Calculating Sparse and Dense Correspondences for Near-Isometric Shapes

Dissertation

zur

Erlangung des Doktorgrades (Dr. rer. nat.)

der

Mathematisch-Naturwissenschaftlichen Fakultät

der

Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von

Dipl.-Inf. Oliver Burghard

aus

Starnberg

Bonn, Oktober 2017

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen
Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter:	Prof. Dr. Reinhard Klein
2. Gutachter:	Prof. Dr. Michael Wand

Tag der Promotion:	15. Mai 2018
Erscheinungsjahr:	2018

Dedicated to my big love, my little one,
mother, father, brother and sister.

"I returned, and saw under the sun, that the race is not to the swift, nor the battle to the strong, neither yet bread to the wise, nor yet riches to men of understanding, nor yet favor to men of skill; but time and chance happens to them all."

(Ecclesiastes 9:2)

Contents

Abstract

The comparison of digital models and their analysis are a basic technique of geometric shape processing with a variety of applications. For example the domain knowledge contained in the growing number of easily obtainable digital models can be used to simplify reoccurring tasks such as modelling of shapes. Furthermore, the increasingly sophisticated methods to digitize physical objects transfers any progress in the analysis of digital models onto the analysis of real objects, which has a variety of applications, such as medical examinations, medical and agricultural research and infrastructure maintenance.

While global shape properties, like volume and surface area, are simple to compare they contain only limited information. Therefore one is often interested in *local* differences, for example where and how a plant grew between successive scans, which contain much more information. Sadly computation of local differences is much more involved, as it requires knowledge of which points on the shapes correspond to each other. A possible representation of this knowledge is the correspondence map, which maps point of one shape onto another. The following article thesis (cumulative dissertation) discusses several recent publications of the author for the creation of correspondence maps. The following topics are discussed:

- Geodesic distances between points, i.e. distances along the surface, are fundamental for several shape processing tasks as well as several shape matching techniques. Chapter 3 introduces and analyses fast but accurate bounds for geodesic distances of arbitrary points.
- When building a shape space on a set of shapes, misaligned correspondences lead to points moving along the surfaces and finally to a larger shape space. Chapter 4 shows that this also works the other way around. That is good correspondences are obtain by optimizing them to generate a compact shape space.
- Representing a correspondence map with a “functional map” has a variety of advantages. Chapter 5 shows that representing the correspondence map as an alignment of Green’s functions of the Laplace operator has similar advantages, but is much less dependent on the number of eigenvectors used for the computations.

- Quadratic assignment problems were recently shown to reliably yield sparse correspondences, while not depending on any additional information than the shapes. Chapter 6 compares state-of-the-art convex relaxations of graphics and vision with methods from discrete optimization on typical quadratic assignment problems emerging in shape matching.

Zusammenfassung

Das Vergleichen von digitalen Modellen und deren Analyse sind grundlegende Techniken der Computer Grafik mit vielfältigen Anwendungen. Zum Beispiel sind Informationen aus der Analyse der immer vielzähligeren, leicht erhältlichen, digitalen Modelle nutzbar um wiederkehrende Aufgaben wie das Modellieren zu vereinfachen. Darüber hinaus transferieren immer ausgefeiltere Methoden zur Digitalisierung physikalischer Objekte jeden Fortschritt bei der Analyse digitaler Modelle in einen Fortschritt bei der Analyse realer Objekte, welche eine Vielzahl von Anwendungen hat, wie zum Beispiel medizinische Untersuchungen, medizinischen und agrikulturelle Forschung und die Aufrechterhaltung von Infrastruktur.

Globale Eigenschaften, wie Volumen und Oberfläche, können leicht verglichen werden, enthalten aber auch nur beschränkte Information. Viel öfter ist man deshalb an *lokalen* Unterschieden interessiert, wie etwa an welcher Stelle und auf welche Art eine Pflanze zwischen zwei Vermessungen wuchs. Lokale Unterschiede enthalten sehr viel mehr Information, sind jedoch auch sehr viel schwieriger zu Berechnen. Und zwar benötigt die Berechnung lokaler Unterschiede das Wissen, welche Punkte der verschiedenen Formen einander entsprechen. Eine mögliche Darstellung dieses Wissens ist die Korrespondenzabbildung, die jeden Punkt der einen Form auf die andere abbildet. Die vorliegende kumulative Dissertation beschreibt mehrere Veröffentlichungen des Autors zur Erstellung von Korrespondenzabbildungen. Die folgenden Themen werden diskutiert:

- Geodätische Abstände zwischen Punkten, d.h. die Abstände entlang der Oberfläche, sind grundlegend zur Formenverarbeitung als auch für einige Techniken zur Korrespondenzenbestimmung zwischen zwei Flächen. Kapitel 3 stellt eine schnelle und trotzdem genaue Approximation geodätischer Abstände vor und analysiert diese.

- Beim lernen eines Formenraumes führen falsche Korrespondenzen zu einem wandern der Punkte entlang der Oberfläche und schließlich zu einem größen Formenraum. Kapitel 4 zeigt, dass das auch umgekehrt funktioniert. Das man also gute Korrespondenzen erhält, wenn man sie darauf optimiert einen möglichst kompakten Formenraum zu erzeugen.

- Die Darstellung der Korrespondenzabbildung als eine „funktionale Abbildung“ hat verschiedene Vorteile. Kapitel 5 zeigt, dass die Darstellung als Abbildung der Green's Funktionen ähnliche Vorteile bringt, dass dabei die Berechnung jedoch viel weniger von der Anzahl der Eigenvektoren der Laplace Basis abhängt.

- Kürzlich wurde gezeigt dass quadratische Zuweisungsprobleme zuverlässig „dünne“ Korrespondenzen erzeugen können ohne dabei auf Zusatzinformationen angewiesen zu sein. Kapitel 6 vergleicht aktuelle konvex Relaxierungen des quadratischen Zuweisungsproblems der Computer Grafik und des Maschinellen Sehen mit etablierten Relaxierungen der Diskreten Optimierung. Dabei werden insbesondere quadratischen Zuweisungsprobleme untersucht, wie sie bei der Korrespondenzberechnung entstehen.

Notations

Symbol	Explanation
$\mathbb{R}$	real numbers
$\mathbb{R}_0^+/\mathbb{R}^+$	non-negative/positive real numbers
$\mathbb{N}$	natural numbers
$\partial_u f(u)$	partial derivative of the function $f(u)$ by u
$f\big\|_V$	restriction of a function $f\colon U \to W$ onto a subset $V \subset U$
$\operatorname{image}(f)$	image of the map f
min, max	minimum/maximum
arg min, arg max	argument of minimum/maximum
$\partial\mathcal{U}$	boundary of the set $\mathcal{U}$
$\operatorname{int}(\mathcal{U})\colon = \mathcal{U}\ \partial\mathcal{U}$	interior of the set $\mathcal{U}$
$\mathbf{a}, \mathbf{b}, \mathbf{c}, \cdots \in \mathbb{R}^n$	vectors
$\mathbf{a} < \mathbf{b}, \mathbf{a} \leq \mathbf{b}$	component-wise less, less or equal
$\mathbf{a}^T$	transposed vector
$\langle \mathbf{a}, \mathbf{b} \rangle := \mathbf{a}^T\mathbf{b}$	inner-product on vectors
$\mathbf{a} \times \mathbf{b}$	cross-product for $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$
$\\|\mathbf{a}\\|_1, \\|\mathbf{a}\\|_2$	one/two norm
$\mathbf{0}/\mathbf{1} \in \mathbb{R}^n$	vector of zeros/ones of suitable dimension
$\mathbf{A}, \mathbf{B}, \mathbf{C}, \cdots \in \mathbb{R}^{n \times m}$	matrices
$\mathbf{A}^T$	transposed matrix
Id	identity matrix
$\operatorname{rank}(\mathbf{A})$	rank of $\mathbf{A}$
$\operatorname{tr}(\mathbf{A})$	trace of $\mathbf{A}$
$\\|\mathbf{A}\\|_F$	Frobenius norm
$\langle \mathbf{A}, \mathbf{B} \rangle := \operatorname{tr}(\mathbf{A}^T\mathbf{B})$	inner-product on matrices
$\mathcal{M}, \mathcal{N}$	shapes, i.e. smooth, compact manifolds embedded in $\mathbb{R}^3$

Symbol	**Explanation**
$\mathcal{L}^2(\mathcal{M})$	Hilbert space of square integrable, real-valued functions on $\mathcal{M}$
$\operatorname{grad} f$	Gradient of $f \in \mathcal{L}^2(\mathcal{M})$
$\operatorname{div} X$	Divergence of vector field X
$\Delta\colon \mathcal{L}^2(\mathcal{M}) \to \mathcal{L}^2(\mathcal{M})$	Laplace-Beltrami operator
$\langle f, g \rangle$	scalar product on $\mathcal{L}^2(\mathcal{M})$ ($f, g \in \mathcal{L}^2(\mathcal{M})$)
$\langle \Delta f, g \rangle$	Dirichlet energy ($f, g \in \mathcal{L}^2(\mathcal{M})$)
δ_p	delta-distribution
$\phi_1, \phi_2, \cdots \in \mathcal{L}^2(\mathcal{M})$	eigenvectors of Δ
$\lambda_1, \lambda_2, \cdots \in \mathbb{R}$	eigenvalues of Δ
$\mathcal{S}^n$	real, symmetric $n \times n$ matrices
$\mathcal{S}^n_+$	real, positive semidefinite, symmetric $n \times n$ matrices
$\mathbf{A} \succcurlyeq \mathbf{B} :\iff \mathbf{A} - \mathbf{B} \in \mathcal{S}^n_+$	Loewner order for $\mathbf{A}, \mathbf{B} \in \mathcal{S}^n$
S_n	Symmetric group of n-element permutations
$[\mathbf{A}] \in \mathbb{R}^{n^2}$	Row-wise unrolling of matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$
$\mathbf{Y}_{pq,rs}$	short for $\mathbf{Y}_{(p-1)\cdot n+q,(r-1)\cdot n+s}$ where $\mathbf{Y} \in \mathbb{R}^{nm \times nm}$ (1-based indices)

Acknowledgments

I am thankful to various people, without whom this thesis would not have been possible. In particular, I thank Prof. Dr. Reinhard Klein for the opportunity to be part of his research group and for his continuous support. Furthermore, I thank Prof. Dr. Michael Wand — also for his continuous support, but as much for challenging and advancing my understanding during various discussions. Finally, I thank my fiancée for her understanding.

1 Introduction

Today digital shapes are more widespread than they used to be and their prevalence is likely to grow further as applications of digital shapes mature. There is a growing availability of low cost acquisition hardware, such as the Microsoft Kinect as well as mobile phone accessories [EOR17; MF17]. On the other hand, there is a growing need of digital shapes from the maturing of 3d-printing, augmented reality and virtual reality (e.g. Oculus Rift, Microsoft HoloLense) as well as from the digitalization of whole industries, such as the film industry. A growing prevalence of digital shapes poses several opportunities for research.

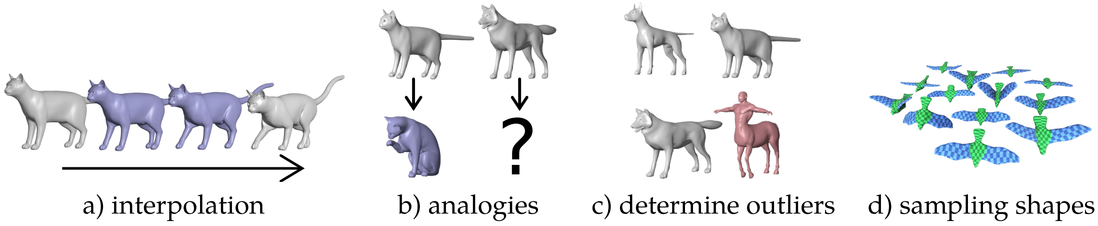
For example, there is a growing need for digital shapes, but modelling shapes still is more cumbersome than it has to be. Instead of taking the prior knowledge of existing and steadily growing shape databases into account, modelling is too often seen as an isolated activity. Why can't the computer *learn* from the existing shapes how to support the designers?

As 3d scanning technologies matured and the creation of digital representations of physical objects becomes more affordable and feasible physical objects are another source of digital shapes. Automating the *analysis* of real world objects through their digital representation will have a large influence in many practical applications. For example, medical devices could report conspicuous organs and visualize where and how they differ from a healthy one. Or biologists can grow plants of differing genomes and under different environmental conditions to determine the effect of the genome and the environment on growth. The large effort required to analyze digital shapes is currently the major obstacle for larger setups.

