacian eigenbases $\bar{\mathbf{\Phi}}_i$ by means of orthonormal matrices $\mathbf{R}_i$, where the quality of the alignment is determined by the coupling and decoupling terms.

The use of the subspace parameterization offers several advantages. Since the Laplacians do not appear explicitly in problem 4.2 but rather their discretized eigenfunctions and eigenvalues, we can employ any method for Laplacian discretization. Also, unlike problems 3.1 and 4.1, the complexity of 4.2 depends only on k, k' and not on n, thus making our method *independent of the Laplacian size* (assuming $\bar{\mathbf{\Phi}}_i$ are precomputed). Since typical values are $n \sim 10^3 - 10^5$, while $k', k \sim 10 - 100$, the reduction of the number of variables can be of several orders of magnitude. Finally, as we represent our coupled basis vectors as linear combinations of the first k' low-frequency eigenvectors, the coupled basis vectors have guaranteed smooth behavior which ensures neighborhood-preservation property typical of Laplacian embeddings.

In applications requiring the computation of many approximate joint eigenvectors ($k \gg 1$), the complexity of optimizing over p large orthonormal matrices might become prohibitively high. In this case, we can split the eigenvectors into non-overlapping bands of size $k'' > 1$ and solve k/k'' problems (4.3) or $k'' \times k''$ matrices $\mathbf{R}_i$ (here we assume that k is a multiple of k'').

4.1.2 Choice of the off-diagonal penalty

An essential component of our problem is the regularization term of the form $\operatorname{off}(\mathbf{R}_i^\top \bar{\mathbf{\Lambda}}_i \mathbf{R}_i)$ or $\operatorname{off}(\hat{\mathbf{\Phi}}_i^\top \mathbf{W}_i \hat{\mathbf{\Phi}}_i)$, ensuring that the coupled bases behave as approximate Laplacian eigenbases. In the joint diagonalization problem, following Cardoso and Souloumiac [1996], we used the term $\operatorname{off}_1(\mathbf{M}) = \|\mathbf{M} - \operatorname{Diag}(\mathbf{M})\|_{\mathrm{F}}^2$ penalizing for off-diagonal elements, with the assumption that an eigenbasis diagonalizes the Laplacian matrix. Note that such a penalty do not impose any *ordering* of the eigenvectors, or in other words, it is invariant to the permutation of the columns of the basis matrix.

In the coupled diagonalization, we are interested in the *first* k eigenvectors, therefore, their order is important. Two possible alternative penalties in this setting are $\operatorname{off}_2(\mathbf{M}) = \operatorname{tr}(\mathbf{M})$ and $\operatorname{off}_3(\mathbf{X}, \mathbf{M}) = \|\mathbf{X}^\top \mathbf{M} \mathbf{X} - \mathbf{\Lambda}_{\mathbf{M}}\|_{\mathrm{F}}^2$ where $\mathbf{M}$ is an input

matrix that we want to diagonalize with the columns of matrix $\mathbf{X}$, and $\mathbf{\Lambda}_\mathbf{M}$ is the diagonal matrix of ordered original eigenvalues of $\mathbf{M}$. Using the latter penalty, we can rewrite our subspace version of the coupled diagonalization problem as

$$\begin{aligned}\min_{\mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}} \sum_{i=1}^{p} \|\mathbf{R}_i^\top \bar{\mathbf{\Lambda}}_i \mathbf{R}_i - \bar{\mathbf{\Lambda}}_i\|_\mathrm{F}^2 \;+\; \mu_\mathrm{c} \sum_{i,j=1}^{p} \|\mathbf{F}_{ij}^\top \mathbf{A}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i - \mathbf{G}_{ij}^\top \mathbf{A}_j \bar{\mathbf{\Phi}}_j \mathbf{R}_j\|_\mathrm{F}^2 \\ -\; \mu_\mathrm{d} \sum_{i,j=1}^{p} \|\mathbf{E}_{ij}^\top \mathbf{A}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i - \mathbf{H}_{ij}^\top \mathbf{A}_j \bar{\mathbf{\Phi}}_j \mathbf{R}_j\|_\mathrm{F}^2.\end{aligned} \tag{4.3}$$

4.2 Numerical implementation

Our coupled diagonalization problem and its subspace version are instances of *manifold-* or *manifold-constrained* optimization, where the variables (matrices $\mathbf{R}_1, \ldots, \mathbf{R}_p$) are restricted to the *Stiefel manifold* of orthonormal matrices. While the solution of such problems can be carried out using standard constrained optimization techniques such as `fmincon` in MATLAB, there are more efficient *manifold optimization* techniques specifically developed for such problems, which we discuss in Chapter 5.

Derivatives. The gradients of the off-diagonal penalties are given by

$$\nabla_\mathbf{X} \mathrm{off}_1(\mathbf{X}, \mathbf{M}) = \nabla_\mathbf{X} \|\mathbf{X}^\top \mathbf{M} \mathbf{X} - \mathrm{Diag}(\mathbf{X}^\top \mathbf{M} \mathbf{X})\|_\mathrm{F}^2 = 4(\mathbf{M}\mathbf{X}\mathbf{X}^\top \mathbf{M}\mathbf{X} - \mathbf{M}\mathbf{X}\,\mathrm{Diag}(\mathbf{X}^\top \mathbf{M} \mathbf{X})),$$

$$\nabla_\mathbf{X} \mathrm{off}_2(\mathbf{X}, \mathbf{M}) = \nabla_\mathbf{X} \,\mathrm{tr}(\mathbf{X}^\top \mathbf{M} \mathbf{X}) = 2\mathbf{M}\mathbf{X},$$

$$\nabla_\mathbf{X} \mathrm{off}_3(\mathbf{X}, \mathbf{M}, \Lambda_\mathbf{M}) = \|\mathbf{X}^\top \mathbf{M} \mathbf{X} - \Lambda_\mathbf{M}\|_\mathrm{F}^2 = 4(\mathbf{M}\mathbf{X}\mathbf{X}^\top \mathbf{M}\mathbf{X} - \mathbf{M}\mathbf{X}\Lambda_\mathbf{M}).$$

The gradient of the coupling/decoupling term is

$$\nabla_\mathbf{X} \|\mathbf{M}_1 \mathbf{X} - \mathbf{M}_2\|_\mathrm{F}^2 = 2(\mathbf{M}_1^\top \mathbf{M}_1 \mathbf{X} - \mathbf{M}_1^\top \mathbf{M}_2).$$

Robustness to initialization. In Figure 4.4, we show examples of runs with different initializations, coupling weight and off-diagonal costs. In our experiments, we used the MATLAB package Manopt [Boumal et al., 2014]. When the coupling weight μ_c increases, the convergence with all the off-diagonal costs is independent on the initialization. Furthermore, the non-isometric case is more

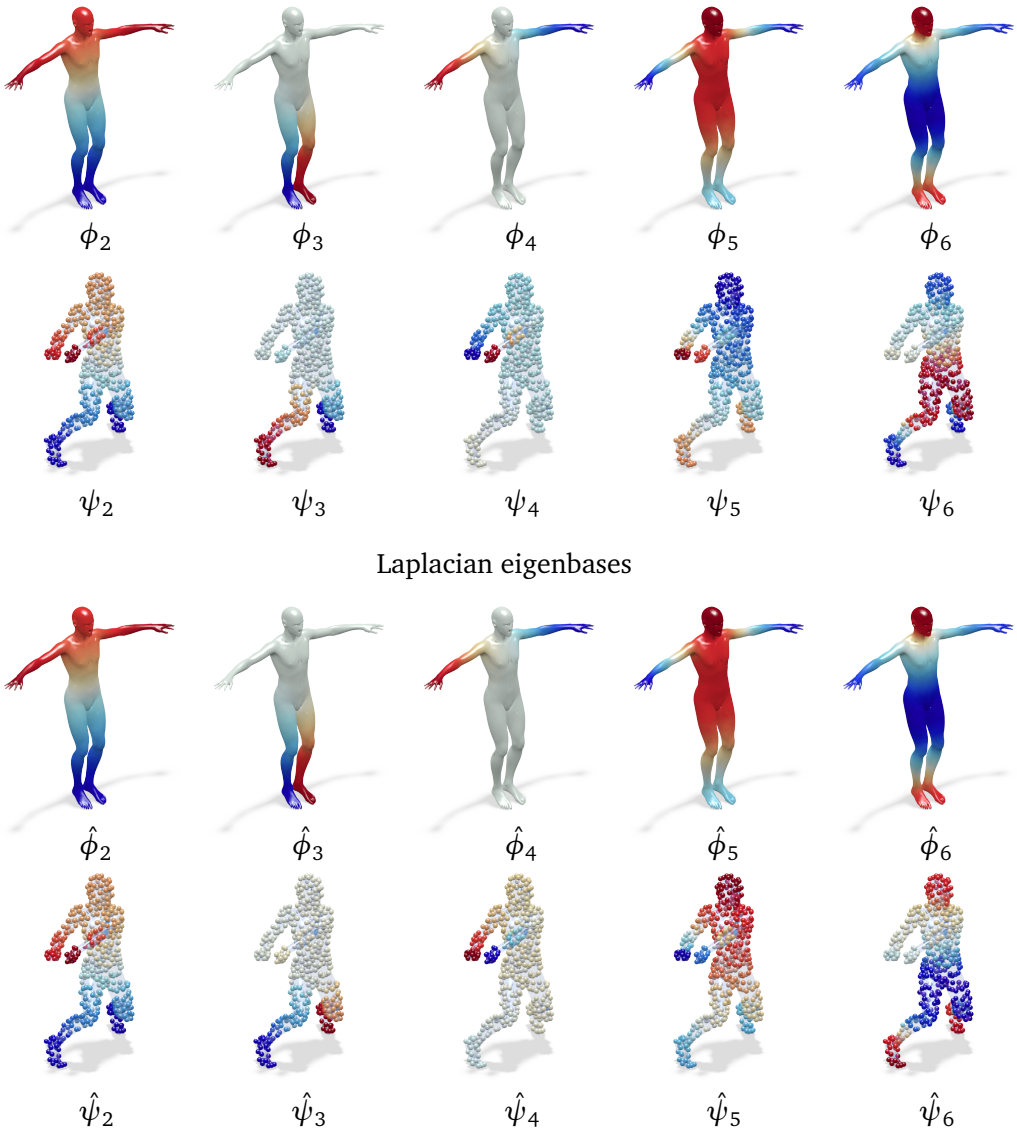


Figure 4.2. First Laplacian eigenfunctions (top) and coupled basis vectors (bottom) computed on a full man mesh and 10-times subsampled point cloud. Hot and cold colors represent positive and negative values, respectively.

robust to initializations, than the isometric case. Intuitively, we can interpret this as follows: for the near-isometric case there are many possible rotations to align the original bases, whereas for non-isometric shapes there are few rotations to

align them.

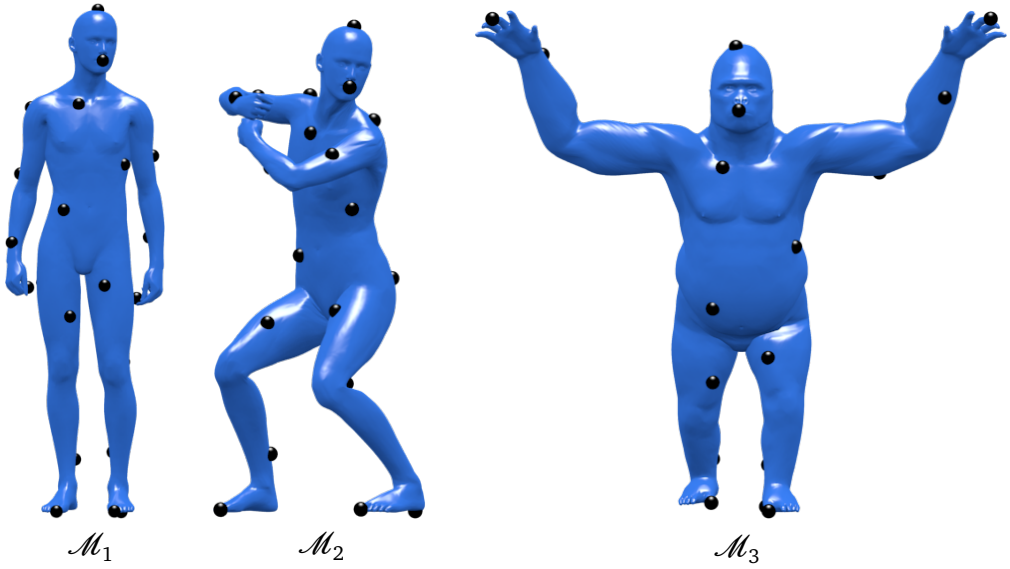


Figure 4.3. Examples of the shapes used for the numerical experiments. Black spheres denote 25 ground-truth point-wise correspondences across the shapes.

Number of corresponding functions. In Figure 4.5, we observe that off_2 is relatively robust to initialization for both isometric and non-isometric cases for varying number of corresponding data. In the non-isometric case, however, also off_3 performs robustly for different number of corresponding functions. The penalty off_1 shows average performance. In the following, we will use off_3 and off_2.

Sensitivity to noise. Figure 4.6 exemplifies the performance of the optimization when the corresponding data is noisy. We observe the general trend that when the amount of noise increases, it becomes more difficult to couple the bases, which is manifested in higher values of the cost function at the minimum.

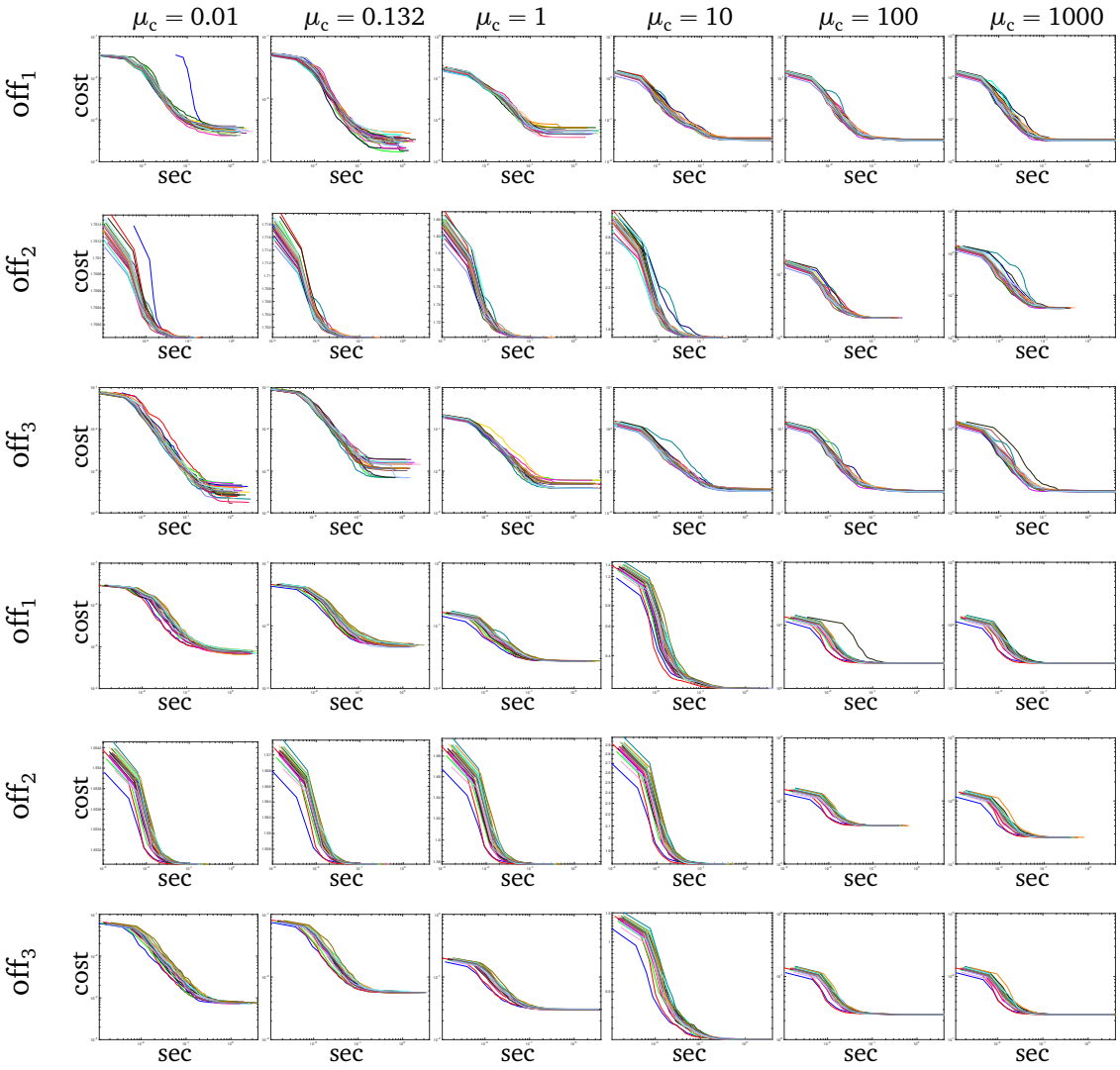


Figure 4.4. Robustness of the parameterized coupled diagonalization problem for different initializations for two near-isometric shapes (the first three rows) and for two non-isometric shapes (the last three rows). For the experiment we considered shapes $\mathscr{M}_1$ and $\mathscr{M}_2$ for the near-isometric case, and shapes $\mathscr{M}_1$ and $\mathscr{M}_3$ for the non-isometric experiment (Figure 4.3). The first and the fourth rows correspond to the off_1 diagonality cost (sum of squared off-diagonal elements), the second and fifth to off_2 (trace) and the third and sixth to off_3 (Frobenius norm distance to the original eigenvalues). Each column corresponds to a different coupling weight μ_c. Lines of different colors denote optimizations with different initializations; the initializations are consistent across different plots, *i.e.*, lines of the same color correspond to the same initialization for each experiment scenario, namely near-isometric and non-isometric. As correspondence data we considered 25 corresponding delta functions illustrated as spheres in Figure 4.3. Number of bases $k' = k = 25$. Logarithm is applied to both axes.

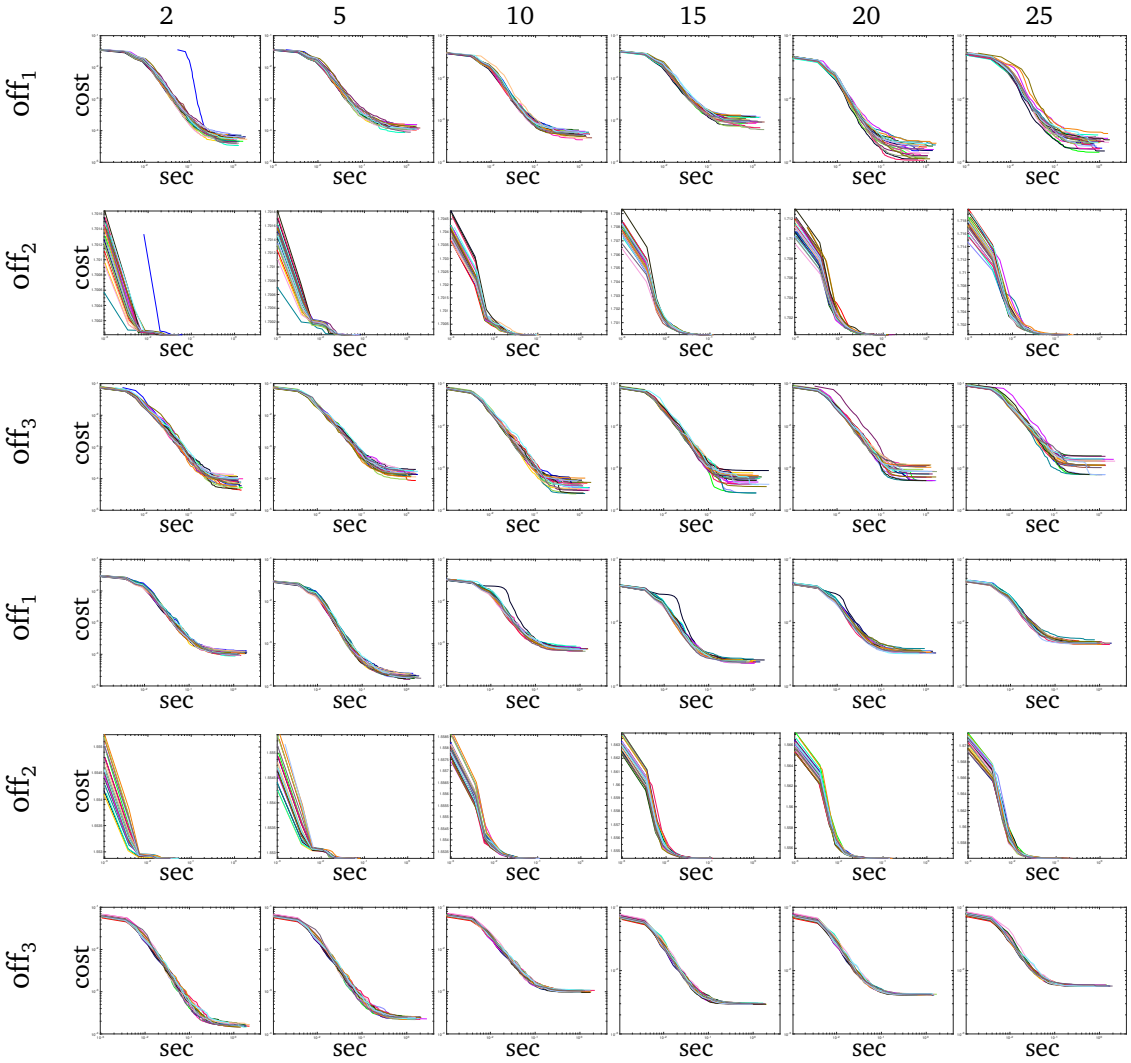


Figure 4.5. Performance of the parameterized coupled diagonalization problem for different initializations and different number of corresponding functions. Numbers for each column denote the number of corresponding functions used for coupling. In all the experiments $\mu_c = 0.132$ and other settings and notations are as in Figure 4.4.

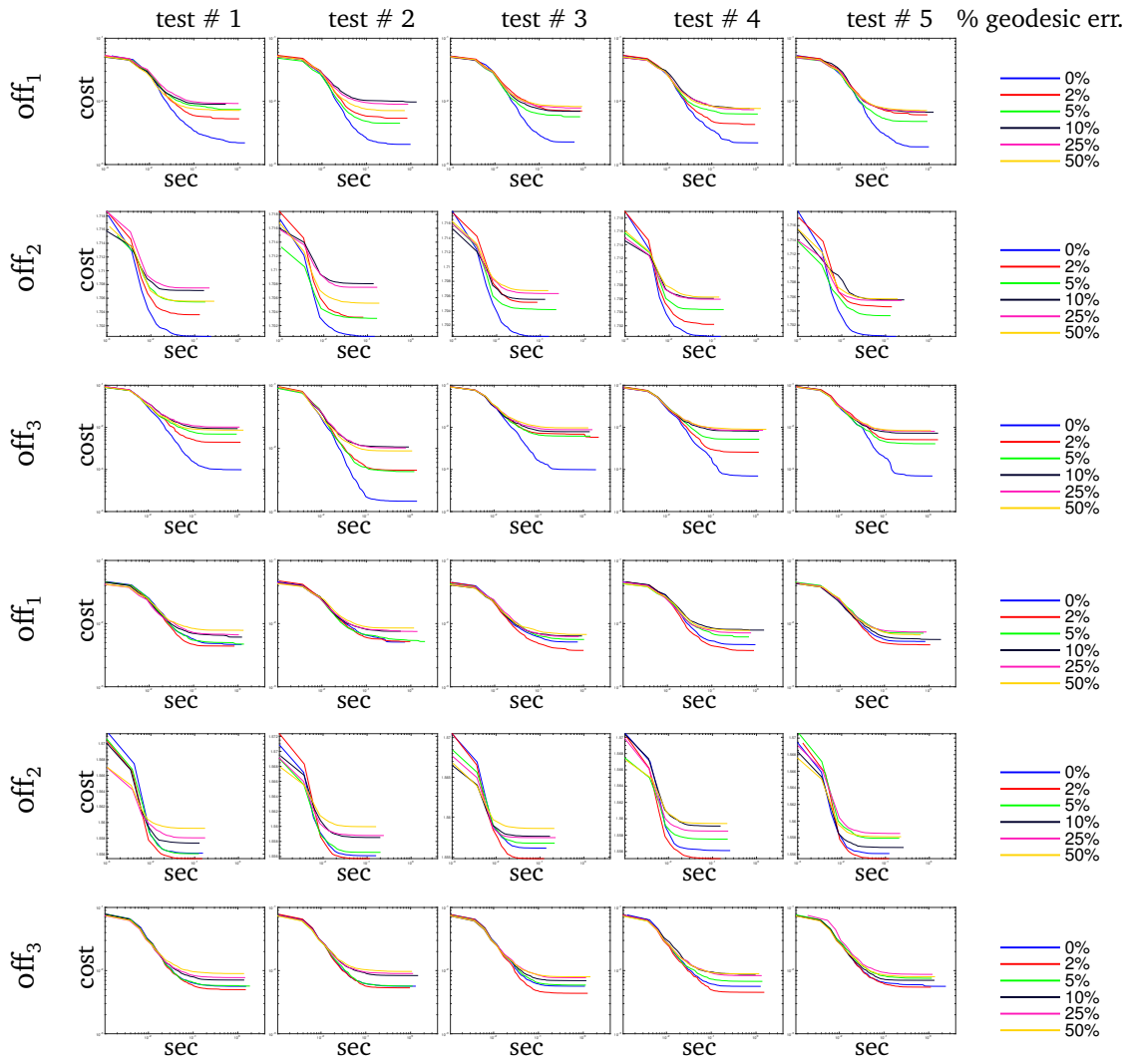


Figure 4.6. Performance of the subspace coupled diagonalization problem for different initializations and different noise magnitude in corresponding data. Numbers for each column denote the index of the test: for each column the initialization is the same across different methods. Colors encode the noise magnitude (deviation from groundtruth in % of geodesic diameter). In all the experiments $\mu_c = 0.132$ and other settings and notations are as in Figure 4.4.

4.3 Robust formulation

Often, the corresponding data (*e.g.*, descriptors, detected compatible regions, etc.) are noisy or have outliers, *i.e.*, irrelevant information, which should be ignored. Replacing the Frobenius norm for the coupling terms with robust norms such as $\|\cdot\|_1$ or $\|\cdot\|_{2,1}$ allows to better cope with outliers in the correspondence data [Huang et al., 2014; Wang and Singer, 2013]. We define the robust coupled diagonalization problem as

$$\begin{aligned}\min_{\mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}} \sum_{i=1}^{p} \|\mathbf{R}_i^\top \bar{\mathbf{\Lambda}}_i \mathbf{R}_i - \bar{\mathbf{\Lambda}}_i\|_{\mathrm{F}}^2 \quad &+ \quad \mu_{\mathrm{c}} \sum_{i,j=1}^{p} \|\mathbf{F}_{ij}^\top \mathbf{A}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i - \mathbf{G}_{ij}^\top \mathbf{A}_j \bar{\mathbf{\Phi}}_j \mathbf{R}_j\|_{2,1} \\ &- \quad \mu_{\mathrm{d}} \sum_{i,j=1}^{p} \|\mathbf{E}_{ij}^\top \mathbf{A}_i \bar{\mathbf{\Phi}}_i \mathbf{R}_i - \mathbf{H}_{ij}^\top \mathbf{A}_j \bar{\mathbf{\Phi}}_j \mathbf{R}_j\|_{2,1}.\end{aligned} \tag{4.4}$$

Note that this robust formulation has a non-differentiable cost function due to $\|\cdot\|_{2,1}$, and is also constrained to a product of Stiefel manifolds. To the best of our knowledge, there are no optimization methods capable of optimizing general non-smooth functions over matrix-manifolds, making the optimization of 4.4 challenging. In Chapter 5, we introduce an efficient algorithm capable of optimizing general non-smooth manifold-constrained problems and, in particular, our problem 4.4.

4.4 Relation to Functional Correspondence

Our final remark is on the relation between our coupled diagonalization problem 4.3 and the functional correspondence problem 2.18 discussed in Chapter 2. In the basic setting of correspondence between two manifolds $\mathscr{X}$ and $\mathscr{Y}$, the functional correspondence problem is stated as a least-squares solution of a linear system,

$$\min_{\mathbf{C} \in \mathbb{R}^{k \times k}} \|\mathbf{G}^\top \mathbf{A}_Y \bar{\mathbf{\Psi}} - \mathbf{F}^\top \mathbf{A}_X \bar{\mathbf{\Phi}} \mathbf{C}\|_{\mathrm{F}}^2 \tag{4.5}$$

where $\mathbf{F}, \mathbf{G}$ are $n_X \times q$ and $n_Y \times q$ matrices of discretized corresponding functions, and $\bar{\mathbf{\Phi}}, \bar{\mathbf{\Psi}}$ are the $n_X \times k$ and $n_Y \times k$ matrices of the respective Laplacian eigenvectors. Ovsjanikov et al. [2012] show that if the functional map originates from an area-preserving map (in particular, an isometry), the $k \times k$ matrix $\mathbf{C}$ is orthonormal and problem (4.5) boils down to the *orthogonal Procrustes problem*, which has a closed-form solution $\mathbf{C} = \mathbf{U}\mathbf{V}^\top$ where $(\mathbf{F}^\top \mathbf{A}_X \bar{\mathbf{\Phi}})^\top \mathbf{G}^\top \mathbf{A}_Y \bar{\mathbf{\Psi}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ denotes singular value decomposition with orthonormal $\mathbf{U}$ and $\mathbf{V}$.

Rewriting problem 4.5 with $\mathbf{C} = \mathbf{U}\mathbf{V}^\top$, we get

$$\min_{\mathbf{U}^\top\mathbf{U}=\mathbf{I},\mathbf{V}^\top\mathbf{V}=\mathbf{I}} \|\mathbf{G}^\top \mathbf{A}_Y \bar{\mathbf{\Psi}}\mathbf{V} - \mathbf{F}^\top \mathbf{A}_X \bar{\mathbf{\Phi}}\mathbf{U}\|_{\mathrm{F}}^2, \tag{4.6}$$

which coincides with our subspace coupled diagonalization problem 4.2 for the setting $k' = k$ and $\mu_{\mathrm{d}} = 0$ and $\mu_{\mathrm{c}} \to \infty$ (no off-diagonal regularization). The geometric interpretation is the same: we try to *rigidly* align the two eigenbases by applying rotation/reflection matrices $\mathbf{U}, \mathbf{V}$ to them.

The setting $k' > k$ of our problem implies that the matrices $\mathbf{U}, \mathbf{V}$ are non-square ($k' \times k$); in this case, the geometric interpretation is of a *non-rigid* alignment of the eigenbases, where the off-diagonality penalty ensures that the deformation is smooth.

In the more general setting of correspondence between p manifolds $\mathscr{M}_1, \dots, \mathscr{M}_p$, the functional map between $\mathscr{M}_i$ and $\mathscr{M}_j$ is an $n_j \times n_i$ matrix $\mathbf{T}_{ij}$ that is approximated by $\mathbf{T}_{ij} \approx \bar{\mathbf{\Phi}}_j \mathbf{C}_{ij}^\top \bar{\mathbf{\Phi}}_i \mathbf{A}_i$, where $\mathbf{C}_{ij}$ is the $k \times k$ matrix translating Fourier coefficients from basis $\bar{\mathbf{\Phi}}_i$ to $\bar{\mathbf{\Phi}}_j$. In this setting, the functional correspondence problem can be written as

$$\min_{\mathbf{C}_{ij}^\top \mathbf{C}_{ij} = \mathbf{I}} \sum_{i \neq j} \|\mathbf{F}_{ij}^\top \bar{\mathbf{\Phi}}_i \mathbf{A}_i \mathbf{C}_{ij} - \mathbf{G}_{ij}^\top \bar{\mathbf{\Phi}}_j \mathbf{A}_j\|_F^2, \tag{4.7}$$

where we assume to be given $n_i \times q_{ij}$ matrix $\mathbf{F}_{ij}$ and $n_j \times q_{ij}$ matrix $\mathbf{G}_{ij}$ of discretized corresponding functions as before.

Imposing a further assumption that $\mathbf{T}_{ij}$ is volume-preserving, $\mathbf{C}_{ij}$ must be an orthonormal matrix, which can be represented as a product of two orthonormal matrices, $\mathbf{C}_{ij} \approx \mathbf{R}_i \mathbf{R}_j^\top$. Plugging in the factorization $\mathbf{C}_{ij} = \mathbf{R}_i \mathbf{R}_j^\top$ and using the unitary invariance of Frobenius norm, we obtain an instance of our subspace

coupled diagonalization problem 4.2,

$$\min_{\mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}} \sum_{i \neq j} \|\mathbf{F}_{ij}^\top \bar{\mathbf{\Phi}}_i \mathbf{A}_i \mathbf{R}_i - \mathbf{G}_{ij}^\top \bar{\mathbf{\Phi}}_j \mathbf{A}_j \mathbf{R}_j\|_F^2, \tag{4.8}$$

in the case $k' = k$ and $\mu_\mathrm{d} = 0$ and $\mu_\mathrm{c} \to \infty$.