What all those examples so far have in common is that once we know which regions on the shapes correspond the tasks become much simpler. Figure 1.1 depicts some illustrative, simple problems, which fall into this category as well:

- Interpolate between two or more shapes preserving some structure, such as intrinsic distances (a).
- Transfer of deformations between shapes, i.e. solve the shape analogy (b):

$$\text{cat1} \leftrightarrow \text{cat2} \qquad \sim \qquad \text{dog1} \leftrightarrow ?$$



a) interpolation b) analogies c) determine outliers d) sampling shapes

Figure 1.1.: Several example applications of dense correspondences.

- Determine outliers in a set of shapes, i.e. the red shape in (c) is different from the other shapes (c).
- Generalize from the shapes of a shape ensemble to randomly create similar ones (d).

All these examples have in common that they can be solved once the points on the different surfaces have been brought in relationship with each other. For each point on each shape we have to know if and where it is located on the other shapes. Mathematically this idea is expressed by the *correspondence* map, which maps points from one shape onto another. A good, global correspondence map is characterized by the following conditions:

- semantically meaningful regions are assigned to each other
- the neighborhood structure is preserved, i.e. the map is continuous
- the map is bijective, i.e. it is well-defined for each source points and each target point is mapped to
- the map exhibits little stretch, which can be defined in terms of changes induced on the metric

The correspondence map can be discretized into point-to-point correspondences, i.e. tuples of corresponding points on the source and target shapes. *Sparse correspondences* assign only a few key source points onto to target and are often a first step for the calculation of *dense correspondences*, which correlate a dense set of points on the source shape. Figure 1.2 shows examples of sparse and dense correspondence.

While humans are especially good finding correspondences, their calculation is harder than it might at first appear. Firstly computers lack the semantic knowledge of humans to compute sparse correspondences, secondly calculation of bijective maps between surfaces is known to be difficult from research on bijective parametrizations - especially if a common measures of stretch is to be minimized.

The past decade brought significant progress to isometric shape matching, so that today isometric and almost isometric shapes, i.e. shapes in different poses such as in

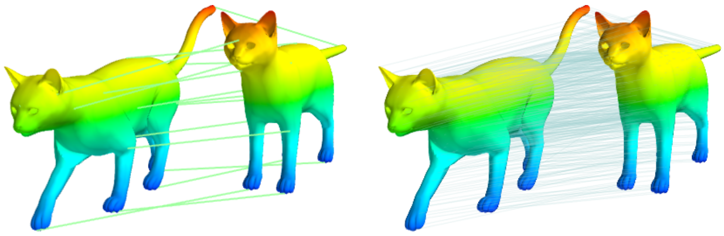


Figure 1.2.: Examples of sparse (left) and dense (right) correspondences.

Figure 1.1a and Figure 1.2, can be reliably matched[Ovs+12; Kov+15; Mar+16a] and that even partial matching[Rod+15; NFS15] is possible. This can not be said of near-isometric shapes, i.e. shapes where the metric allows a global alignment but is far from isometric such as in Figure 1.1b-d, where the computation of correspondences is much more difficult and progress was naturally slower. This is unfortunate as applications on near-isometric shapes would be especially interesting. In the following article thesis (cumulative dissertation) I present my findings to advance the state-of-the-art in near-isometric shape matching.

1.1. Publications

This article thesis (cumulative dissertation) contains the following first author publications:

- Oliver Burghard et al. "Compact Part-Based Shape Spaces for Dense Correspondences". In: *arXiv:1311.7535* (2013)
- Oliver Burghard and Reinhard Klein. "Simple, Robust, Constant-Time Bounds on Surface Geodesic Distances using Point Landmarks". In: *Vision, Modeling & Visualization*. 2015
- Oliver Burghard, Alexander Dieckmann, and Reinhard Klein. "Embedding shapes with Green's functions for global shape matching". In: *Computers & Graphics* (2017)
- Oliver Burghard and Reinhard Klein. "Efficient Lifted Relaxations of the Quadratic Assignment Problem". In: *Vision, Modeling & Visualization*. 2017

While working on the thesis the following second author publications were created:

- Alexander Berner et al. *A Morphable Part Model for Shape Manipulation*. eng. Research Report MPI-I-2011-4-005. Saarbrücken: Max-Planck-Institut für Informatik, 2011

- Z Lähner et al. "SHREC'16: Matching of deformable shapes with topological noise". In: *Proc. 3DOR* 2.8 (2016), p. 11

1.2. Contributions and outline

This introduction motivates the need for correspondences, presents the contributions of the covered publications and relates them to the previous methods. Chapter 2 presents the theoretical foundations of shape matching and covers extrinsic and intrinsic shape alignment, functional maps and sparse correspondence generation with assignment problems. The next chapters present the contributions of the thesis:

- Geodesic distances on surfaces are an important tool in geometric shape processing used for example to prune correspondences [Hua+08a], formulate linear and quadratic assignment problems [Ves+17], average points via geodesic Karcher means[Pan+13] and cluster point with geodesic k-means[KSC07; FA07; AM08].

  In chapter 3 we introduce and analyze novel lower and upper bounds on geodesic distances. Derived from the triangle inequality they are simple to implement, quick to compute, continuous and have a small absolute error. The lower bound has a bounded relative error as well.

- The principle of minimum description length was successfully used to optimize correspondences in shape ensembles [KT98; Dav+02b].

  In chapter 4 we apply this principle to the morphable-part model of Berner et al. [Ber+11]. Our approach allows matching shapes of different topology and has much fewer artifacts from part-wise rotations. Furthermore, our part-wise models and the loose coupling of the part shape spaces generalizes better from few examples than holistic models. We further add a bi-Laplacian regularizer [Yeh+11] to correct for the sampling bias caused by the entropy minimization. Minimizing a twice differentiable cost function with a quasi-Newton method allows us to compute much larger models than the previous methods. Given sufficient prior correspondences our evaluation shows that the method delivers high quality correspondences especially well suited for building shape spaces.

- Solving for a point-wise functional map is equivalent to the computation of an alignment of *dual*-delta-distributions as noted in chapter 5 and section 2.4.3. Such an alignment requires L_2 distances on (dual-)delta-distributions, which are not well-defined. Previous works circumvents this problems by projecting (dual-)delta-distributions onto the first k eigenvectors of the Laplace operator, after which distances are defined, but strongly depend on k.

Chapter 5 explores the representation of point-wise maps as an alignment of Green's functions of the Laplace operator instead. Such an alignment is well-defined and it's calculation over the first k Laplace eigenvectors converges quickly as k grows. The Green's alignment and the pullback functional map are connected by a linear relation, which allows us to transfer functional constraints and operator commutativity into our setting and to use the Green's alignment as a drop-in replacement of the pullback functional map.

Furthermore, we show how to include conformality into the functional setting by additional functional constraints. Finally, we observe that Green's functions change little under near-isometric, conformal maps, which can exploited by future work to construct conformal maps between shapes of arbitrary topology.

- Recently there is growing usage of quadratic assignment problems (QAP) to compute sparse correspondences between shapes, which they compute *without* any prior information apart from the shapes. While posing a QAP is often simple, solving it is NP-hard. Literature describes several convex relaxations to approximate the solutions in polynomial time. Among the tightest known are convex relaxations by semidefinite programs over the lifted permutations.

  Chapter 6 compares the feasible sets of several convex relaxations and shows that the semi-definite relaxation of Zhao [Zha+98; PR09] is at least as accurate as the ones recently used in graphics and vision. Interestingly dropping the semi-definite constraint from the relaxation of Zhao results in a linear program[FY83; AJ94], which provides slightly weaker bounds but is significant faster to solve for almost all instances of up to fifteen points. This is despite already approximating the semi-definite programs with the recent method from Wang [Wan+16], which often is a magnitude faster than the MOSEK [Mos10] state-of-the-art interior-point solver.

  Furthermore, we show that typical shape matching problems result in notably "simple" QAP instances, in so far as *all* examined relaxations find the optimal solution of all our shape matching instances. We also extend our results to quadratic assignment matching (QAM), which is a generalization of the QAP to partial permutations, by reducing any QAM into a QAP. Finally we show that the product manifold filter [Ves+17] can be seen as a heuristic to solve a shape matching QAP. This insight allows the automatic computation of correspondences of several hundred points.

Chapter 7 presents a conclusion and an outlook onto future research directions. Finally, appendix A present further results for the chapters 3 and 4 not contained in the original publications.

1.3. Relation of our publications to previous results

Geodesic distances On triangle meshes exact geodesic distances from a single point can be computed as on any triangle the squared distances are the minimum of a finite number of quadratic functions [MMP87; CH90; XW09; Xu+15]. On a triangle mesh with n triangles the minimal computational complexity is $\mathcal{O}(n^2)$. The early methods have either optimal average or optimal worst case complexity, while the method of Xu [Xu+15] has both.

Several methods trade speed with precision. Fast marching methods [KS98] exploit the Eikonal equation to approximate a geodesic distance field in $\mathcal{O}(n \log n)$ operations, while Crane et al. [CWW13] approximate it in $\mathcal{O}(n)$ operations (excluding a one-time matrix factorization) by predicting the field's gradients from heat-diffusion.

Often applications do not require entire distances fields but only point-to-point distances, which can be approximated even faster. For example point-to-point distances are sufficient to prune correspondences [Hua+08a] in non-rigid shape matching, to formulate linear and quadratic assignment problems [Ves+17], to average points via geodesic Karcher means[Pan+13] and to cluster points with geodesic k-means[KSC07; FA07; AM08].

In chapter 3 we present continuous constant time bounds on geodesic distances. Xin et al. [XYH12] approximate point-to-point in constant time as well. They precompute equally spaced landmark points, their pair-wise distances as well as a coarse intrinsic Delaunay triangulation. They can predict distances from any point within a triangle to any landmark point by projecting the quadrilateral of a coarse triangle and another point into the plane. A second step generalizes this approximation to arbitrary points. On the edges of the coarse triangulation their distances are not continuous in comparison to our method. Their distances are not a proper distance metric and they give no bounds on their approximation's error. Aflalo et al. [AK13] compute a Euclidean embedding, such that Euclidean distances approximate geodesic distances. First they derive a compact representation of the all-pair distances from a set of precomputed distance fields and a smoothness prior. Then they create the embedding by generalizing multidimensional scaling to their compact representation. A more detailed comparison of their method with ours is in appendix A.1.

Extrinsic and intrinsic shape alignment Iterative closest points (ICP) method alternates between determining corresponding points between aligned point sets and aligning those point sets with the correspondences. First introduced to align rigid shapes [RL01b], it was soon generalized to deformable shapes by replacing the rigid alignment with non-rigid deformations [ACP03a; SP04; ARV07; Hua+08a;

Yeh+11; Bur+13b; Yos+14]. Such an extrinsic shape alignment with a bi-Laplacian regularizer [Yeh+11] is used by our optimization in chapter 4.

Deformable ICP methods can be further adopted to the matching of articulated shapes by first embedding shapes such that Euclidean distances approximate intrinsic distances [JZ06; SY12; AK13; BGB14; RMC15]. Nevertheless, the above ICP methods are local optimizations of the non-convexity ICP cost function and find a *local* minimum only. They therefore depend on the initial, predefined correspondences and on heuristics to prune correspondences.

The functional map framework [Ovs+12] presented in section 2.4 relaxes the pointwise maps to the vector space of functional maps, so that solving for a functional map from functional constraints becomes a *convex* problem. This successful approach initiated a series of publications, for computing functional maps on shapes or shape ensembles[Ngu+11b; COC15; RMC15; Kov+13; Kov+15; Ngu+11b; HWG14].

While functional maps help to infer some point-wise correspondences computing a functional map equivalent to a point-wise map is equivalent to an alignment of delta-distributions and can be solved in an ICP like manner. In contrast to the alignment methods above, in the functional maps setting articulated shapes can be aligned with a *linear* alignment, which is even *orthogonal* for area preserving maps. On the downside distances on delta-distributions are defined only after projection into a finite dimensional subspace. In chapter 5 we therefore explore replacing delta-distributions by the Green's functions of the Laplace operator, whose L_2 distances are the well-defined biharmonic distances[LRF10b]. Our alignment of Green's functions can substitute functional maps in most applications as one can be expressed as an affine function of the other. We are the first to explore this alignment which also allows aligning articulated shapes with a linear function, while being well-defined.

Furthermore, the relationship between Green's functions and conformal maps explored in chapter 5 is based on the preservation of the Dirichlet energy by conformal maps (section 2.2, [Rus+13a]).

Parameterization methods Another line of research parameterizes both shapes onto a common domain to build the correspondence map [KS04b; APL14; APL15; AL15; AL16; Kur+13; Kur+12; Lag+16]. For completeness, we summarize these methods although our contributions do not depend on them nor are these techniques required to understanding our contributions. A major advantage of these methods is the computation of a bijective correspondence map simply by ensuring that both parametrizations are bijective. Usually the parameterization is initially guided by predefined point-to-point constraints and then optimized to minimize a stretch functional from the source to the target shape. The non-convex stretch functional is difficult to minimize so that some methods substitute it with a convex cost function. In any case the final map strongly depends on the initial correspondences and is not assured to be a global minimum of the stretch functional.

Uniformization allows efficient computation and sampling of conformal maps, which can locally be a good approximation of stretch minimizing maps. Sampling a multitude of conformal maps, was used to determine point-to-point correspondences [LF09b] and to construct a global map by a blending of local maps [KLF11a].

Ensemble optimization On a shape ensemble, which is a set of several similar shapes, the registration task is to construct all pairwise correspondence maps, as for example required for statistical shape processing [BV99a; ACP03b; Has+09b]. Good correspondences lead to shape spaces that generalize, i.e. produce other likely shapes. Interestingly this works the other way around as well. Recreating the data with a compact model is a machine learning heuristic [Ris78] to overcome the bias-variance dilemma and to create models that generalize. And indeed, good correspondences are obtain by optimizing them to generate a compact shape space [HT94].

In computer vision and medical imaging compact multivariate normal distributions of point distributions were used for this purpose [Coo+95; KT98; Dav+02b; EÅ03; TO03; Hei+05]. For manifolds with sphere topology these methods were combined with a spherical parameterization [Dav+02a; Hei+05; Dav+10], which additionally guarantees bijectivity. Cates et al. [Cat+06] extended the approach to regularly sampled manifolds, removing topological restrictions and assuring a uniform sampling by an elegant complementary entropy term.

Cycle consistency [Ngu+11a; Hua+12; Kim+12; Kez+15a] is another principle to optimize correspondences in shape ensembles. Maps whose source and target shapes match can be concatenated. Concatenating maps along a closed path, i.e. a path whose first and last shapes match, must result in the identity map.

In chapter 4 we introduce an ensemble optimization based on the entropy-based approach of Kotcheff and Taylor [KT98]. While other methods use a single linear shape space we model the shapes with the morphable-part model of Berner et al.[Ber+11]. Our approach allows matching shapes of different topology, has much fewer artifacts from part-wise rotations and generalizes from fewer examples. We use a bi-Laplacian regularizer [Yeh+11] to avoid the sampling bias caused by the entropy minimization and our quasi-Newton method allows to process much larger models.

Assignment problems Linear assignment problems are one of the fundamental combinatorial optimization problems and are used in shape matching to find one-to-one correspondences between aligned point sets[BMP02; Ves+17]. Linear assignment problems are well understood and the Hungarian algorithm[PS13] solves an instance of n points in $\mathcal{O}(n^3)$ operations.

Quadratic assignment problems are another important discrete optimization problem that have been used to match features points in images and on shapes [LH05; BBM05; FS06; SRS07; ADK13; Kez+15b]. While posing a quadratic assignment problem is simple, approximating quadratic assignment problems to any precision is NP-hard[SG76] and solving instances with as little as thirty points is typically not considered practical. Since their introduction in 1957 by Koopmans[KB57] there was much research on quadratic assignment problems well summarized in several surveys[Bur+98; Loi+07; Cel13].

For example, given enough time, branch and bound methods[Gil62; Law63; Ans03] traverse the solution space and prune subspaces in which the cost is larger than the cost of the best currently known solution. One of the earliest methods to estimate the minimal cost of a subspace is the Gilmore-Lawler bound[Gil62; Law63], which is still used due to its fast computation. Due to the NP-hardness many methods approximate quadratic assignment problems in polynomial time, such as convex relaxations of the cost function, which estimate lower bounds of the cost function as well as a solution. Various relaxations were proposed, such as spectral relaxations[LH05; ADK13], relaxations as linear programs[HG98; Kar+99], relaxations as mixed linear integer programs [KB78; FY83; AJ94], relaxations as quadratic constraint quadratic programs[Luo+10] and relaxations as semidefinite programs [GW95; LS91; Kar95; Zha+98; PR09]. The relaxations as semidefinite programmings over the lifted permutation matrices using $\mathcal{O}(n^4)$ variables are known for their tightness.

Recently there is also growing interest in convex relaxations with copositive programs[PVZ15; Bur12; PR09; BMP12]. A matrix $\mathbf{A}$ is copositive iff $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$ for all $\mathbf{x} \geq 0$. A copositive programs minimizes a linear objective under linear constraints over the convex set of copositive matrices. Solving copositive programs is itself NP-hard, but copositive programs can be approximated by semidefinite programs up to any accuracy.

The variety of relaxations used in graphics, vision and discrete optimization is the motivation of chapter 6 to compare the established convex relaxations of discrete optimization[AJ94; Zha+98; PR09] to the tightest relaxations used in graphics and vision [Kez+15b; Wan+16].

It additionally evaluates the approximation of fixed trace semidefinite programs by a quasi-Newton minimization of the Lagrange dual[Wan+16], which often is an order of magnitude faster than current interior-point solvers. By reducing quadratic assignment matching, which generalizes the quadratic assignment problem to partial permutations, to a quadratic assignment problem the results are transferred onto quadratic assignment matching[SRS07; Kez+15b].

The product manifold filter[Ves+17] is a recent, very successful, iterative technique to improve correspondences by solving linear assignment problems. Chapter 6 shows that the product manifold filter can be seen as a heuristic to minimize a quadratic assignment problem.

2 Foundations

There is a large body of relevant research for the important problem of shape matching. In the following we introduce a theoretical framework for our work together with a review of relevant prior works.

2.1. Functional analysis on meshes

Unless stated otherwise we assume that $\mathcal{M}$ is a compact, connected, differential 2-manifold embedded into the $\mathbb{R}^3$ with a differential function $\mathbf{p}\colon \mathcal{M} \to \mathbb{R}^3$. Let f be a real-valued functions f on $\mathcal{M}$, i.e. a function which maps $\mathcal{M}$ onto $\mathbb{R}$, and $\phi\colon (u,v) \in U \mapsto \mathcal{M}$ be a local parameterization over an open set $U \subset \mathbb{R}^3$. f can be integrated on U: $\int_U f(u,v)\ \det(\partial_u \mathbf{p}, \partial_v \mathbf{p})\ dudv$. The area form $dp = \det(\partial_u \mathbf{p}, \partial_v \mathbf{p})\ dudv$ is independent of the local parameterization and thus defines an integral over $\mathcal{M}$.

Real-valued functions with a finite square integral are called square integrable: $\mathcal{L}^2(\mathcal{M}) := \{f\colon \mathcal{M} \to \mathbb{R} \mid \int_{\mathcal{M}} f^2(p)\ dp < \infty\}$. The area form dp defines a scalar product and norm on $\mathcal{L}^2(\mathcal{M})$:

$$\langle f, g\rangle := \int_{\mathcal{M}} f(p) g(p)\ dp \qquad \|f\|_2 = \sqrt{\langle f, f\rangle} \tag{2.1}$$

On $\mathcal{L}^2(\mathcal{M})$ both, $\langle f, g\rangle$ and $\|f\|_2$, are finite due to the definition of $\mathcal{L}^2$ and the Cauchy-Schwarz inequality.

The directions in which the surface can be traversed from a fixed point p are called *Tangent vectors*. Riemannian geometry usually represents tangent vectors by derivative functionals. For simplicity, we represent tangent vectors by the derivation of the embedding function $\mathbf{p}$, i.e. by tangent vectors on the surface in the classical sense. This choice simplifies the exposition at the cost of representing the intrinsic tangent vectors dependent on the embedding $\mathbf{p}$[Car16; FC13].

Any differentiable curve $x\colon (-\epsilon, \epsilon) \subset \mathbb{R} \to \mathcal{M}$ can be extended into the embedding by $\mathbf{p} \circ x$. The tangent vectors $[\partial_t (\mathbf{p} \circ x)]_{t=0}$ of these extended curves are tangent to the embedding of the surface and are called the *tangent vectors* of the surface. For every tangent vector $X \in \mathbb{R}^3$ at the point p there is a curve x as above with

$$X = [\partial_t(\mathbf{p} \circ x)]_{t=0} \text{ with } x(0) = p \tag{2.2}$$

and every such curve induces a tangent vector. Every curve represents exactly one tangent vector, but a single tangent vector is represented by several curves. The tangent vectors at a point $p \in \mathcal{M}$ are orthogonal to the surface normal $n_p \in \mathbb{R}^3$ at p and span a two-dimensional subspace of the $\mathbb{R}^3$ called the *tangent space* $T_p\mathcal{M}$. With a slight abuse of notation we write $x \in T_p\mathcal{M}$ as well for a differential curve x, which represents a tangent vector in $T_p\mathcal{M}$. On a non-degenerate triangle embedded with a linear map, the embedded edges are tangent vectors spanning the tangent space.

The *gradient* $(\nabla f)_p \in T_p\mathcal{M}$ at a point $p \in \mathcal{M}$ of a real-valued function f is the tangent vector defined by:

$$\left\langle (\nabla f)_p, [\partial_t(\mathbf{p} \circ x)]_{t=0} \right\rangle = [\partial_t(f \circ x)]_{t=0} \quad \forall x \in T_p\mathcal{M} \tag{2.3}$$

on *vector fields* $V, W \colon \mathcal{M} \to \mathbb{R}^3$, which assign every point p a vector, like the gradient of a function above, we define the scalar product:

$$\langle V, W \rangle := \int_{\mathcal{M}} V(p)^T W(p) dp \tag{2.4}$$

2.1.1. Smoothness

Many methods in geometry processing use the concept of smoothness. Its usage is pervasive and often concealed in implicit formulations. One possible definition of a smooth function is one with small variations. In other words its gradients are small as measured with the *Dirichlet energy*:

$$E_{\text{Dirichlet}}[f] := \frac{1}{2} \int_{\mathcal{M}} \|(\nabla f)(p)\|_2^2 \, dp = \frac{1}{2} \langle \nabla f, \nabla f \rangle \tag{2.5}$$

Smooth functions are characterized as minima of the Dirichlet energy, which is a quadratic, positive semi-definite functional that is minimal on a linear subspace of $\mathcal{L}^2(\mathcal{M})$. When solving for the function values in an open subset $U \subset \mathcal{M}$ the solution can be further restricted by boundary conditions such as:

- Dirichlet boundary conditions: $f(x) = f_0(x)$
- Neumann boundary conditions: $(\partial f / \partial v_x)(x) = g_0(x)$ where v_x is the outwards pointing normal vector of the boundary
- Natural boundary conditions: Neumann boundary conditions with $g_0 = 0$

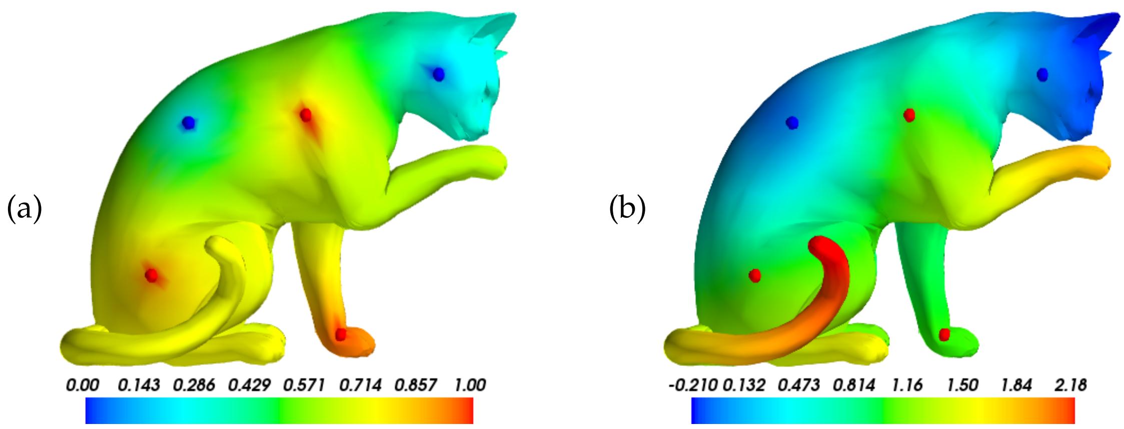


Figure 2.1.: Harmonic (a) and biharmonic (b) scalar field interpolation.