Chapter 5

MADMM: a Generic Algorithm For Non-smooth Optimization on Manifolds

The subspace formulation of the coupled diagonalization and the joint approximate diagonalization problems require optimization of some cost function over the set of orthonormal matrices that is known as the *Stiefel manifold*. Such class of optimization problems is known as *manifold optimization* and can be addressed by a special type of algorithms. The Manopt Matlab toolbox [Boumal et al., 2014] is the most comprehensive toolbox we are aware of, implementing several solvers and supporting several types of manifolds in a very convenient manner.

Yet, the methods in the Manopt toolbox, and more generally, most of the methods known in this community, have been developed for *smooth* functions. In our applications, when dealing with noisy data, one is interested in robust formulations involving, *e.g.*, a non-smooth norm such as $\|\cdot\|_{2,1}$ in the coupling term of 4.4.

In this chapter, we develop a general framework for non-smooth manifold optimization, which was presented at ECCV [Kovnatsky et al., 2016]. Our method is very simple and straightforward to implement. While focusing our discussion mostly on problems of simultaneous diagonalization involving optimization on the Stiefel manifold, we stress that the method is generic and can be applied to

other manifolds as well. An interested reader is referred to additional examples in [Kovnatsky et al., 2016].

5.1 Manifold Optimization

The term *manifold-* or *manifold-constrained optimization* refers to a class of problems of the form

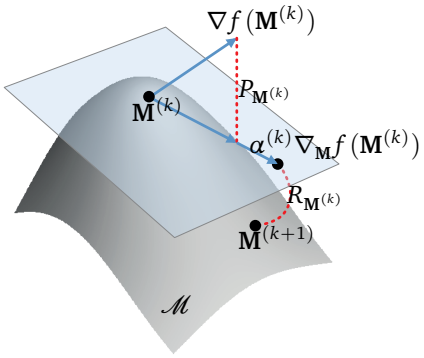
$$\min_{\mathbf{M}\in\mathscr{M}} f(\mathbf{M}), \tag{5.1}$$

where f is a smooth real-valued function, $\mathbf{M}$ is an $n' \times n$ real matrix, and $\mathscr{M}$ is some Riemannian submanifold of $\mathbb{R}^{n'\times n}$. The main idea of manifold optimization is to treat the objective as a function $f : \mathscr{M} \to \mathbb{R}$ defined *on the manifold*, and perform descent on the manifold itself rather than in the ambient Euclidean space representing manifold structure as appropriate constraints on $\mathbf{M}$.

repeat
- Compute the extrinsic gradient $\nabla f(\mathbf{M}^{(k)})$
- Projection: $\nabla_{\mathscr{M}} f(\mathbf{M}^{(k)}) = P_{\mathbf{M}^{(k)}}(\nabla f(\mathbf{M}^{(k)}))$
- Compute the step size $\alpha^{(k)}$ along the descent direction
- Retraction: $\mathbf{M}^{(k+1)} = R_{\mathbf{M}^{(k)}}(-\alpha^{(k)}\nabla_{\mathscr{M}} f(\mathbf{M}^{(k)}))$

until *convergence*;

Algorithm 2: Conceptual algorithm for smooth optimization on manifold $\mathscr{M}$.



A conceptual gradient descent-like manifold optimization is presented in Algorithm 2 (we refer the reader to [Absil et al., 2009] for a comprehensive introduction to manifold optimization). On a manifold, the *intrinsic* (Riemannian) gradient $\nabla_{\mathscr{M}} f(\mathbf{M})$ is a tangent vector and can be obtained by projecting the standard (Euclidean) gradient $\nabla f(\mathbf{M})$ onto tangent plane $T_{\mathbf{M}}\mathscr{M}$ at point $\mathbf{M}$ by means

of a *projection* operator $P_{\mathbf{M}}$ (see an illustration in Figure 2). During an iteration of a manifold optimization algorithm, a step along the intrinsic gradient direction is performed in the tangent plane. In order to obtain the next iterate, the point in the tangent plane is mapped back to the manifold by means of a *retraction* operator $R_{\mathbf{M}}$, which is typically an approximation of the *exponential map* (see Section 2.1). For many manifolds, the projection P and retraction R operators have a closed form expression.

5.1.1 Applications of manifold optimization

A wide range of problems in machine learning, pattern recognition, computer vision, and signal processing is formulated as manifold optimization problems. For example, optimization on the *Grassman manifold* comes up in multi-view clustering [Dong et al., 2014] and matrix completion [Keshavan and Oh, 2009]. Optimization on the *sphere* is used in principle geodesic analysis [Zhang and Fletcher, 2013], a generalization of the classical PCA to non-Euclidean domains.

Optimization on the *Stiefel manifold* arises in a plethora of applications ranging from classical ones such as eigenvalue problems, assignment problems, and Procrustes problems [Berge, 1977], to more recent ones such as 1-bit compressed sensing [Boufounos and Baraniuk, 2008]. Problems involving *products of Stiefel manifolds* are our main focus here. Their various applications include shape correspondence [Kovnatsky et al., 2013], manifold learning [Eynard et al., 2015], sensor localization [Cucuringu et al., 2012a], structural biology [Cucuringu et al., 2012b], and structure from motion recovery [Arie-Nachimson et al., 2012].

Optimization over the manifold of *fixed-rank matrices* arises in maxcut problems [Journée et al., 2010], sparse principal component analysis [Journée et al., 2010], regression [Meyer et al., 2011], matrix completion [Boumal and Absil, 2011; Tan et al., 2014], and image classification [Shalit et al., 2010]. *Oblique manifolds* are encountered in problems such as independent component analysis and joint diagonalization [Absil and Gallivan, 2006], blind source separation [Kleinsteuber and Shen, 2012], and prediction of stock returns [Higham, 2002].

Though some instances of manifold optimization such as eigenvalues problems have been treated extensively in the distant past, the first general purpose

algorithms appeared only in the 1990s [Smith, 1994; Wen and Yin, 2013]. With the emergence of numerous applications during the last decade, especially in the machine learning community, there has been an increased interest in general-purpose optimization on different manifolds [Absil et al., 2009], leading to several manifold optimization algorithms such as conjugate gradients [Edelman et al., 1998], trust regions [Absil et al., 2007], and Newton [Smith, 1994; Alvarez et al., 2008].

5.2 Non-smooth manifold optimization

More recently, there has been an interest in optimizing non-smooth functions on manifolds. Typical applications, again, arise mostly from problems in machine learning and pattern recognition communities, where the need to deal with noise requires the use of non-smooth robust cost functions.

Broadly speaking, there are three classes of optimization methods for non-smooth functions. First, *smoothing* methods replace the non-differentiable function with its smooth approximation [Chen, 2012] (*e.g.*, $|x| \approx \sqrt{x^2 + \epsilon}$). Such methods typically suffer from a tradeoff between accuracy (how close the approximation function to the original objective) and convergence speed (less smooth functions are usually harder to optimize). The second class of methods use *subgradients* as a generalization of derivatives of non-differentiable functions. In the context of manifold optimization, several subgradient approaches have been proposed [Ferreira and Oliveira, 1998; Ledyaev and Zhu, 2007; Kleinsteuber and Shen, 2012].

The third class of methods are *splitting* approaches, studied mostly for problems involving the minimization of matrix functions with orthogonality constraints, which limits their application solely to Stiefel manifolds. Lai and Osher [2014] proposed the method of splitting orthogonal constraints (SOC) based on the Bregman iteration . A similar approach was independently developed in [Rosman et al., 2011, 2014]. Neumann *et al.* [Neumann et al., 2014] used a different splitting scheme for the same class of problems.

Splitting methods offer a bunch of nice properties, making them very popular

in machine learning and signal processing applications. First, they split complicated large problems into simple sub-problems, where each subproblem has a simple solution (in some cases, even closed-form). In some settings, the splitting lends itself to parallelization on distributed computational architectures. Second, they make very few assumptions about the properties of the objective function. Third, they are very easy to implement. Finally, splitting methods typically demonstrate fast convergence, often superior to other methods.

5.3 Manifold ADMM

Here, we consider general non-smooth manifold optimization problems of the form

$$\min_{\mathbf{M}\in\mathcal{M}} f(\mathbf{M}) + h(\mathbf{SM}), \tag{5.2}$$

where f and h are smooth and non-smooth real-valued functions, respectively. $\mathbf{S}$ is a $p \times n'$ matrix, and the rest of the notation is as in problem 5.1. Typical examples of h often used in machine learning application are nuclear-, $\|\cdot\|_1$, or $\|\cdot\|_{2,1}$ norms. Because of non-smoothness of the objective functions, Algorithm 2 cannot be used directly to minimize (5.2).

We propose treating this class of problems using the alternating directions method of multipliers (ADMM). The key idea is that problem (5.2) can be equivalently formulated as

$$\min_{\mathbf{M}\in\mathcal{M},\mathbf{Z}\in\mathbb{R}^{p\times n}} f(\mathbf{M}) + h(\mathbf{Z}) \quad \text{s.t.} \quad \mathbf{Z} = \mathbf{SM} \tag{5.3}$$

by introducing an artificial variable $\mathbf{Z}$ and a linear constraint. The method of multipliers [Hestenes, 1969; Powell, 1969], applied to only the linear constraints in (5.3), leads to the minimization problem

$$\min_{\mathbf{M}\in\mathcal{M},\mathbf{Z}\in\mathbb{R}^{p\times n}} f(\mathbf{M}) + h(\mathbf{Z}) + \tfrac{\rho}{2}\|\mathbf{SM} - \mathbf{Z} + \mathbf{U}\|_{\mathrm{F}}^2 \tag{5.4}$$

where $\rho > 0$ and $\mathbf{U} \in \mathbb{R}^{p\times n}$ have to be chosen and updated appropriately (see

below).

This formulation now allows splitting the problem into two optimization sub-problems w.r.t. to $\mathbf{M}$ and $\mathbf{Z}$, which are solved in an alternating manner, followed by an updating of $\mathbf{U}$ and, if necessary, of ρ. Observe that in the first sub-problem w.r.t. $\mathbf{M}$ we minimize a *smooth* function with manifold constraints, and in the second sub-problem w.r.t. $\mathbf{Z}$ we minimize a non-smooth function without manifold constraints. Thus, the problem breaks down into two well-known sub-problems. This method, which we call *Manifold alternating direction method of multipliers* (MADMM), is summarized in Algorithm 3.

Initialize $k \leftarrow 1$, $\mathbf{Z}^{(1)} = \mathbf{S}\mathbf{M}^{(1)}$, $\mathbf{M}^{(1)} = 0$, $\mathbf{U}^{(1)} = 0$.
repeat
  $\mathbf{M}$-step: $\mathbf{M}^{(k+1)} = \operatorname{argmin}_{\mathbf{M}\in\mathscr{M}} f(\mathbf{M}) + \frac{\rho}{2}\|\mathbf{S}\mathbf{M} - \mathbf{Z}^{(k)} + \mathbf{U}^{(k)}\|_{\mathrm{F}}^2$
  $\mathbf{Z}$-step: $\mathbf{Z}^{(k+1)} = \operatorname{argmin}_{\mathbf{Z}\in\mathbb{R}^{p\times n}} h(Z) + \frac{\rho}{2}\|\mathbf{S}\mathbf{M}^{(k+1)} - \mathbf{Z} + \mathbf{U}^{(k)}\|_{\mathrm{F}}^2$
  Update $\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + \mathbf{S}\mathbf{M}^{(k+1)} - \mathbf{Z}^{(k+1)}$ and $k \leftarrow k+1$
until *convergence*;

Algorithm 3: Generic MADMM method for non-smooth optimization on manifold $\mathscr{M}$.

Note that MADMM is extremely simple and easy to implement. The $\mathbf{M}$-step is the setting of Algorithm 2 and can be carried out using any standard smooth manifold optimization method. Similarly to common implementation of ADMM algorithms, there is no need to solve the $\mathbf{M}$-step problem *exactly*; instead, only a few iterations of manifold optimization are done. Furthermore, for some manifolds and some functions f, the $\mathbf{M}$-step has a closed-form solution.

The $\mathbf{Z}$-step is the proximity operator of $\frac{1}{\rho}h$ at $\mathbf{S}\mathbf{M}^{(k+1)} + \mathbf{U}^{(k)}$ [Parikh and Boyd, 2014]. In many cases it has a closed-form solution: for example, when h is the L_1-norm, the $\mathbf{Z}$-step boils down to simple shrinkage, and when h is nuclear norm, the $\mathbf{Z}$-step is performed by singular value shrinkage. ρ is the only parameter of the algorithm and its choice is not critical for convergence. In our following experiments, we used a rather arbitrary fixed value of ρ, though in the ADMM literature it is common to adapt ρ at each iteration, *e.g.*, using the strategy described in [Boyd et al., 2010].

5.3.1 MADMM for Coupled Diagonalization

We show the application of MADMM to robust formulations of the coupled diagonalization problem discussed in Chapter 4. For additional examples and comparisons of MADMM to state-of-the-art methods applied to other problems, we refer the reader to our report [Kovnatsky et al., 2016]. Our notation follows that of Chapter 4. We assume to be given a collection of p shapes $\mathcal{M}_1, \dots, \mathcal{M}_p$, each discretized at n_i points and equipped with a Laplacian $\mathbf{L}_i = \mathbf{A}_i^{-1}\mathbf{W}_i$ represented as an $n_i \times n_i$ matrix; $\mathbf{A}_i$ and $\mathbf{W}_i$ are the mass and stiffness matrices as in (2.7), respectively. The first k eigenvectors of these Laplacians arranged as $n_i \times k$ matrices $\bar{\mathbf{\Phi}}_i$; the corresponding eigenvalues are represented as $k \times k$ diagonal matrices $\bar{\mathbf{\Lambda}}_i$. The functional map between manifolds $\mathcal{M}_i$ and $\mathcal{M}_j$ is approximated by $\mathbf{T}_{ij} \approx \bar{\mathbf{\Phi}}_j \mathbf{C}_{ij}^\top \bar{\mathbf{\Phi}}_i \mathbf{A}_i$; imposing a further assumption that $\mathbf{T}_{ij}$ is volume-preserving, we get an orthonormal matrix $\mathbf{C}_{ij} \approx \mathbf{R}_i \mathbf{R}_j^\top$, where $\mathbf{R}_1, \dots, \mathbf{R}_p$ are orthonormal. For each pair of manifolds $\mathcal{M}_i$, $\mathcal{M}_j$, we assume to be given a set of q_{ij} corresponding vectors arranged as columns of an $n_i \times q_{ij}$ matrix $\mathbf{F}_{ij}$ and $n_j \times q_{ij}$ matrix $\mathbf{G}_{ij}$.

Assuming for simplicity no decoupling term ($\mu_\mathrm{d} = 0$), the robust version of our subspace coupled diagonalization problem takes the form

$$\min_{\mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}} \sum_i \mathrm{off}(\mathbf{R}_i^\top \bar{\mathbf{\Lambda}}_i \mathbf{R}_i) + \mu_\mathrm{c} \sum_{i \neq j} \| \mathbf{F}_{ij}^\top \bar{\mathbf{\Phi}}_i \mathbf{A}_i \mathbf{R}_i - \mathbf{G}_{ij}^\top \bar{\mathbf{\Phi}}_j \mathbf{A}_j \mathbf{R}_j \|_{2,1}. \tag{5.5}$$

Problem 5.5 is an instance of non-smooth manifolds optimization, since the cost function has a non-smooth component $h = \sum \| \cdot \|_{2,1}$, and the variables $\mathbf{R}_1, \dots, \mathbf{R}_p$ are restricted to the Stiefel manifold. We can apply our MADMM approach, as summarized in Algorithm 4. The M-step of MADMM is performed using a few iterations of some standard manifold optimization on a product of Stiefel manifolds. The Z-step in this case boils down to the standard sparse coding or Lasso problem, which is carried out by a single shrinkage iteration, given in closed form by applying the *shrinkage operator*

$$\mathrm{Shrink}(\mathbf{M}, \tau) = \mathrm{sign}(\mathbf{M}) \odot \max\{|\mathbf{M}| - \tau, \mathbf{0}\}$$

in an element-wise manner.

Input $n_j \times q_{ij}$ corresponding matrices $\mathbf{F}_{ij}, \mathbf{G}_{ij}$, $n_i \times k$ eigenbases matrices $\bar{\mathbf{\Phi}}_i$, $k \times k$ diagonal matrices $\bar{\mathbf{\Lambda}}_i$ of corresponding eigenvalues, and parameter $\mu_c > 0$

Output orthonormal matrices $\mathbf{R}_1, \dots, \mathbf{R}_p$ aligning the bases $\bar{\mathbf{\Phi}}_1, \dots, \bar{\mathbf{\Phi}}_p$ and allowing to express the functional correspondences as $\mathbf{T}_{ij} \approx \bar{\mathbf{\Phi}}_j \mathbf{R}_j \mathbf{R}_i^\top \bar{\mathbf{\Phi}}_i^\top \mathbf{A}_i$.

Initialize $k \leftarrow 1$, $\mathbf{R}_i^{(1)} \leftarrow \mathbf{I}$, $Z_{ij}^{(1)} \leftarrow \mathbf{F}_{ij}\mathbf{A}_i\bar{\mathbf{\Phi}}_i\mathbf{R}_i^{(1)} - \mathbf{G}_{ij}\mathbf{A}_j\bar{\mathbf{\Phi}}_j\mathbf{R}_j^{(1)}$, $\mathbf{U}_{ij}^{(1)} \leftarrow \mathbf{I}$.

repeat

$$(\mathbf{R}_1^{(k+1)}, \dots, \mathbf{R}_p^{(k+1)}) = \underset{\mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}}{\operatorname{argmin}} \sum_i \operatorname{off}(\mathbf{R}_i^\top \bar{\mathbf{\Lambda}}_i \mathbf{R}_i) + \frac{\rho}{2} \sum_{i \neq j} \|\mathbf{F}_{ij}\mathbf{A}_i\bar{\mathbf{\Phi}}_i\mathbf{R}_i - \mathbf{G}_{ij}\mathbf{A}_j\bar{\mathbf{\Phi}}_j\mathbf{R}_j - \mathbf{Z}_{ij}^{(k)} + \mathbf{U}_{ij}^{(k)}\|_F^2$$

$$\mathbf{Z}_{ij}^{(k+1)}(:,l) = \operatorname{Shrink}_{\frac{\mu_c}{\rho}}(\mathbf{F}_{ij}\bar{\mathbf{\Phi}}_i\mathbf{A}_i\mathbf{R}_i^{(k+1)}(:,l) - \mathbf{G}_{ij}\bar{\mathbf{\Phi}}_j\mathbf{A}_j\mathbf{R}_j^{(k+1)}(:,l) + \mathbf{U}_{ij}^{(k)}(:,l))$$

$$\mathbf{U}_{ij}^{(k+1)} = \mathbf{U}_{ij}^{(k)} + \mathbf{F}_{ij}\bar{\mathbf{\Phi}}_i\mathbf{A}_i\mathbf{R}_i^{(k+1)} - \mathbf{G}_{ij}\bar{\mathbf{\Phi}}_j\mathbf{A}_j\mathbf{R}_j^{(k+1)} - \mathbf{Z}_{ij}^{(k+1)}$$

$k \leftarrow k + 1$

until *convergence*;

Algorithm 4: MADMM method for robust coupled diagonalization on multiple manifolds. $\mathbf{M}(:,i)$ denotes the ith column of matrix $\mathbf{M}$.

5.3.2 Convergence

Our final note is on the convergence of the proposed method. MADMM belongs to the class of multiplier algorithms that can be considered as 'methods with partial elimination of constraints' [Bertsekas, 1982] and as 'augmented Lagrangian methods with general lower-level constraints' [Andreani et al., 2007]. We note that the convergence results of [Bertsekas, 1982; Andreani et al., 2007] do not apply in our case due to non-differentiability of the function h in (5.2). Furthermore, MADMM is an alternating method and thus is not covered by theoretical results on 'pure' multiplier methods. An avenue for obtaining convergence results for (a regularized version of) MADMM is the recently developed theory by

Attouch et al. [2010], which is applicable to non convex and non-differentiable functions f and h. Attouch et al. [2010] show convergence results for the class of *semi algebraic objects*, which includes Stiefel and other matrix manifolds. Wang et al. [2015] prove global convergence of ADMM in convex and non-smooth scenarios, however the non-smooth and non-convex parts should belong to a specific class of functions (piecewise linear functions, ℓ_q quasi-norms ($0 \leqslant q \leqslant 1$), etc.), which limits the use of their convergence results. We defer a deeper study of convergence properties to future work.

Chapter 6

Applications in Manifold Learning & Computer Vision

In Chapters 3 and 4, we prosed an extension of spectral methods to multiple domains through simultaneous diagonalization of the respective Laplacian matrices. This naturally extends classical data analysis tools such as diffusion maps and spectral clustering. In this chapter, we provide an experimental evaluation on synthetic and real data in problems of manifold learning, object classification, and clustering. We show that joint spectral geometry better captures inherent structure of multi-modal data leading to a superior performance. Finally, we show the relation of many existing approaches for multimodal manifold analysis to our framework. The results in this chapter are from our PAMI paper [Eynard et al., 2015].

6.1 Multimodal spectral methods

We start with an overview of other spectral methods dealing with multimodal data and try to establish their relation to our approach. In our analysis, we distinguish between two broad groups of approaches: those assuming the *full coupling* setting (*i.e.*, equal number of samples/vertices $n_i = n$ and known bijective correspondence between them) and *sparse coupling* setting (each modality might have a different number of samples/vertices, and the correspondence is

known only between a few of them). The notation follows sections 3, 4.

6.1.1 Full coupling setting

Laplacian averaging. Assuming the full coupling setting between p modalities and that the first k eigenvalues of the Laplacians are zero, we want to find $\breve{\mathbf{\Phi}} \in \mathbb{R}^{n\times k}$ such that $\mathbf{L}_i\breve{\mathbf{\Phi}} = 0$ for all $i = 1,\dots,p$ and $\breve{\mathbf{\Phi}}^\top\breve{\mathbf{\Phi}} = \mathbf{I}$ by reformulating (Equation 3.3) as

$$\min_{\breve{\mathbf{\Phi}}^\top\breve{\mathbf{\Phi}}=\mathbf{I}} \sum_{i=1}^{p} \|\mathbf{L}_i\breve{\mathbf{\Phi}}\|_{\mathrm{F}}^2 \tag{6.1}$$

Since $\sum_{i=1}^p \|\mathbf{L}_i\breve{\mathbf{\Phi}}\|_{\mathrm{F}}^2 = \mathrm{tr}(\breve{\mathbf{\Phi}}^\top(\sum_{i=1}^p \mathbf{L}_i^\top\mathbf{L}_i)\breve{\mathbf{\Phi}})$, the problem can be equivalently recast as finding the null eigenvectors of the "average" Laplacian matrix $\bar{\mathbf{L}} = \sum_{i=1}^p \mathbf{L}_i^\top\mathbf{L}_i$. One can also consider other averaging operators, such as arithmetic mean $\bar{\mathbf{L}} = \frac{1}{p}\sum_{i=1}^p \mathbf{L}_i$ or harmonic mean $\bar{\mathbf{L}} = (\sum_{i=1}^p \mathbf{L}_i^{-1})^{-1}$. In what follows, we will show that many approaches for multimodal manifold alignment boil down to simple Laplacian averaging in their limit cases. Laplacian averaging methods seem to be the most 'naïve' way of producing multi-modal spectral geometry and have been used in several applications such as clustering [Ma and Lee, 2008]. [1]

Matrix factorization. Tang et al. [2009] proposed graph clustering through low-rank factorization of the weight matrix (section 2.3), trying to find a common factor $\breve{\mathbf{\Phi}}$ such that $\mathbf{W}_i \approx \breve{\mathbf{\Phi}}\breve{\mathbf{\Lambda}}_i\breve{\mathbf{\Phi}}^\top$ by solving

$$\min_{\breve{\mathbf{\Phi}}\in\mathbb{R}^{n\times k},\breve{\mathbf{\Lambda}}_i\in\mathbb{R}^{k\times k}} \sum_{i=1}^{p} \|\mathbf{W}_i - \breve{\mathbf{\Phi}}\breve{\mathbf{\Lambda}}_i\breve{\mathbf{\Phi}}^\top\|_{\mathrm{F}}^2, \tag{6.2}$$

using the quasi-Newton method. Besides the fact that the factorization is applied to the weight matrix (it can be equivalently applied to the Laplacian), we see here a (non-orthogonal) joint diagonalization problem with an off-diagonality criterion considered by Yeredor [2002].

MVSC. Cai et al. [2011] proposed a method for *multi-view spectral clustering*

[1] We refer the reader to [Bronstein and Glashoff, 2013] for a recent attempt to generalize Laplacian averaging methods to a more general setting of sparse coupling, through heat kernel coupling.

(MVSC) by solving[2]

$$\min_{\breve{\mathbf{\Phi}}_i, \breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \sum_{i=1}^{p} \operatorname{tr}(\breve{\mathbf{\Phi}}_i^\top \mathbf{L}_i \breve{\mathbf{\Phi}}_i) + \mu \|\breve{\mathbf{\Phi}}_i - \breve{\mathbf{\Phi}}\|_{\mathrm{F}}^2 \tag{6.3}$$

and show that this problem can be equivalently posed as

$$\max_{\breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \operatorname{tr}\left(\breve{\mathbf{\Phi}}^\top \textstyle\sum_{i=1}^{p} (\mathbf{L}_i + \mu \mathbf{I})^{-1} \breve{\mathbf{\Phi}}\right) \tag{6.4}$$

We observe that problem (6.3) consists of p minimum-eigenvalue problems w.r.t. bases $\breve{\mathbf{\Phi}}_i$, with the addition of a coupling term, encouraging $\breve{\mathbf{\Phi}}_i$ to be as close as possible to some common basis $\breve{\mathbf{\Phi}}$ (note that the authors do not impose orthogonality constraints $\breve{\mathbf{\Phi}}_i^\top \breve{\mathbf{\Phi}} = \mathbf{I}$, but for $\mu \to \infty$, the proximity to orthogonal $\breve{\mathbf{\Phi}}$ makes $\breve{\mathbf{\Phi}}_i$ approximately orthogonal). Thus, it is possible to interpret (6.3) as a kind of joint diagonalization criterion similar to manifold alignment discussed in Section 6.1.2.

Problem (6.4) can be rewritten as a minimum eigenvalue problem

$$\min_{\breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \operatorname{tr}\left(\breve{\mathbf{\Phi}}^\top \left(\textstyle\sum_{i=1}^{p} (\mathbf{L}_i + \mu \mathbf{I})^{-1}\right)^{-1} \breve{\mathbf{\Phi}}\right), \tag{6.5}$$

whose solution is given by the first k eigenvectors of the matrix $\left(\sum_{i=1}^{p} (\mathbf{L}_i + \mu \mathbf{I})^{-1}\right)^{-1}$. For $\mu = 0$, this is simply the harmonic mean of the Laplacians. In order to obtain the limit case $\mu \to \infty$, observe that

$$\begin{aligned} \sum_{i=1}^{p} (\mathbf{L} + \mu \mathbf{I})^{-1} &= \tfrac{1}{\mu} \sum_{i=1}^{p} (\tfrac{1}{\mu} \mathbf{L} + \mathbf{I})^{-1} \approx \tfrac{1}{\mu} \sum_{i=1}^{p} \mathbf{I} - \tfrac{1}{\mu} \mathbf{L} \\ &= \tfrac{p}{\mu} \mathbf{I} - \tfrac{1}{\mu} \sum_{i=1}^{p} \mathbf{L}. \end{aligned} \tag{6.6}$$

[2] Cai et al. [2011] also impose a non-negativity constraint on the matrix $\breve{\mathbf{\Phi}}$ in order to obtain cluster indicators directly and bypass the K-means clustering stage [Hochbaum and Shmoys, 1985]. We ignore this additional constraint for the simplicity of discussion; such a constraint can be added to all the problems discussed in the following.

Plugged into (6.4) and normalized, in the limit $\mu \to \infty$ expression (6.6) becomes

$$\min_{\breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \operatorname{tr}\left(\breve{\mathbf{\Phi}}^\top \sum_{i=1}^{p} \mathbf{L}_i \breve{\mathbf{\Phi}}\right) \tag{6.7}$$

thus essentially boiling down to arithmetic mean of the Laplacians. The same result can be obtained by analyzing (6.3) and noticing that for $\mu \to \infty$ we have $\breve{\mathbf{\Phi}}_i = \breve{\mathbf{\Phi}}$ and the problem becomes

$$\min_{\breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \operatorname{tr}\left(\breve{\mathbf{\Phi}}^\top \sum_{i=1}^{p} \mathbf{L} \breve{\mathbf{\Phi}}\right).$$

Co-regularization. Kumar et al. [2011] proposed the *centroid co-regularization* approach for multimodal clustering based on the minimization:

$$\min_{\breve{\mathbf{\Phi}}_i^\top \breve{\mathbf{\Phi}}_i = \mathbf{I},\ \breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \sum_{i=1}^{p} \operatorname{tr}(\breve{\mathbf{\Phi}}_i^\top \mathbf{L} \breve{\mathbf{\Phi}}_i) - \mu \operatorname{tr}(\breve{\mathbf{\Phi}}_i \breve{\mathbf{\Phi}}_i^\top \breve{\mathbf{\Phi}} \breve{\mathbf{\Phi}}^\top). \tag{6.8}$$

This function is alternatingly minimized, first with respect to the $n \times k$ matrices $\breve{\mathbf{\Phi}}_i$, then with respect to $\breve{\mathbf{\Phi}}$. The term $\operatorname{tr}(\breve{\mathbf{\Phi}}_i \breve{\mathbf{\Phi}}_i^\top \breve{\mathbf{\Phi}} \breve{\mathbf{\Phi}}^\top)$ measures the Grassmanian distance between the column subspaces $\operatorname{span}\{\breve{\boldsymbol{\phi}}_{i1}, \ldots, \breve{\boldsymbol{\phi}}_{ik}\}$ and $\operatorname{span}\{\breve{\boldsymbol{\phi}}_1, \ldots, \breve{\boldsymbol{\phi}}_k\}$ [Dong et al., 2014] and has an effect similar to our coupling term.

SC-ML. Dong et al. [2014] proposed an approach for *spectral clustering on multi-layer graphs* (SC-ML), trying to find an $n \times k$ matrix $\breve{\mathbf{\Phi}}$ which minimizes that Laplacian quadratic form and is closest to the subspaces spanned by the first k eigenvectors $\mathbf{\Phi}_i$ of the Laplacians $\mathbf{L}$,

$$\min_{\breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \sum_{i=1}^{p} \operatorname{tr}(\breve{\mathbf{\Phi}}^\top \mathbf{L} \breve{\mathbf{\Phi}}) - \mu \operatorname{tr}(\mathbf{\Phi}_i \mathbf{\Phi}_i^\top \breve{\mathbf{\Phi}} \breve{\mathbf{\Phi}}^\top). \tag{6.9}$$

Rewriting (6.9) as

$$\min_{\breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}} \operatorname{tr}\left(\breve{\mathbf{\Phi}}^\top \left(\sum_{i=1}^{p} \mathbf{L} - \mu \mathbf{\Phi}_i \mathbf{\Phi}_i^\top\right) \breve{\mathbf{\Phi}}\right), \tag{6.10}$$

one obtains a closed-form solution to $\breve{\mathbf{\Phi}}$ by finding the first k eigenvectors of the

matrix $\sum_{i=1}^{p} \mathbf{L} - \mu \boldsymbol{\Phi}_i \boldsymbol{\Phi}_i^\top$.

We must stress that these methods were developed for clustering problems where one has to find the null eigenvectors, and do not adapt easily to other applications of diffusion geometry where one has to find many or all joint eigenvectors of the Laplacians (*e.g.*, computation of diffusion distances). In particular, iterative solvers used in [Tang et al., 2009; Kumar et al., 2011; Cai et al., 2011] do not scale up to such cases.

CCO. Using the relation between joint diagonalizability and commutativity [Lin, 1997; Glashoff and Bronstein, 2013], Bronstein et al. [2013] considered a class of problems referred to as *closest commuting operators* (CCO). Bronstein et al. [2013] find the closest commuting matrices $\tilde{\mathbf{L}}_X, \tilde{\mathbf{L}}_Y$ to the given $\mathbf{L}_X$ and $\mathbf{L}_Y$,

$$C(\mathbf{L}_X, \mathbf{L}_Y) = \min_{\tilde{\mathbf{L}}_X, \tilde{\mathbf{L}}_Y \in \mathscr{M}} \|\tilde{\mathbf{L}}_X - \mathbf{L}_X\|_{\mathrm{F}}^2 + \|\tilde{\mathbf{L}}_Y - \mathbf{L}_Y\|_{\mathrm{F}}^2 \quad \text{s.t. } \left[\tilde{\mathbf{L}}_X, \tilde{\mathbf{L}}_Y\right] = 0, \tag{6.11}$$

where $\mathscr{M}$ is some space of matrices. The matrices $\tilde{\mathbf{L}}_X, \tilde{\mathbf{L}}_Y$ minimizing 6.11 are called the *closest commuting operators* (CCO). In the case $\mathscr{M} = \mathbb{R}^{n \times n}$, if $\mathbf{L}_X, \mathbf{L}_Y$ approximately commute, then $C(\mathbf{L}_X, \mathbf{L}_Y)$ is guaranteed to be small [Lin, 1997]. Moreover, in this case the JADE and CCO problems are equivalent [Bronstein et al., 2013], in the following sense: since the minimizers of (6.11) are commuting matrices, they are jointly diagonalizable, and their joint eigenbasis is the minimizer of the JADE optimization problem (3.3) [Bronstein et al., 2013].

The main advantage of problem 6.11 is that we are able to find matrices from the *desired* space $\mathscr{M}$ (*e.g.*, space of all admissible Laplacians) that are close to the input matrices and which are jointly diagonalizable. The solution of (6.11) is carried out by parametrizing $\tilde{\mathbf{L}}_i$ through the non-zero elements of the adjacency matrix $\mathbf{W}_i$ (section 2.3). The complexity of the problem depends both on the *size* and the *structure* of $\mathbf{W}_i$: assuming that each row of the adjacency matrix has at most s non-zero elements, computing the cost function and the constraints and their gradients requires $\mathscr{O}(sn^2)$ operations.

6.1.2 Sparse coupling setting

Manifold alignment. Ham et al. [2005] introduced *manifold alignment* as

a way to construct embeddings that are consistent in two different modalities. Let us be given two weighted adjacency graphs with n vertices sampled from manifolds $\mathscr{M}_1$ and $\mathscr{M}_2$, and let us assume w.l.o.g. that the points are ordered such that the first l points in the two modalities correspond. The main idea of manifold alignment is to construct a big graph with $2n$ vertices where the edges connecting corresponding points in different modalities have some weight μ. The joint Laplacian of such a graph is a $2n \times 2n$ matrix of the form

$$\hat{\mathbf{L}} = \begin{pmatrix} \mathbf{L}_1 + \mu\mathbf{Q} & -\mu\mathbf{Q} \\ -\mu\mathbf{Q} & \mathbf{L}_2 + \mu\mathbf{Q} \end{pmatrix} \tag{6.12}$$

where $\mathbf{Q}$ is an $n \times n$ diagonal matrix with first l diagonal elements equal to one and the rest to zero. Ham et al. [2005] then compute the eigenmap of the joint Laplacian

$$\min_{\breve{\mathbf{\Phi}} \in \mathbb{R}^{2n \times k}} \operatorname{tr}(\breve{\mathbf{\Phi}}^\top \hat{\mathbf{L}} \breve{\mathbf{\Phi}}) \text{ s.t. } \breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}} = \mathbf{I}, \tag{6.13}$$

and use the rows $1, \dots, n$ and $n+1, \dots, 2n$ of $\breve{\mathbf{\Phi}}$ as the k-dimensional embeddings of manifolds $\mathscr{M}_1$ and $\mathscr{M}_2$, respectively. Larger values of μ ensure that the embedding coordinates of the corresponding points coincide.

Denoting $\breve{\mathbf{\Phi}} = \left[\breve{\mathbf{\Phi}}_1; \breve{\mathbf{\Phi}}_2\right]$[3] we can rewrite (6.13) as

$$\begin{aligned} \min_{\breve{\mathbf{\Phi}}_i \in \mathbb{R}^{n \times k}} & \operatorname{tr}(\breve{\mathbf{\Phi}}_1^\top \mathbf{L}_1 \breve{\mathbf{\Phi}}_1) + \operatorname{tr}(\breve{\mathbf{\Phi}}_2^\top \mathbf{L}_2 \breve{\mathbf{\Phi}}_2) + \mu \|\mathbf{Q}\breve{\mathbf{\Phi}}_1 - \mathbf{Q}\breve{\mathbf{\Phi}}_2\|_{\mathrm{F}}^2 \\ & \text{s.t. } \breve{\mathbf{\Phi}}_1^\top \breve{\mathbf{\Phi}}_1 + \breve{\mathbf{\Phi}}_2^\top \breve{\mathbf{\Phi}}_2 = \mathbf{I}, \end{aligned} \tag{6.14}$$

We recognize in the first terms the cost used in the classical eigenmap (Equation 2.12). In the case $\mu = 0$, $\hat{\mathbf{L}}$ becomes a block-diagonal matrix and $\breve{\mathbf{\Phi}}_i = \mathbf{\Phi}_i$ the eigenvectors of the Laplacians $\mathbf{L}_i$. For $\mu > 0$, the problem 6.14 becomes similar to our coupled diagonalization problem (Equation 4.1) with $\mathrm{off}_2 = \operatorname{tr}(\cdot)$ as the off-diagonality penalty. An important difference, however, is that $\breve{\mathbf{\Phi}}_i$ in this formulation are not orthonormal (the orthogonality constraint is on their sum).

Finally, observe that in the limit case $\mu \to \infty$ and $\mathbf{Q} = \mathbf{I}$, problem (6.13)

[3] We follow the Matlab notation here.

becomes (up to scaling) the minimum eigenvalue problem

$$\min_{\breve{\mathbf{\Phi}}^\top \breve{\mathbf{\Phi}}=\mathbf{I}} \quad \mathrm{tr}(\breve{\mathbf{\Phi}}^\top(\mathbf{L}_1+\mathbf{L}_2)\breve{\mathbf{\Phi}}) \tag{6.15}$$

for the matrix $\mathbf{L}_1+\mathbf{L}_2$. Thus, in this case manifold alignment boils down to arithmetic mean of the Laplacians.

Procrustes analysis. Wang and Mahadevan [2008] proposed to align the low-dimensional embeddings of two modalities by solving the orthogonal Procrustes problem. Note that this problem is a particular setting of our coupled joint diagonalization problem (4.2) for $p=2, k'=k$ and $\mu_c \to \infty$: in this case, we can ignore the off-diagonality penalty and obtain

$$\min_{\mathbf{R}_i^\top \mathbf{R}_i=\mathbf{I}} \|\mathbf{F}_1^\top \mathbf{A}_1 \bar{\mathbf{\Phi}}_1 \mathbf{R}_1 - \mathbf{F}_2^\top \mathbf{A}_2 \bar{\mathbf{\Phi}}_2 \mathbf{R}_2\|_{\mathrm{F}}^2. \tag{6.16}$$

Using the invariance of the Frobenius norm under orthogonal transformation, we can rewrite problem 6.16 as the orthogonal Procrustes problem

$$\min_{\mathbf{\Omega}^\top \mathbf{\Omega}=\mathbf{I}} \|\mathbf{F}_1^\top \mathbf{A}_1 \bar{\mathbf{\Phi}}_1 - \mathbf{F}_2^\top \mathbf{A}_2 \bar{\mathbf{\Phi}}_2 \mathbf{\Omega}\|_{\mathrm{F}}^2, \tag{6.17}$$

where $\mathbf{\Omega}=\mathbf{R}_2\mathbf{R}_1^\top$.

The problem 6.17 has an analytical solution (Section 4.4), where $\mathbf{R}_1=\mathbf{\Omega}_{\mathrm{R}}$, $\mathbf{R}_2=\mathbf{\Omega}_{\mathrm{L}}$ are right and left singular vectors of matrix $\mathbf{M}=\mathbf{F}_2^\top \mathbf{A}_2 \bar{\mathbf{\Phi}}_2 \bar{\mathbf{\Phi}}_1^\top \mathbf{A}_1 \mathbf{F}_1^\top$ [Schönemann, 1966], respectively.

Matrices $\mathbf{R}_1$, $\mathbf{R}_2$ have a geometric interpretation of rotations of respective Laplacian bases to coincide in the best way at the corresponding points ($\mathbf{F}_1$, $\mathbf{F}_2$ are delta functions of corresponding points). Note that the solution is not unique but ambiguous up to a rotation $\mathbf{P}$, i.e., $\mathbf{R}_1\mathbf{P}, \mathbf{R}_2\mathbf{P}$ is also a solution.

Note the resulting coupled bases are not guaranteed to be quasi-harmonic (*e.g.*, they possess a low-pass filter property) since the off-diagonality penalty on diagonalization of Laplacian matrices is not used.

6.2 Experiments and Results

In this section, we show several examples of the application of our simultaneous diagonalization approach in machine learning and computer vision. The main context of the experiments is, given several graphs ('modalities') representing same or related data in slightly different ways, to take advantage of this multimodal information.

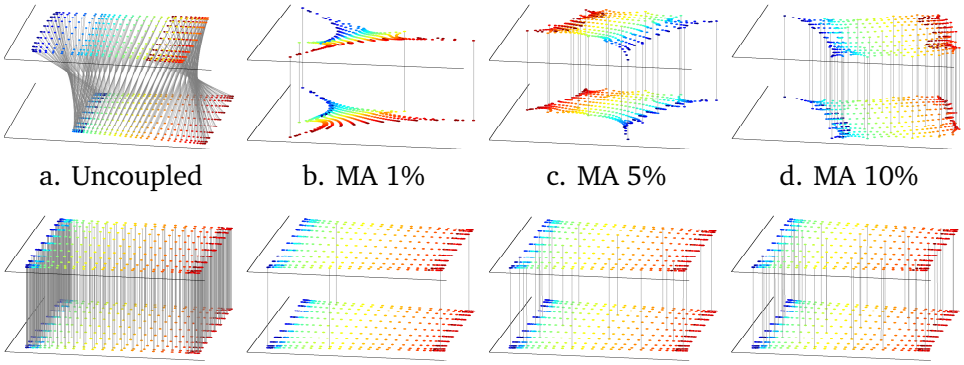
The experiments are structured as follows: Firstly, we show examples of dimensionality reduction by embedding the multimodal data into low-dimensional spaces using joint Laplacian eigenvectors. We show that the eigenmaps of different modalities are well aligned in this way. These results are mostly qualitative. Then, we compare different state-of-the-art multimodal clustering algorithms on standard datasets. Afterwards, we compute diffusion distances in the joint Laplacian eigenspaces and use them to classify objects. Later, we show the example of applying diffusion distances for meaningful subsampling of the datasets using the farthest point sampling (FPS) technique [Hochbaum and Shmoys, 1985]. Finally, we analyze the complexity of our approach.

6.2.1 Dimensionality reduction

Swiss rolls. In this experiment, we used two Swiss roll surfaces with slightly different embedding as two different data modalities. The rolls were constructed in such a way that in each modality there is topological noise (connectivity "across" the roll loops) at different points. The rolls contained $n = 451$ points. Laplacians were constructed as in [Belkin and Niyogi, 2002], using 8-neighbor connectivity and Gaussian weights with scale parameter $\sigma = \sqrt{\frac{5}{2}}$ (Figure 2.1, leftmost). Using individual modalities, the difference in the connectivity produces different eigenvectors (*e.g.*, Figure 3.1, top), with respect to both their order and their behavior (values across two different faces are closer where there are links). When using joint eigenvectors (*e.g.*, Figure 3.1, bottom), instead, we are able to correctly capture the intrinsic structure of the data (*i.e.*, links across faces are not influent anymore), and the eigenvectors behave the same way. This effect is evident in Figure 6.1, where the second and third uncoupled (a) and joint (e)

eigenvectors of the same rolls are plotted.

Sparsely-coupled Swiss rolls. Next, we repeat the same experiment using correspondence between a small subset of vertices (sparse coupling) rather than all the points. The corresponding sparse points were sampled using FPS [Hochbaum and Shmoys, 1985]. Since the rolls are (up to topological noise) isometric to a plane, their ideal embeddings should be rectangular patches. Figure 6.1 (f-h) shows the result of joint embedding using our CD with sparse pointwise correspondence. With as little as 1% correspondences, we obtain results similar to JADE (which uses full coupling). Figure 6.1 (b-d) shows the result of manifold alignment (MA) [Ham et al., 2005] with the same sparse correspondences. It is evident that MA requires many more points to achieve results similar to CD.



a. Uncoupled b. MA 1% c. MA 5% d. MA 10%

e. JADE f. CD 1% g. CD 5% h. CD 10%

Figure 6.1. Simultaneous two-dimensional embedding of two Swiss rolls with slightly different connectivity using joint diagonalization with generalized Jacobi method (JADE), coupled diagonalization (CD) and manifold alignment (MA). Ideally, the embedding should 'unroll' the rolls into rectangular regions, and the embeddings of the two modalities should coincide. Using the same sparse coupling (from 10% to 1% of the points, shown in gray lines), CD produces a significantly better alignment than MA.

Alignment of visual manifolds. As an additional comparison of CD and MA, we reproduce the problem of alignment of two visual manifolds using the data of [Ham et al., 2005]: 831 120×100 images of a face and 698 64×64 images of a statue. The datasets were coupled sampling 25 points from the statue dataset with FPS [Hochbaum and Shmoys, 1985] and then manually matching them

with corresponding images in the faces dataset. Figure 6.2 shows the result of the alignment of face (green) and statue (blue) manifolds. As an example, we took 6 face pictures in different poses (green circles) and showed them, for both methods, next to their closest counterparts on the statue manifold (blue circles). We observe that, with the same number of correspondences, the alignment of the two manifolds is significantly better using CD compared to MA: as a consequence, pictures in the statue dataset tend to be closer to pictures of faces in the same pose.

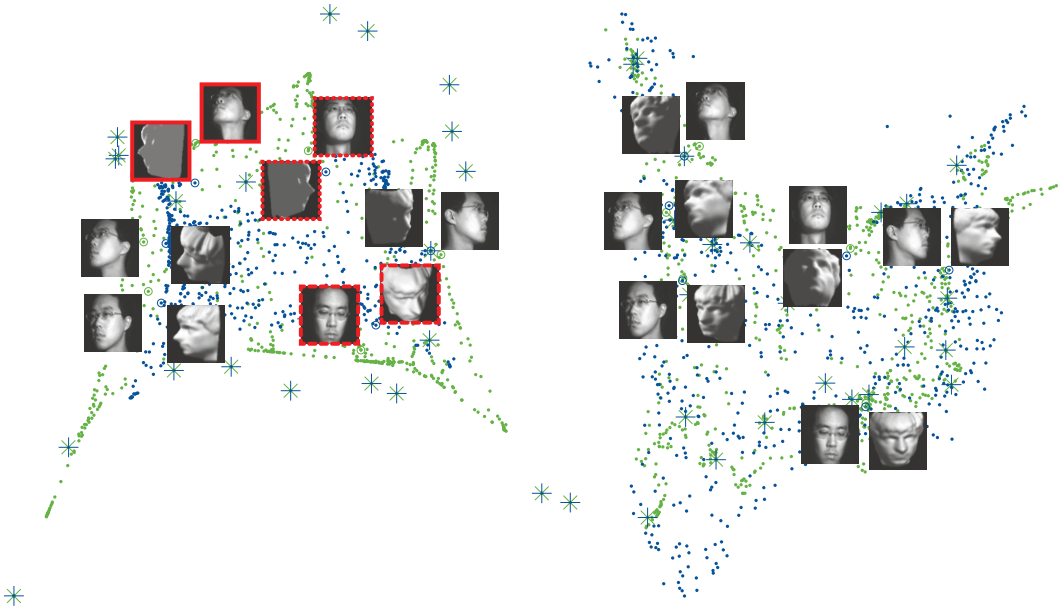


Figure 6.2. Alignment of face (green) and statue (blue) manifolds. Each point represents an image in the respective dataset; circles represent corresponding poses of the statue and face images shown. Crosses denote the data points used for coupling. Note some significant misalignment between the manifolds in the MA results (marked in red).

6.2.2 Multimodal clustering

We performed multimodal spectral clustering on six different multimodal datasets. *Circles* and *Text* are two synthetic datasets purposely built to be noisy in each modality (overlapping clusters) and to have modalities that disambiguate each other (clusters which are close in one modality are far apart in the other one, see

Figure 6.3).

NUS is a subset of the NUS-WIDE dataset [Chua et al., 2009] containing images (represented by 64-dimensional color histograms) and their text annotations (represented by 1000-dimensional bags of words). Images were purposely selected to have ambiguous content and annotations (*e.g.*, swimming tigers are also tagged as "water" making them confuse, *e.g.*, with whales). *Caltech* is a subset of the Caltech-101 dataset with the same 7 image classes as in Cai et al. [2011]. For each image, kernels arising from different visual descriptors were given [Pinto, 2009]: we chose the bio-inspired features and 4x4 pyramid histogram of visual words (PHOW) as different modalities. *Digits* is the UCI Handwritten Digits dataset [Alpaydin and Kaynak, 1998; Liu et al., 2013], represented using 76 Fourier coefficients and the 240 pixel averages in 2×3 windows. *Reuters* is a subset of the Reuters multilingual text collection [Amini et al., 2009; Liu et al., 2013] using the English and French languages as two different modalities.

Laplacians were constructed using the Gaussian weight selected with a self-tuning scale [Perona and Zelnik-Manor, 2004]. Spectral clustering was performed independently on each modality (Uncoupled), on the joint eigenspace calculated with JADE (Section 3.1, [Cardoso and Souloumiac, 1996]), and on the coupled bases calculated using CD with coupling only (pos, $\mu_d = 0$; Section 4.1, Equation 4.3) as well as decoupling (pos+neg) terms. Sparse sets of corresponding points for coupling were generated using FPS [Hochbaum and Shmoys, 1985] on each cluster with random initial point. The results were averaged over ten runs with different sampling. Negatives were generated by choosing blobs of points belonging to ambiguous sets (*e.g.*, clusters 5, 6, and 7 for NUS, and clusters 4 and 5 for Caltech).

For reference, we show the performance of the following state-of-the-art multiview clustering methods: Comraf [Bekkerman and Jeon, 2007], MVSC [Cai et al., 2011], MultiNMF [Liu et al., 2013], and SC-ML [Dong et al., 2014].[4] We further compare with the two Laplacian averaging methods (harmonic mean and arithmetic mean). Clustering quality was measured using two standard cri-

[4] Since Comraf and Multi-NMF methods require explicit coordinates of the data points, while *Caltech* data is represented implicitly as kernels, we could not measure performance on this dataset.

teria used in the evaluation of clustering algorithms: the *micro-averaged accuracy* [Bekkerman and Jeon, 2007] and the *normalized mutual information* (NMI) [Manning et al., 2008].

Figure 6.3 visualizes the behavior of different multimodal clustering algorithms (all assuming the full coupling setting) on the synthetic datasets *Circles* and *Text*: due to the non-globular shapes of the clusters and their overlap, both the unimodal approach and the non-spectral multimodal ones perform poorly on these datasets. Clusters found by multimodal spectral clustering methods, instead, are all quite accurate, and JADE performs the best among them.

Figures 6.4 and 6.5 visualize the results of unimodal spectral clustering with its multimodal extension (calculated using JADE) on the NUS and Caltech dataset. One can easily see the advantage of using simultaneously information from both modalities: images which are ambiguous in either modality (*e.g.*, due to their colors, tags, or other visual features) are made unambiguous in the multimodal case.

Finally, Table 6.1 summarizes the quantitative evaluation of different clustering methods. In the full coupling setting, we observe that multimodal spectral methods perform consistently better on non-globular clusters and very noisy datasets. In particular, methods that might look naive such as harmonic and arithmetic mean provide surprisingly good results, competing with other much more elaborate approaches. In the sparse coupling setting (using correspondence between 10%-100% points), CD is able to obtain performances favourably comparable to the ones of full coupling methods.

6.2.3 Object classification

In this experiment, we used the diffusion distances computed using Laplacian eigenvectors (individual and joint). The distances were computed with the first 100 eigenvectors according to (Equation 2.14) using heat diffusion kernel $K(\lambda) = e^{-5\lambda}$ (Equation 2.13). Figure 6.6 shows the distance matrices between the objects in the *Caltech* (top) and *NUS* (bottom) datasets. Ideally, the distance matrix should contain zero blocks on the diagonal (objects of the same class) and non-zero elsewhere (objects from different classes). Thresholding these distances at

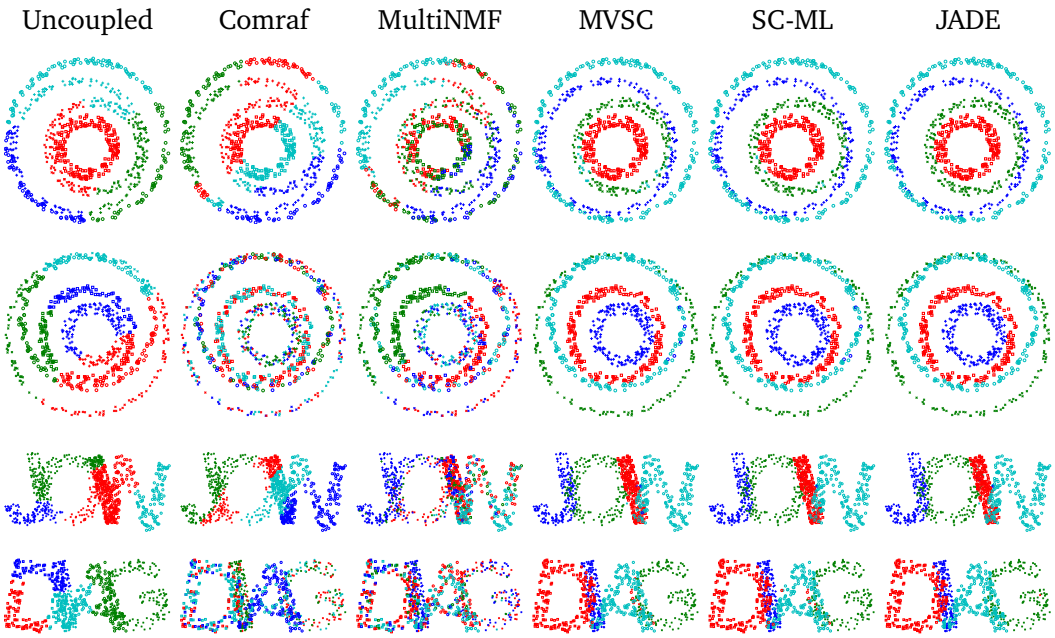


Figure 6.3. Clustering of synthetic multimodal datasets *Circles* (two modalities shown in first and second rows) and *Text* (third and fourth rows). Marker shape represents ground truth clusters; marker color represents the clustering results produced by different methods (ideally, all markers of one type should have only one color).

a set of levels and measuring the false positives/true positive rates (FPR/TPR), we produce the ROC curves that clearly indicate the advantage of using multiple modalities (see Figure 6.7).

6.2.4 Manifold subsampling

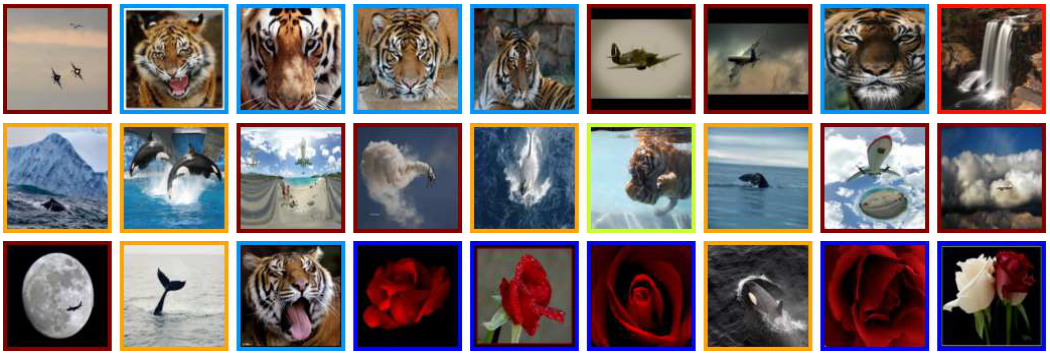
Next, we used the same diffusion distances to progressively sample the *Caltech* (top) and *NUS* (bottom) datasets using the farthest point sampling (FPS) strategy [Hochbaum and Shmoys, 1985]: starting with some point, pick up the second one as most distant from the first; then the third as the most distant from the first and second, and so on. Such sampling is almost-optimal and is known to produce a progressively refined r-covering of the dataset [Hochbaum and Shmoys, 1985]. Figure 6.8 shows that the first seven samples produced in this way cover all the classes present in the dataset, providing thus a meaningful subsampling.

	Method		Accuracy/NMI (%) Circles	Text	Caltech	NUS	Digits	Reuters
		#points	800	800	105	145	2000	600
		*Uncoupled**	53.0/39.5	60.4/50.9	78.1/77.0	80.7/81.9	78.9/82.2	52.3/41.1
		Harmonic Mean	95.6/90.1	97.2/91.0	87.6/82.5	89.0/83.8	87.0/86.3	52.3/40.9
		Arithmetic Mean	96.5/91.2	96.9/89.6	87.6/82.4	95.2/92.1	82.8/84.8	52.2/41.4
		Comraf	40.8/16.9	60.8/41.7	–[4]	86.9/84.3	81.6/77.0	53.2/30.7
		MVSC	95.6/90.1	97.2/91.0	81.0/79.4	89.0/83.4	83.1/84.9	52.3/40.9
		MultiNMF	41.1/14.2	50.5/23.2	–[4]	77.4/79.3	87.2/79.3	53.1/40.9
		SC-ML	98.2/94.6	97.6/92.1	88.6/82.6	94.5/90.7	87.8/85.3	52.8/38.4
		JADE	100/100	98.4/94.1	86.7/80.6	93.1/87.5	82.5/85.1	52.3/40.9
CD*	pos	10%	52.5/26.0	54.5/26.2	78.7/75.3	78.6/77.9	94.2/87.8	53.7/34.4
		20%	61.3/40.2	60.0/41.9	80.8/76.0	82.9/78.2	94.1/87.4	54.2/33.7
		60%	93.7/85.4	86.5/69.7	87.0/80.0	87.2/78.9	93.9/87.1	54.7/36.5
		100%	98.9/95.5	96.8/89.4	89.5/83.3	94.5/90.6	93.9/87.1	54.8/36.9
	pos+neg	10%	67.3/46.5	63.6/42.1	86.5/80.9	92.7/86.2	94.9/88.9	59.0/37.7
		20%	69.6/50.2	67.8/50.0	87.9/81.2	93.3/87.0	94.8/88.7	57.6/37.1
		60%	95.2/87.9	87.0/68.5	89.2/84.0	94.5/88.5	94.8/88.7	57.0/38.8

Table 6.1. Performance of different multimodal clustering methods of different datasets (accuracy / normalized mutual information in %, the higher the better). *Best performing modality is shown.

This is an indication of the presence of data clusters in the coupled eigenspace which are cohesive (points in the same class are close to each other) and at the same time well separated (points in different classes are far from each other).

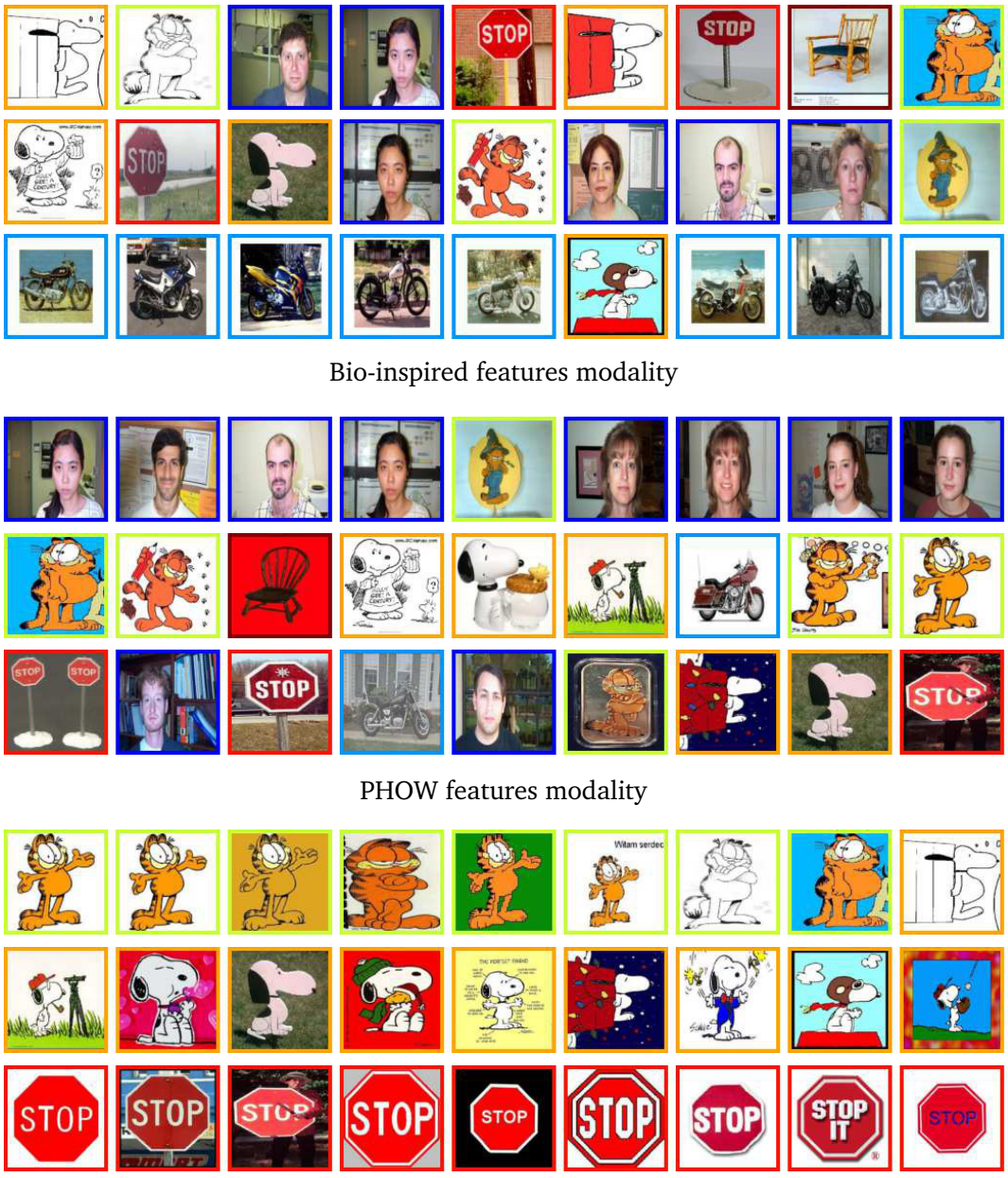
Tags modality



Color histogram modality

Multimodal (JADE)

Figure 6.4. Spectral clustering of the NUS dataset. Shown are a few images (randomly sampled) attributed to a single cluster by spectral clustering using the Tags modality only (top), the Color modality only (middle) and the Tags + Color multimodal clustering using JADE (bottom). Groundtruth clusters are shown in different colors. Note the ambiguities in the Tag-based clustering (*e.g.*, swimming tigers and underwater scenes) and Color-base clustering (*e.g.*, yellowish tigers and autumn scenes).



Bio-inspired features modality

PHOW features modality

Multimodal (JADE)

Figure 6.5. Spectral clustering of the Caltech-7 dataset. Shown are a few images (randomly sampled) attributed to a single cluster by spectral clustering using the Bio-inspired modality only (top), the PHOW modality only (middle) and the multimodal clustering using JADE (bottom). Groundtruth clusters are shown in different colors. Ideally, a cluster should contain images from a single class only.

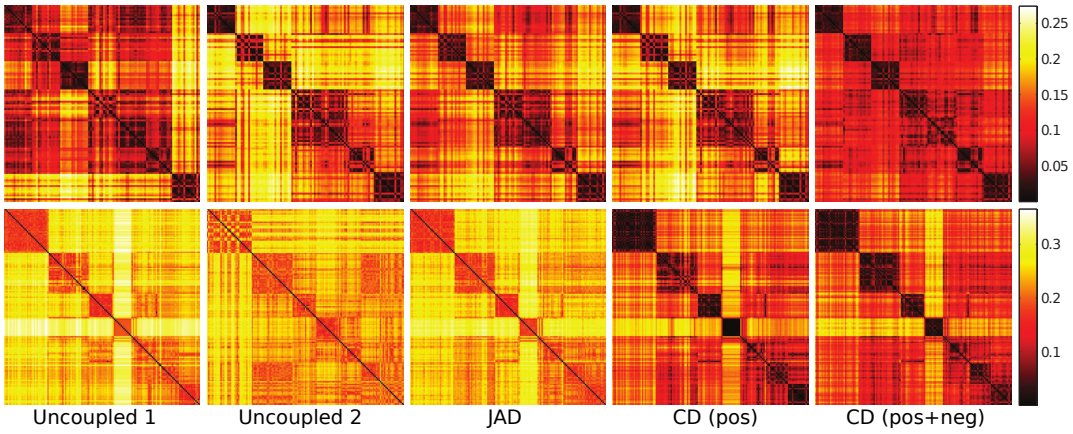


Figure 6.6. Diffusion distances between objects from the Caltech (top) and NUS (bottom) datasets using separate modalities (first and second columns), JADE (third column) and CD with coupling (fourth column) and coupling + decoupling (fifth column) terms. Note the ambiguities between different classes of objects (marked in cyan) when using a single modality.

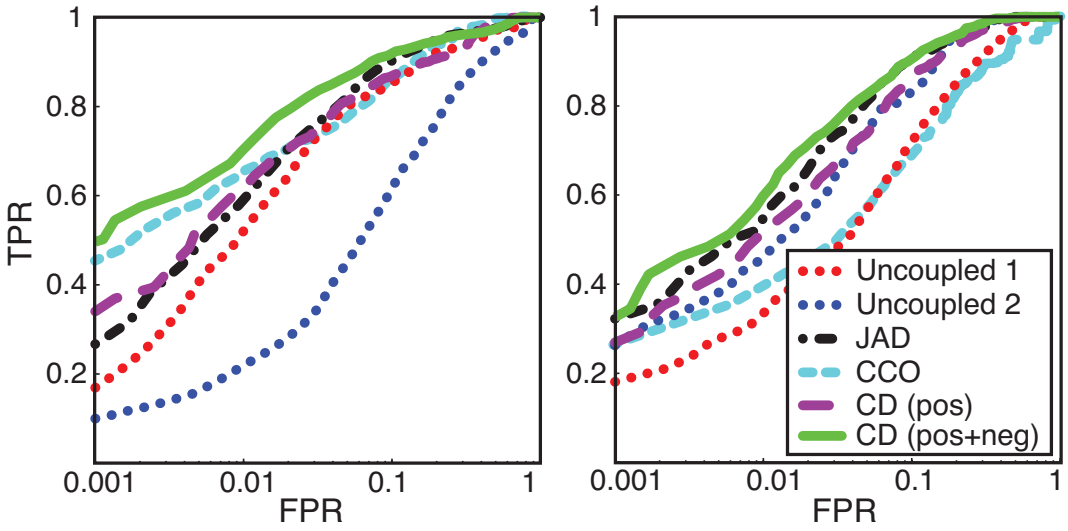


Figure 6.7. Object classification performance on Caltech (left) and NUS (right) datasets using diffusion distances computed in each modality separately (Uncoupled), a joint eigenspace (JADE), coupled eigenspaces produced by CD with coupling (pos) and coupling + decoupling (pos + neg) terms, and the joint eigenspace of the closest commuting Laplacians (CCO). Note that CD (pos + neg) performs better than each modality on its own and outperforms the other methods.