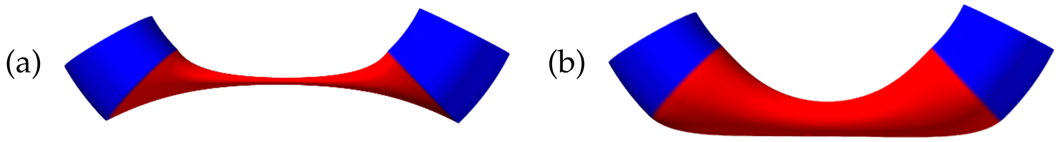


Figure 2.2.: Mesh completion with minimal area (a) and thin-plate energy (b).

Different sections of the boundary of U can have different boundary constraints and the boundary of U does not have to be the boundary of $\mathcal{M}$. For example interpolation of function values at discrete points can be modeled by Dirichlet boundary conditions on the points and natural boundary conditions elsewhere.

The extremal functions of the Dirichlet energy on $U \subset \mathcal{M}$ with fixed boundary conditions are called *harmonic functions*. Figure 2.1a shows smooth interpolation of five discrete function values, depicted with small balls, with a harmonic function. Their smoothness is confirmed by the following maximum principle:

Proposition 1 (Maximum Principle). *Within a compact, connected, open subset $V \subset U$ a harmonic function $f\colon U \to \mathbb{R}$ has no local extrema unless it is constant.*

Harmonic functions can approximate minimal surfaces. Parameterized with local orthogonal, coordinates $(u, v) \in U \subset \mathbb{R}^2 \to \mathcal{M}$ the embedding $\mathbf{p}\colon U \to \mathbb{R}^3$ has the area:

$$\int \|\partial_u p \times \partial_v p\| \;\; dudv \tag{2.6}$$

Minimization of the non-convex functional is difficult, also for its invariance to reparameterization. Pinkall [PP93] built an algorithm on the idea to minimize the

Dirichlet energy $\langle \Delta \mathbf{p}, \mathbf{p} \rangle$ instead:

$$\langle \Delta p, p \rangle = \int \|\partial_u p\|^2 \, \|\partial_v p\|^2 \;\; du\, dv \leq \int \|\partial_u p \times \partial_v p\|^2 \;\; du\, dv =: R(p)\ . \tag{2.7}$$

The inequality becomes an equality for a conformal, that is angle preserving, parameterization. Therefore, minimizing the Dirichlet energy yields an embedding minimizing the area and in which the parameterization is a conformal one. Figure 2.2a shows such the result of minimizing the Dirichlet energy in the blue area while fixing the read area with Dirichlet boundary constraints.

2.1.2. Laplace-Beltrami operator

A function is harmonic if and only if it solves the Euler-Lagrange equation of the Dirichlet energy, which we derive in the following using variational calculus. From the divergence theorem follows:

$$\int_{\mathcal{U}} g \operatorname{div} \nabla f + (\nabla f)^T (\nabla g)\ dp = \int_{\mathcal{U}} \operatorname{div}(g \nabla f)\ dp = \oint_{\partial \mathcal{U}} \langle (g \nabla f)(p), v_p \rangle\ dp$$

where v_p is the outwards pointing boundary normal vector as above and $\oint$ is the integral along the boundary. For any g with $g(\partial \mathcal{U}) = 0$ the right side is zero so that:

$$\langle -\operatorname{div} \nabla f, g \rangle = \langle \nabla f, \nabla g \rangle \qquad \forall g \in \mathcal{L}^2(\mathcal{M}) \colon g(\partial \mathcal{U}) = 0$$

The linear operator $\Delta \colon \mathcal{L}^2(\mathcal{M}) \to \mathcal{L}^2(\mathcal{M});\ f \mapsto \operatorname{div} \nabla(f)$ is known as the Laplace-Beltrami - or simply Laplace operator or Laplacian.

The harmonic functions over an open subset $U \subset \mathcal{M}$ are exactly the functions $f \colon U \to \mathbb{R}$ that adhere to the boundary conditions as well as to the Euler-Lagrange equation of the Dirichlet energy:

$$(\Delta f)(x) = 0 \quad \forall x \in \mathcal{U}\ \partial \mathcal{U} \tag{2.8}$$

Proof. From variational calculus follows:

$$0 \overset{!}{=} \left[\frac{d}{d\epsilon} \frac{1}{2} \langle \nabla (f + \epsilon g), \nabla (f + \epsilon g) \rangle \right]_{\epsilon = 0} = \langle \nabla f, \nabla g \rangle = \langle \Delta f, g \rangle \quad \forall g \in \mathcal{L}^2(\mathcal{M}) \colon g(\partial \mathcal{U}) = 0$$

so that the Euler-Lagrange equation of Eq. 2.5 is Eq. 2.8. □

The definition of the Dirichlet energy, the Laplace operator and all derived quantities such as harmonic maps depend only on *intrinsic* quantities, i.e. they only depend on distances on the shape but not on how the shape actually is embedded. Intrinsic quantities are important for the matching of articulated shapes as articulation changes them little. Figure 2.3 shows several harmonic and biharmonic

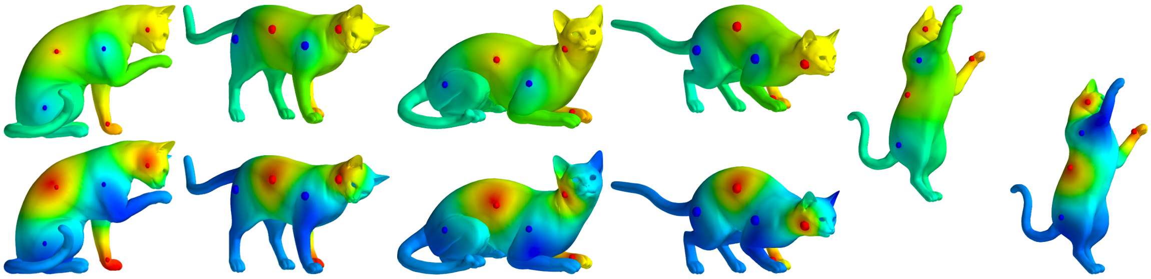


Figure 2.3.: Harmonic and biharmonic functions on different instances of the cat.

functions on several articulated versions of a mesh. As the calculation of these functions depends only on intrinsic quantities the functions on the meshes are very similar.

2.1.3. Biharmonic functions

Harmonic functions on $\mathcal{U}$ are smooth within $\mathcal{U}$, but not necessarily on the boundary, e.g. the harmonic function in Figure 2.1a is not smooth at the constraint vertices. Another definition for smoothness stems from the thin-plate energy, defined for functions $f\colon \mathbb{R}^2 \to \mathbb{R}$ on the Euclidean plane by:

$$\frac{1}{2}\left(\frac{\partial^2}{\partial_x^2} f\right)^2 + \left(\frac{\partial^2}{\partial_x \partial_y} f\right)^2 + \frac{1}{2}\left(\frac{\partial^2}{\partial_y^2} f\right)^2 \qquad \left[= \frac{1}{2}\|\Delta f\|_F^2\right] \tag{2.9}$$

The thin-plate energy on surfaces is defined analog:

$$\frac{1}{2}\|\Delta f\|_F^2 \,. \tag{2.10}$$

A function $f\colon \mathcal{U} \subset \mathcal{M} \to \mathbb{R}$ of extremal thin-plate energy is called a *biharmonic functions*. Biharmonic functions are precisely the ones fulfilling the Euler-Lagrange equation of the thin-plate energy, which is:

$$(\Delta^2 f)(x) = 0 \quad \forall x \in \mathcal{U}\ \partial\mathcal{U} \tag{2.11}$$

Proof. Again we use variational calculus for the Euler-Lagrange equation: $0 \overset{!}{=} \left[\frac{d}{d\epsilon}\,\frac{1}{2}\|\Delta(f+\epsilon g)\|^2\right]_{\epsilon=0} = \langle \Delta^2 f, g\rangle \qquad \forall g \in \mathcal{L}^2(\mathcal{M}) \colon g(\partial\mathcal{U}) = 0$ □

The class of biharmonic functions is *larger* than the class of harmonic functions. For example, every harmonic function is also biharmonic:

$$\Delta f\Big|_{\mathrm{int}(\mathcal{U})} = 0 \Rightarrow \Delta^2 f\Big|_{\mathrm{int}(\mathcal{U})} = 0 \,.$$

	Euler-Lagr. eq.	**Effect**
Dirichlet energy $\|\nabla f\|_F^2$	harmonic $\Delta f = \mathbf{0}$	minimum principle on f, i.e. in an open neighborhood f is constant or has no extrema
Thin-plate energy $\|\Delta f\|_F^2$	biharmonic $\Delta^2 f = \mathbf{0}$	minimum principle on Δf, i.e. in an open neighborhood Δf is constant or has no extrema

Table 2.1.: Dirichlet energy vs thin-plate energy

Furthermore, a function $f\colon \mathcal{U} \to \mathbb{R}$ is biharmonic if and only if there is a harmonic function $h\colon \mathcal{U} \to \mathbb{R}$, such that:

$$\Delta f = h\Big|_{\mathrm{int}\,\mathcal{U}} \qquad\qquad \Delta h = 0\Big|_{\mathrm{int}\,\mathcal{U}} \tag{2.12}$$

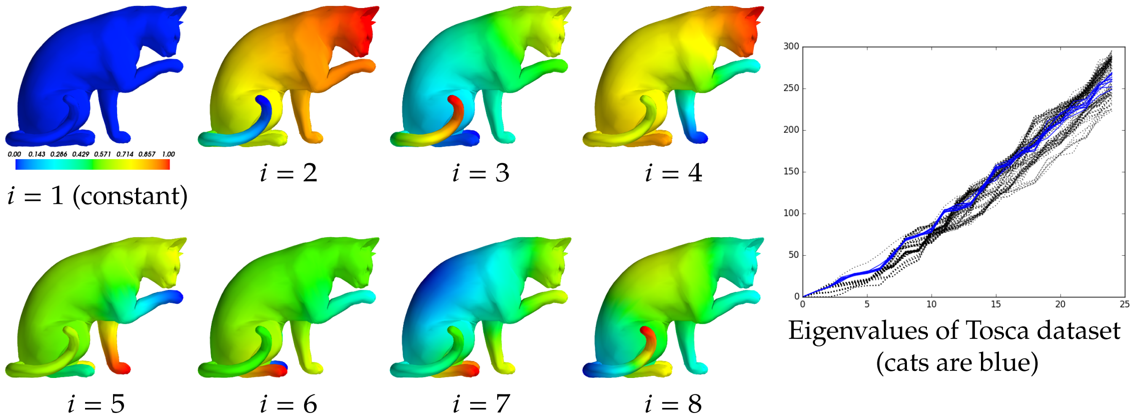
To construct a biharmonic function we can therefore enforce boundary constraints on f *as well as* on h. The additional degrees of freedom allow smoother interpolation with biharmonic functions. A typical choice is to restrict the Laplace operator at the boundary as well $\Delta f\Big|_{\partial\mathcal{U}} = 0$, which led to the smooth interpolations of Figure 2.1b and Figure 2.2b [BK04b]. Table 2.1 gives an overview over both concepts.

2.1.4. Spectral decomposition of the Laplace operator

For natural boundary conditions Δ is a self-adjoint operator, i.e. $\langle \Delta f, g\rangle = \langle f, \Delta g\rangle$. Therefore, it has a spectral decomposition with real eigenvalues λ_i and eigenvectors ϕ_i that build a basis of the $\mathcal{L}^2(\mathcal{M})$. Δ is positive definite as $\langle \Delta f, f\rangle = \langle \nabla f, \nabla f\rangle \geq 0$, so that all eigenvalues are larger or equal to 0. With natural boundary conditions the multiplicity of the eigenvalue 0 is equal to the number of components. Eigenvalues asymptotically converge to linear growth [RWP06]:

$$\exists \kappa \in \mathbb{R}_+\colon\ \lim_{i\to\infty} \frac{|\lambda_i - \kappa \cdot i|}{\kappa \cdot i} = 0 \tag{2.13}$$

Figure 2.4 shows several eigenvectors on an example surface and the growth of the eigenvalues of various shapes after normalization by their area. When smoothing a function via gradient descent on the Dirichlet energy, the next section shows that the coefficients in the eigenbasis change by $\exp(-t\lambda_i)$, thus the smaller the eigenvalue of an eigenspace, the later the corresponding signal diminishes. The eigenvalues of the eigenvectors thus corresponds to their smoothness. In the Euclidean plane the Laplace eigenvectors and eigenvalues equal the Fourier basis. On general manifolds the Laplace eigenvectors are a generalization of the Fourier transformation to manifolds.