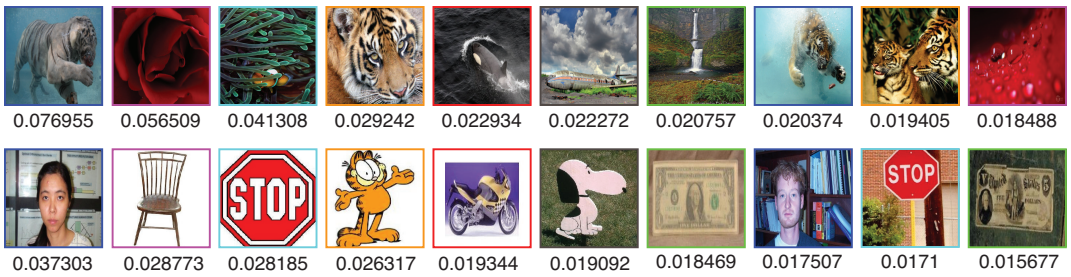


Figure 6.8. Farthest point sampling of NUS (top) and Caltech-7 (bottom) datasets using the diffusion distance in the joint eigenspace computed by JADE. First point is on the left. Numbers indicate the sampling radius. Note that in both cases, the first seven samples cover all the image classes, providing a meaningful subsampling of the respective datasets.

Chapter 7

Applications in Shape Analysis

In this chapter, we apply the coupled diagonalization method developed in Chapter 4 to several challenging problems from the domain of computer graphics. In particular, we consider the problems of shape correspondence, similarity and pose transfer. The chapter is based on our EUROGRAPHICS paper [Kovnatsky et al., 2013]. Some additional results showing multiple shape correspondence with a non-smooth version of the coupled diagonalization problem are from [Kovnatsky et al., 2016].

7.1 Spectral methods in shape analysis

The Laplace-Beltrami operator is perhaps one of the most widely used differential operators in the computer graphics field. Constructions based on the Laplace-Beltrami operator have been used in a wide range of applications, including remeshing [Kobbelt, 1997; Nealen et al., 2006], parametrization [Floater and Hormann, 2005], compression [Karni and Gotsman, 2000; Váša et al., 2014], shape recognition [Reuter et al., 2006; Rustamov, 2007], deformation transfer [Zhou et al., 2005], mesh editing [Sorkine et al., 2004], watermarking for copyright protection [Ohbuchi et al., 2001, 2002], triangle mesh optimization [Nealen et al., 2006], filtering [Vallet and Lévy, 2008; Kovnatsky et al., 2012], shape correspondence and similarity [Reuter et al., 2006; Bronstein et al., 2010a; Ovsjanikov et al., 2012; Kovnatsky et al., 2013], shape segmentation [Reuter, 2010;

Sharma et al., 2009], isometric embeddings of shapes [Belkin and Niyogi, 2002; Rustamov, 2007], diffusion metrics [Coifman et al., 2005], and shape descriptors [Sun et al., 2009; Bronstein and Kokkinos, 2010; Aubry et al., 2011]. For a comprehensive review of the applications of the Laplace-Beltrami operator in computer graphics, we refer the reader to [Lévy and Zhang, 2009; Sorkine, 2006]. Many of the above methods draw inspiration from physics; notable examples include heat diffusion [Coifman et al., 2005] or wave propagation [Aubry et al., 2011] phenomena on manifolds.

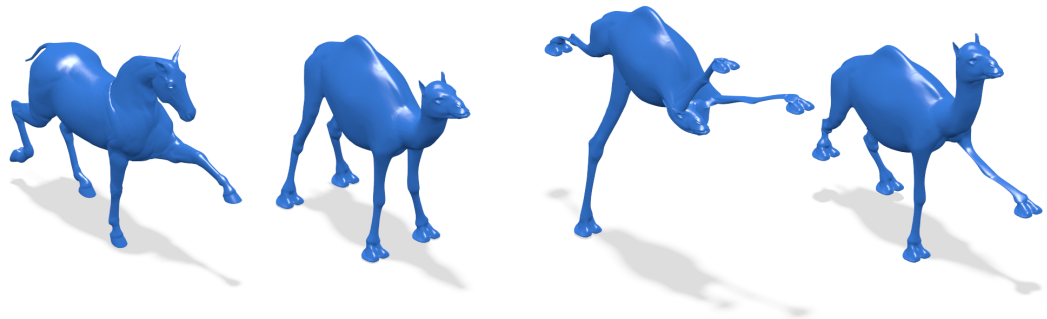
In a seminal paper, Taubin [1995] uses the Laplace-Beltrami eigenfunctions as an orthonormal basis (often referred to in this field as *manifold harmonics*) to perform Fourier analysis on manifolds, analogously to classical signal processing. The orthonormal Laplacian eigenfunctions replace the standard Fourier basis $e^{-i\omega x}$, and their corresponding eigenvalues play the role of frequencies. In fact, the standard Fourier basis functions are the eigenfunctions of the Euclidean 1D Laplacian operator, and spherical harmonics of that of the sphere. The analogy between manifold harmonics and classical signal processing was exploited in [Kim and Rossignac, 2005; Lévy, 2006; Vallet and Lévy, 2008; Lévy and Zhang, 2009] to perform shape filtering and editing in the spectral domain.

Of particular interest in this thesis are settings dealing with multiple shapes. In Chapter 1, we already mentioned the work of Lévy [2006] proposing a pose transfer approach based on the Fourier decomposition of the manifold embedding coordinates. Given two shapes represented as manifolds $\mathscr{X}$ and $\mathscr{Y}$ embedded in $\mathbb{R}^3$. We denote by $\mathbf{X}$ and $\mathbf{Y}$ the Euclidean embeddings of manifolds $\mathscr{X}$ and $\mathscr{Y}$ (which we consider here as 3-dimensional vector fields on the manifolds), and by $\{\boldsymbol{\phi}_i\}_{i\geqslant 1}$ and $\{\boldsymbol{\psi}_i\}_{i\geqslant 1}$ their Laplacian eigenbases, respectively. Decomposing $\mathbf{X}$ and $\mathbf{Y}$ in the respective eigenbases, one obtains

$$\mathbf{X} = \sum_{i\geqslant 1} \mathbf{a}_i \boldsymbol{\phi}_i, \tag{7.1}$$

$$\mathbf{Y} = \sum_{i\geqslant 1} \mathbf{b}_i \boldsymbol{\psi}_i, \tag{7.2}$$

where $\mathbf{a}_i$, $\mathbf{b}_i$ denote the three-dimensional vectors of the Fourier coefficients corresponding to each embedding coordinate. Next, a new shape is composed ac-



Source pose Target shape Eigenvectors Coupled bases

Figure 7.1. Pose transfer from horse (leftmost) to camel shape (second from left) by substituting the first 6 Fourier coefficients in the decomposition of extrinsic coordinates of the shape in the Laplacian eigenbasis as done in [Lévy, 2006] (second from right) and coupled basis (rightmost).

cording to

$$\mathbf{Z} = \sum_{i=1}^{k} \mathbf{a}_i \boldsymbol{\phi}_i + \sum_{i>k} \mathbf{b}_i \boldsymbol{\psi}_i, \tag{7.3}$$

with the first k low frequency coefficients taken from $\mathscr{X}$, and higher frequencies taken from $\mathscr{Y}$. This transfers the "layout" (pose) of the shape $\mathscr{X}$ to the shape $\mathscr{Y}$ while preserving the geometric details of $\mathscr{Y}$. Lévy [2006] notes that this method works only when the corresponding Laplacian eigenfunctions behave consistently, *i.e.*, when the coefficients $\mathbf{a}_i$ and $\mathbf{b}_i$ are expressed in the same "language". Since the two sets of eigenvectors are computed independently, this cannot be guaranteed: it is sufficient to have sign flips to produce a meaningless result (Figure 7.1, third from left).

This is exactly the problem that our coupled diagonalization procedure attempts to cure: using coupled bases instead of the standard Laplacian eigenbases, the new shape becomes meaningful (pose of the horse and details of the camel, see Figure 7.1, rightmost).

7.2 Experiments and Results

In this section, we show several examples of coupled bases construction, and applications of the proposed approach. We used shapes from publicly available datasets [Bronstein et al., 2008; Sumner and Popović, 2004; Shilane et al., 2004]. From these shapes we created a dataset with assigned manually ground-truth point-wise correspondences. This dataset is available at JAD-data. Mesh sizes vary widely between $600 - 25$K vertices. In the following, unless we state otherwise, when we write that q corresponding points were used for coupling, we refer to the q points sampled with FPS algorithm [Hochbaum and Shmoys, 1985] and the ground-truth correspondence.

Discretization of the Laplace-Beltrami operator was done using the cotangent formula (Equation (2.9)) [Meyer et al., 2003]. In all our examples, we constructed coupled bases solving the coupled diagonalization problem (4.3) with off-diagonality penalty off_3, as described in Section 4.1; the values of $\mu_c = 0.132$ and $\mu_d = 0$ were used in all experiments unless stated otherwise. We use the off_3 off-diagonality cost because it outputs ordered coupled bases (in our case, we are interested in a few eigenvectors corresponding to the low part of the spectrum, rather than arbitrary few eigenvectors).

In our experiments, we used several solvers for the orthogonally-constrained coupled diagonalization problem: MATLAB `fmincon` with default settings (`active-set`) and Manopt optimization package [Boumal et al., 2014] (`conjugate gradient`) on Stiefel manifold. Typical time to compute 15 joint eigenvectors was about 45-60 sec for `fmincon` and 1-5 sec for Manopt. The codes for reproducing the experiments may be downloaded here: MATLAB JAD-code, Manopt JAD-code.

7.2.1 Coupled diagonalization

Isometric vs non-isometric. Figure 7.3 (top) shows examples of coupled diagonalization of Laplacians of different shapes: near isometric (two poses of an elephant) and non-isometric (elephant and horse), as well as multiple non-isometric shapes (four humanoids, where the coupling done using only 15 points). We computed the first $k = 20$ joint approximate eigenvectors. The

Laplacians are almost perfectly diagonalized by the obtained coupled bases in the case of near-isometric shapes. For non-isometric shapes, off-diagonal elements are more prominent; nevertheless, a clear diagonally-dominant structure is present.

Sensitivity to correspondence error. In this experiment, we computed the coupled bases for two near-isometric human shapes using point-wise coupling at $q = 10$ points with noisy correspondence that deviated from groundtruth correspondence by up to 15% of the geodesic diameter of the shape. Table 7.1 shows the obtained diagonalization quality (measured as the ratio of the norm of the off-diagonal and diagonal values averaged on two shapes), and Figure 7.4 depicts the approximately diagonalized Laplacians. This experiment illustrates that very few roughly corresponding points are required for the coupling term in our problem, and that the proposed procedure is robust to correspondence noise.

0%	3%	6%	15%
0.0417	0.0534	0.0892	0.1439

Table 7.1. Sensitivity of joint diagonalization quality (measured as the ratio of the norms of the off-diagonal and diagonal elements, $\|\text{off}\|/\|\text{diag}\|$) to correspondence error (measured as % of geodesic diameter).

Sensitivity to representation and sampling. Taking the shapes from the previous experiment, we downsampled and re-meshed one of them from 8K vertices to 800 vertices. In addition, we removed the triangulation and computed a Laplacian on the point cloud, using 7 nearest neighbors and Gaussian weight $w_{ij} = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2/3}$ (Equation (2.6)). Table 7.2 shows the quality of the joint diagonalization of the Laplacians on the original mesh (first row), subsampled mesh (second row) and the point cloud (third row). Figure 7.6 shows the obtained coupled bases. We see that the coupled bases still behave consistently even for a challenging scenario: different discretization and 10-times less sampling density.

7.2.2 Shape correspondence

Shape correspondence. One of the key applications of our approach is for the computation of functional correspondence [Ovsjanikov et al., 2012]. According

	full	**90%**	**75%**	**50%**	**10%**
mesh-mesh	0.0417	0.0414	0.0384	0.0429	0.0625
mesh-cloud	0.0934	0.1399	0.1265	0.1303	0.1424

Table 7.2. Sensitivity of the joint diagonalization quality (measured as the ratio of the norms of the off-diagonal and diagonal elements, $\|\text{off}\|/\|\text{diag}\|$) to sampling density for two meshes and mesh/point cloud.

to this approach, which we overviewed in Chapter 2, one has to solve the linear system of kq equations in k^2 variables,

$$\mathbf{G}^\top \mathbf{A}_Y \bar{\mathbf{\Psi}} = \mathbf{F}^\top \mathbf{A}_X \bar{\mathbf{\Phi}} \mathbf{C}_k, \tag{7.4}$$

where $\bar{\mathbf{\Phi}}^\top \mathbf{A}_X \mathbf{F}$ and $\bar{\mathbf{\Psi}}^\top \mathbf{A}_Y \mathbf{G}$ are $k \times q$ matrices of Fourier coefficients of q corresponding vectors $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_q]$ and $\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_q]$ on shapes $\mathscr{X}$ and $\mathscr{Y}$, respectively (we assume $\mathbf{TF} \approx \mathbf{G}$). The $k \times k$ matrix $\mathbf{C}_k$ is the spectral representation of the functional map (Equation 2.16; see details in Section 2.5). Ovsjanikov et al. [2012] also imposed some additional constraints on $\mathbf{C}_k$, *e.g.*, commutativity with the Laplacian in case of nearly-isometric shapes.

The main tradeoff in problem (7.4) is between k and q. The number of eigenvectors k used to represent the functional map has a direct impact on its quality: ideally, one wants to have $k = n$ or at least as large as possible. On the other hand q is the number of corresponding vectors, which in practical applications have to be found automatically (the could be, for example, dimensions of different descriptors or some sparse point-wise correspondences). Only small q correspondences can usually be reliably found. Finally, in order for the system (7.4) to be determined, we must have $k \geqslant q$, which in practice imposes limitations on how large k can be. Due to noise, one typically tries to make the system *overdetermined* ($k < q$) and solve it in the least-squares sense.

In [Kovnatsky et al., 2013], we proposed expressing $\mathbf{C}_k$ using coupled bases $\hat{\mathbf{\Phi}}, \hat{\mathbf{\Psi}}$ instead of the standard Laplacian eigenbases $\bar{\mathbf{\Phi}}, \bar{\mathbf{\Psi}}$. In the coupled bases, $\mathbf{C}_k$ is approximately diagonal, which can be used as regularization. The simplest way of imposing this structure is by ignoring the off-diagonal elements of $\mathbf{C}_k$ altogether and rewriting problem (7.4) for diagonal elements only, yielding a

linear system with qk equations with k variables (instead of k^2 as in the original formulation),

$$\begin{bmatrix} \operatorname{diag}(\mathbf{f}_1^\top \mathbf{A}_X \hat{\mathbf{\Phi}}) \\ \vdots \\ \operatorname{diag}(\mathbf{f}_q^\top \mathbf{A}_X \hat{\mathbf{\Phi}}) \end{bmatrix} \begin{bmatrix} c_{11} \\ \vdots \\ c_{kk} \end{bmatrix} = \begin{bmatrix} \operatorname{diag}(\hat{\mathbf{\Psi}} \mathbf{A}_Y \mathbf{g}_1) \\ \vdots \\ \operatorname{diag}(\hat{\mathbf{\Psi}} \mathbf{A}_Y \mathbf{g}_q) \end{bmatrix}. \tag{7.5}$$

Note that system (7.5) is always determined: theoretically, we need only $q = 1$ corresponding vector to find $c_{11}, \dots, c_{kk}$. In practice, since this is an approximation (*e.g.*, in Figure 7.3 one may observe non-zero off-diagonal elements) and due to noise, one would still need to have multiple corresponding vectors. Yet, compared to the original formulation (7.4), problem (7.5) allows to use significantly less data and is also more computationally efficient.

For obtaining a point-wise correspondence from a functional map, we used the ICP-like refinement approach [Ovsjanikov et al., 2012] discussed in Section 2.5.1. Our results show that with the same corresponding data, we obtain better correspondence with our approach compared to the original formulation of Ovsjanikov et al. [2012]. The difference in performance is especially pronounced in case of finding correspondence between non-isometric shapes.

Figure 7.9 shows an example of finding functional correspondence between non-isometric shapes of human and gorilla. As corresponding vectors, we used binary indicator functions of $q = 7$ regions automatically detected using the method of [Rodolà et al., 2014]. The ordering of the regions was assumed to be given (corresponding regions are denoted by similar color in Figure 7.7). We compared the method described in [Ovsjanikov et al., 2012] for computing $\mathbf{C}_k$ by solving the system (7.4) in the standard Laplacian eigenbases (Figure 7.9, left) and the diagonal-only approximation (2.18) in the coupled bases (Figure 7.9, center). The coupled bases were computed using the same data for the coupling term (see Figure 7.8). In all the cases, we used $k = 25$ first basis vectors.

Figure 7.10 measures the correspondence quality using the Princeton protocol [Kim et al., 2011b]: y-axis is the percentage of correspondences falling within a geodesic ball of increasing radius (x-axis) w.r.t. the groundtruth correspondence (higher curves represent better correspondence). Our method shows

significantly better performance compared to [Ovsjanikov et al., 2012].

We also performed an experiment with generating correspondence data by random union and intersection of the subsets of the detected seven regions (thus increasing q). The performance of classical functional maps based on Laplacian eigenbases increases in this case (Figure 7.10, right), but the proposed approach is still better. We can conclude that our method is especially advantageous when the correspondence data are scarce.

Robust correspondence between multiple shapes. In this experiment, we test the robust functional correspondence between multiple manifolds (problem 5.5) introduced in Section 4.4, using the MADMM solver discussed in Section 5.3.1. The M-step of MADMM in our experiments was performed using 4 iterations of the manifold conjugate gradients solver. We used $\rho = 1$ and initialized all $\mathbf{R}_i = \mathbf{I}$. We computed functional correspondences between $p = 6$ human 3D shapes from the TOSCA dataset [Bronstein et al., 2008] using $k = 25$ basis functions and $q = 25$ seeds as correspondence data, contaminated for some pairs of shapes by 16% of outliers. Figure 7.12 (left) analyzes the resulting correspondence quality using the Princeton protocol [Kim et al., 2011b]. For comparison, we show the results of the least-squares solution [Ovsjanikov et al., 2012] (see Figure 7.11).

Figure 7.12 (right) shows the convergence of MADMM in a correspondence problem with $p = 2$ shapes. For comparison, in Figure 7.12 (left) we show the convergence of a smoothed version of the $\|\cdot\|_{2,1}$ norm, $\|\mathbf{M}\|_{2,1} \approx \sum_j \left(\sum_i m_{ij}^2 + \epsilon\right)^{1/2}$, for various values of the smoothing parameter ϵ.

7.2.3 Pose transfer

Rong et al. [2008] proposed an approach for mesh editing based on elastic energy minimization. Given a shape with embedding coordinates $\mathbf{X}$, the method attempts to find a deformation field $\mathbf{d}$ producing a new shape $\mathbf{X}' = \mathbf{X} + \mathbf{d}$, providing a set of user-defined n' anchor points for which the displacement is known (w.l.o.g. assuming to be the first n' points, $\mathbf{d}_i = \mathbf{d}'_i$ for $i = 1, \ldots, n'$), as a solution

of the system of equations

$$\begin{bmatrix} \kappa_b \mathbf{L}_X^2 - \mu \mathbf{L}_X & \mathbf{Q} \\ \mathbf{Q}^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{d} \\ \zeta \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{d}' \end{bmatrix}, \tag{7.6}$$

where $\mathbf{Q} = (\mathbf{I},\ \mathbf{0})^\top$ is an $n \times n'$ identity matrix, ζ are unknown Lagrange multipliers corresponding to the constraints on anchor points, and κ_b, μ are parameters trading off between resistance to bending and stretching, respectively [Rong et al., 2008] . The system of equations can be expressed in the frequency domain using $k \ll n$ first harmonic basis functions,

$$\begin{bmatrix} \bar{\mathbf{\Phi}}^\top (\kappa_b \mathbf{L}_X^2 - \mu \mathbf{L}_X) \bar{\mathbf{\Phi}} & \bar{\mathbf{\Phi}}^\top \mathbf{Q} \\ \mathbf{Q}^\top \bar{\mathbf{\Phi}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{b} \\ \zeta \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{d}' \end{bmatrix} \tag{7.7}$$

where $\mathbf{b} = \bar{\mathbf{\Phi}}^\top \mathbf{A}_X \mathbf{d}$ are the k Fourier coefficients. The desired deformation field is obtained by solving the system of equations for $\mathbf{b}$ and transforming it to the spatial domain $\mathbf{d} = \bar{\mathbf{\Phi}}\mathbf{b}$ (for details, the reader is referred to [Rong et al., 2008]).

Using coupled bases, it is possible to extend this approach to simultaneous editing of multiple shapes, solving the system (7.7) with the coupled basis $\hat{\mathbf{\Phi}}$ in place of $\bar{\mathbf{\Phi}}$, and applying the deformation to the second mesh using $\mathbf{d} = \bar{\mathbf{\Phi}}\mathbf{b}$. Figure 7.13 exemplifies this idea, showing how a deformation of the cat shape is automatically transferred to the lion shape, which accurately and naturally repeats the cat pose.

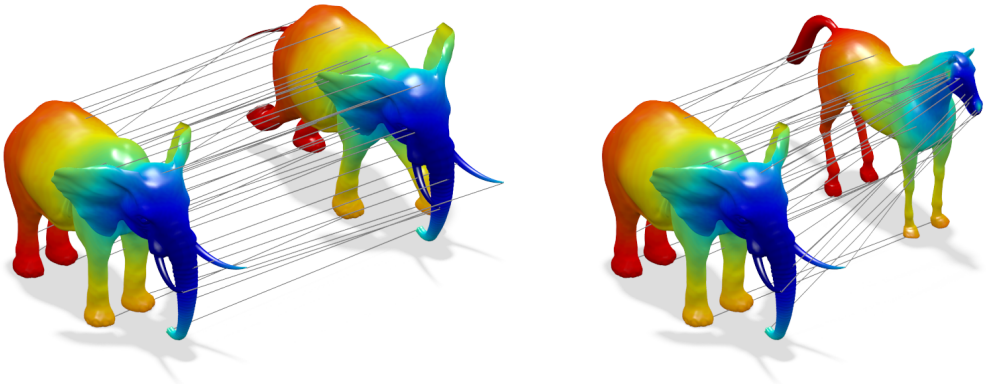
7.2.4 Shape similarity

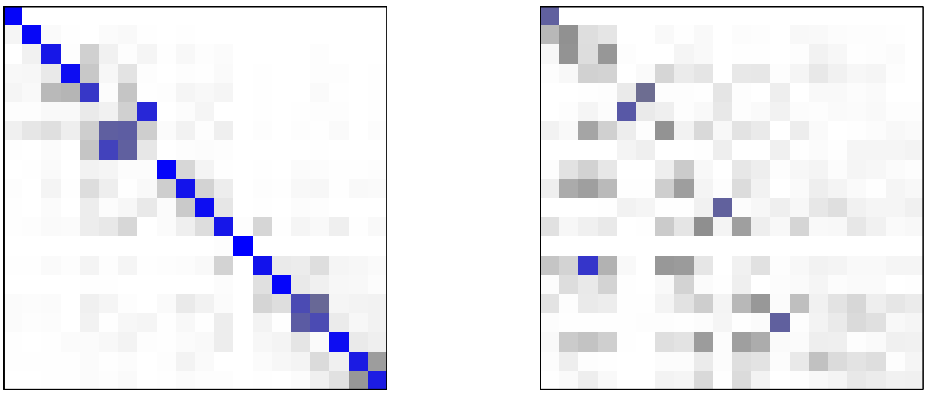
As the last application for coupled bases, we consider the problem of discovering similarities in shape collections. The joint diagonalizability of Laplacians can be used as a criterion for shape similarity, with isometric shapes having jointly diagonalizable Laplacians. With this approach, it is possible to compare two shapes from a small number of inaccurate correspondences, which are used in the coupling term of our simultaneous diagonalization problem.

Figure 7.14 shows the similarity matrix between 25 shapes belonging to 8 different classes. Each shape is present with 3-4 near-isometric deformations.

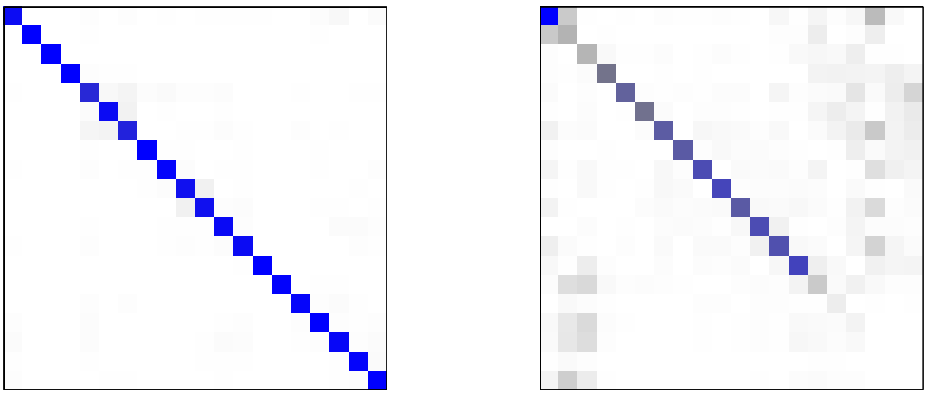
We used 25 point correspondences for point-wise coupling; dissimilarity of a pair of shapes was computed by jointly diagonalizing the respective Laplacians and then computing the average ratio of the norms of the off-diagonal and diagonal elements of both matrices. We use $k = 10$ first coupled eigenvectors.

We observe that besides the ability to discover correctly near-isometric shapes (dark blocks on the diagonal of the similarity matrix), we were able to discover two higher classes of two-legged (green rectangle) and four-legged shapes (blue rectangle). Also note that the size of the coupled basis k used to measure the joint diagonalizability determines the smallest "feature size" of our similarity criterion (using more vectors allows to capture finer "high frequency" behaviors). In our example, this is noticeable in the inability to distinguish between male and females shapes; the feature size allowing to make this finer distinction requires more than $k = 10$ eigenvectors.





Laplacian eigenbases



Coupled bases

Figure 7.2. Matrix **C** of coefficients expressing a given correspondence between two poses of an elephant (left) and elephant and horse (right) in the Laplacian eigenbases (second row) and coupled bases (third row). First row: correspondence between shapes shown with similar colors.

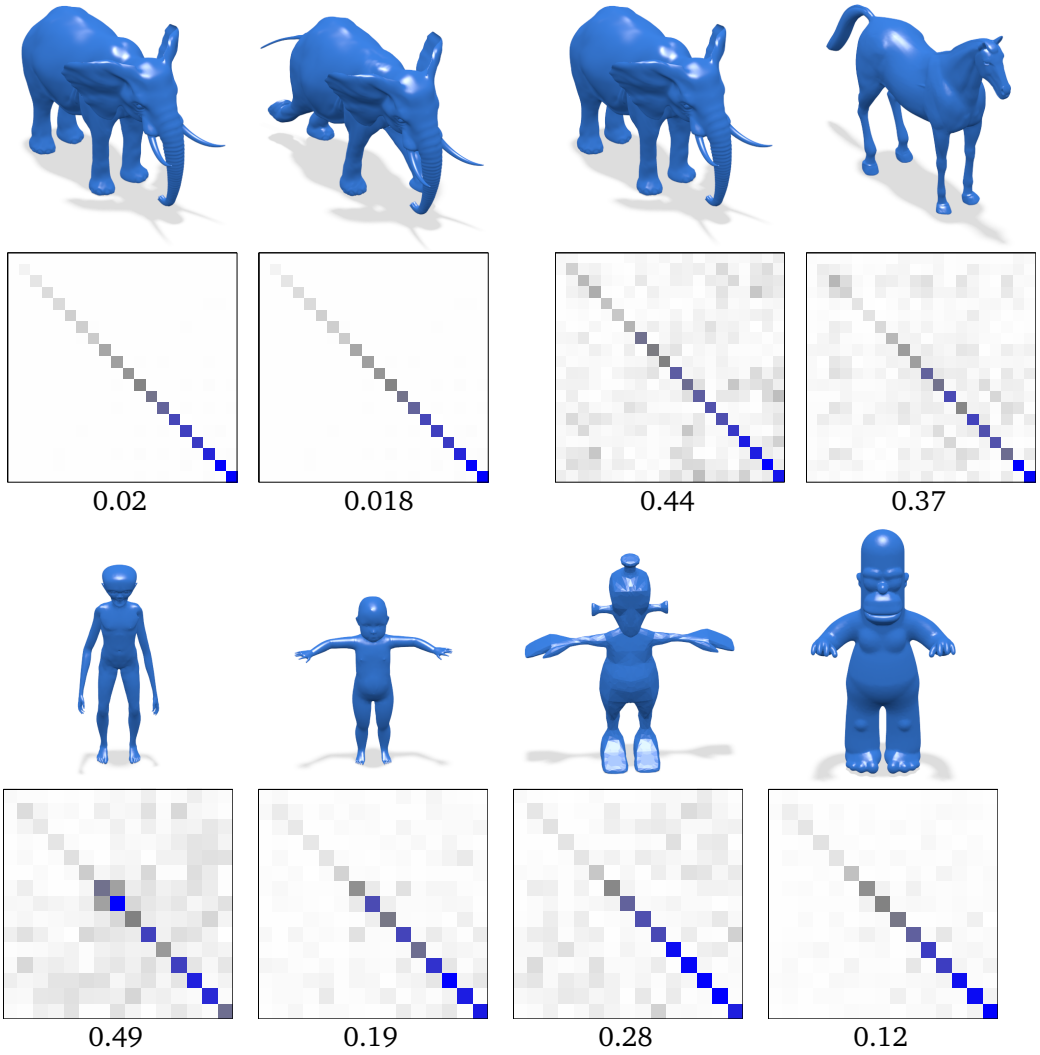


Figure 7.3. Examples of joint diagonalization of Laplacians of near-isometric shapes (two poses of an elephant, top right) and non-isometric shapes (elephant and horse, top right; four humanoids, bottom). Second and fourth rows show the approximate diagonalization of the respective Laplacians. Coupling was done using 40 points for the elephant and horse shapes and 25 for the humanoid shapes. Numbers show the ratio of the norm of the diagonal and off-diagonal values, $\|\text{off}\|/\|\text{diag}\|$.

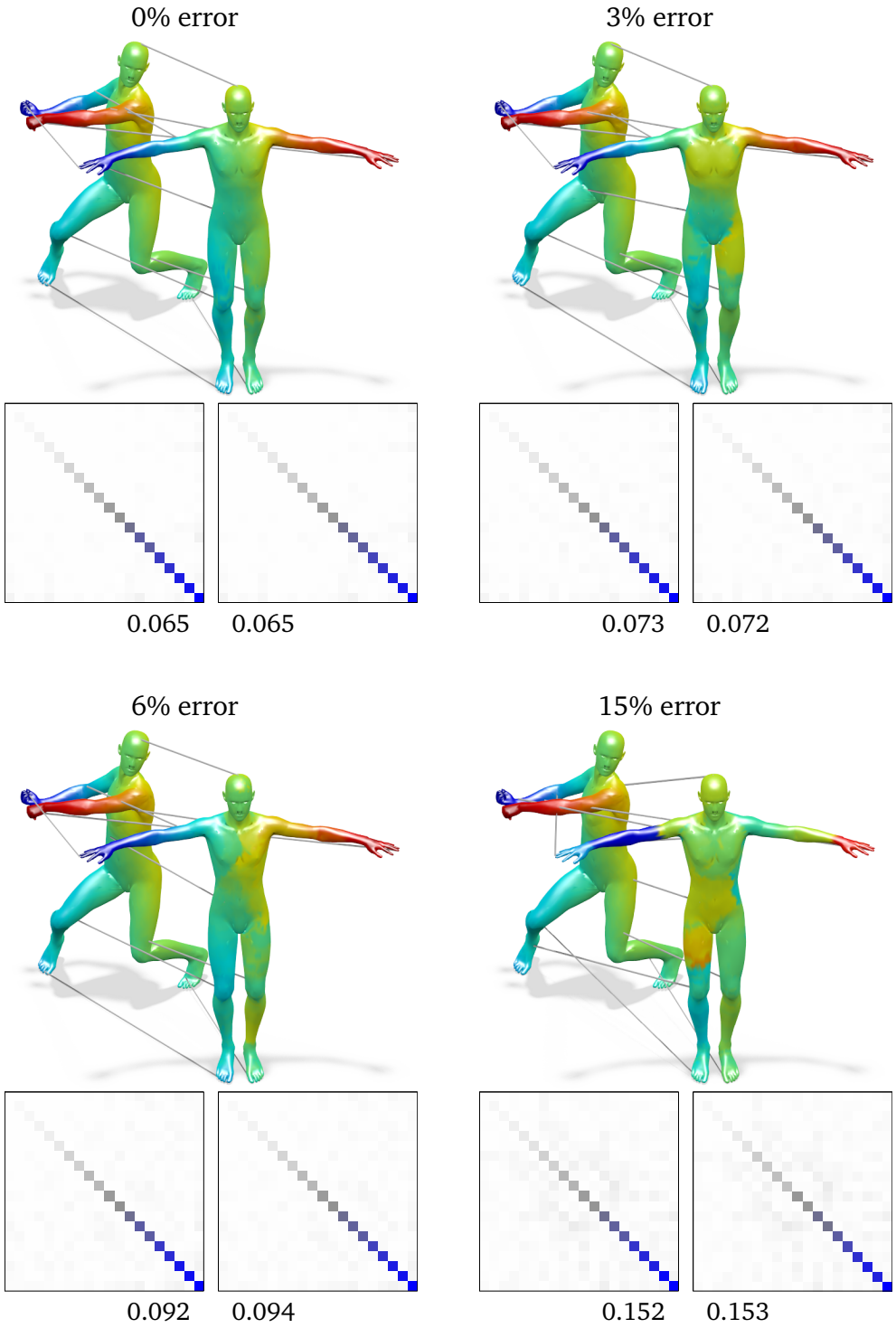


Figure 7.4. Sensitivity of joint diagonalization to errors in correspondence (in % of geodesic diameter of the shape) used in the coupling term. Shapes are shown with similar colors representing corresponding points. Correspondences between 10 points used for coupling are shown with lines. Numbers show the ratio of the norm of the diagonal and off-diagonal values, $\|\text{off}\|/\|\text{diag}\|$.

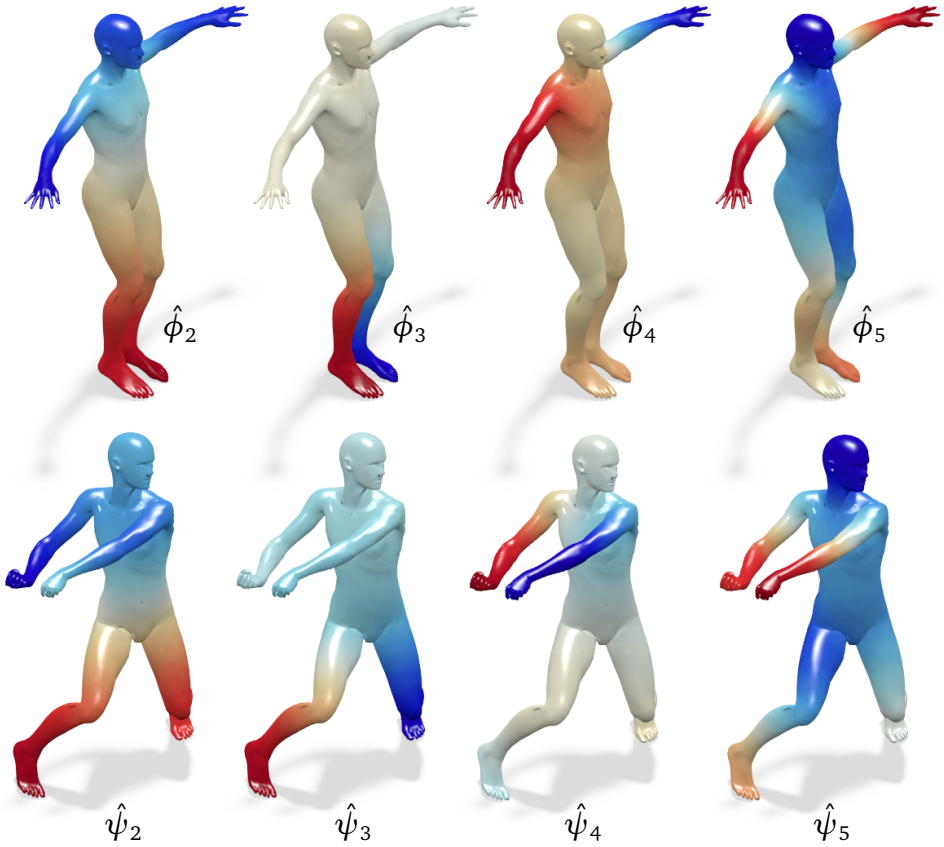


Figure 7.5. Coupled bases elements of the human shapes from Figure 7.4 obtained using 10 points with inaccurate correspondence (error of 15% geodesic diameter) for coupling.

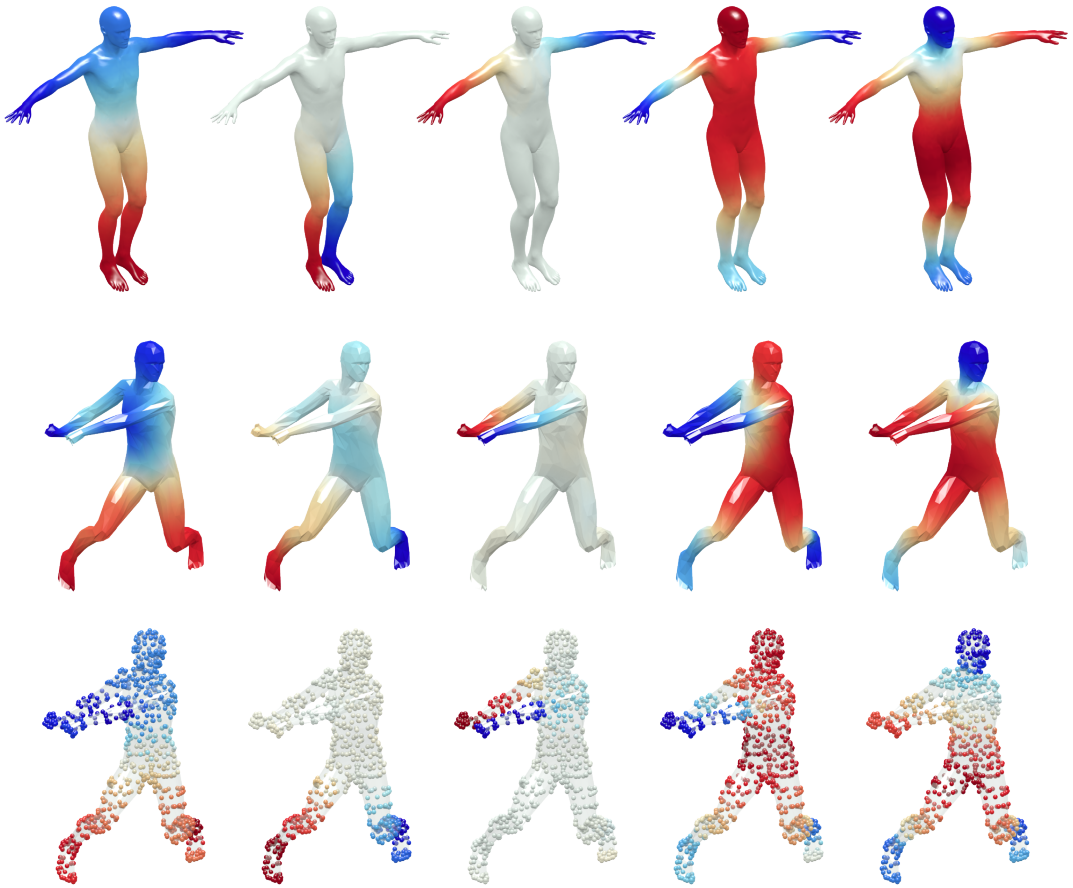


Figure 7.6. Coupled bases computed with 15 corresponding points on 3 meshes: the full mesh (8K vertices; the first row), a 10-times subsampled mesh (the second row), and a 10-times subsampled mesh represented as a point cloud (the last row).

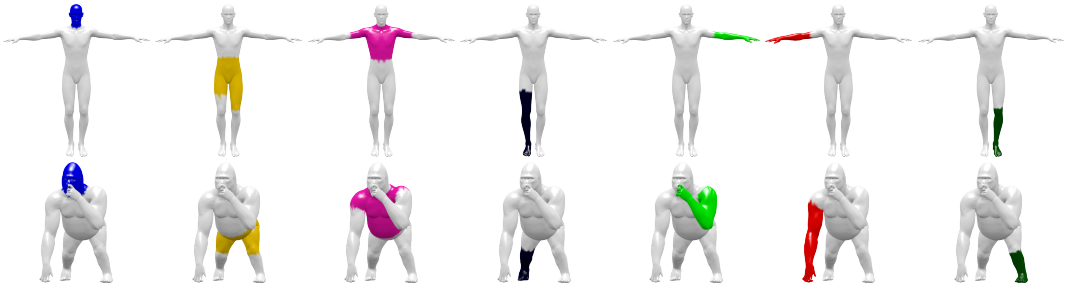


Figure 7.7. Regions in the human and gorilla shapes using the method of [Rodolà et al., 2014].

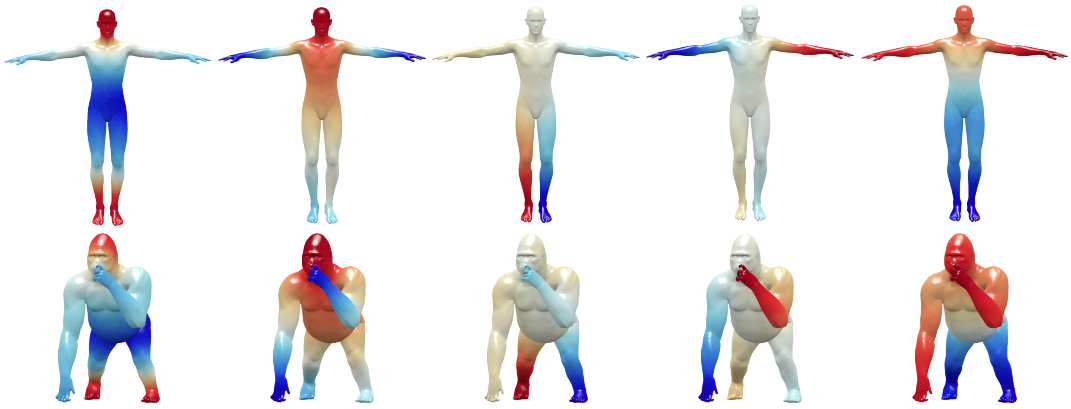


Figure 7.8. Coupled bases elements of the human and gorilla shapes obtained with Fourier coupling using the regions from Figure 7.7.

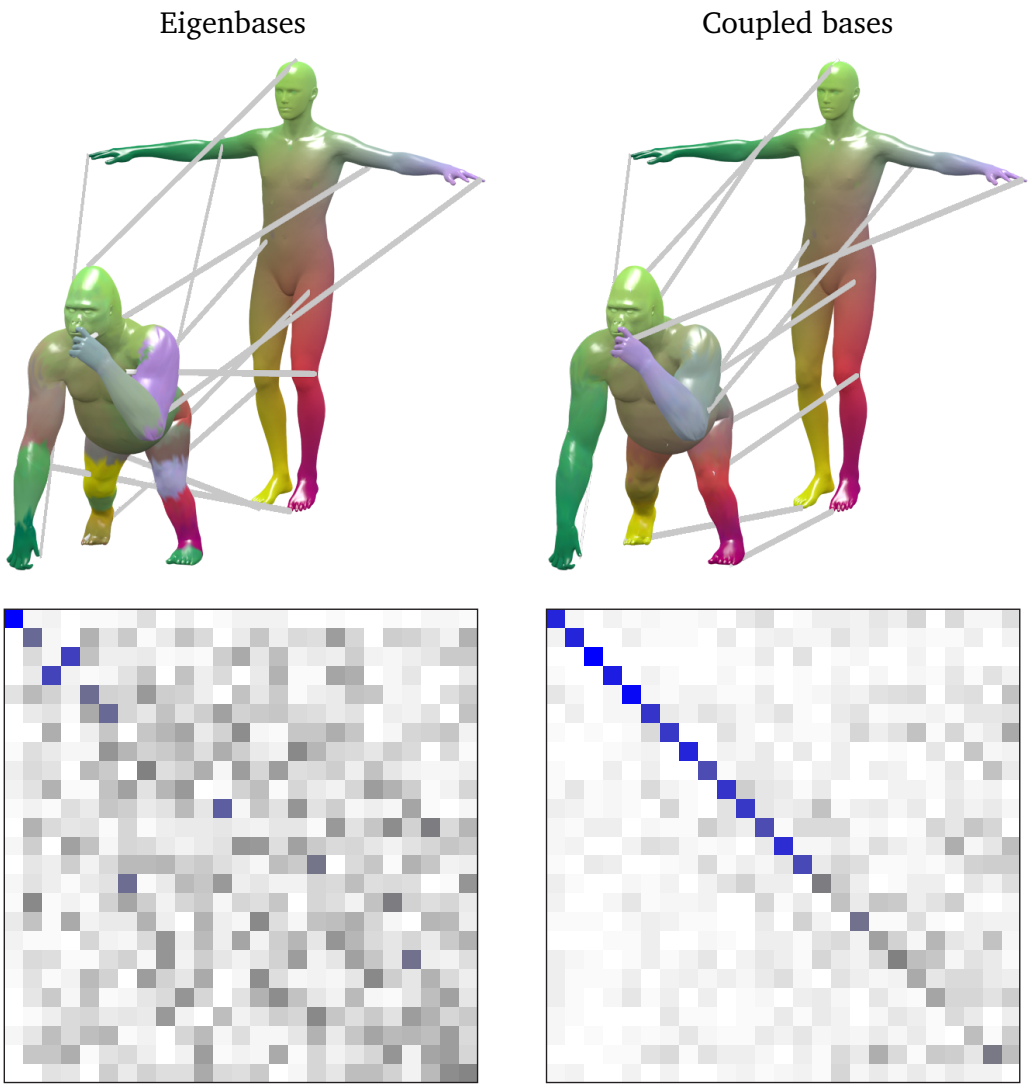


Figure 7.9. Functional correspondence computed by solving system (7.4) with the Laplacian commutativity constraint (left) [Ovsjanikov et al., 2012] and system (7.5) with coupled bases (right); the results are shown after the ICP-like refinement method. Corresponding points are shown in similar color. The resulting matrix $\mathbf{C}$ is shown in the second row (note that the ICP refinement alters the diagonal-only structure).

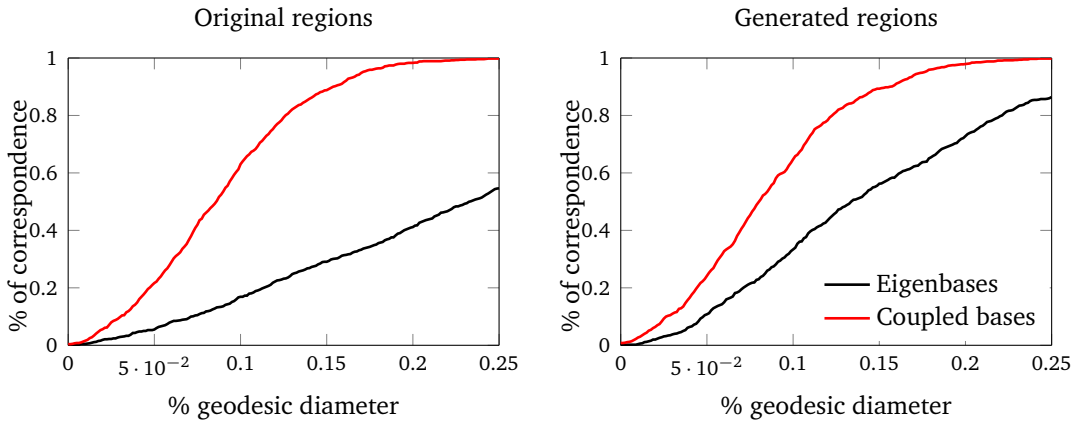


Figure 7.10. Evaluation of the functional correspondence obtained with the approaches illustrated in Figure 7.9. Shown is the percentage of correspondences falling within a geodesic ball of increasing radius w.r.t. the groundtruth correspondence. Left: correspondence obtained from 7 regions; right: correspondence obtained from 251 generated regions (random intersection and union of the subsets of 7 regions). We observe the dramatic improvement of the Laplacian eigenbases approach [Ovsjanikov et al., 2012] and similar performance of diagonal-only approach with coupled bases (Equation 7.5). This suggests that the off-diagonality terms act as the regularization, which is crucial in the case of the scarce data given.

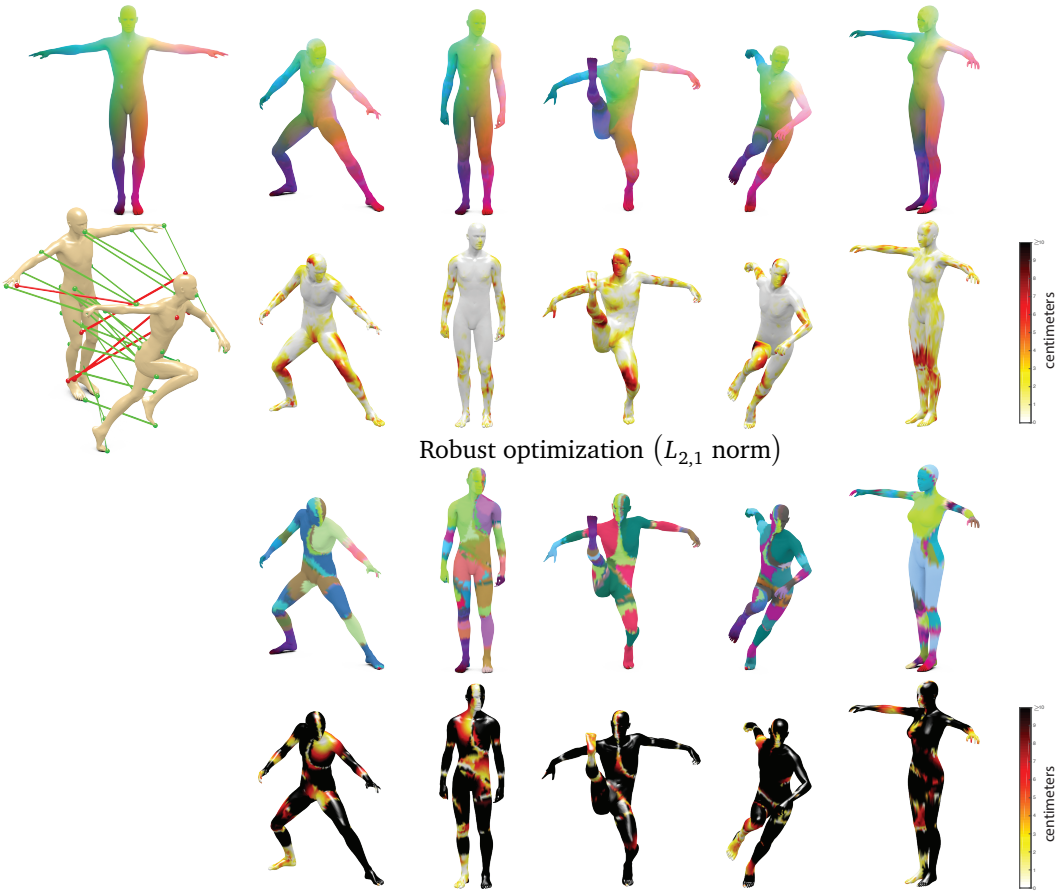


Figure 7.11. Examples of correspondences obtained with MADMM (top two rows) and least-squares solution (bottom two rows). Rows 1 and 3: similar colors encode corresponding points; rows 2 and 4: color encodes the correspondence error (distance in centimeters to the ground-truth). Leftmost column, 1st row: the reference shape; 2nd row: examples of correspondence between a pair of shapes (outliers are shown in red).

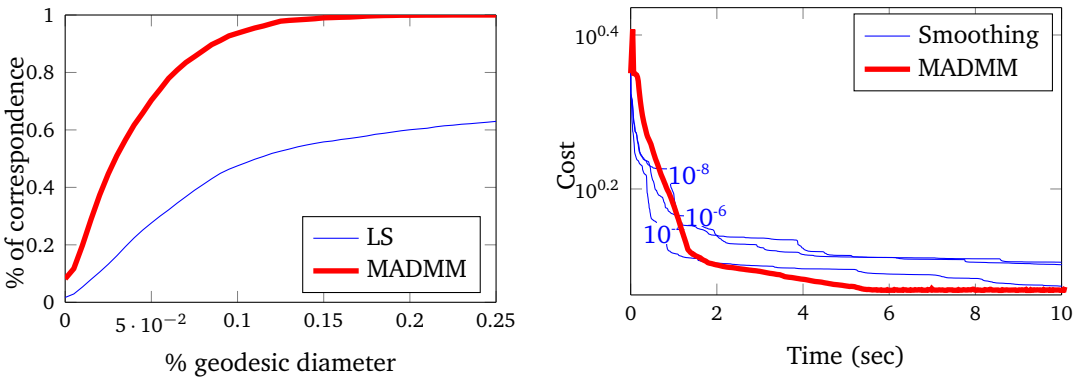


Figure 7.12. Functional correspondence problem. Left: evaluation of the functional correspondence obtained using MADMM and least squares. Right: convergence of MADMM and smoothing method for various values of the smoothing parameter. LS stands for the Least-Squares optimization [Ovsjanikov et al., 2012]..

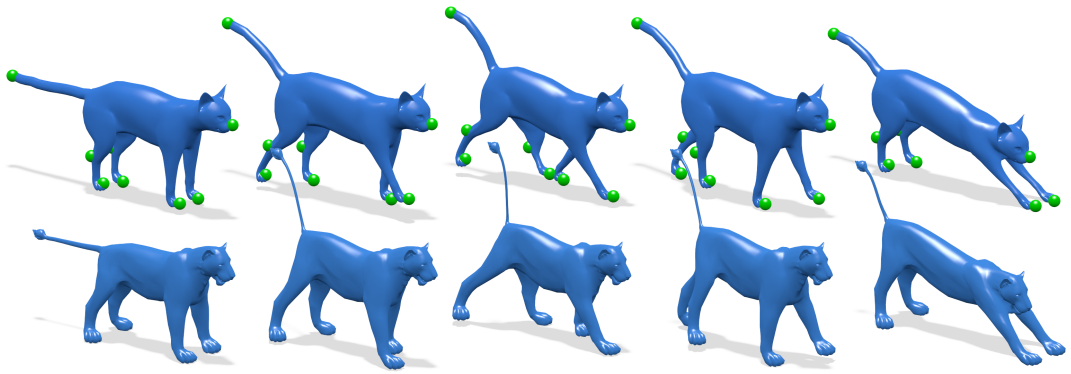


Figure 7.13. Simultaneous shape editing in the frequency domain using the approach of [Rong et al., 2008]. Top: editing the cat shape (anchor vertices in problem (7.6) shown in green). Bottom: the same pose is transferred to the lion shape using coupled basis.

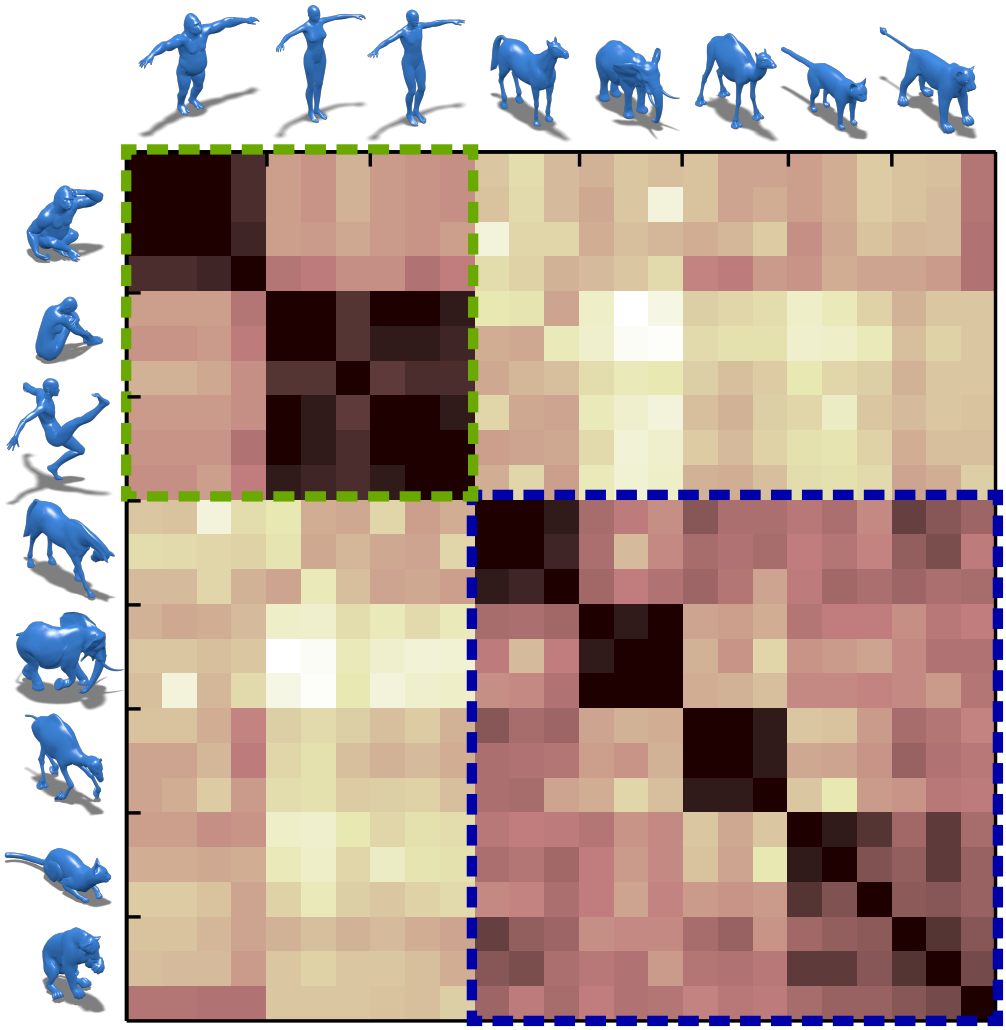


Figure 7.14. Shape similarity using joint diagonalization. Darker colors represent more similar shapes. One can clearly distinguish blocks of isometric shapes. Also, two classes of two- and four-legged shapes (marked with green and blue) are visible. Small figures show representative shapes from each class.

Chapter 8

Applications in Image Processing

In this chapter, we present *Laplacian colormaps*, a generic framework for computing structure-preserving color transformations using joint diagonalizability of image Laplacians as a criterion for structural similarity of the underlying images. Our framework is based on the relation between joint approximate diagonalization and approximate commutativity described in Section 3.3). This chapter is mainly based on our EUROGRAPHICS paper [Eynard et al., 2014]; gamut mapping results were further extended in [Kovnatsky et al., 2014]. The MATLAB code used to produce the presented experiments is available at CommuteJAD-Code.

8.1 Color manipulations in image processing

A wide class of image processing problems relies on transformations between color spaces. Some notable examples include gamut mapping, image optimization for color-deficient viewers, and multispectral image fusion. Often these transformations imply a reduction in the dimensionality of the original color space, resulting in information loss and ambiguities.

Decolorization or *color-to-gray conversion* is a classical example one frequently encounters when printing a color image on a black-and-white printer. The ambiguity of such a conversion (called *metamerism*, when many different RGB colors are mapped to the same gray level) may result in a loss of important structure in the image (see Figure 8.5). Preserving salient characteristics of the original

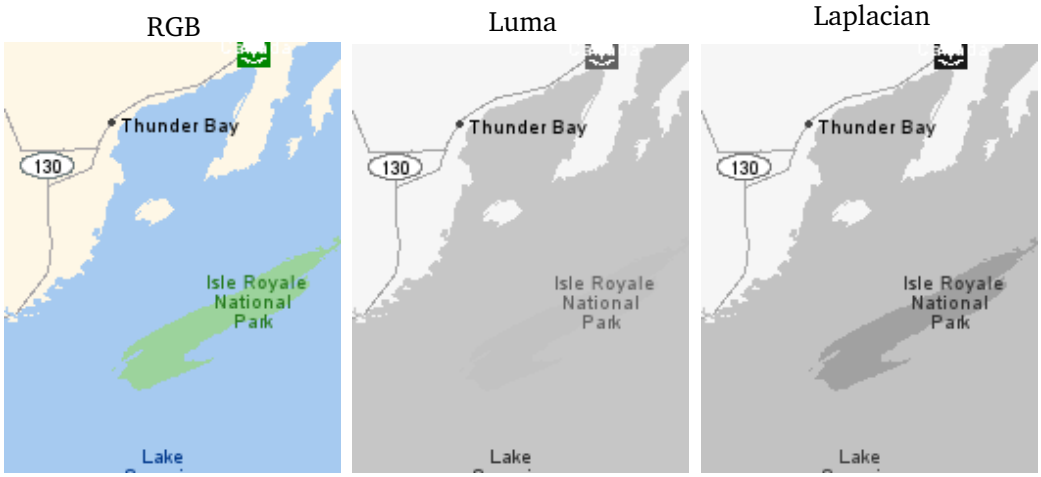


Figure 8.1. RGB to gray conversion performed by keeping the luma channel may result in loss of image structure due to metamerism (the green island disappears, middle). The proposed Laplacian colormap preserves all structures (right).

image is thus crucial for a quality color transformation process. These characteristics can be represented in different ways, *e.g.*, as contrasts between color pixels in terms of their luminance and chrominance [Gooch et al., 2005], color distances [Grundland and Dodgson, 2007], image gradients [Zhou and Feng, 2012] and Laplacians [Bansal and Daniilidis, 2013].

Color-to-gray maps can be classified into *global* (using the same map for each pixel) and *local* (or spatial, allowing different pixels with the same color to be mapped to different gray values, at the advantage of a better perception of color contrasts). Members of the first group include the pixel-based approaches by Gooch et al. [2005] and Grundland and Dodgson [2007], and the color-based ones by Rasche et al. [2005a], Kuhn et al. [2008a], Kim et al. [2009], Lu et al. [2012]. Among local methods [Neumann et al., 2007; Kuk et al., 2010; Zhou and Feng, 2012], several try to preserve information in the gradient domain. Smith et al. [2008] presented a hybrid (local+global) approach that relies on both an image-independent global mapping and a multiscale local contrast enhancement. Lau et al. [2011] proposed an approach defined as 'semi-local', as it clusters pixels based on both their spatial and chromatic similarities. The color mapping problem is solved with an optimization aimed at finding optimal cluster colors such that the contrast between clusters is preserved.

Gamut mapping is the process of adjusting the colors of an input image into

the set of colors a given device (monitor, printer, *etc*.) can reproduce. Gamut mapping algorithms can be mainly divided into *clipping* and *compression* approaches [Morovič, 2008]. The former ones change the source colors that fall outside of the destination gamut (*e.g*., HPMINDE [CIE, 2004; Bonnier et al., 2006]); the latter also modify the in-gamut colors [Kimmel et al., 2005; Balasubramanian et al., 2000; Nakauchi et al., 1999]. Similarly to color-to-gray conversion, gamut mapping methods can also be categorized as global and local. To address metamerism in gamut mapping, local approaches [Balasubramanian et al., 2000; Nakauchi et al., 1999; Kimmel et al., 2005] allow two spatially-distant pixels of equal color to be mapped to different in-gamut colors. Global approaches, conversely, will always apply the same map to two pixels of the same color, regardless of their location. Many gamut mapping algorithms optimize some image difference criterion [Nakauchi et al., 1999; Kimmel et al., 2005; Alsam and Farup, 2009; Lau et al., 2011].

Color-blind viewers cannot perceive differences between some given colors, due to the lack of one or more types of cone cells in their eyes [dal; Meyer and Greenberg, 1988]. Image perception by a color-deficient observer is typically simulated by first applying a linear transformation from a standard color space such as RGB [Kim et al., 2012; Brettel et al., 1997; Viénot et al., 1999], XYZ [Meyer and Greenberg, 1988; Rasche et al., 2005b], or CIE Lab* [Kuhn et al., 2008b; Huang et al., 2007] to a special LMS space, which specifies colors in terms of the relative excitations of the cones. Then, the color domain is reduced in accordance with the color deficiency (typically, by means of a linear transformation in the LMS space [Viénot et al., 1999; Kim et al., 2012; Huang et al., 2007]). Finally, the reduced LMS space is mapped back to RGB.

When trying to adapt an image for a color-blind viewer, one has to ensure that the structure of the original image is not lost due to color ambiguities. Kuhn et al. [2008b] focus on obtaining natural images by preserving, as much as possible, the original image colors.

Rasche et al. [2005b], instead, try to maintain distance ratios during the reduction process. Lau et al. [2011] is aimed at preserving both the contrast between color clusters and the reduced image colors.

Multispectral image fusion aims to combine a collection of images captured

at different wavelengths into a single one, containing details from several spectra. Zhang et al. [2008] and Lau et al. [2011] present a method that adaptively adjusts the contrast of photographs by using the contrast and texture information from near-infrared (NIR) image. Kim et al. [2011a] show how to use different bands of the invisible spectrum to improve the visual quality of old documents. Süsstrunk and Fredembach [2010] provide a good introduction to the topic and present, as examples of image enhancements, haze removal and realistic skin smoothing.