$i = 1$ (constant)

$i = 2$

$i = 3$

$i = 4$

$i = 5$

$i = 6$

$i = 7$

$i = 8$

Eigenvalues of Tosca dataset (cats are blue)

Figure 2.4.: Eigenvectors of the Laplace operator on a single and its eigenvalues on multiple meshes.

2.1.5. Diffusion processes

When the distribution of a quantity on a surface changes over time, such that quantities move along the surface opposite to the gradient direction and proportional to its length, this is called a linear diffusion process. A typical example is the evolution of an initial temperature distribution $F_0 \in \mathcal{L}^2(\mathcal{M})$ into its equilibrium described by the heat-equation:

$$F_t \colon \mathbb{R} \to \mathcal{L}^2(\mathcal{M}) \qquad f_0 = F_0 \qquad -\partial_t f_t = \Delta(f_t) \tag{2.14}$$

Thus an intuitive meaning of the Laplace operator is the temperature change in the next infinitesimal step of a heat diffusion process. Previously we defined smoothness of a function by small Dirichlet energy. Smoothing a function by infinite steps against the gradient of the Dirichlet energy results in the same equation:

$$\partial_t f_t = -\left[\frac{d}{dt}\,\frac{1}{2}\langle \nabla f, \nabla f\rangle\right]_{f=f_t} = -\Delta(f_t)$$

Thus the heat-equation describes a gradient descent on the Dirichlet energy thereby steadily increasing the smoothness of f_t.

The solution of the partial differential equation Eq. (2.14) with initial temperature F_0 is:

$$f_t(x) = (\exp(-t\Delta)\; F_0)(x) = \sum_i \exp(-t\lambda_i)\phi_i(x)\langle \phi_i, F_0\rangle \quad \exp(A) := \sum_i \frac{A^i}{i!} \tag{2.15}$$

The heat-kernel $h_t \colon \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is defined such that $h_t(\cdot, y) \colon \mathcal{M} \to \mathbb{R}$ is the

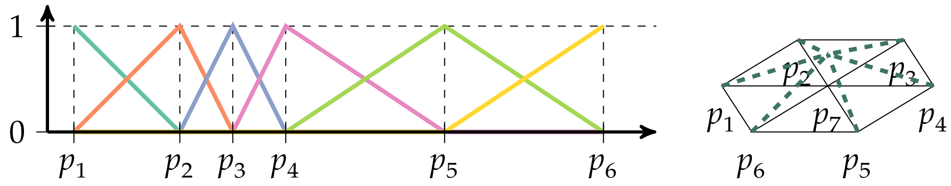


Figure 2.5.: Nodal basis on a 1d domain and a single basis vector on a 2d domain.

solution of the heat-diffusion for a delta-distribution at $y \in \mathcal{M}$:

$$h_t(x,y) = \sum_i \exp(-t\lambda_i)\phi_i(x)\langle \phi_i, \delta_y \rangle = \sum_i \exp(-t\lambda_i)\phi_i(x)\phi_i(y) \tag{2.16}$$

and f_t can also be written as:

$$f_t(x) = \int_p h_t(x,p)F_0(p)\, dp \tag{2.17}$$

2.1.6. Discretization

After arranging the n vertices in a predefined order, a function $f \in \mathcal{L}^2(\mathcal{M})$ can be represented by a vector $\hat{f} \in \mathbb{R}^n$ of function values. The underlying basis is called the "hat" or *nodal basis* and is depicted in Figure 2.5. A basis element $b_p : \mathcal{M} \to \mathbb{R}$ of the point p is defined to be 1 on p, 0 on all other vertices and linearly interpolated on the triangles. The linear space spanned by the nodal basis contains all continuous functions, which are linear on the triangles.

Cotan-Laplace The cotan formula [PP93; Mey+03] is a well-known discretization of the Laplace operator. Let p, q be two points in a triangle of area A and let α_{pq} be the angle opposite the edge pq. The scalar products of the nodal basis on the single triangle and the scalar product of their gradients are

$$\langle b_p, b_p \rangle = \frac{A}{6} \qquad \langle \nabla b_p, \nabla b_q \rangle = \frac{\cot \alpha_{pq}}{2}$$
$$\langle b_p, b_q \rangle = \frac{A}{12} \qquad \langle \nabla b_p, \nabla b_p \rangle = -\langle \nabla b_p, \nabla b_q \rangle - \langle \nabla b_p, \nabla b_r \rangle$$

Summation over all triangles leads to a matrix representation $W \in \mathbb{R}^{n \times n}$ of the scalar product $\langle \cdot, \cdot \rangle$, which is also called the "mass matrix", and a matrix representation $C \in \mathbb{R}^{n \times n}$ of $\langle \nabla \cdot, \nabla \cdot \rangle$, which is also called the "stiffness matrix". Let $\mathcal{N}'(p)$ be the adjacent triangles to vertex p, $\mathcal{N}(p)$ be the adjacent vertices to vertex p, let A_k be the area of triangle k. For edge pq let A_{pq} and B_{pq} be the areas of the adjacent triangles and let α_{pq} and β_{pq} be the opposing angles in the adjacent triangles. Then

W and C are

$$W_{pq} = \begin{cases} \frac{\sum_{k \in \mathcal{N}'(p)} |A_k|}{6} & p = q \\ \frac{|A_{pq}|+|B_{pq}|}{12} & p \neq q \end{cases} \qquad C_{pq} = \begin{cases} \frac{\cot \alpha_{pq} + \cot \beta_{pq}}{2} & p \neq q \\ -\sum_{k \in \mathcal{N}(p)} C_{pk} & p = q \end{cases} \tag{2.18}$$

and the Laplace-Beltrami operator is $L = W^{-1}C$. Often the mass matrix is replaced by the diagonal "lumped-mass matrix", which simplifies further calculations:

$$W_{pq} = \begin{cases} \frac{\sum_{k \in \mathcal{N}'(p)} |A_k|}{3} & p = q \\ 0 & p \neq q \end{cases} \tag{2.19}$$

Spectral Decomposition The spectral decomposition of the Laplace operator Δ can be written by a generalized eigenvalue problem over the matrices W and C:

$$C\phi_i = \lambda_i W \phi_i \text{ so that } \phi_i^T W \phi_j = \delta_{ij} \text{ and } \lambda_1 \leq \lambda_2 \leq \ldots$$

As long as C and W are symmetric and W is positive definite the generalized eigenvalue problem has real eigenvalues and yields a basis of $\mathcal{L}^2(\mathcal{M})$. If W is positive definite, the generalized eigendecomposition can be reduced to the standard eigenvalue problem of the symmetric matrix $W^{-1/2}CW^{-1/2}$, whose eigenvectors ψ_i are related to ϕ_i by $\phi_i = W^{-1/2}\psi_i$.

Graph Laplacian For equally sampled meshes the Laplace operator can be well approximated from the connectivity alone, while neglecting the geometry of the embedding. Let $N(i)$ be the indices of vertices adjacent to vertex i and let $|N(i)|$ be the number of neighbors of vertex i. Then the *graph Laplacian* is defined by:

$$(Cf)_i = \sum_{j \in N(i)} \frac{f(j) - f(i)}{|N(i)|} \tag{2.20}$$

Point Cloud Laplacian Smoothness is a concept of small Dirichlet energy, defined by the eigenvectors belonging to the smallest eigenvalues. Although, from the derivation of the cotan weighting scheme above might hint that the local shape of the manifold is important in the definition of smoothness it is not. Indeed, the Graph Laplacian, which simply connects all neighboring vertices with uniform weights, often results in very similar smoothing. It is therefore not surprising that generalizations of the Laplace operator to point clouds, which lack connectivity information, are feasible as well.

Belkin introduced such a discretization [BSW09; LPG12]. They model Δ so that for a small, fixed time step t the heat-diffusion with Δ equals Euclidean heat-diffusion:

$$(Cu)_i = \frac{1}{4\pi t^2} \sum_j \exp\left(-\frac{\|p_i - p_j\|^2}{4t}\right)(u_i - u_j) \,, \tag{2.21}$$

where $p_i \in \mathbb{R}^3$ are the point positions. Point areas and thereby W are calculated from a local estimation of the surface.

Discretization of the biharmonic equation As discussed above, harmonic functions also solve the biharmonic equation. After discretization this is true if care is taken. The evaluation of $(W^{-1}C)^2$, which is the discretization of Δ^2, at vertex requires the values of its two-ring neighborhood. It therefore can not be computed in the one-ring of the boundary. The discretization of Eq. 2.12 does not have this limitation.

2.1.7. Further results

The cotan weights are independent from the triangulation in the sense that the results converge as the triangulation is correctly refined [Xu04]. They are often used and have been derived several times, for example using finite elements methods as above or the discrete exterior calculus [Hir03; KS13]. There are also discretizations to other primitives, such as to general polygons[AW11]. Still, there is no discretization on triangles, which exhibits all properties of the continuous Laplace operator [War+07]: symmetry, locality, linearity, semi-definite positivity.

2.2. Mapping between shapes

In the following we investigate maps between two shapes $\mathcal{M}$ and $\mathcal{N}$ embedded with the differentiable maps $\mathbf{p}\colon \mathcal{M} \to \mathbb{R}^3, \mathbf{q}\colon \mathcal{N} \to \mathbb{R}^3$. A point-wise map $T\colon \mathcal{M} \to \mathcal{N}$ between the shapes induces several maps and functions, which help understand the properties of T.

Functional maps For example, T can be used to map functions defined on $\mathcal{N}$ onto functions defined on $\mathcal{M}$: $f \in \mathcal{L}^2(\mathcal{N}) \mapsto f \circ T \in \mathcal{L}^2(\mathcal{M})$. This mapping is called the *pullback functional map F_T*, as it pull function values along T:

$$F_T\colon \mathcal{L}^2(\mathcal{N}) \to \mathcal{L}^2(\mathcal{M}), f \mapsto f \circ T \tag{2.22}$$

Note, while T is from $\mathcal{M}$ onto $\mathcal{N}$, the functional map is the other way around.

The transportation of functions from $\mathcal{N}$ onto $\mathcal{M}$ defines a new integration of functions on $\mathcal{N}$ by first projecting them onto $\mathcal{M}$ and integrating there:

$$\int_{\mathcal{N}} f(q)\, dq^* := \int_{\mathcal{M}} F_T(f)(p)\, dp \tag{2.23}$$

The new integral is defined by the area form dq^* and induces a new scalar product $\langle f, g \rangle_{\mathcal{M}\to\mathcal{N}}$ as well as a new norm for functions $f, g \in \mathcal{L}^2(\mathcal{N})$:

$$\langle f, g \rangle_{\mathcal{M}\to\mathcal{N}} := \int_{\mathcal{N}} f(q) g(q)\, dq^* \qquad \|f\|_{\mathcal{M}\to\mathcal{N},2} := \sqrt{\langle f, f \rangle_{\mathcal{M}\to\mathcal{N}}} \tag{2.24}$$

Area-preserving maps The map T is called *area-preserving* if the old and new area forms on $\mathcal{N}$ match:

$$\langle f, g \rangle_{\mathcal{M}\to\mathcal{N}} = \langle f, g \rangle_{\mathcal{N}} \qquad \forall f, g \in \mathcal{L}^2(\mathcal{N}) \tag{2.25}$$

and a point-wise functional map is area-preserving if and only if it is orthogonal:

$$T \text{ area-preserving} \quad \Leftrightarrow \quad F_T^T = F_T^{-1} \tag{2.26}$$

Proof. $\langle F_T(f), F_T(g) \rangle_{\mathcal{M}} = \langle f, g \rangle_{\mathcal{N}} \forall f, g \in \mathcal{L}^2(\mathcal{N}) \Leftrightarrow F_T^T F_T = \mathrm{Id}$. □

Differential Another map induced by T is the *differential* dT_p. It maps tangent vectors from $T_p\mathcal{M}$ onto $T_{T(p)}\mathcal{N}$ by mapping the underlying differentiable curves $x \in T_p\mathcal{M}$:

$$dT_p \colon T_p\mathcal{M} \to T_{T(p)}\mathcal{N},\ X = [\partial_t(\mathbf{p} \circ x)]_{t=0} \mapsto [\partial_t(\mathbf{q} \circ T \circ x)]_{t=0} \tag{2.27}$$

The differential is a linear map. It is defined only on the tangent space $T_p\mathcal{M}$ and it is undefined in the normal direction $n_p \in \mathbb{R}^3$. The differential at a point is defined by the image of the linear independent tangent vectors as they span a basis of the tangent space. In applications T as well as the embeddings $\mathbf{p}, \mathbf{q}$ are often linear on the triangles. In this case, the differential is constant on the triangles and is defined by the embedded edges, which span the tangent spaces.