General approaches. We should stress that despite a significant corpus of research on color transformations, most of the methods are targeted to specific applications and lack the generality of a framework that could be applied to different classes of problems. At the same time, there is an obvious common denominator between the aforementioned problems: for example, both color-blind transformations [Rasche et al., 2005a] and color-to-gray conversions [Cui et al., 2010; Zhao and Tamimi, 2010; Zhou and Feng, 2012] can be regarded as mappings to a gamuts of lower dimension [Gooch et al., 2005]. To the best of our knowledge, only the recent work of Lau et al. [2011] introduces a comprehensive approach that works with generic color transformations and easily adapts to different applications.

8.2 Laplacian colormaps

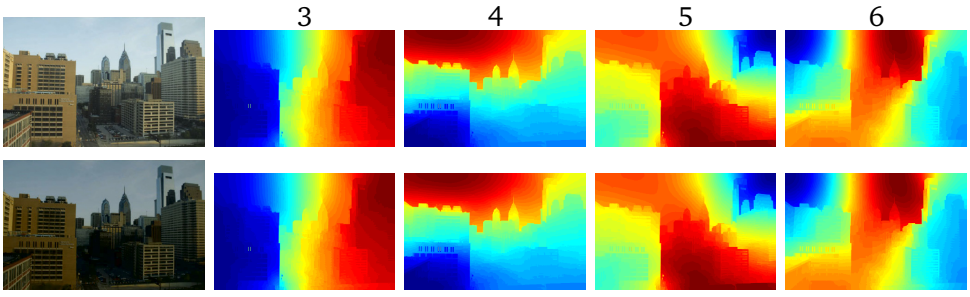


Figure 8.2. Image structure similarity is conveyed by the eigenstructure of their Laplacians. Top: the original image; bottom: its darken version. Left-to-right: original image, first four eigenvectors with non-zero eigenvalues of the corresponding Laplacian.

Our main motivation comes from recent work of Bansal and Daniilidis [2013], who used the eigenvectors of image Laplacians to perform matching of images taken in different illumination conditions, arguing that the Laplacian acts as a self-similarity descriptor [Shechtman and Irani, 2007] of the image. Using Laplacians as image structure descriptors, we observe that an ideal color transformation should preserve the Laplacian eigenstructure, implying that the Laplacians of the original and color-converted image should be jointly diagonalizable (Figures 8.2, 8.3). Employing the relation between joint diagonalizability and commutativity of matrices [Glashoff and Bronstein, 2013; Bronstein et al., 2013], discussed in Section 3.3, we use Laplacians commutativity as a criterion of image structure preservation. We try to find such a colormap η that would produce a converted image whose Laplacian commutes as much as possible with the Laplacian of the original image. Since Laplacians can be defined in any colorspace, our approach is generic and applicable to any kind of color conversions (in particular, color-to-gray, gamut mapping, color-blind optimization, etc.). Furthermore, we can work with both global and local colormaps.

As we will show in our experiments, even approximately jointly diagonalizable Laplacians serve the purpose of structure preserving color transformation, giving us the freedom to apply different transformations with different constraints on their parameters. Let us summarize the main assumptions so far, which will motivate our approach described in the following. First, Laplacians can be used as structural descriptors of images. Second, two images having similar structures translates into having the corresponding Laplacians jointly diagonalizable. Third, joint diagonalizability is equivalent to commutativity, as discussed in Chapter 3.

Let us be given an $N \times M$ image with d color channels, column-stacked into an $NM \times d$ matrix $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_{NM}]^\top$. The problem of color conversion is creating a new image $\mathbf{Y} = \eta(\mathbf{X})$ with d' color channels, by means of a *colormap* $\eta : \mathbb{R}^{NM \times d} \to \mathbb{R}^{NM \times d'}$. In particular, we are interested in parametric colormaps $\eta_{\boldsymbol{\theta}}$. In the simplest case, $\eta_{\boldsymbol{\theta}}$ is a global color transformation applied pixel-wise, *i.e.*, each pixel $\mathbf{x}_i \in \mathbb{R}^d$ of the original image is mapped by means of the same $\eta_{\boldsymbol{\theta}} : \mathbb{R}^d \to \mathbb{R}^{d'}$ such that $\eta_{\boldsymbol{\theta}}(\mathbf{X}) \triangleq [\eta_{\boldsymbol{\theta}}(\mathbf{x}_1), \dots, \eta_{\boldsymbol{\theta}}(\mathbf{x}_{NM})]^\top$ (a simple example is linear RGB to gray mapping, where $d = 3$, $d' = 1$, $n = 3$ and $\eta_{\boldsymbol{\theta}}(\mathbf{x}_i) = \sum_{j=1}^{d} \boldsymbol{\theta}_i \mathbf{x}_{ij}$, where in

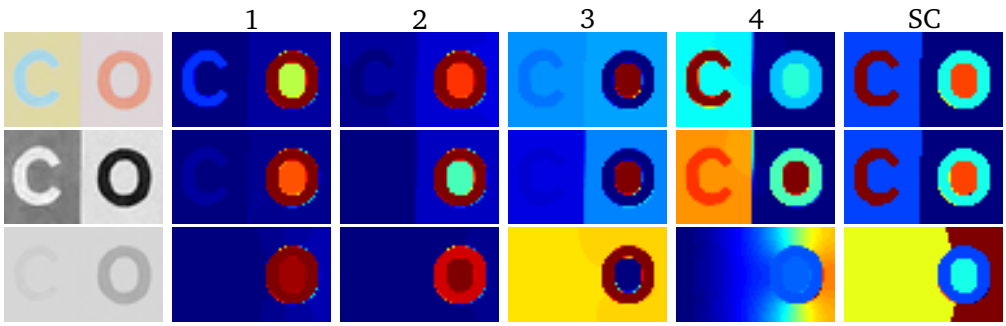


Figure 8.3. Image structure similarity is conveyed by the eigenstructure of their Laplacians. Top: original RGB image; middle: grayscale conversion by our method; bottom: luma only conversion. Left-to-right: original image, first four eigenvectors of the corresponding Laplacian, result of spectral clustering.

addition we require $\eta_{\boldsymbol{\theta}} \geqslant 0$ and $\sum_{i=1}^{d} \boldsymbol{\theta}_i = 1$).

Let $\{k_1, \ldots, k_L\} \subseteq \{1, \ldots, NM\}$ denote a subset of the image pixel indices (this subset can be the whole set of NM pixels, a regularly subsampled $M/s \times N/s$ image, 'representative' pixels obtained by clustering the image, etc.). Considering these pixels as vertices of a graph, we define edge weights (*adjacencies*) in the most general case as a combination of spatial and 'radiometric' distances (as in Figure 2.1, left, where the embedding consists of both spatial and 'radiometric' parts),

$$w_{ij} = e^{-\frac{\delta_{ij}^2}{2\sigma_s^2}} e^{-\frac{\|\mathbf{x}_{k_i} - \mathbf{x}_{k_j}\|_2^2}{2\sigma_r^2}}, \tag{8.1}$$

where δ_{ij} is the spatial distance between pixels k_i and k_j, and $\sigma_s, \sigma_r \geqslant 0$ are parameters (more generally, the 'radiometric' part of the adjacency w_{ij} does not have to work on pixel-wise colors, and one can consider some local features, the simplest of which are patches [Wetzler and Kimmel, 2012]). For practical computations, it is usually assumed that $w_{ij} \approx 0$ between spatially-distant pixels, so they are disconnected.

Given the adjacency matrix $\mathbf{W}$, we define the (unnormalized) *Laplacian* $\mathbf{L}$ as in Section 2.6. One of the strengths of our method is that it can cope with any operator capturing the image structure and not only the (unnormalized) *Laplacian*. Here, we consider the unnormalized Laplacian merely for the sake of simplicity.

8.2.1 Colormap optimization problem

Our goal is to find a set of parameters $\boldsymbol{\theta}$ such that the structures of the input image $\mathbf{X}$ and the modified image $\mathbf{Y} = \eta_{\boldsymbol{\theta}}(\mathbf{X})$ are as similar as possible, where the similarity is judged by the commutativity of the corresponding Laplacians. This leads us to a class of optimization problems of the form

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^n} \quad & \mu_0 \|[\mathbf{L}_{\mathbf{X}}, \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}]\|_{\mathrm{F}}^2 + \mu_1 \|\mathbf{L}_{\mathbf{X}} - \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}\|_{\mathrm{F}}^2 \\ & + \mu_2 \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_2^2 + \mu_3 \|\eta_{\boldsymbol{\theta}}(\mathbf{X}_{\mathrm{c}}) - \mathbf{Y}_{\mathrm{c}}\|_{\mathrm{F}}^2 \\ & \text{s.t.} \quad \text{constraints on } \boldsymbol{\theta}. \end{aligned} \tag{8.2}$$

One can easily recognize in problem (8.2) a parametric version of the closest commuting operators (CCO) problem (6.1.1) with one of the Laplacians fixed. Note that the Laplacian $\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}$ is parametrized by a small number of degrees of freedom $l \ll MN$ (number of non-zero elements), and thus it would be usually impossible to make it exactly commute with the given $\mathbf{L}_{\mathbf{X}}$ - hence, unlike the CCO problem, the commutator norm appears as a penalty rather than a constraint.

Additional regularization (third and fourth terms in (8.2)) is used if we have some 'nominal' parameters $\boldsymbol{\theta}_0$ representing a standard color transformation, or if some colors $\mathbf{X}_{\mathrm{c}} = (\mathbf{x}_1, \dots, \mathbf{x}_p)^\top$ should be mapped into some $\mathbf{Y}_{\mathrm{c}} = (\mathbf{y}_1, \dots, \mathbf{y}_p)^\top$ known in advance (for example, in some cases it is important to preserve black and white colors). Finally, depending on the type of the colormap $\eta_{\boldsymbol{\theta}}$, one may impose some constraints on the parameters $\boldsymbol{\theta}$ (*e.g.*, in linear RGB-to-gray conversion, $\boldsymbol{\theta} \geqslant \mathbf{0}$ and $\boldsymbol{\theta}^\top \mathbf{1} = 1$).

Local maps. Our approach imposes no limitations on the complexity of the colormap $\eta_{\boldsymbol{\theta}}$; in particular, this map does not have to be global. Let us assume that the source image is partitioned into q (soft) regions, represented by weight vectors $\boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_q$ of size $NM \times d'$, such that $\sum_{i=1}^q \boldsymbol{\omega}_i = \mathbf{1}$ and $\boldsymbol{\omega}_i \geqslant \mathbf{0}$($\boldsymbol{\omega}_i$ can be interpreted as an indication function of $\eta_{\boldsymbol{\theta}_i}$ for the ith region). In each region i, we allow for a different colormap $\eta_{\boldsymbol{\theta}_i}$. Then, the overall colormap is given as $\eta_{\boldsymbol{\theta}}(\mathbf{X}) = \sum_{i=1}^q \boldsymbol{\omega}_i \eta_{\boldsymbol{\theta}_i}(\mathbf{X})$, parametrized by $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_q)$. Optimization w.r.t. to the parameters of the local colormap is performed in exactly the same manner as described above.

Multiple Laplacians. In some applications like multispectral image fusion, one may wish to impose structural similarity between the output image and *multiple* images, $\mathbf{X}_1, \ldots, \mathbf{X}_K$ with colorspaces of dimensionality $d_1, \ldots, d_K$. The input image $\mathbf{X}$ may be one of the K images or a merged image with $\sum_{k=1}^{K} d_k$-dimensional colorspace. In this case, our optimization problem (8.2) assumes the form

$$\begin{aligned} \min_{\boldsymbol{\theta} \in \mathbb{R}^n} \quad & \sum_{k=1}^{K} \mu_{0k} \|[\mathbf{L}_{\mathbf{X}_k}, \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}]\|_{\mathrm{F}}^2 + \mu_{1k} \|\mathbf{L}_{\mathbf{X}_k} - \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}\|_{\mathrm{F}}^2 \\ & + \mu_2 \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_2^2 + \mu_3 \|\eta_{\boldsymbol{\theta}}(\mathbf{X}_{\mathrm{c}}) - \mathbf{Y}_{\mathrm{c}}\|_{\mathrm{F}}^2 \\ \text{s.t.} \quad & \text{constraints on } \boldsymbol{\theta}, \end{aligned} \tag{8.3}$$

where $\mu_{01}, \ldots, \mu_{0K}, \mu_{11}, \ldots, \mu_{1K}, \mu_2, \mu_3 \geqslant 0$ are constants determining the trade-off between different penalties.

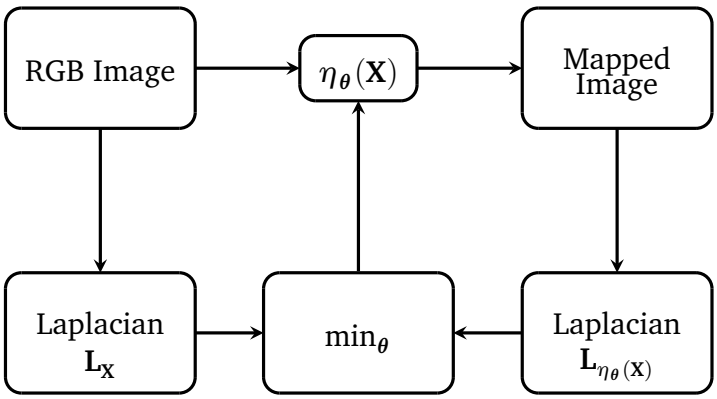


Figure 8.4. The pipeline of Laplacian colormap method.

8.3 Experiments and Results

In this section, we show several applications of *Laplacian colormaps* for decolorization, image optimization for color-blind people, gamut mapping, and multispectral image fusion, providing extensive comparison to previous works. As a quantitative criterion of the colormap quality, we use the *root weighted mean square* (RWMS) error proposed by Kuhn et al. [2008a], measuring the distortion

of relative color distances in two images,

$$\mathrm{err}_i = \left(\frac{1}{NM} \sum_{j=1}^{NM} \frac{(R_{\mathbf{Y}} \|\mathbf{x}_i - \mathbf{x}_j\| - R_{\mathbf{X}} \|\mathbf{y}_i - \mathbf{y}_j\|)^2}{R_{\mathbf{Y}}^2 \|\mathbf{x}_i - \mathbf{x}_j\|^2} \right)^{1/2}, \tag{8.4}$$

where $N \times M$ is the image size, $\mathbf{x}_i \in \mathbb{R}^d$ and $\mathbf{y}_i \in \mathbb{R}^{d'}$ denote the ith pixel of the input and output images, respectively, and $R_{\mathbf{X}} = \max_{ij} \|\mathbf{x}_i - \mathbf{x}_j\|$ is the color range of image $\mathbf{X}$.

Given the size of the problem, we calculate relative color distances on randomly chosen sets of points with size$<NM$, repeat the process 10 times and average the results. Plotting the pixel-wise RWMS error err_i as an image allows to see which pixels are most affected by the color transformation. The average $\mathrm{err}(\eta) = \frac{1}{NM} \sum_{i=1}^{NM} \mathrm{err}_i$ is used as a single number representing the quality of the colormap η.

All experiments share a common setup: first of all, RGB values are scaled by 255. Then we calculate a weighted adjacency matrix according to Section 2.3 using all pixels ($L = MN$) if the images are small enough, and resizing the image to have long side of 300 pixels otherwise. Adjacency matrices were calculated with fixed 4-neighbors connectivity and parameters $\sigma_r = 1$, $\sigma_s = 0$ (as all the neighbors have the same distance in this case). Unless differently specified in the following, default weights for the cost function are $\mu_0 = \mu_1 = \mu_2 = 1, \mu_3 = 0$, and regularization term $\boldsymbol{\theta}_0 = \mathbf{0}$. Parameters are initialized randomly and normalized to satisfy the condition $\boldsymbol{\theta}^\top \mathbf{1} = 1$. As a last step, since mapping might produce color values out of the $[0, 1]$ range, output channels are normalized.

8.3.1 Decolorization

For RGB-to-gray mapping, we used a global colormap, applying in each pixel $\mathbf{x}_i$ the following transformation: $y_i = \alpha + \sum_{j=1}^{3} \beta_i x_{ij}^{\gamma_i}$, where x_{ij} is the jth RGB channel of the ith pixel, y_i is the grayscale output, and $\boldsymbol{\theta} = (\alpha, \beta_1, \gamma_1, \ldots, \beta_3, \gamma_3)$ are the colormap parameters w.r.t. which the optimization is performed.

Images used for this experiment were taken from Čadík [2008]. Figure 8.6 shows the results of our transformations, compared to previous works evaluated

by Čadik Gooch et al. [2005]; Rasche et al. [2005a]; Grundland and Dodgson [2007]; Neumann et al. [2007]; Smith et al. [2008] and more recent ones [Lu et al., 2012]. Results were evaluated using two different metrics: quantitative (RWMS) and qualitative perceptual evaluation following Čadík [2008]. In the perceptual evaluation conducted through a Web survey, 124 volunteers were shown the original RGB image together with a pair of its gray conversions, and were asked which of the two results better preserved the original image. Then, we used Thurstone's law of comparative judgments to convert the 2884 pairwise evaluations into interval *z-score* scales [Thurstone, 1927; Tsukida and Gupta, 2011]. Table 8.1 provides average RWMS values and z-scores calculated on an 8-images subset of Čadik's. Our approach performs the best w.r.t. both criteria.

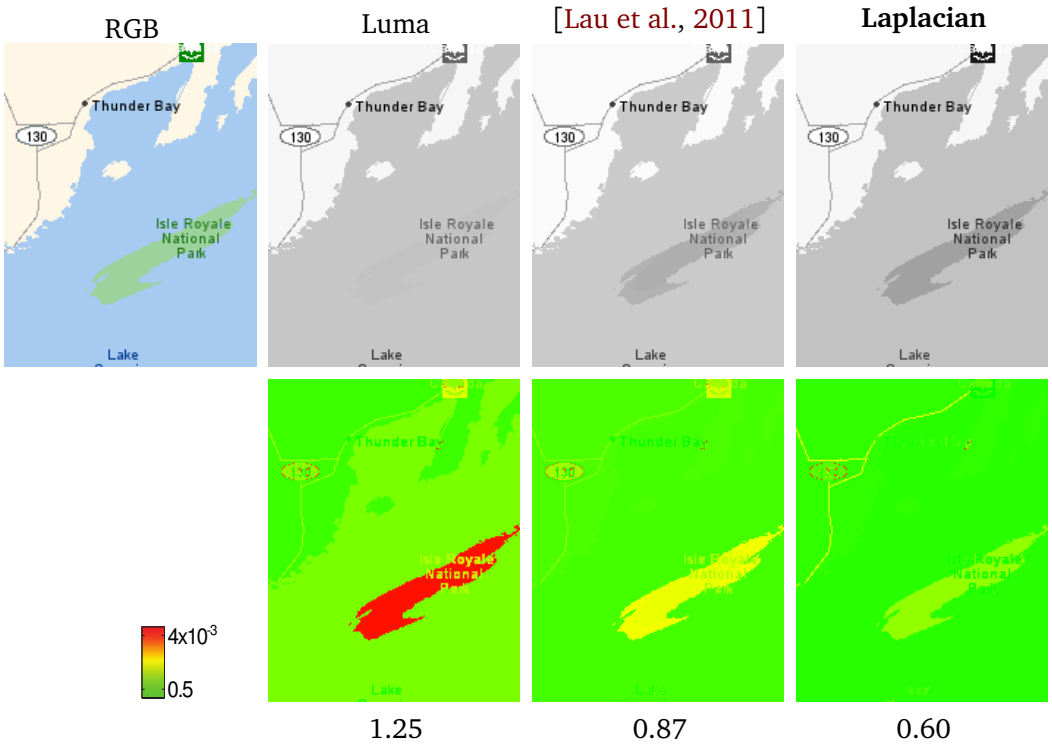


Figure 8.5. Decolorization experiment results. Left: original RGB image, right (first row): grayscale conversion results using the method of Lau et al. [2011] and our Laplacian colormap. The second row: RWMS error images and mean RWMS / perceptual score values.

	CIE Y	[1]	[2]	[3]	[4]	[5]	[6]	**Laplacian**
RWMS	2.84	2.31	2.46	2.20	4.85	2.94	1.90	**1.33**
z-score	-0.17	-0.31	-0.63	0.55	-0.53	-0.09	0.34	**0.84**

Table 8.1. Comparison of color-to-gray conversions in terms of mean RWMS value and z-score, averaged on all images. [1][Gooch et al., 2005], [2][Rasche et al., 2005a], [3][Grundland and Dodgson, 2007], [4][Neumann et al., 2007], [5][Smith et al., 2008], [6][Lu et al., 2012]

8.3.2 Computational complexity

Computationally, pixel-wise relationships appear to be the main bottleneck of our approach: a benchmark we ran on all the 25 pictures from [Čadík, 2008] using a MacBook Pro with 8GB RAM showed that the average computation time for color-to-grayscale conversion was 117 seconds. However, one of the strengths of our approach is that the construction of the Laplacian used to find the parameters of the colormap is completely flexible. The Laplacian does not have to be built on full-resolution image, and not even on uniform sampling (image superpixels [Ren and Malik, 2003] can be used as graph vertices), allowing for a trade-off between efficiency and accuracy. Then, once parameters are found, the colormap is applied to the original image in full resolution. Note this approximation is the reason why we have few pixels out of gamut in gamut mapping experiments (Figure 8.11).

Figure 8.8 shows the results of a 616x596 image decolorization, using a linear mapping function with different settings. In practice, we can achieve <1sec performance without sacrificing accuracy.

8.3.3 Image optimization for color-blind viewers

We model the color distortion of an RGB image $\mathbf{X}$ as perceived by a color-blind person by means of a map $\Pi : \mathbb{R}^{NM\times d} \rightarrow \mathbb{R}^{NM\times d}$. Since Π is given and beyond our control, we try to 'pre-transform' the original image by means of $\eta_{\boldsymbol{\theta}} : \mathbb{R}^{NM\times d} \rightarrow \mathbb{R}^{NM\times d}$ in such a way that the image $(\Pi \circ \eta_{\boldsymbol{\theta}})(\mathbf{X})$ that appears to the color-blind person has the structure of the original image $\mathbf{X}$. In our experiment $\eta_{\boldsymbol{\theta}}$ applies the same non-linear transformation used for grayscale to generate three output

channels ($\boldsymbol{\theta}$ thus holds 9 parameters). Π is a $3x3$ linear transformation akin to the ones in [Kim et al., 2012], used to replicate the results of [Lau et al., 2011]. We extend our problem formulation so that the transformed image maintains its structure both *when seen by a color-blind observer* and *when seen by a regular observer*. In our optimization problem, this translates into requiring the two pairs of Laplacians $\mathbf{L}_{\mathbf{X}}, \mathbf{L}_{(\Pi\circ\eta_{\boldsymbol{\theta}})(\mathbf{X})}$ and $\mathbf{L}_{\mathbf{X}}, \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}$ to commute. The cost function is similar to the multiple Laplacians setting (8.3):

$$\begin{aligned}\min_{\boldsymbol{\theta}\in\mathbb{R}^n} \quad & \mu_{01}\|[\mathbf{L}_{\mathbf{X}}, \mathbf{L}_{(\Pi\circ\eta_{\boldsymbol{\theta}})(\mathbf{X})}]\|_{\mathrm{F}}^2 + \mu_{02}\|[\mathbf{L}_{\mathbf{X}}, \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}]\|_{\mathrm{F}}^2 \\ & + \mu_{11}\|\mathbf{L}_{\mathbf{X}} - \mathbf{L}_{(\Pi\circ\eta_{\boldsymbol{\theta}})(\mathbf{X})}\|_{\mathrm{F}}^2 + \mu_{12}\|\mathbf{L}_{\mathbf{X}} - \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}\|_{\mathrm{F}}^2 \\ & + \mu_2\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_2^2 \end{aligned} \tag{8.5}$$

Figures 8.9, 8.10 show Laplacian colormaps results for two different types of color blindness (protanopia and tritanopia, respectively). Qualitatively, our result appears to be much closer to the original image compared to [Lau et al., 2011] (this is especially apparent in the tritanopia case) such that a 'normal' viewer sees less distorted colors, while a color-deficient viewer can clearly see the structure in the image (digit 6 and different candies) which otherwise would disappear. Quantitatively, we obtain smaller RWMS error, suggesting that our mapping better preserves the original structure of the image, even in those areas that are critical for other approaches.

8.3.4 Gamut mapping

Gamut mapping is a problem similar to the previous one. The function $\eta_{\boldsymbol{\theta}}$ is exactly the same, then a transformation Π which maps colors from RGB to the XY chromaticity space and a color gamut Γ (a convex polytope, and in this particular experiment a triangle) are given. Our goal is to find $\boldsymbol{\theta}$ minimizing the cost (8.2) subject to $(\Pi \circ \eta_{\boldsymbol{\theta}})(\mathbf{X}) \subseteq \Gamma$, which is imposed as a set of linear constraints. We used the parameters $\mu_{01} = 1, \mu_{11} = 0.25, \mu_2 = 0.1$, and $\mu_{02} = \mu_{12} = 0$. Figure 8.11 compares our results with the outputs of HPMINDE [CIE, 2004] and of the method of Lau et al. [2011]. Qualitatively, the output of Laplacian colormaps preserves more details of the original picture (see, *e.g.*, the plumage on the red

parrot's head). Quantitatively, our algorithm outperforms the other methods in terms of percentage of out-of-gamut pixels.

8.3.5 Multispectral image fusion

Multispectral image fusion can be seen as a color-to-color transformation with $d > 3$ input channels and $d' = 3$ output channels. The mapping function is thus just an extension of η_θ dealing with more channels. We use the cost function (8.3), with $\mu_{01} = \mu_{02} = \mu_{11} = \mu_{12} = \mu_2 = 1$.

Figure 8.12 shows multispectral to RGB transformations where the input space is the concatenation of RGB and NIR ($d = 4$). In this specific example, the NIR channel is used to enhance the RGB image with an additional source of information. Constraints on output colors are provided (with parameter $\mu_3 = 1.7e3$) so that the conversion preserves colors for five given image features (dark and light trees, water, mountains, sky). This does not only act as regularization, but also provides us a way to automatically order the three output channels. Comparing our result with the method of Lau et al. [2011], we can see that Laplacian colormap provides an enhanced version of RGB while preserving the correct colors (*e.g.*, trees on the mountains have more detail than in RGB, but at the same time they do not present the blue-green halo that appears in [Lau et al., 2011]). Finally, in Figure 8.13 we show a fusion of four photos of a city in different lighting conditions into a single image, which looks visually plausible. Here μ_3=0.25 with penalization colors taken from the third image.

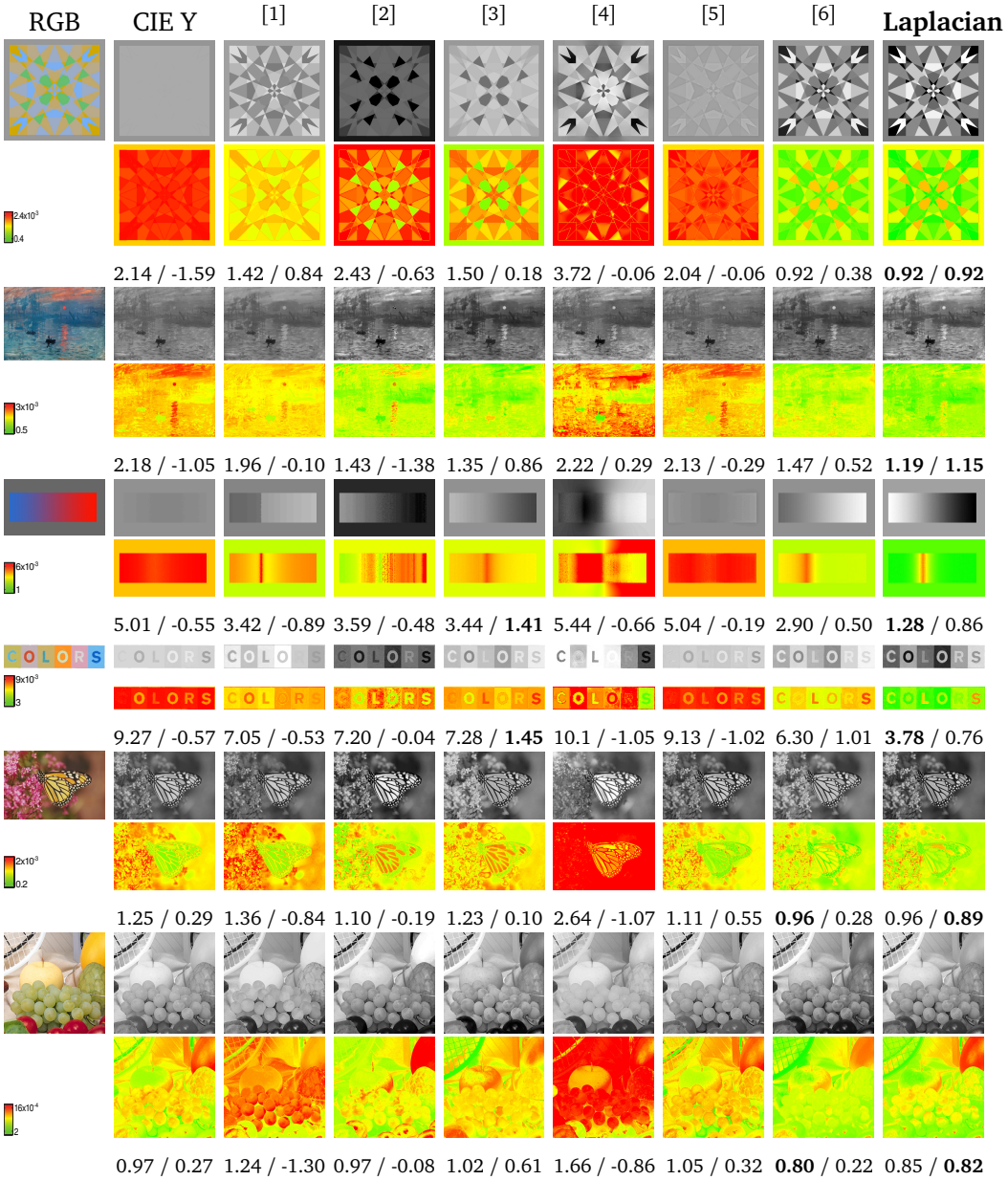


Figure 8.6. Decolorization experiment results. Left: original RGB image, right: grayscale conversion results. Rows 2, 5, : RWMS error images and mean RWMS (the smaller the better) / z-score (the larger the better) values. Our Laplacian colormap method performs the best in most cases. [1][Gooch et al., 2005], [2][Rasche et al., 2005a], [3][Grundland and Dodgson, 2007], [4][Neumann et al., 2007], [5][Smith et al., 2008], [6][Lu et al., 2012]

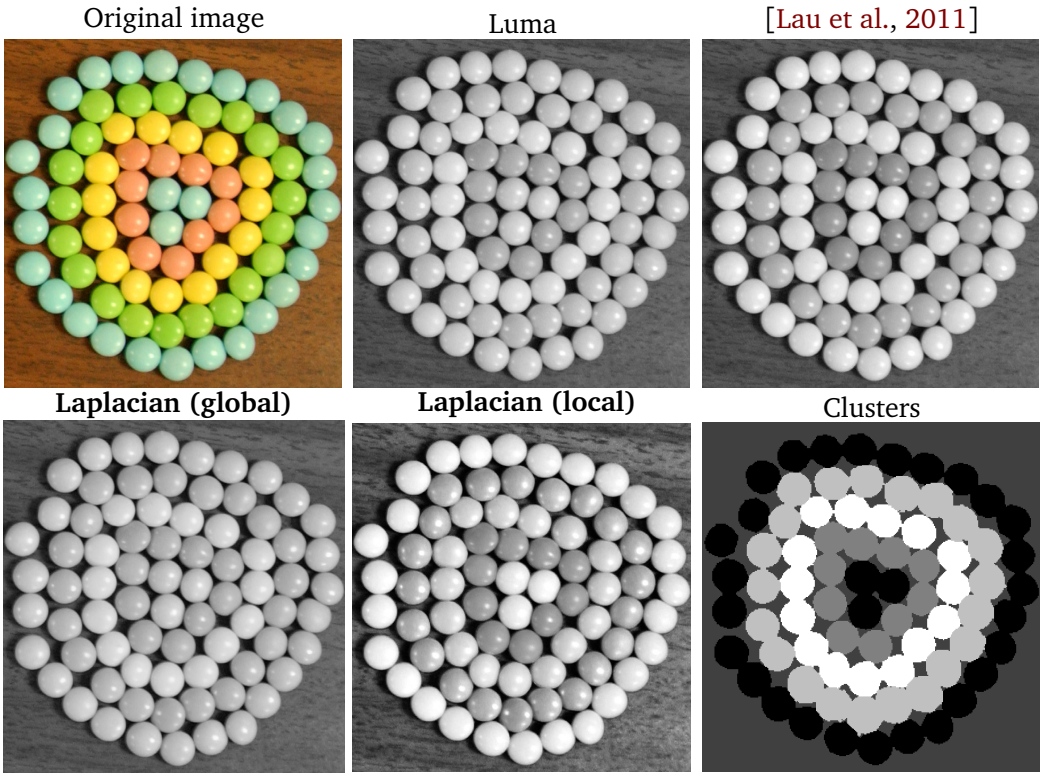


Figure 8.7. Global vs Local maps results. Top row, left-to-right: original image, Luma, result of Lau et al. [2011]. Bottom row: Laplacian colormaps using a global (left) and local (middle) map; the spatial weights used in the latter.