The differential describes the local change of the surface metric. For example the new area form dq^* is related to the old area form dq by $dq^* = \det(dT_p) \cdot dq$ and area-preserving maps are the maps whose differential has determinant 1.

Proof. For a local parameterization of $\mathcal{M}$ over an open set $U \subset \mathbb{R}^2$ the area forms are $dq^* = \det(\partial_u(\mathbf{q} \circ T), \partial_v(\mathbf{q} \circ T))\ dudv$ and $dp = \det(\partial_u\mathbf{p}, \partial_v\mathbf{p})\ dudv$, so that the equation above follows. □

The adjunct of the differential dT_p^T is defined by the equation

$$\langle dP_p^T(f), g\rangle = \langle f, dP_p(g)\rangle \ \forall f, g \in \mathbb{R}^3$$

and maps the gradient of a function on $\mathcal{N}$ onto the gradient of the same function on $\mathcal{M}$:

$$\nabla_{\mathcal{M}}(F_T(f)) = dT_p^T(\nabla_{\mathcal{N}} f) \tag{2.28}$$

Proof. For any $x \in T_p\mathcal{M}$ follows from Eq. (2.3): $\langle \nabla_{\mathcal{M}}(f \circ T), [\partial_t(\mathbf{p} \circ x)]_{t=0}\rangle_{\mathcal{M}} = [\partial_t(f \circ T \circ x)]_{t=0} = \langle \nabla_{\mathcal{N}} f, [\partial_t(\mathbf{q} \circ T \circ x)]_{t=0}\rangle_{\mathcal{N}} = \langle \nabla_{\mathcal{N}} f, dT_p([\partial_t(\mathbf{p} \circ x)]_{t=0})\rangle_{\mathcal{N}}$, so that $\langle \nabla_{\mathcal{M}}(f \circ T), X\rangle_{\mathcal{M}} = \langle \nabla_{\mathcal{N}} f, dT_p(X)\rangle_{\mathcal{N}}$ for any $X \in T_p\mathcal{M}$. □

Furthermore, the scalar product on the tangent vectors and on vector fields can be pulled from $\mathcal{N}$ onto $\mathcal{M}$ using the differential:

$$\langle X, Y\rangle_{\mathcal{N}\to\mathcal{M},p} := \langle dT_p(X), dT_p(Y)\rangle_{T(p)} = X^T\, dT_p^T dT_p\, Y \qquad \forall X, Y \in T_p\mathcal{M} \tag{2.29}$$

$$\langle X, Y\rangle_{\mathcal{N}\to\mathcal{M}} := \int_{\mathcal{N}} \left\langle dT_p(X_p), dT_p(Y_p)\right\rangle_{T(p)} d(T(p))^* \qquad \forall X, Y \in T_p\mathcal{M} \tag{2.30}$$

Conformal maps A map T is called a *conformal* if it preserves the angles between tangent vectors:

$$\left.\langle X, Y\rangle_{\mathcal{M}}\middle/\|X\|_{\mathcal{M}}\|Y\|_{\mathcal{M}}\right. = \left.\langle X, Y\rangle_{\mathcal{N}\to\mathcal{M}}\middle/\|X\|_{\mathcal{N}\to\mathcal{M}}\|Y\|_{\mathcal{N}\to\mathcal{M}}\right. . \tag{2.31}$$

A map is conformal if and only if its differential is a similarity transform at any point, i.e. for every point p there is a scaling factor γ_p so that $(dT_p)^T(dT_p) = \gamma_p^2$ Id. The scaling factor $\gamma_p \in \mathbb{R}$ is called the *conformal factor*.

Proof. "⇐" If $dT_p^T dT_p = \gamma_p^2$ then $\langle X, Y\rangle_{\mathcal{N}\to\mathcal{M}} = \gamma_p^2\langle X, Y\rangle_{\mathcal{M}}$, thus follows Eq. (2.31). "⇒" For a conformal map and $X, Y \in T_p\mathcal{M}$ holds $\langle X, Y\rangle_{\mathcal{M}} = 0 \Leftrightarrow \langle X, Y\rangle_{\mathcal{N}\to\mathcal{M}} = 0$. Let U, V be a orthonormal basis of $T_p\mathcal{M}$. Then from $U^TV = 0$, $(U+V)^T(U-V) = 0$ follows $0 = \langle U+V, U-V\rangle_{\mathcal{N}\to\mathcal{M}} = \langle U, U\rangle_{\mathcal{N}\to\mathcal{M}} - \langle V, V\rangle_{\mathcal{N}\to\mathcal{M}}$, i.e. $\langle U, U\rangle_{\mathcal{N}\to\mathcal{M}} = \langle V, V\rangle_{\mathcal{N}\to\mathcal{M}} = \gamma_p^2$. Therefore, for any vectors $X, Y \in T_p\mathcal{M}$ we have $\langle X, Y\rangle_{\mathcal{N}\to\mathcal{M}} = \begin{pmatrix}\langle X, U\rangle \\ \langle X, V\rangle\end{pmatrix}^T \gamma_p^2 \text{ Id} \begin{pmatrix}\langle Y, U\rangle \\ \langle Y, V\rangle\end{pmatrix}$. □

An interesting property of conformal maps is their preservation of the scalar product on gradients and the Dirichlet energy:

$$\langle \nabla_{\mathcal{M}} F_T(f), \nabla_{\mathcal{M}} F_T(g)\rangle_{\mathcal{M}} = \langle \nabla_{\mathcal{N}} f, \nabla_{\mathcal{N}} g\rangle_{\mathcal{N}} \ \forall f, g \in \mathcal{L}^2(\mathcal{N}) \tag{2.32}$$

Proof. $\int_{\mathcal{M}} \nabla_{\mathcal{M}}(f \circ T)(p)^T \nabla_{\mathcal{M}}(g \circ T)(p)\; dp = \int_{\mathcal{M}} \gamma_p^2 (\nabla_{\mathcal{N}}(f)(p)^T \nabla_{\mathcal{N}}(g)(p)) \circ T\; dp$
$= \int_{\mathcal{N}} \frac{\gamma_p^2}{\gamma_p^2} (\nabla_{\mathcal{N}}(f)(q)^T \nabla_{\mathcal{N}}(g)(q))\; dq$ □

Remark: We mentioned that a linear diffusion process can be described as a gradient descent on the Dirichlet energy. Linear diffusion processes can be very different on conformally equivalent shapes, i.e. shapes mapped onto each other by a conformal map. That is despite the preservation of the Dirichlet energy along the conformal maps. The apparent contradiction vanishes as the gradient of the Dirichlet energy, which is the direction of fastest change, depends on the area form as well.

Indeed, T is a conformal map if and only if F_T pulls the Laplace operator from $\mathcal{M}$ onto $\mathcal{N}$:

$$T \text{ conformal} \quad \Leftrightarrow \quad \Delta_{\mathcal{N}} = F_T^T \Delta_{\mathcal{M}} F_T \tag{2.33}$$

Proof. ⇒ follows directly from Eq. (2.32), for ⇐ see [Rus+13a]. □

Isometries A map T preserving intrinsic (that is geodesic) distances is called *isometric*, that is a map which preserves the scalar product on the tangent spaces:

$$\langle X, Y \rangle_{\mathcal{M},p} = \langle X, Y \rangle_{\mathcal{N}\to\mathcal{M},p} \; \forall X, Y \in T_p\mathcal{M} \forall p \in \mathcal{M} \tag{2.34}$$

Two shapes are called isometric if they can be mapped by an isometry. There are several other equivalent descriptions of isometric maps. For a point-wise map T the following are equivalent statements:

(i) T is an isometry according to Eq. (2.34).
(ii) T preserves the length $L[x] := \int_0^1 \left\| \left[\frac{d}{dt} x \right]_{t=\tau} \right\|_2 \, d\tau$ of curves $x\colon [0,1] \to \mathcal{M}$, i.e. $L[x] = L[T \circ x]$.
(iii) dT_p is a rotation for all $p \in \mathcal{M}$.
(iv) T is area-preserving and conformal.
(v) F_T is orthogonal and $\Delta_{\mathcal{N}} = F_T^T \Delta_{\mathcal{M}} F_T$.

Proof. (i) ⇔ (ii): L only depends on the scalar product of the tangent spaces. (i) ⇔ (iii): From the definition of $\langle X, Y \rangle_{\mathcal{N}\to\mathcal{M},p}$. (iii) ⇔ (iv): area-preserving maps are exactly the ones with $\det(dT_p) = 1$ and conformal maps are exactly the ones where dT_p is a similarity transform. (iv) ⇔ (v): equivalent as shown above. □

Isometries transport the intrinsic metric and therefore preserve all intrinsic quantities, such as angles between curves, lengths of curves, areas of sets. The corresponding functional maps preserve the scalar product on functions ($\langle f, g \rangle_{\mathcal{M}} = \langle f, g \rangle_{\mathcal{N}\to\mathcal{M}}$), the scalar product on gradients of functions ($\langle \nabla_{\mathcal{M}} f, \nabla_{\mathcal{M}} g \rangle_{\mathcal{M}} = \langle \nabla_{\mathcal{N}} f, \nabla_{\mathcal{N}} g \rangle_{\mathcal{N}\to\mathcal{M}}$) and the Dirichlet energy. Furthermore, the Laplace operator (Eq. (2.35)) commutes with the functional map F_T:

$$\Delta_{\mathcal{M}} \circ F_T = F_T \circ \Delta_{\mathcal{N}} \tag{2.35}$$

As the Dirichlet energy is preserved as well as the scalar product on functions, linear diffusion processes, which are a gradient descent on the Dirichlet energy, are the same on both shapes. And also the eigenvalues and eigenvectors of the Laplace operator match.

2.3. Extrinsic shape alignment

The goal of this section is to derive a low distortion correspondence map $T\colon \mathcal{M} \to \mathcal{N}$ of two shapes $\mathcal{M}$ and $\mathcal{N}$ with the embeddings $\mathbf{p}\colon \mathcal{M} \to \mathbb{R}^3$ and $\mathbf{q}\colon \mathcal{N} \to \mathbb{R}^3$. This can be done by finding a new embedding $\hat{\mathbf{p}}$ such that $\hat{\mathbf{p}}$ and $\mathbf{q}$ are aligned and that the deformation $\mathbf{d}_\mathrm{p} = \hat{\mathbf{p}} \circ \mathbf{p}^{-1}$ has low distortions. How well shapes are aligned can be measured by their Hausdorff distance, which is the smallest $\delta > 0$ such that:

$$\begin{aligned} &\|\mathbf{p}(p) - \mathbf{q}(q)\|_2 \leq \delta \quad \forall p \in \mathcal{M}\, \exists q \in \mathcal{N} \quad \text{and} \\ &\|\mathbf{p}(p) - \mathbf{q}(q)\|_2 \leq \delta \quad \forall q \in \mathcal{N}\, \exists p \in \mathcal{M} \end{aligned}$$

If $(\mathcal{M}, \hat{\mathbf{p}})$ and $(\mathcal{N}, \mathbf{q})$ are aligned and have small Hausdorff distance then the new correspondences can be calculated by (note that embeddings are injective):

$$T\colon p \mapsto \operatorname*{arg\,min}_{q \in \mathcal{N}} \|\hat{\mathbf{p}}(p) - \mathbf{q}(q)\|_2 \tag{2.36}$$

If on the other hand we have a correspondence map T then the embeddings $\hat{\mathbf{p}} := \mathbf{q} \circ T$ and $\mathbf{q}$ are aligned.

The distortion of $\mathbf{d}$ is closely related to the differential of the identity map from $(\mathcal{M}, \mathbf{p})$ to $(\mathcal{M}, \hat{\mathbf{p}})$ and is some measure how much lengths were changed, i.e. how much the differential deviates from a rotation.

2.3.1. Rigid iterative closest points

The Iterative Closest Points (ICP) method aligns shapes by repeatedly alternating between optimizing the deformation and the correspondences. Usually it assumes predefined initial correspondences. From these it calculates a new alignment, which then yield new correspondences using Eq. (2.36), which again lead to another alignment, and so on.

The earliest instances of the shape matching problem was the alignment of multiple partial point cloud scans of a static scene, for which rigid deformations are sufficient [RL01b]. For a set of *fixed* set of point-to-point correspondences $(c_1, d_1), \ldots, (c_s, d_s) \in \mathcal{M} \times \mathcal{N}$ the optimal rigid alignment $x \to \mathbf{R}x + t$ in the least squares sense is the minimum of the quadratic cost function:

$$E = \sum_i (\mathbf{R} \cdot \mathbf{p}(c_i) + t - \mathbf{q}(d_i))^2 \quad \text{with } \mathbf{R}\mathbf{R}^T = \text{Id} \tag{2.37}$$

Let $\bar{\mathbf{c}} = \frac{1}{n} \sum \mathbf{p}(c_i)$, $\bar{\mathbf{d}} = \frac{1}{n} \sum \mathbf{q}(d_i)$, $\mathbf{C} = [\mathbf{p}(c_1) - \bar{\mathbf{c}}, \ldots, \mathbf{p}(c_n) - \bar{\mathbf{c}}]$, $\mathbf{D} = [\mathbf{q}(d_1) - \bar{\mathbf{d}}, \ldots, \mathbf{q}(d_n) - \bar{\mathbf{d}}]$ and $\mathbf{U} \operatorname{diag}(\boldsymbol{\lambda}) \mathbf{V}^T$ be the singular value decomposition of $\mathbf{C}\mathbf{D}^T$. Then the optimal alignment is

$$\mathbf{R} = \mathbf{V}^T \mathbf{U}^T \qquad\qquad t = \bar{\mathbf{q}} - \mathbf{R}\bar{\mathbf{p}} \tag{2.38}$$

Proof. From $\partial_{\mathbf{t}} E = 0$ follows $2 \sum (\mathbf{R}\bar{\mathbf{c}} + \bar{\mathbf{d}} + \mathbf{t}) = 0$ and $\mathbf{t}$ as above. Substituting the optimal $\mathbf{t}$ into E yields $E = \operatorname{tr}\left((\mathbf{RC} - \mathbf{D})(\mathbf{RC} - \mathbf{D})^T\right) = \operatorname{tr}\left(\mathbf{C}\mathbf{C}^T + \mathbf{D}\mathbf{D}^T\right) - 2 \operatorname{tr}\left(\mathbf{RCD}^T\right)$. R follows from maximizing $\operatorname{tr}\left(\mathbf{RCD}^T\right) = \operatorname{tr}(\mathbf{VRU} \operatorname{diag} \lambda)$. The rotation matrix $\mathbf{VRU}$ has diagonal entries strictly less than 1. Then $\operatorname{tr}(\mathbf{VRU} \operatorname{diag} \boldsymbol{\lambda}) \leq \sum_i \boldsymbol{\lambda}_i$ with equality for $\mathbf{VRU} = \text{Id}$. □

Analysis. Rigid ICP is an iterative, deformation-based correspondence algorithm. From its simple formulation follow a number of properties, which many advanced methods share as well:

- ICP is the minimization of the non-convex cost function Eq. 2.37.
- Optimizing critically depends on the initial correspondences.
- Heuristics to prune correspondences are crucial for stability.
- Nearest neighbor search in $\mathbb{R}^3$ tends to match extrinsically close point to extrinsically close points. This is an advantage when matching rigid shapes, but becomes a disadvantage when matching articulated shapes.

Further work. Several methods address the stability issues of the original ICP method. L_1 and Huber distances were used to circumvent the sensitivity of L_2 distance to outliers [Hub92; Fit03]. Binary nearest neighbor assignments were relaxed to soft assignments [Gol+98; TK04]. Coherent point-drift assumes that neighboring points should move in a similar direction [M+07; MS10]. The initial correspondence were determined by Graph matching as described in section 2.5. And some methods find the globally best alignment by following a branch and bound approach over the rigid transformations of the $\mathbb{R}^3$ [CP16; Yan+16; ZPK16].

2.3.2. Deformable iterative closest points

Alignment of articulated shapes requires non-rigid deformation methods. *Deformable ICP* methods generalize the above rigid alignment method to articulated shapes by substituting the rigid deformation with a non-rigid deformation model. In the following we present several non-rigid deformation models and the resulting deformable ICP methods.

Let again $\mathcal{M}$ and $\mathcal{N}$ be two shapes embeddings by $\mathbf{p}\colon \mathcal{M} \to \mathbb{R}^3$ and $\mathbf{q}\colon \mathcal{N} \to \mathbb{R}^3$, let further $(c_1, d_1), \dots, (c_s, d_s) \in \mathcal{M} \times \mathcal{N}$ be known sparse correspondences and we solve for a new coordinate function $\hat{\mathbf{p}}$ so that $\hat{\mathbf{p}}$ and $\mathbf{q}$ are aligned.

Smooth coordinate functions A first deformation model constraints $\hat{\mathbf{p}}$ at the correspondences target points and minimizes the smoothness of $\hat{\mathbf{p}}$ elsewhere. Smoothness is measured with the Dirichlet energy (Eq. (2.5), Figure 2.2a), or the thin-plate energy (Eq. 2.10, Figure 2.2b):

$$\min_{\hat{\mathbf{p}}_i} \quad w_d \|\nabla \hat{\mathbf{p}}_i\|_F^2 + w_t \ \|\Delta \hat{\mathbf{p}}_i\|_F^2 \ + \frac{1}{s} \sum_{i=1,\dots,s} \left\|\hat{\mathbf{p}}_{c_i} - \mathbf{q}_{d_i}\right\|^2 ,$$

where the parameters w_d and w_t allow deciding which smoothness to use and allow trading interpolation quality against smoothness. The quadratic cost function typically has a unique minimum, which can be efficiently computed solving the normal equations. If the constraint indices are fixed, the normal equations can be prefactorized, which greatly decreases the time to solve for an alignment from the constrained positions. The deformation depends on the constraint positions as well as on the intrinsic metric of $\mathbf{p}$ as captured by its Laplace operator Δ and the function norm $\|\cdot\|_F$.

Smooth differential of the deformation The previous deformation model looses many details of the embedding $\mathbf{p}$ as it depends on the embedding only indirectly via the intrinsic metric. The next deformation model restricts the deformation $\mathbf{d} = \hat{\mathbf{p}} \circ \mathbf{p}^{-1}$ and preserves the extrinsic shapes of $\mathbf{p}$ much better. At every point $p \in \mathcal{M}$ we approximate $\mathbf{d}$ with an affine map $A_p\colon \mathbb{R}^3 \to \mathbb{R}^3$, $A_p(x) = \mathbf{L}_p(x - \mathbf{p}(x)) + \mathbf{t}_p$ so that

(i) $A_p(\mathbf{p}(p)) = \hat{\mathbf{p}}(p)\ \forall p \in \mathcal{M}$, i.e. $\mathbf{t}_p = \hat{\mathbf{p}}(p)$ and
(ii) that the restriction of $\mathbf{L}_p$ to the tangent space $T_p\mathcal{M}$ is the differential of $\mathbf{d}$ (Section 2.2), i.e. tangent vectors on $\mathbf{p}$ are mapped onto the corresponding tangent vectors on $\hat{\mathbf{p}}$.

A common deformation model determines $\hat{\mathbf{q}}$ by minimizing the thin-plate energy of the L_p[ARV07; SSP07; Yos+14]:

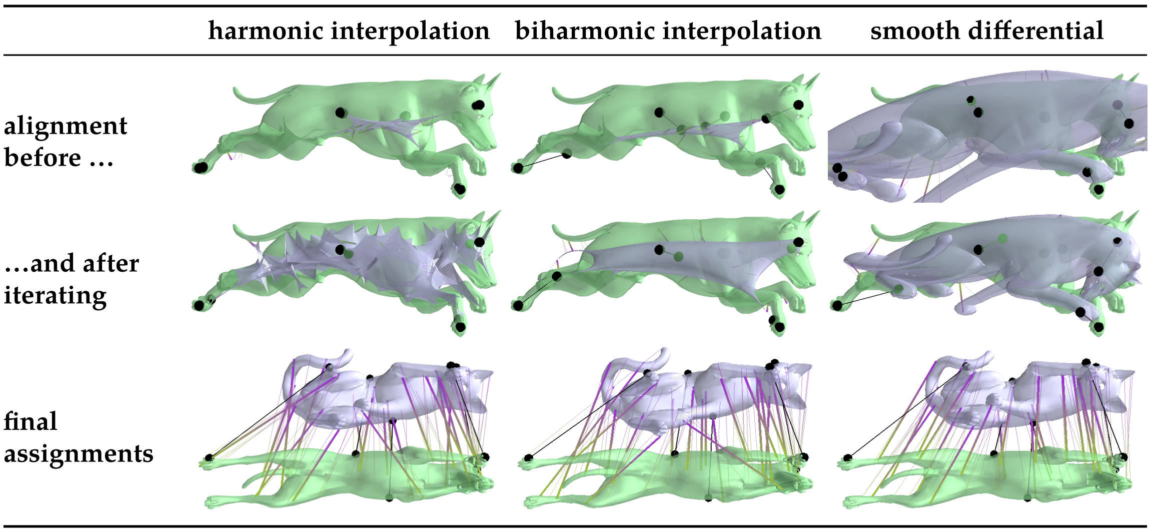
$$w_s \|\Delta \mathbf{L}_p\|_F^2 + \frac{1}{s} \sum_{i=1,\ldots,s} \left\| \hat{\mathbf{q}}_{c_i} - \mathbf{q}_{d_i} \right\|^2 \tag{2.39}$$

Over the variables $\hat{\mathbf{p}}$ and $\mathbf{L}_p$ the quadratic cost function with linear constraints can be solved with Lagrange multipliers or via normal equations after converting the hard to soft constraints. The parameter w_c allows trade-off interpolation quality for smoothness.

Discretization. Let $L_1, L_2, \cdots \in \mathbb{R}^{3\times3}$ be the linear maps on each triangle, further let $\mathcal{H}$ be the set of oriented edges, $a(i,j)$ be the index of the clockwise-oriented triangle which contains the edge $i \to j$ and let $\mathcal{D}$ be a set of dual edges, i.e. $(i,j) \in \mathcal{D}$ iff triangles i, j are adjacent. For the triangle valued functions L_i we discretize the Laplace operator with the graph Laplacian and determine the new positions by minimizing:

$$\begin{aligned} \min_{\hat{\mathbf{q}};L_1,L_2,\ldots} \quad & \sum_{(i,j)\in\mathcal{D}} \|L_i - L_j\|^2 + w_c \sum_{i=1,\ldots,s} \left\| \hat{\mathbf{q}}_{c_i} - \mathbf{q}_{d_i} \right\|^2 \\ \text{s.t.} \quad & T_{a(i,j)}(\mathbf{p}_i - \mathbf{p}_j) = \hat{\mathbf{p}}_i - \hat{\mathbf{p}}_j \quad \forall (i,j) \in \mathcal{H} \end{aligned}$$

A comparison of the different deformation methods is shown in Figure 2.3.2. It shows that deformable ICP as a *local* optimization is easily stuck in local minima.



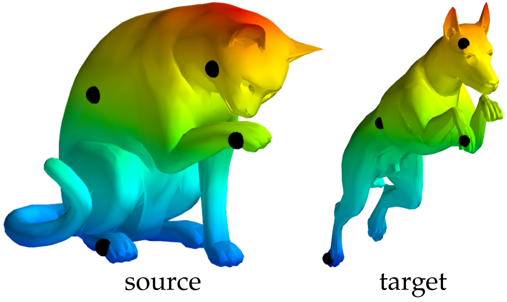


Figure 2.6.: Registration of non-rigid shapes with deformable ICP using different deformation models and from five predefined correspondences (black markers).