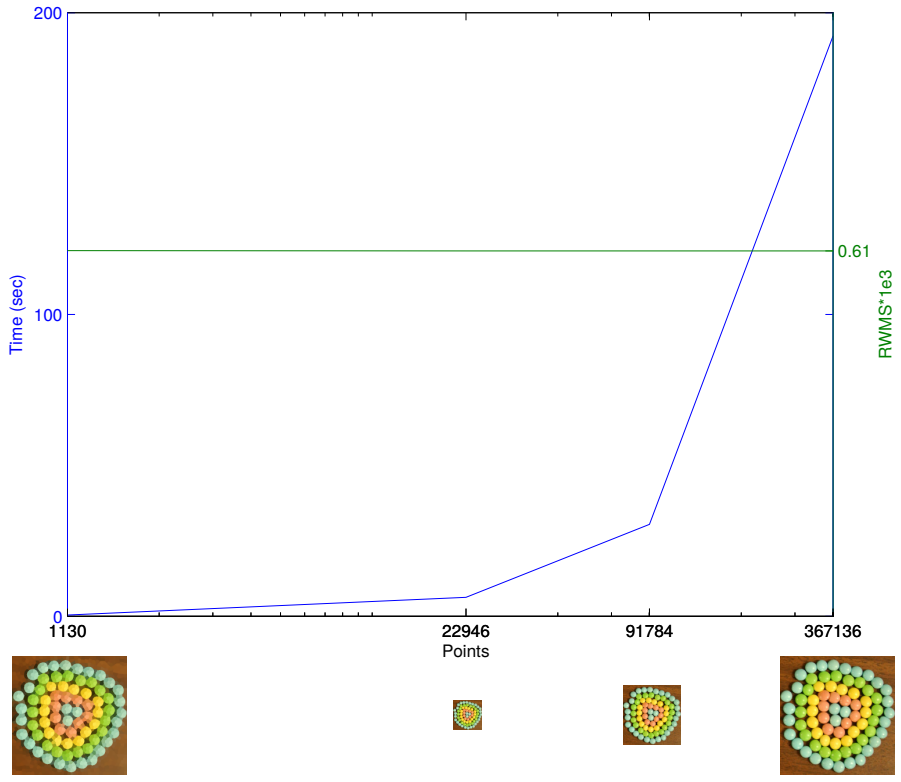


Figure 8.8. Computational complexity of Laplacian colormaps as function of the graph size.

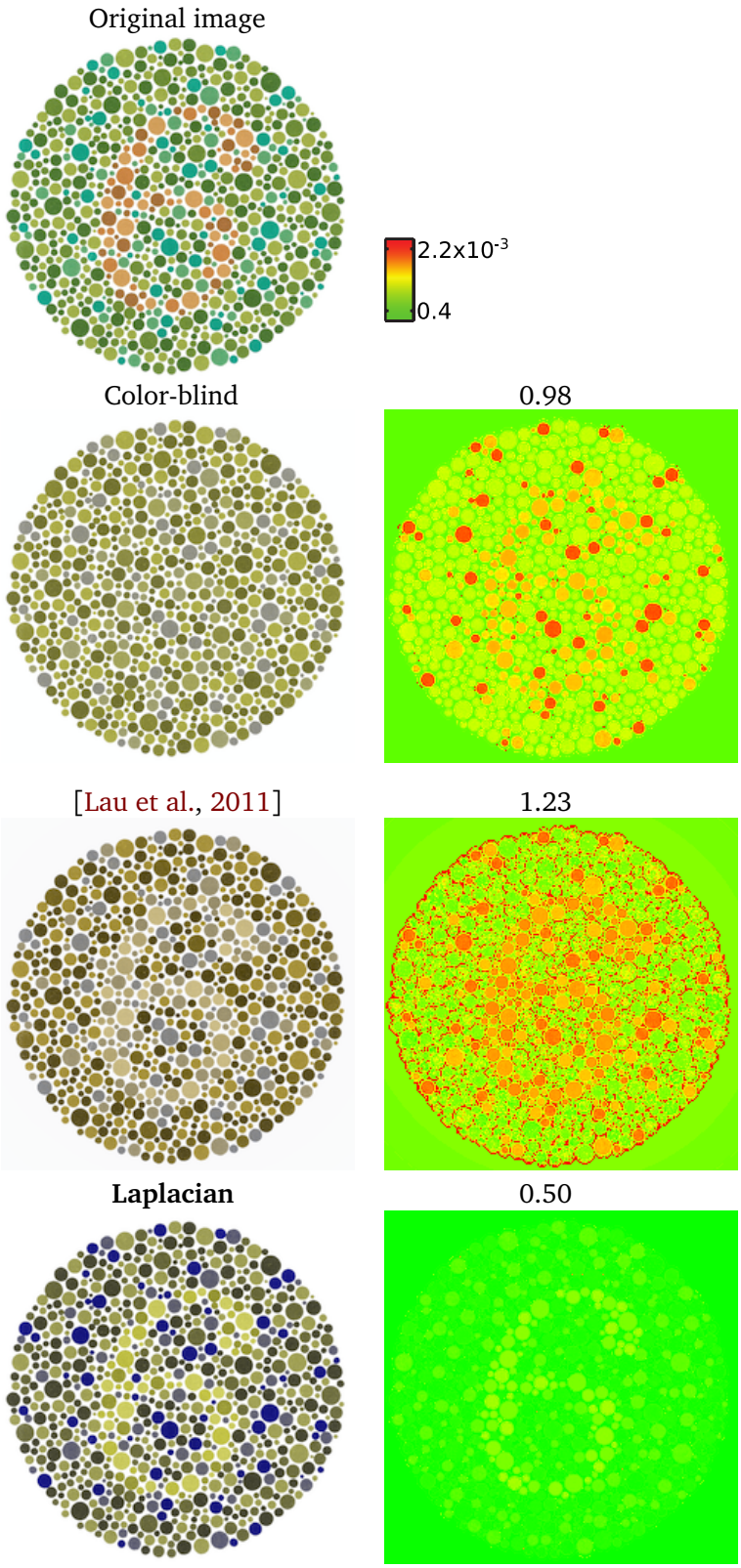


Figure 8.9. Color mapping for protanope color-blind observer. The first column, from top to bottom: original image, simulated color-blind, result from [Lau et al., 2011], and our result; the second column are their respective RWMS error images and mean RWMS values.

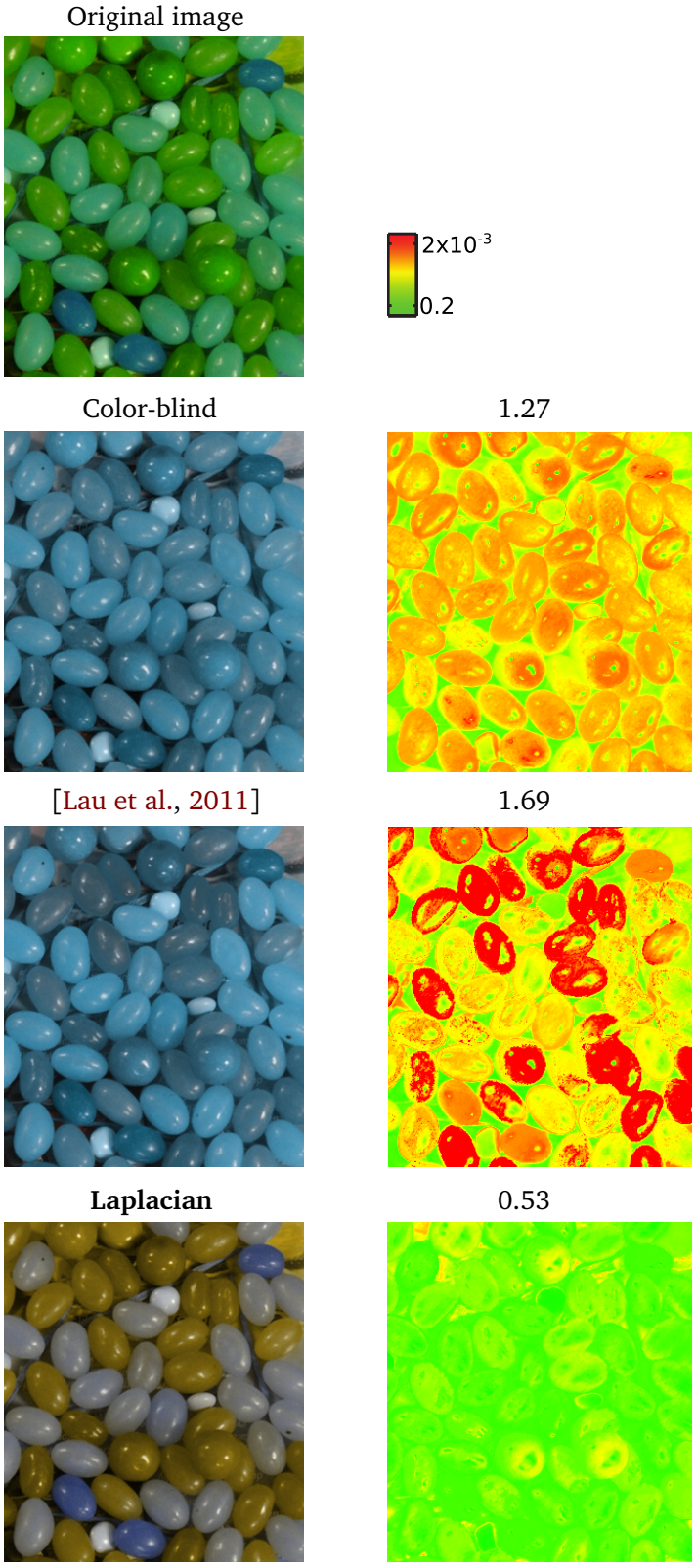


Figure 8.10. Color mapping for tritanope color-blind observer. The first column, from top to bottom: original image, simulated color-blind, result from [Lau et al., 2011], and our result; the second column are their respective RWMS error images and mean RWMS values.

Figure 8.11. Gamut mapping results. Right (top to bottom): original image, HPMINDE [CIE, 2004], the method of Lau et al. [2011], our approach. Left: gamut alerts for the images above (green shows the out-of-gamut pixels).

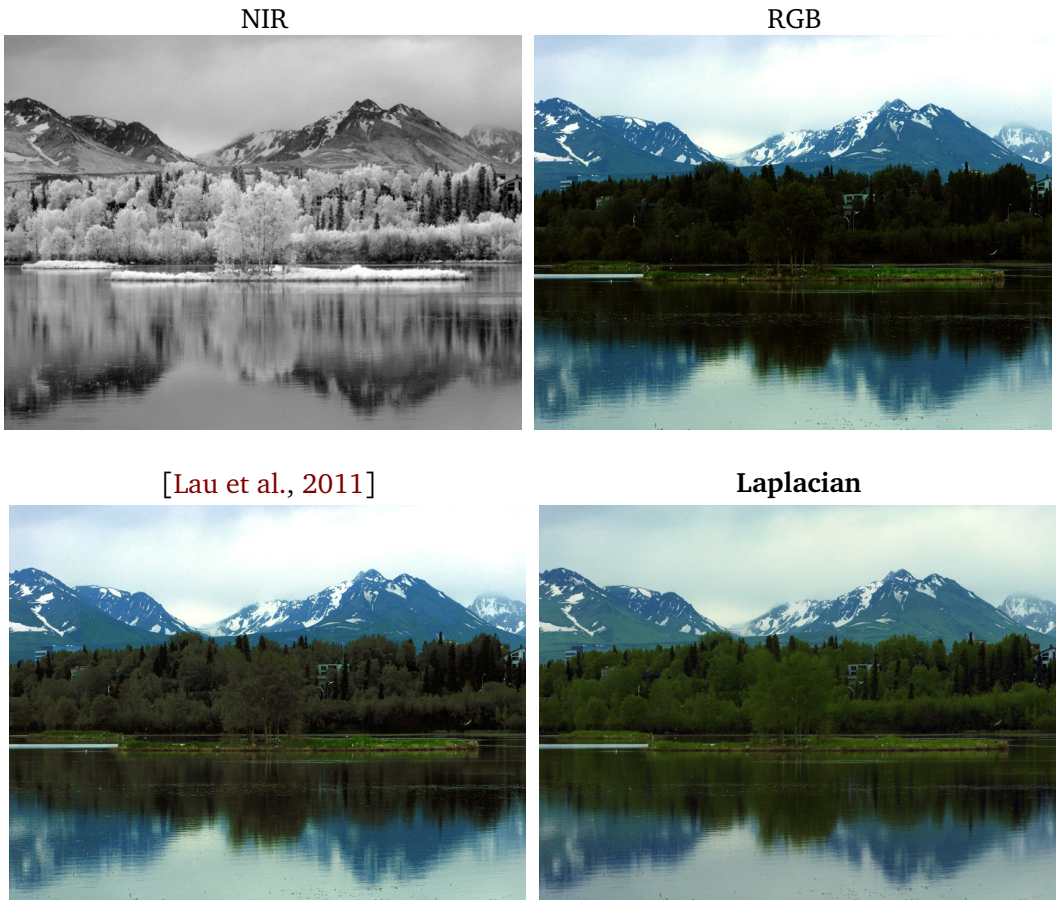


Figure 8.12. Multispectral (RGB + NIR) fusion results.

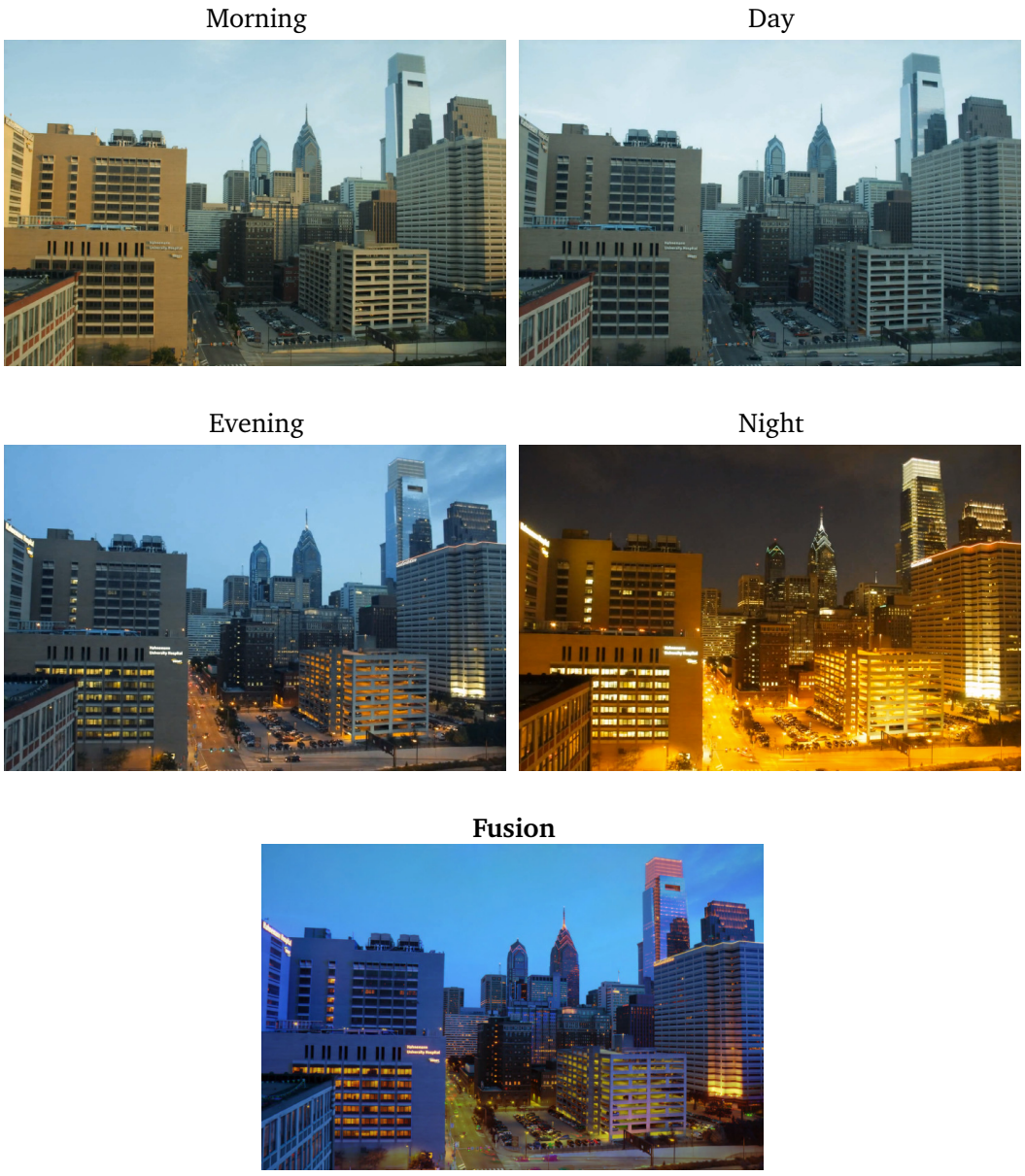


Figure 8.13. Fusion of images of four different illuminations of Philadelphia skyline into a single RGB image.

Chapter 9

Conclusions and Discussions

The successful application of spectral methods to problems in many fields made them a classical machinery for data analysis. Unfortunately, this success has so far not been replicated in more challenging problems of multimodal data analysis involving multiple non-Euclidean geometric domains. The growing number of such multimodal applications (such as multi-view clustering, shape correspondence, and multichannel image processing) calls for a generalization of spectral methods to the multimodal setting.

In Chapter 3, we introduced a general mathematically principled framework for such an extension. Our baseline approach is based on approximate joint diagonalization of Laplacian matrices, allowing a straightforward extension of various spectral data analysis tools such as spectral clustering and Laplacian eigenmaps.

The next step discussed in Chapter 4 is a more general coupled problem, extending the joint diagonalization approach by relaxing two main assumptions: equal dimensionality and a known bijective correspondence. Additionally, this approach allows to calculate a subset of joint Laplacian eigenvectors, rather than the whole orthonormal Laplacian basis as the standard joint diagonalization approaches. We also proposed an efficient numerical scheme for calculating these bases based on parameterized optimization in the subspace of the eigenvectors of the Laplacians using manifold optimization.

In machine learning applications considered in Chapter 6, the proposed ap-

proach achieved state-of-the-art performance on some standard multi-view clustering benchmarks. Moreover, we established that many existing works on multi-view clustering and manifold alignment can be considered as particular instances of our framework. In shape analysis applications presented in Chapter 7, we used the proposed method to establish similarity and dense intrinsic correspondence between deformable shapes in different representation.

Exploring noise- and outliers- robust formulation of the coupled diagonalization problem, in Chapter 5 we came up with a generic method (Manifold ADMM) for optimizing non-smooth matrix functions with manifold constraints. To the best of our knowledge, this is the first general optimization framework of this kind.

Finally, using the relation between joint diagonalizability and commutativity of matrices we introduced a new structural similarity measure of images, and applied it to structure-preserving color image manipulations (Chapter 8).

We believe that the important paradigm of simultaneous diagonalization of Laplacians that was introduced in this thesis and applied to a selection of problems from the domains of machine learning, computer vision, computer graphics and image processing, would allow addressing novel challenging problems on the one hand, and give a new look on classical problems on the other.

9.1 Follow-up Work

To much of our satisfaction, the proposed framework has been adopted by the community and already exploited in several important problems in different domains. We would like to mention a few interesting follow-up works we are aware of at the moment of writing this thesis, which exemplify the use of our methods in various applications.

One of the applications is in functional magnetic resonance imaging (fMRI) group studies, where the ability to establish correspondence across individuals is of key importance, as it enables location specific comparison of functional brain characteristics. Traditional registration approaches are based on brain morphology and do not take variability of functional localization into account. Nenning

et al. [2015] used our coupled diagonalization for multi-subject functional registration (Figure 9.1). In another follow-up from the medical domain, Cai et al. [2015] applied the joint diagonalization for multi-modal, multi-chamber, and multi-subject heart segmentation in cardiac imaging, a very challenging medical application (Figure 9.2).

In the domain of computer graphics, Zhu et al. [2015] used coupled Laplacian eigenbases for flexible and efficient example-based elastic deformation (Figure 9.3). Similarly to our pose transfer example, the main idea of this paper is that complex deformations from given examples are represented in the spectral domain and transferred to a previously unseen shape.

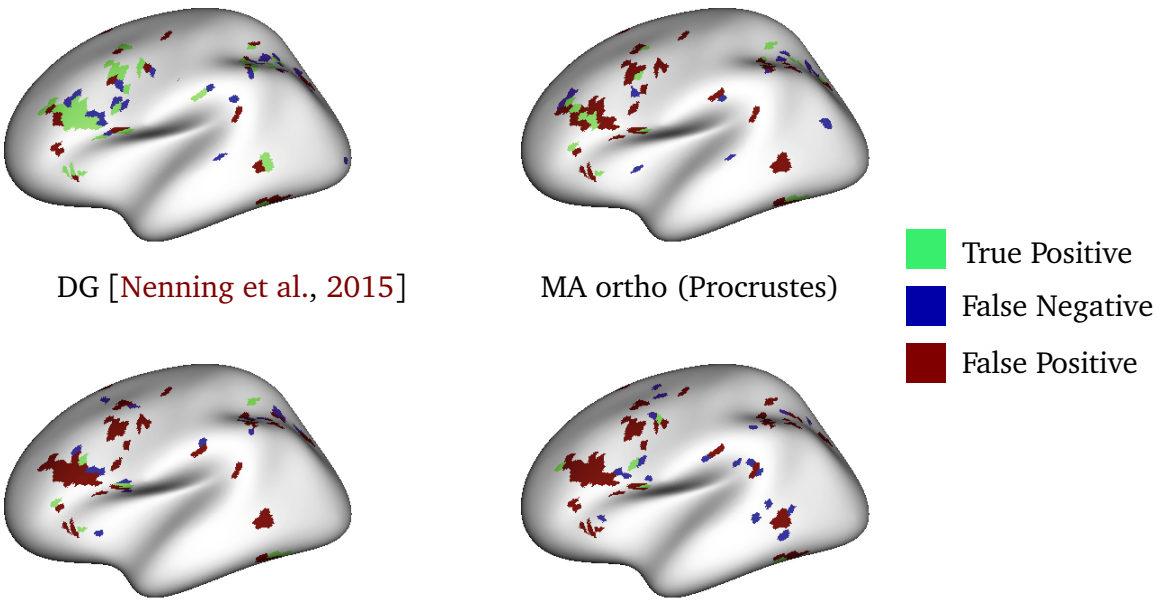


Figure 9.1. Nenning et al. [2015] use our method for multi-subject alignment of functional networks representing brain activations in fMRI. Shown here is the mapped activation on the left hemisphere. Our application of coupled diagonalization (DG) maps core regions related to language function successfully, with false positive and false negative mappings on the border areas. Figure reproduced from [Nenning et al., 2015].

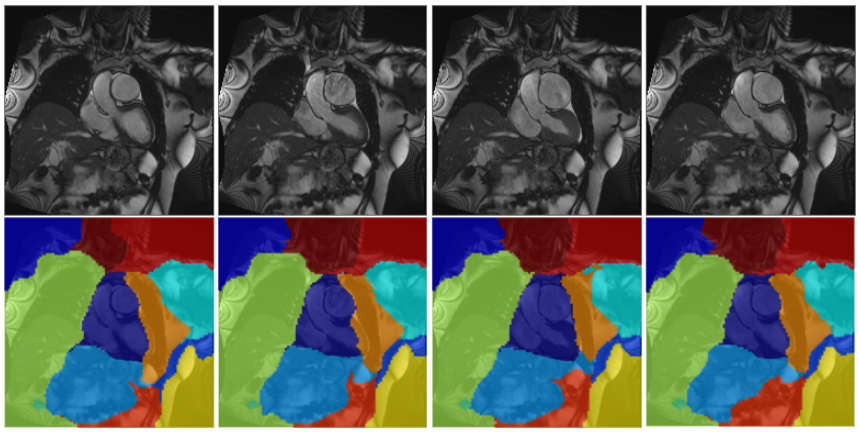


Figure 9.2. Cai et al. [2015] use our coupled diagonalization method for group-wise segmentation for coronal MRI scans. Our method successfully segments and identifies the hearts (shown in blue). Figure reproduced from [Cai et al., 2015].

In a follow-up paper [Kovnatsky et al., 2014] that is not included as part of this thesis, we ourselves extended the gamut mapping approach to devices with any number of primaries (rather than three primaries considered here).

9.2 Future Work

There are several promising avenues for a future extension of our work. On the theoretical front, it would be important to show bounds on the relation between the coupled eigenvectors and the original ones, similarly for the result of Theorem 1 for joint eigenvectors. Additional open problem we left in this thesis is the convergence proof of the MADMM algorithm, which would theoretically justify its use in many additional applications.

The need to know some kind of correspondence between the domains for eigenbasis coupling could be a limitation of our approach in some settings. Though automatic detection of correspondence between manifolds and graphs is a problem of its own extending way beyond the scope of this thesis, a rather straightforward extension of coupled diagonalization would be to combine it with the permutation recovery approach of [Pokrass et al., 2013]. In such a formulation, the corresponding functions are given up to an unknown order (with possibly

missing correspondences) and optimization is done both on the linear combination coefficients and the permutation matrix. This would allow to employ our approach in correspondence problems where only a set of automatically detected stable regions is given [Litman et al., 2011; Rodolà et al., 2012, 2014]. Another approach is to use information from a collection of manifolds for finding correspondence between them rather than relying only on a pair of manifolds, *e.g.*, of the recent promising work [Cosmo et al., 2016]. Though we considered only Laplacians in this thesis, our approach would naturally apply to other differential operators defined on manifolds, such as [Hildebrandt et al., 2012].

Finally, the structural image similarity measure based on Laplacian commutativity we used for color manipulations could also be employed for image alignment. We are aware of a current effort of performing multi-modal medical image alignment using this approach, where it successfully competes with traditional alignment criteria such as mutual information.

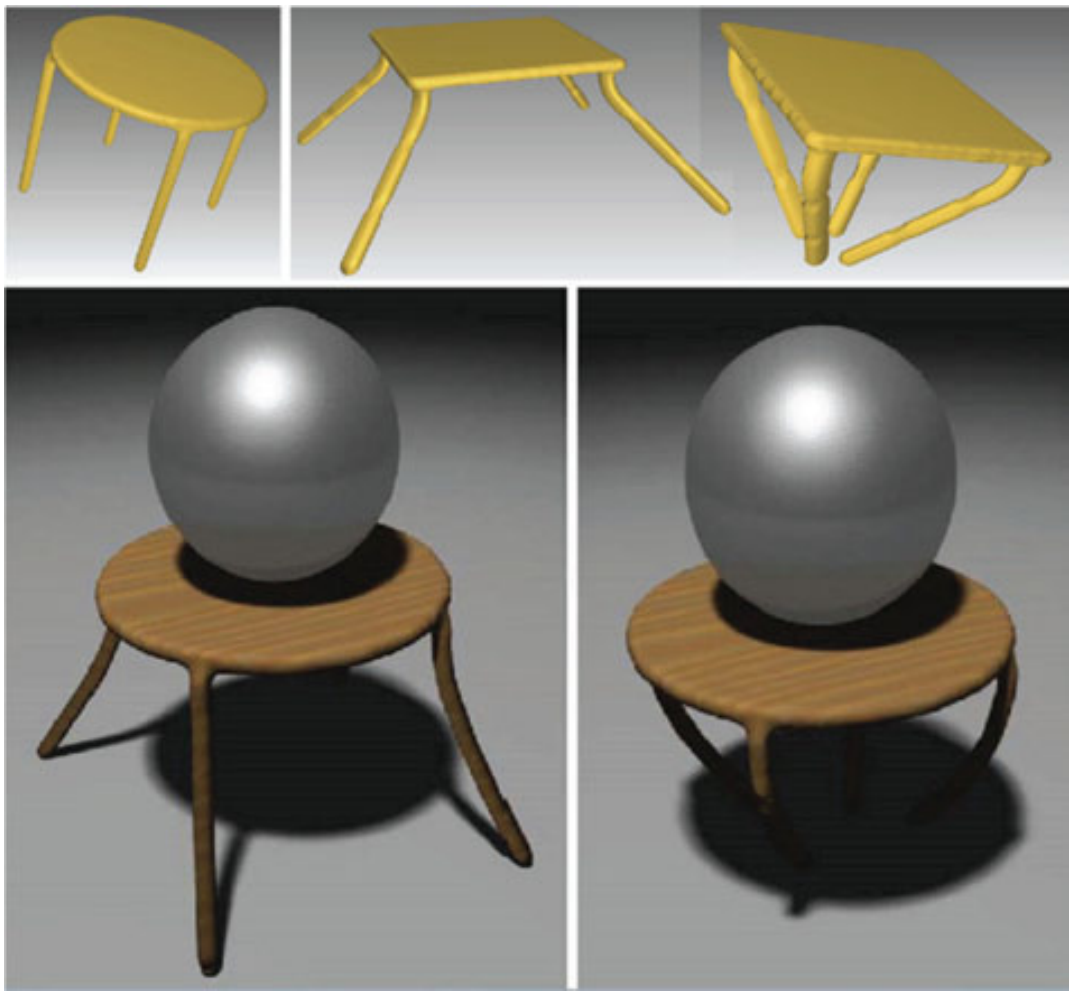


Figure 9.3. Zhu et al. [2015] use our method for example-based shape deformation. A round stool bends its legs while compressed by a steel sphere (bottom). The deformation is transferred from square stool deformations provided as deformation examples (top) using coupled Laplacian eigenbases. Figure reproduced from [Zhu et al., 2015].

Appendix A

Notation

A.1 General

$\mathbb{R}^d$	d-dimensional vector space of real numbers
a	scalar
$\mathbf{a} = (a_i)$	vector
$\mathbf{A} = (a_{ij})$	matrix
$\mathcal{M}$	manifold
id	identity map

A.2 Analysis

∇	(extrinsic) gradient
Δ	Laplacian operator

A.3 Algebra

$\mathbf{I}$	identity matrix

$\mathbf{1}$	column vector of ones
$\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$	eigendecomposition of symmetric square matrix $\mathbf{A}$ with eigenvectors $\mathbf{U}$ and eigenvalues $\boldsymbol{\Lambda}$
$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$	singular value decomposition of matrix $\mathbf{A}$ with singular vectors $\mathbf{U}, \mathbf{V}$ and singular values $\boldsymbol{\Sigma}$
$\|\mathbf{a}\|_2 = \mathbf{a}^\top\mathbf{a}$	Euclidean norm of vector $\mathbf{a}$
$\|\mathbf{a}\|_\mathbf{M} = \mathbf{b}^\top\mathbf{M}\mathbf{a}$	$\mathbf{M}$-norm of vector $\mathbf{a}$, for some $\mathbf{M} > 0$
$\|\mathbf{a}\|_1 = \sum_i \|a_i\|$	L_1 norm of vector $\mathbf{a}$
$\|\mathbf{A}\|_\mathrm{F} = \left(\sum_{ij} a_{ij}^2\right)^{1/2}$	Frobenius norm of matrix $\mathbf{A}$
$\|\mathbf{A}\|_{2,1} = \sum_i \|\mathbf{a}_i\|_2$	$L_{2,1}$ norm of matrix $\mathbf{A}$
$\|\mathbf{A}\|_* = \sum \sigma_i(\mathbf{A})$	Nuclear norm of matrix $\mathbf{A}$ ($\sigma_i(\mathbf{A})$ denotes the ith singular value)
$\times$	cross product
$\mathrm{diag}(\mathbf{A})$	Column vector comprising diagonal elements of $\mathbf{A}$
$\mathrm{Diag}(\mathbf{A})$	Matrix with diagonal of matrix $\mathbf{A}$ and other elements 0
$\mathrm{diag}([a_1, a_2, \ldots, a_k])$	Diagonal matrix with vector $[a_1, a_2, \ldots, a_k]$ at its diagonal
$\mathrm{vec}(\mathbf{A})$	Column vector obtained by column-wise stacking elements of matrix $\mathbf{A}$
$\mathrm{tr}(\mathbf{A}) = \sum_i a_{ii}$	Trace of matrix $\mathbf{A}$
$\mathbf{A} \odot \mathbf{B} = (a_{ij} b_{ij})$	Element-wise product of matrices
$[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$	Matrix commutator

A.4 Differential geometry

$T_m\mathcal{M}$	tangent space to the manifold $\mathcal{M}$ at point m
$\partial_{u_1}, \ldots, \partial_{u_p}$	basis tangent vectors of $T_m\mathcal{M}$
$\langle \cdot, \cdot \rangle_m$ or $g_m(\cdot, \cdot)$	Riemannian metric tensor on $\mathcal{M}$

$\exp_m$	exponential map at point m
γ	smooth curve on the manifold
$l(\gamma)$	length of the curve
$\langle\!\langle f,h \rangle\!\rangle_{\mathscr{M}}$	inner product on Riemannian manifold $\mathscr{M}$
$\mathscr{F}(\mathscr{M})$	space of scalar functions on $\mathscr{M}$
$L^2(\mathscr{M})$	space of square-integrable scalar functions on Riemannian manifold $\mathscr{M}$
$\nabla_{\mathscr{M}}$	intrinsic (Riemannian) gradient
$\Delta_{\mathscr{M}}$	Laplace-Beltrami operator on Riemannian manifold $\mathscr{M}$

A.5 Simultaneous diagonalization

$\mathrm{J}(\mathbf{A},\mathbf{B})$	measure of joint diagonalizability of two matrices
$\mathrm{off}(\mathbf{A}) = \sum_{i \neq j} a_{ij}^2$	off-diagonality penalty
$\tilde{\mathbf{\Phi}}$	joint approximate eigenvectors
$\hat{\mathbf{\Phi}}$	coupled eigenvectors

Appendix B

Numerical Implementation of Laplacian Colormaps

The algorithm starts by calculating a Laplacian $\mathbf{L}_\mathbf{X}$ from the RGB image $\mathbf{X}$, as described in Section 2.3 and illustrated in Figure 8.4 (left). Then, both $\mathbf{L}_\mathbf{X}$ and the image are passed as inputs to the optimization function, together with an initialization for $\boldsymbol{\theta}$. At each step, the optimization function calculates $\eta_{\boldsymbol{\theta}}(\mathbf{X})$ and its Laplacian $\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}$, then it tries to minimize the cost of 8.2 w.r.t. the parameters $\boldsymbol{\theta}$. The optimization was implemented in MATLAB, using interior-point method from the Optimization Toolbox.

Derivation of the gradients. The notations and discretizations are as in Section 2.3. Let $\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})} = \mathbf{D}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})} - \mathbf{W}_{\Phi_{\boldsymbol{\theta}}(\mathbf{X})}$ be the image Laplacian with weights as defined in (8.1). We denote by $|W|$ the number of non-zero elements in the adjacency matrix $\mathbf{W}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}$, and by n the number of parameters $\boldsymbol{\theta}$ of the colormap, respectively. The non-zero elements $w_{ij} > 0$ are indexed as $\mathbf{w}_{\boldsymbol{\theta}} = \left[w_1, \ldots, w_{|W|}\right] = \left[w_{i_1,j_1}, \ldots, w_{i_{|W|},j_{|W|}}\right]$. $\eta^i_{\boldsymbol{\theta}} : \mathbb{R}^d \to \mathbb{R}$ denotes the ith channel of the colormap, such that $\eta_{\boldsymbol{\theta}}(\mathbf{x}) = \left[\eta^1_{\boldsymbol{\theta}}(\mathbf{x}), \cdots, \eta^{d'}_{\boldsymbol{\theta}}(\mathbf{x})\right]^\top$, and $\nabla_{\boldsymbol{\theta}} \phi^i_{\boldsymbol{\theta}}$ is its gradient w.r.t. $\boldsymbol{\theta}$.

We now derive the gradients of the cost function (8.2). The gradient of the μ_2-term is trivial,

$$\nabla_{\boldsymbol{\theta}} \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_2^2 = 2(\boldsymbol{\theta} - \boldsymbol{\theta}_0).$$

Denote by $\mathbf{M}^i_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}$ the matrix of size $NM \times n$, whose jth row is the gradient of the ith channel at the jth pixel, $\nabla_{\boldsymbol{\theta}} \eta^i_{\boldsymbol{\theta}}(\mathbf{x}_j)^\top$, and define $NMd' \times n$ matrix

$\mathbf{M}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})} = \left[(\mathbf{M}^1_{\eta_{\boldsymbol{\theta}}(\mathbf{X})})^\top, \cdots, (\mathbf{M}^{d'}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})})^\top\right]^\top$. Differentiating the μ_3-term w.r.t. $\boldsymbol{\theta}$ gives

$$\nabla_{\boldsymbol{\theta}} \|\eta_{\boldsymbol{\theta}}(\mathbf{X}_c) - \mathbf{Y}_c\|_F^2 = 2\mathbf{M}^\top_{\eta_{\boldsymbol{\theta}}(\mathbf{X})} \left(\mathrm{vec}(\eta_{\boldsymbol{\theta}}(\mathbf{X}_c)) - \mathrm{vec}(\mathbf{Y}_c)\right).$$

The gradients of the first two terms of (8.2) are obtained by applying the chain rule. First, we differentiate the terms w.r.t. $\mathbf{w}_{\boldsymbol{\theta}}$, obtaining a gradient of size $|W| \times 1$. Next, we differentiate w.r.t. $\boldsymbol{\theta}$. The gradient of the adjacency matrix elements w_{ij} w.r.t. $\boldsymbol{\theta}$ is

$$\nabla_{\boldsymbol{\theta}} w_{ij} = -\frac{w_{ij}}{\sigma_r^2} \sum_{k=1}^{d'} (\eta^k_{\boldsymbol{\theta}}(\mathbf{x}_i) - \eta^k_{\boldsymbol{\theta}}(\mathbf{x}_j))(\nabla_{\boldsymbol{\theta}} \eta^k_{\boldsymbol{\theta}}(\mathbf{x}_i) - \nabla_{\boldsymbol{\theta}} \eta^k_{\boldsymbol{\theta}}(\mathbf{x}_j)).$$

The gradient of the commutator (μ_0-term) is

$$\frac{\partial}{\partial w_{ij}} \|[\mathbf{L}_\mathbf{X}, \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}]\|_F^2 = -2\left(\mathbf{O}_1 - \mathbf{L}^\top_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}[\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}, \mathbf{L}_\mathbf{X}] - \mathbf{O}_2 + [\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}, \mathbf{L}_\mathbf{X}]\mathbf{L}^\top_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}\right)_{ij}.$$

The gradient of the μ_1-term is

$$\frac{\partial}{\partial w_{ij}} \|\mathbf{L}_\mathbf{X} - \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}\|_F^2 = 2\left(\mathbf{O} + \mathbf{L}_\mathbf{X} - \mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}\right)_{ij},$$

where $\mathbf{O}, \mathbf{O}_k$ are matrices with equal columns given by

$$\begin{aligned}
\mathbf{O} &= (\mathrm{diag}(\mathbf{L}_\mathbf{X}), \cdots, \mathrm{diag}(\mathbf{L}_\mathbf{X})), \\
\mathbf{O}_1 &= (\mathrm{diag}(\mathbf{L}^\top_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}[\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}, \mathbf{L}_\mathbf{X}]), \cdots, \mathrm{diag}(\mathbf{L}^\top_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}[\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}, \mathbf{L}_\mathbf{X}])) \\
\mathbf{O}_2 &= (\mathrm{diag}([\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}, \mathbf{L}_\mathbf{X}]\mathbf{L}^\top_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}), \cdots, \mathrm{diag}([\mathbf{L}_{\eta_{\boldsymbol{\theta}}(\mathbf{X})}, \mathbf{L}_\mathbf{X}]\mathbf{L}^\top_{\eta_{\boldsymbol{\theta}}(\mathbf{X})})).
\end{aligned}$$

Finally, the gradient of the colormap appearing in the expressions above depends on the choice of the colormap. For all the experiments using the colormap $\eta_{\boldsymbol{\theta}}(\mathbf{x}) = \left[\eta^1_{\boldsymbol{\theta}}(\mathbf{x}), \cdots, \eta^{d'}_{\boldsymbol{\theta}}(\mathbf{x})\right]^\top$ defined previously in Section 8.2, the differentiation is straightforward.

In experiments simulating color blindness, the colormap is assumed to be $(\Pi \circ \eta_{\boldsymbol{\theta}})(\mathbf{x})$, where the transformation Π simulates the deficient observer is assumed

to be linear $\Pi(\mathbf{x}) = \mathbf{A}\mathbf{x}$. The Jacobian of the colormap is given as

$$\mathbf{J}_{\boldsymbol{\theta}}(\Pi \circ \eta_{\boldsymbol{\theta}})(\mathbf{x}) = \mathbf{J}_{\Pi}(\eta_{\boldsymbol{\theta}}(\mathbf{x}))\mathbf{J}_{\eta_{\boldsymbol{\theta}}}(\mathbf{x}) = \mathbf{A}\left[\nabla_{\boldsymbol{\theta}}\eta_{\boldsymbol{\theta}}^{1}(\mathbf{x}), \ldots, \nabla_{\boldsymbol{\theta}}\eta_{\boldsymbol{\theta}}^{d'}(\mathbf{x})\right]^{\top}.$$

Bibliography

Daltonize. http://www.daltonize.org/ (accessed: 01.2016).

P.-A. Absil and K. A. Gallivan. Joint diagonalization on the oblique manifold for independent component analysis. In *Proc. ICASSP*, 2006.

P-A Absil, C. G. Baker, and K. A. Gallivan. Trust-region methods on Riemannian manifolds. *Foundations of Computational Mathematics*, 7(3):303–330, 2007.

P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

X. Alameda-Pineda, V. Khalidov, R. Horaud, and F. Forbes. Finding audio-visual events in informal social gatherings. In *Proc. ICMI*, 2011.

E. Alpaydin and C. Kaynak. Cascading classifiers. *Kybernetika*, 34(4):369–374, 1998.

A. Alsam and I. Farup. *Colour gamut mapping as a constrained variational problem*, pages 109–118. Springer, 2009.

F. Alvarez, J. Bolte, and J. Munier. A unifying local convergence result for Newton's method in Riemannian manifolds. *Foundations of Computational Mathematics*, 8(2):197–226, 2008.

M. R. Amini, N. Usunier, and C. Goutte. Learning from multiple partially observed views-an application to multilingual text categorization. In *Proc. NIPS*, 2009.

R. Andreani, E. G Birgin, J. M. Martínez, and M. L. Schuverdt. On augmented Lagrangian methods with general lower-level constraints. *SIAM Journal on Optimization*, 18(4):1286–1309, 2007.

M. Arie-Nachimson, S. Z. Kovalsky, I. Kemelmacher-Shlizerman, A. Singer, and R. Basri. Global motion estimation from point matches. In *Proc. 3DIMPVT*, 2012.

H. Attouch, J. Bolte, P. Redont, and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Lojasiewicz inequality. *Mathematics of Operations Research*, 35 (2):438–457, 2010.

M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: a quantum mechanical approach to shape analysis. In *Proc. 4DMOD*, 2011.

R. Balasubramanian, R. L. de Queiroz, R. Eschbach, and W. Wu. Gamut mapping to preserve spatial luminance variations. In *Proc. CIC*, 2000.

M. Bansal and K. Daniilidis. Joint spectral correspondence for disparate image matching. In *Proc. CVPR*, 2013.

R. Bekkerman and J. Jeon. Multi-modal clustering for multimedia collections. In *Proc. CVPR*, 2007.

M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2002.

M. Belkin, J. Sun, and Y. Wang. Discrete Laplace operator on meshed surfaces. In *Proc. SCG*, 2008.

P. Bèrard, G. Besson, and S. Gallot. Embedding Riemannian manifolds by their heat kernel. *Geometric and Functional Analysis*, 4(4):373–398, 1994.

J. M. F. Berge. Orthogonal Procrustes rotation for two or more matrices. *Psychometrika*, 42(2):267–276, 1977.

D. P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 1982.

N. Bonnier, F. Schmitt, H. Brettel, and S. Berche. Evaluation of spatial gamut mapping algorithms. In *Proc. CIC*, 2006.

M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy. *Polygon Mesh Processing*. Ak Peters Series. Taylor & Francis, 2010.

P. T. Boufounos and R. G. Baraniuk. 1-bit compressive sensing. In *Proc. CISS*, 2008.

N. Boumal and P.-A. Absil. RTRMC: A Riemannian trust-region method for low-rank matrix completion. In *Proc. NIPS*, 2011.

N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre. Manopt, a Matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15(1): 1455–1459, 2014.

S. Boyd, N. Parikk, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

H. Brettel, F. Viénot, and J. D. Mollon. Computerized simulation of color appearance for dichromats. *Journal of the Optical Society of America A*, 14(10): 2647–2655, 1997.

A. M. Bronstein, M. M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer, 2008.

A. M. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro. A Gromov-Hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching. *International Journal of Computer Vision*, 89(2-3):266–286, 2010a.

M. M. Bronstein and K. Glashoff. Heat kernel coupling for multiple graph analysis. *arXiv:1312.3035*, 2013.

M. M. Bronstein and I. Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Proc. CVPR*, 2010.

M. M. Bronstein, A. M. Bronstein, F. Michel, and N. Paragios. Data fusion through cross-modality metric learning using similarity-sensitivg hashing. In *Proc. CVPR*, 2010b.

M. M. Bronstein, K. Glashoff, and T. A. Loring. Making Laplacians commute. *arXiv:1307.6549*, 2013.

A. Bunse-Gerstner, R. Byers, and V. Mehrmann. Numerical methods for simultaneous diagonalization. *SIAM Journal on Matrix Analysis and Applications*, 14 (4):927–949, 1993.

D. Burago, Y. D. Burago, and S. Ivanov. *A Course in Metric Geometry*. American Mathematical Society, 2001.

M. Čadík. Perceptual evaluation of color-to-grayscale image conversions. *Computer Graphics Forum*, 27(7):1745–1754, 2008.

X. Cai, F. Nie, H. Huang, and F. Kamangar. Heterogeneous image feature integration via multi-modal spectral clustering. In *Proc. CVPR*, 2011.

Y. Cai, A. Islam, M. Bhaduri, I. Chan, and S. Li. Unsupervised free-view groupwise segmentation for m3 cardiac images using synchronized spectral network. In *Proc. MICCAI*, 2015.

J. F. Cardoso. Perturbation of joint diagonalizers. Technical report, Télécom Paris, 1995.

J. F. Cardoso and A. Souloumiac. Blind beamforming for non-gaussian signals. *Radar and Signal Processing*, 140(6):362–370, 1993.

J. F. Cardoso and A. Souloumiac. Jacobi angles for simultaneous diagonalization. *SIAM Journal on Matrix Analysis and Applications*, 17(1):161–164, 1996.

X. Chen. Smoothing methods for nonsmooth, nonconvex minimization. *Mathematical Programming*, 134(1):71–99, 2012.

T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng. NUS-WIDE: A real-world web image database from national university of singapore. In *Proc. CIVR*, 2009.

CIE. Guidelines for the evaluation of color gamut mapping algorithms. Technical Report 156, 2004.

A. Clements and H. Zhang. Robust 3D shape correspondence in the spectral domain. In *Proc. SMI*, 2006.

R. R. Coifman, S. Lafon, A. Lee, M. Maggioni, F. Warner, and S. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the National Academy of Science*, 102(21): 7426–7431, 2005.

L. Cosmo, E. Rodolà, A. Albarelli, F. Memoli, and D. Cremers. Consistent partial matching of shape collections via sparse modeling. *Computer Graphics Forum*, 2016. to appear.

M. Cucuringu, Y. Lipman, and A. Singer. Sensor network localization by eigenvector synchronization over the Euclidean group. *ACM Transactions on Sensor Networks*, 8(3):19:1–19:42, 2012a.

M. Cucuringu, A. Singer, and D. Cowburn. Eigenvector synchronization, graph rigidity and the molecule problem. *Information and Inference*, 1(1):21–67, 2012b.

M. Cui, J. Hu, A. Razdan, and P. Wonka. Color-to-gray conversion using isomap. *Visual Computer*, 26(11):1349–1360, 2010.

C. H. Q. Ding, H. Xiaofeng, Z. Hongyuan, G. Ming, and H. D. Simon. A min-max cut algorithm for graph partitioning and data clustering. In *Proc. CDM*, 2001.

M. P. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, 1976.

M. P. do Carmo. *Riemannian Geometry*. Birkhäuser, 1992.

X. Dong, P. Frossard, P. Vandergheynst, and N. Nefedov. Clustering on multi-layer graphs via subspace analysis on Grassmann manifolds. *IEEE Transactions on Signal Processing*, 62(4):905–918, 2014.

A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20 (2):303–353, 1998.

A. C. Evans, S. Marrett, P. Neelin, L. Collins, K. Worsley, W. Dai, S. Milot, E. Meyer, and D. Bub. Anatomical mapping of functional activation in stereotactic coordinate space. *Neuroimage*, 1(1):43–53, 1992.

D. Eynard, A. Kovnatsky, and M. M. Bronstein. Laplacian colormaps: a framework for structure-preserving color transformations. *Computer Graphics Forum*, 33 (2):215 – 224, 2014.

D. Eynard, A. Kovnatsky, M. M. Bronstein, K. Glashoff, and A. M. Bronstein. Multimodal manifold analysis using simultaneous diagonalization of Laplacians. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(12):2505–2517, 2015.

O. P. Ferreira and P. R. Oliveira. Subgradient algorithm on Riemannian manifolds. *Journal of Optimization Theory and Application*, 97(1):93–104, 1998.

M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In *Advances in Multiresolution for Geometric Modelling*, Mathematics and Visualization, pages 157–186. Springer, 2005.

K. Glashoff and M. M. Bronstein. Matrix commutators: their asymptotic metric properties and relation to approximate joint diagonalization. *Linear Algebra and its Applications*, 438(8):2503–2513, 2013.

A. A. Gooch, S. C. Olsen, J. Tumblin, and B. Gooch. Color2gray: salience-preserving color removal. *ACM Transactions on Graphics*, 24(3):634–639, 2005.

M. Grundland and N. A. Dodgson. Decolorize: Fast, contrast enhancing, color to grayscale conversion. *Pattern Recognition*, 40(11):2891–2896, 2007.

J. Ham, D. Lee, and L. Saul. Semisupervised alignment of manifolds. In *Proc. UAI*, 2005.

M. R. Hestenes. Multiplier and gradient methods. *Journal of Optimization Theory and Applications*, 4(5):303–320, 1969.

N. J. Higham. Computing the nearest correlation matrix – a problem from finance. *IMA Journal of Numerical Analysis*, 22(3):329–343, 2002.

K. Hildebrandt, C. Schulz, C. von Tycowicz, and K. Polthier. Modal shape analysis beyond Laplacian. *Computer Aided Geometric Design*, 29(5):204 – 218, 2012.

D. S. Hochbaum and D. B. Shmoys. A best possible heuristic for the k-center problem. *Mathematics of Operations Research*, 10(2):180–184, 1985.

R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1990.

J. Hu and J. Hua. Salient spectral geometric features for shape matching and retrieval. *Visual Computing*, 25(5-7):667–675, 2009.

J.-B. Huang, Y.-C. Tseng, S.-I. Wu, and S.-J. Wang. Information preserving color transformation for protanopia and deuteranopia. *Signal Processing Letters*, 14 (10):711–714, 2007.

Q. Huang, F. Wang, and L. Guibas. Functional map networks for analyzing and exploring large shape collections. *ACM Transactions on Graphics*, 33(4):36:1–36:11, 2014.

C. J. G. Jacobi. Über ein leichtes Verfahren, die in der Theorie der Säkularstörungen vorkommenden Gleichungen numerisch aufzulösen. *Journal für reine und angewandte Mathematik*, 30:51–95, 1846.

A. Jacobson and O. Sorkine. A cotangent Laplacian for images as surfaces. Technical Report 757, ETH Zurich, 2012.

P. W. Jones, M. Maggioni, and R. Schul. Manifold parametrizations by eigenfunctions of the laplacian and heat kernels. *Proceedings of the National Academy of Science*, 105(6):1803–1808, 2008.

M. Journée, F. Bach, P.-A. Absil, and R. Sepulchre. Low-rank optimization on the cone of positive semidefinite matrices. *SIAM Journal on Optimization*, 20(5): 2327–2351, 2010.

Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *Proceedings of SIGGRAPH*, 2000.

R. H. Keshavan and S. Oh. A gradient descent algorithm on the Grassman manifold for matrix completion. *arXiv:0910.5260*, 2009.

E. Kidron, Y. Y. Schechner, and M. Elad. Pixels that sound. In *Proc. CVPR*, 2005.

B. M. Kim and J. Rossignac. Geofilter: Geometric selection of mesh filter parameters. *Computer Graphics Forum*, 24(3):295–302, 2005.

H.-J. Kim, J.-Y. Jeong, Y.-J. Yoon, Y.-H. Kim, and S.-J. Ko. Color modification for color-blind viewers using the dynamic color transformation. In *Proc. ICCE*, 2012.

S. J. Kim, F. Deng, and M. S. Brown. Visual enhancement of old documents with hyperspectral imaging. *Pattern Recognition*, 44(7):1461–1469, 2011a.

V. G. Kim, Y. Lipman, and T. Funkhouser. Blended intrinsic maps. *ACM Transactions on Graphics*, 30(4):79:1–79:12, 2011b.

Y. Kim, C. Jang, J. Demouth, and S. Lee. Robust color-to-gray via nonlinear global mapping. *ACM Transactions on Graphics*, 28(5):161:1–161:4, 2009.

R. Kimmel, D. Shaked, M. Elad, and I. Sobel. Space-dependent color gamut mapping: a variational approach. *Transactions on Image Processing*, 14(6): 796–803, 2005.

M. Kleinsteuber and H. Shen. Blind source separation with compressively sensed linear mixtures. *Signal Processing Letters*, 19(2):107–110, 2012.

L. Kobbelt. Discrete fairing. In *Proc. IMA MS*, 1997.

A. Kovnatsky, M. M. Bronstein, and A. M. Bronstein. Stable spectral mesh filtering. In *Proc. NORDIA*, 2012.

A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, K. Glashoff, and R. Kimmel. Coupled quasi-harmonic bases. *Computer Graphics Forum*, 32(2):439–448, 2013.

A. Kovnatsky, D. Eynard, and M. M. Bronstein. Gamut mapping with image Laplacian commutators. In *Proc. ICIP*, 2014.

A. Kovnatsky, K. Glashoff, and M. M. Bronstein. MADMM: a generic algorithm for non-smooth optimization on manifolds. In *Proc. ECCV*, 2016.

G. R. Kuhn, M. M. Oliveira, and L. A. F. Fernandes. An improved contrast enhancing approach for color-to-grayscale mappings. *Visual Computing*, 24(7): 505–514, 2008a.

G. R. Kuhn, M. M. Oliveira, and L. A. F. Fernandes. An efficient naturalness-preserving image-recoloring method for dichromats. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1747–1754, 2008b.

J. G. Kuk, J. H. Ahn, and N. I. Cho. A color to grayscale conversion considering local and global contrast. In *Proc. ACCV*, 2010.

A. Kumar, P. Rai, and H. Daumé III. Co-regularized multi-view spectral clustering. In *Proc. NIPS*, 2011.

R. Lai and S. Osher. A splitting method for orthogonality constrained problems. *Journal of Scientific Computing*, 58(2):431–449, 2014.

C. Lau, W. Heidrich, and R. Mantiuk. Cluster-based color space optimizations. In *Proc. ICCV*, 2011.

Y. Ledyaev and Q. Zhu. Nonsmooth analysis on smooth manifolds. *Transactions of the American Mathematical Society*, 359(8):3687–3732, 2007.

B. Lévy. Laplace-Beltrami eigenfunctions towards an algorithm that 'understands' geometry. In *Proc. SMI*, 2006.

B. Lévy and R. H. Zhang. Spectral geometry processing. In *ACM Transactions on Graphics Asia Course Notes*, 2009.

H. Lin. Almost commuting selfadjoint matrices and applications. *Fields Institute Communications American Mathematical Society*, 13:193–233, 1997.

R. Litman, A. M. Bronstein, and M. M. Bronstein. Diffusion-geometric maximally stable component detection in deformable shapes. *Computers & Graphics*, 35 (3):549 – 560, 2011.

J. Liu, C. Wang, J. Gao, and J. Han. Multi-view clustering via joint nonnegative matrix factorization. In *Proc. SDM*, 2013.

C. Lu, L. Xu, and J. Jia. Real-time contrast preserving decolorization. In *Proc. SIGGRAPH Asia*, 2012.

C. Ma and C.-H. Lee. Unsupervised anchor shot detection using multi-modal spectral clustering. In *Proc. ICASSP*, 2008.

C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

B. McFee and G.R.G. Lanckriet. Learning multi-modal similarity. *Journal of Machine Learning Research*, 12:491–523, 2011.

G. Meyer, S. Bonnabel, and R. Sepulchre. Linear regression under fixed-rank constraints: a Riemannian approach. In *Proc. ICML*, 2011.

G. W. Meyer and D. P. Greenberg. Color-defective vision and computer graphics displays. *IEEE Computer Graphics and Applications*, 8(5):28–40, 1988.

M. Meyer, M. Desbrun, P. Schröder, and A. H. Barr. *Visualization and Mathematics III*, chapter Discrete Differential-Geometry Operators for Triangulated 2-Manifolds, pages 35–57. Springer, 2003.

J. Morovič. *Color Gamut Mapping*. Wiley, 2008.

B. Nadler, S. Lafon, R. R. Coifman, and I.G. Kevrekidis. Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators. In *Proc. NIPS*, 2005.

S. Nakauchi, S. Hatanaka, and S. Usui. Color gamut mapping based on a perceptual image difference measure. *Color Research & Application*, 24(4):280–291, 1999.

A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa. Laplacian mesh optimization. In *Proc. GRAPHITE*, 2006.

K. H. Nenning, K. Kollndorfer, V. Schöpf, D. Prayer, and G. Langs. Multi-subject manifold alignment of functional network structures via joint diagonalization. In *Proc. IPMI*, 2015.

L. Neumann, M. Čadík, and A. Nemcsics. An efficient perception-based adaptive color to gray transformation. In *Proc. CAGVI*, 2007.

T. Neumann, K. Varanasi, C. Theobalt, M. Magnor, and M. Wacker. Compressed manifold modes for mesh processing. *Computer Graphics Forum*, 33(5):35–44, 2014.

A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proc. NIPS*, 2001.

R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama. Watermarking 3D polygonal meshes in the mesh spectral domain. In *Proc. GI*, 2001.

R. Ohbuchi, A. Mukaiyama, and S. Takahashi. A frequency-domain approach to watermarking 3D shapes. *Computer Graphics Forum*, 21(3):373–382, 2002.

M. Ovsjanikov, J. Sun, and L. Guibas. Global intrinsic symmetries of shapes. *Computer Graphics Forum*, 27(5):1341–1348, 2008.

M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, and L. Guibas. Functional maps: A flexible representation of maps between shapes. *ACM Transactions on Graphics*, 31(4):30:1–30:11, 2012.

N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):127–239, 2014.

B. N. Parlett. *The symmetric eigenvalue problem*. Prentice-Hall, 1998.

P. Perona and L. Zelnik-Manor. Self-tuning spectral clustering. In *Proc. NIPS*, 2004.

U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2(1):15–36, 1993.

N. Pinto. UCSD-MIT Caltech-101-mkl dataset, 2009. URL http://mkl.ucsd.edu/dataset/ucsd-mit-caltech-101-mkl-dataset.

J. Pokrass, A. M. Bronstein, M. M. Bronstein, P. Sprechmann, and G. Sapiro. Sparse modeling of intrinsic correspondences. *Computer Graphics Forum*, 32 (2):459–468, 2013.

M. J. D. Powell. A method for nonlinear constraints in minimization problems. In R. Fletcher, editor, *Optimization*. Academic Press, 1969.

K. Rahbar and J. P. Reilly. Geometric optimization methods for blind source separation of signals. In *Proc. ICA*, 2000.

K. Rasche, R. Geist, and J. Westall. Re-coloring images for gamuts of lower dimension. *Computer Graphics Forum*, 24(3):423–432, 2005a.

K. Rasche, R. Geist, and J. Westall. Detail preserving reproduction of color images for monochromats and dichromats. *IEEE Computer Graphics and Applications*, 25(3):22–30, 2005b.

N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G.R.G. Lanckriet, R. Levy, and N. Vasconcelos. A new approach to cross-modal multimedia retrieval. In *Proc. ICM*, 2010.

X. Ren and J. Malik. Learning a classification model for segmentation. In *Proc. ICCV*, 2003.

M. Reuter. Hierarchical shape segmentation and registration via topological features of Laplace-Beltrami eigenfunctions. *International Journal of Computer Vision*, 89(2):287–308, 2010.

M. Reuter, F. E. Wolter, and N. Peinecke. Laplace-Beltrami spectra as "Shape-DNA" of surfaces and solids. *Computer-Aided Design*, 38(4):342–366, 2006.

E. Rodolà, A. M. Bronstein, A. Albarelli, F. Bergamasco, and A. Torsello. A game-theoretic approach to deformable shape matching. In *Proc. CVPR*, 2012.

E. Rodolà, S. R. Bulo, and D. Cremers. Robust region detection via consensus segmentation of deformable shapes. *Computer Graphics Forum*, 33(5):97–106, 2014.

G. Rong, Y. Cao, and X. Guo. Spectral mesh deformation. *The Visual Computer*, 24(7):787–796, 2008.

G. Rosman, Y. Wang, X.-C. Tai, R. Kimmel, and A. M. Bruckstein. Fast regularization of matrix-valued images. In *Proc. ECCV*, 2011.

G. Rosman, X.-C. Tai, R. Kimmel, and A. M. Bruckstein. Augmented-Lagrangian regularization of matrix-valued maps. *Methods and Applications of Analysis*, 21 (1):121–138, 2014.

R. M. Rustamov. Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In *Proc. SGP*, 2007.

P. H. Schönemann. A generalized solution of the orthogonal Procrustes problem. *Psychometrika*, 31(1):1–10, 1966.

U. Shalit, D. Weinshall, and G. Chechik. Online learning in the manifold of low-rank matrices. In *Proc. NIPS*, 2010.

A. Sharma, R. Horaud, D. Knossow, and E. von Lavante. Mesh segmentation using Laplacian eigenvectors and gaussian mixtures. In *Proc. AAAI*, 2009.

E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *Proc. CVPR*, 2007.

J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The Princeton shape benchmark. In *Proc. SMI*, 2004.

A. Shtern and R. Kimmel. Matching the LBO eigenspace of non-rigid shapes via high order statistics. *Axioms*, 3(3):300, 2014.

K. Smith, P.-E. Landes, J. Thollot, and K. Myszkowski. Apparent greyscale: A simple and fast conversion to perceptually accurate images and video. *Computer Graphics Forum*, 27(2):193–200, 2008.

S. T. Smith. Optimization techniques on Riemannian manifolds. *Fields Institute Communications*, 3(3):113–135, 1994.

O. Sorkine. Differential representations for mesh processing. *Computer Graphics Forum*, 25(4):789–807, 2006.

O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian surface editing. In *Proc. SGP*, 2004.

R. W. Sumner and J. Popović. Deformation transfer for triangle meshes. *ACM Transactions on Graphics*, 23(3):399–405, 2004.

J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum*, 28(5): 1383–1392, 2009.

S. Süsstrunk and C. Fredembach. Enhancing the visible with the invisible. In *Proc. SID*, 2010.

M. Tan, I. W. Tsang, L. Wang, B. Vandereycken, and S. J. Pan. Riemannian pursuit for big matrix recovery. In *Proc. ICML*, 2014.

W. Tang, Z. Lu, and I. S. Dhillon. Clustering with multiple graphs. In *Proc. ICDM*, 2009.

G. Taubin. *A signal processing approach to fair surface design*. ACM, 1995.

L. L. Thurstone. A law of comparative judgment. *Psychological Review*, 34(4): 273–286, 1927.

K. Tsukida and M. R. Gupta. How to analyze paired comparison data. Technical report, DTIC Document, 2011.

B. Vallet and B. Lévy. Spectral geometry processing with manifold harmonics. *Computer Graphics Forum*, 27(2):251–260, 2008.

L. Váša, S. Marras, K. Hormann, and G. Brunnett. Compressing dynamic meshes with geometric Laplacians. *Computer Graphics Forum*, 33(2):145–154, 2014.

F. Viénot, H. Brettel, and J. D. Mollon. Digital video colourmaps for checking the legibility of displays by dichromats. *Color Research & Application*, 24(4): 243–252, 1999.

U. von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17 (4):395–416, 2007.

C. Wang and S. Mahadevan. Manifold alignment using Procrustes analysis. In *Proc. ICML*, 2008.

L. Wang and A. Singer. Exact and stable recovery of rotations for robust synchronization. *Information and Inference*, 2013.

Y. Wang, Y. Wotao, and Z. Jinshan. Global convergence of ADMM in nonconvex nonsmooth optimization. *arXiv:1511.06324*, 2015.

M. Wardetzky, S. Mathur, F. Kälberer, and E. Grinspun. Discrete Laplace operators: no free lunch. In *Proc. SGP*, 2008.

Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In *Proc. NIPS*, 2008.

Z. Wen and W. Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434, 2013.

J. Weston, S. Bengio, and N. Usunier. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine Learning*, 81(1):21–35, 2010.

A. Wetzler and R. Kimmel. Efficient Beltrami flow in patch-space. In *Proc. SSVM*. 2012.

A. Yeredor. Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation. *IEEE Transactions on Signal Processing*, 50(7):1545 –1553, 2002.

M. Zhang and P. T. Fletcher. Probabilistic principal geodesic analysis. In *Proc. NIPS*, 2013.

X. Zhang, T. Sim, and X. Miao. Enhancing photographs with near infrared images. In *Proc. CVPR*, 2008.

Y. Zhao and Z. Tamimi. Spectral image decolorization. In *Proc. ISVC*, 2010.

B. Zhou and J. Feng. Gradient domain salience-preserving color-to-gray conversion. In *Proc. SA*, 2012.

K. Zhou, J. Huang, J. Snyder, X. Liu, H. Bao, B. Guo, and H.-Y. Shum. Large mesh deformation using the volumetric graph Laplacian. *ACM Transactions on Graphics*, 24(3):496–503, 2005.

F. Zhu, S. Li, and G. Wang. Example-based materials in Laplace-Beltrami shape space. *Computer Graphics Forum*, 34(1):36–46, 2015.

A. Ziehe. *Blind Source Separation based on Joint Diagonalization of Matrices with Applications in Biomedical Signal Processing*. PhD thesis, Universität Potsdam, 2005.

Index