2.4. Functional maps

The alignment of articulated shapes becomes much more stable once shifted from extrinsic to *intrinsic* alignment, i.e. invariant of the shape articulation. This is one of the advantages of the functional maps framework, which describes maps between shapes by their effect on real-valued functions. Let $\mathcal{M}$ and $\mathcal{N}$ be two shapes with the embeddings $\mathbf{p}\colon \mathcal{M} \to \mathbb{R}^3$ and $\mathbf{q}\colon \mathcal{N} \to \mathbb{R}^3$. For symbols whose belonging to either $\mathcal{M}$ or $\mathcal{N}$ is not obvious we mark the belonging with $\mathcal{M}$ and $\mathcal{N}$. A functional map is a mapping between the function spaces of $\mathcal{M}$ and $\mathcal{N}$:

$$F\colon \mathcal{L}^2(\mathcal{N}) \to \mathcal{L}^2(\mathcal{M}) \tag{2.40}$$

A correspondence map $T\colon \mathcal{M} \to \mathcal{N}$, which is also called a point-wise map, is tightly coupled to the functional map [Ovs+12] pulling functions from $\mathcal{L}^2(\mathcal{N})$ onto $\mathcal{L}^2(\mathcal{M})$ using T (Section 2.2):

$$F_T\colon \mathcal{L}^2(\mathcal{N}) \to \mathcal{L}^2(\mathcal{M}) \qquad f \mapsto f \circ T \tag{2.41}$$

The point-wise induced functional maps F_T are a proper subset of all the functional maps. A point-wise induced functional map F_T maps a function with a distinguished maximum onto a function maximal at the corresponding point, so that T can be recovered from F_T. For example, in Figure 2.7 a point-wise induced functional map maps a function from $\mathcal{N}$ onto $\mathcal{M}$ and the maxima of both functions coincide. In the functional maps setting we solve for a functional map F using constraints that are valid for F_T and derive the point-wise map T only as a last step, if at all. In contrast to point-wise maps T the functional maps are proper linear maps and form a *vector space* with a well-defined addition and multiplication with real numbers. We can therefore describe functional maps as minima of convex functions and use standard linear algebra tools for their solution and analysis [Ovs+13; Rus+13b]. Figure 2.8 illustrates the interpolation of functional maps, i.e. a simple application of the vector space property. The concatenation of functional maps,

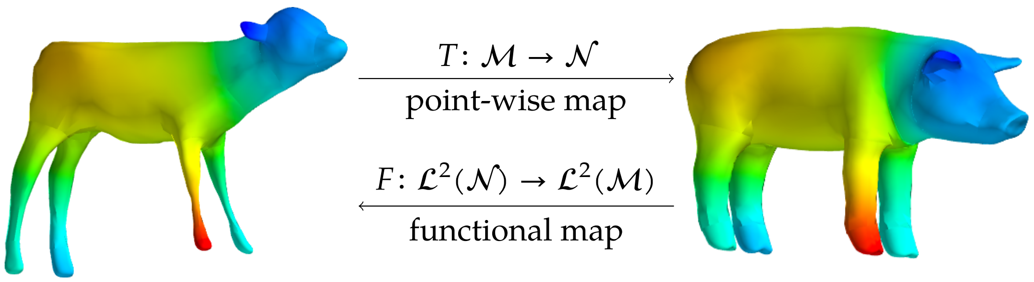

$T\colon \mathcal{M} \to \mathcal{N}$
point-wise map
$F\colon \mathcal{L}^2(\mathcal{N}) \to \mathcal{L}^2(\mathcal{M})$
functional map

Figure 2.7.: Some function f on $\mathcal{N}$ is mapped onto $\mathcal{M}$ (red region is maximal).

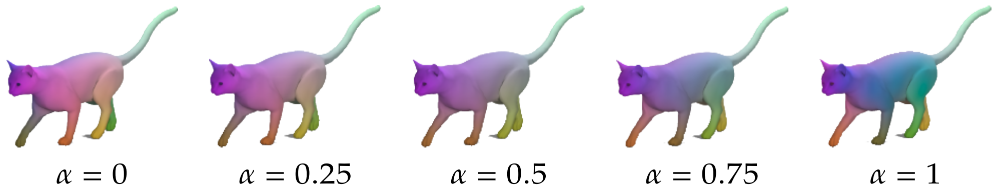


Figure 2.8.: Interpolating two functional maps using $(1-\alpha)F_1 + \alpha F_2$ and mapping a scalar field. F_1 is the identity map, F_2 maps onto the intrinsic symmetry, i.e. flipping the left and the right side (from [Ovs+12]).

whose target and source shape match, is well-defined, so that joint optimization of a collection of maps for a ensemble of shapes is possible [Hua+12; HWG14].

Solving Solving for a functional map usually involves *functional constraints*. These constrain the image of a function $f \in \mathcal{L}^2(\mathcal{N})$ onto another function $f \in \mathcal{L}^2(\mathcal{M})$:

$$F(f) = h \qquad f \in \mathcal{L}^2(\mathcal{N}), h \in \mathcal{L}^2(\mathcal{M}) \tag{2.42}$$

Examples of functional constraints are matching indicator functions of point-to-point constraints or matching heat-kernel signatures for isometric maps. Functional constraints are *linear* in the functional map F. When solving for a functional maps, functional constraints can be included as hard constraints, for example using Lagrange multipliers. Usually functional constraints are not exact but approximative and they are minimized in a least squares sense.

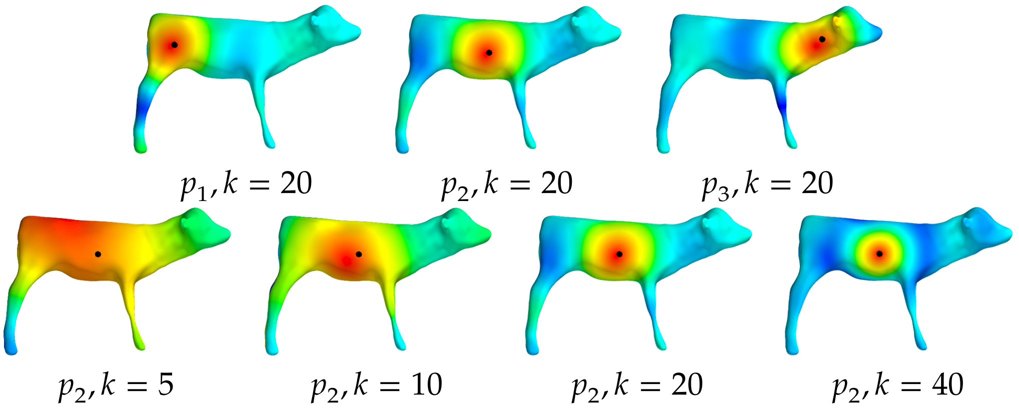
Two linear, functional operators $O_{\mathcal{M}}\colon \mathcal{L}^2(\mathcal{M}) \to \mathcal{L}^2(\mathcal{M})$ and $O_{\mathcal{N}}\colon \mathcal{L}^2(\mathcal{N}) \to \mathcal{L}^2(\mathcal{N})$ commute with the functional map F iff application of $O_{\mathcal{N}}$ follows by a projection with F is equivalent with application of $O_{\mathcal{M}}$ after a projection with F. Two operators commuting with F lead to the linear constraint:

$$O_{\mathcal{M}}F = FO_{\mathcal{N}} \tag{2.43}$$

For example, on isometric shapes the Laplace operators commute with F: $\Delta_{\mathcal{M}}F = F\Delta_{\mathcal{N}}$ (Eq. 2.35). Another example are operators mapping onto intrinsic symmetries, such as mapping the cat of Figure 2.8 onto itself while changing the left and right sides.

For the functional constraints $(f_1, g_1), \ldots, (f_s, g_s) \in \mathcal{L}^2(\mathcal{N}) \times \mathcal{L}^2(\mathcal{M})$ and commutativity with the two operators $(\Delta_{\mathcal{M}}, \Delta_{\mathcal{N}})$, the optimal functional map in the least squares sense is:

$$\min_{F\colon \mathcal{L}^2(\mathcal{N}) \to \mathcal{L}^2(\mathcal{M})} \sum_i \|F(f_i) - g_i\|_2^2 \quad + \|F\Delta_{\mathcal{N}} - \Delta_{\mathcal{M}}F\|_F^2 \tag{2.44}$$



$p_1, k = 20$ $p_2, k = 20$ $p_3, k = 20$

$p_2, k = 5$ $p_2, k = 10$ $p_2, k = 20$ $p_2, k = 40$

Figure 2.9.: Projection of delta-distributions, located at the points p_1, p_2, p_3, into the basis of the first k eigenvectors.

2.4.1. Representation of points

Point-to-point constraints are often included and point-to-point correspondences are often extracted by identification of points with primal or dual delta-distributions. For a point p on $\mathcal{M}$ the dual delta-distribution is the linear functional:

$$D_p^{\mathcal{M}} \colon C^\infty(\mathcal{M}) \to \mathbb{R}, \quad D_p^{\mathcal{M}}(f) \mapsto f(p)$$

For a linear subspace $\mathcal{T} \subset \mathcal{L}^2(\mathcal{M})$ and a point p the function $\delta_p \in \mathcal{T}$ is a delta-distribution if and only if:

$$\langle \delta_p, f \rangle = D_p(f) = f(p) \quad \forall f \in \mathcal{T} \tag{2.45}$$

For $\mathcal{T} = \mathcal{L}^2(\mathcal{M})$ such a function $\delta_p \in \mathcal{L}^2(\mathcal{M})$ does not exist, as it can be shown to be 0 almost everywhere except for p where it is unbounded. For a finite dimensional subspace spanned by the orthonormal basis $\phi_1, \dots, \phi_k$ delta-distributions the delta-distributions are well-defined (Riesz representation theorem):

$$\delta_p = \sum \phi_i(p)\phi_i \tag{2.46}$$

Proof. Let $f = \sum_j \alpha_j \phi_j$ then $\langle \delta_p, f \rangle = \langle \delta_p, \sum_j \alpha_j \phi_j \rangle = \sum_j \alpha_j \phi_j(p) = f(p)$ □

Figure 2.9 shows some delta-distributions for different points and subsets spanned by the Laplace eigenvectors.

The delta-distributions and dual-delta-distributions are related by the pullback functional map F_T:

$$D_p^{\mathcal{M}} \circ F_T = D_{T(p)}^{\mathcal{N}} \tag{2.47}$$

$$F_T^T \delta_p^{\mathcal{M}} = \delta_{T(p)}^{\mathcal{N}} \tag{2.48}$$

Proof. $D_p^{\mathcal{M}} \circ F_T(f) = F_T(f)(p) = f(T(p)) = D_{T(p)}^{\mathcal{N}}(f)$. Because $D_p^{\mathcal{M}}(f) = \langle \delta_p^{\mathcal{M}}, f \rangle_{\mathcal{M}}$ we have $\langle \delta_p^{\mathcal{M}}, F_T(f) \rangle_{\mathcal{M}} = D_p^{\mathcal{M}} \circ F_T(f) = D_{T(p)}^{\mathcal{N}}(f) = \langle \delta_{T(p)}^{\mathcal{N}}, f \rangle_{\mathcal{N}}$. □

Furthermore, area-preserving point-wise functional maps $F_T^{-1} = F_T^T$ map delta-- distributions: [1]

$$\delta_p^{\mathcal{M}} = F_T \delta_{T(p)}^{\mathcal{N}}$$

The above observations allow building functional constraints from the point-to-point constraints $(c_1, d_1), \dots, (c_s, d_s) \in \mathcal{M} \times \mathcal{N}$ by representing points with dual-delta-distributions:

$$\sum_i \|F^T \delta_{d_i} - \delta_{c_i}\|_2^2 \tag{2.49}$$

And for area-preserving maps additionally with delta-distributions:

$$\sum_i \|F \delta_{c_i} - \delta_{d_i}\|_2^2 \tag{2.50}$$

Extracting a correspondence map T from a functional map is possible by projecting delta-distributions and determining the most similar delta-distribution on the other shape:

$$T(p) = \arg\min_{q \in \mathcal{N}} \|F^T \delta_q - \delta_p\|_2 \tag{2.51}$$

And for area-preserving maps additional with:

$$T(p) = \arg\min_{q \in \mathcal{N}} \|F \delta_p - \delta_q\|_2 \tag{2.52}$$

This only works if L_2 distances on delta-distributions provide a reasonable distance measure on the shape. Indeed, if represented with the first k eigenvectors of the

[1] Various publications assume Eq. 2.49 without restricting themselves to area-preserving maps, e.g. [Ovs+12](Section 6.1), but interestingly result in good maps for area-distorting shapes as well. One reason might be that often area changes are small. Another reason is that those methods solve for F_T^T instead of F_T.

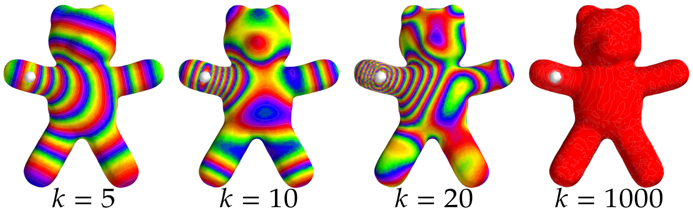


Figure 2.10.: L_2 distances of delta-distributions projected into a basis of k eigenvectors.

Laplace operator L_2 distances on delta-distributions measure intrinsic, articulation invariant distances in a small neighborhood as Figure 2.10 shows.

Functional maps provide an intrinsic shape matching in the following sense. If the constraints on the functional map are intrinsic, i.e. only depend on the distances on the shape but not on its embedding and articulation, then calculating a functional map representing and extracting points with delta-distributions is intrinsic and does not depend on the articulation of the shapes.

2.4.2. Choice of basis

Computation of functional maps and representation of functions and delta-distributions requires a functional basis on both shapes. The linear functional map is represented over the bases $\phi_1^{\mathcal{M}}, \dots, \phi_k^{\mathcal{M}}$ and $\phi_1^{\mathcal{N}}, \dots, \phi_k^{\mathcal{N}}$ by the matrix $\mathbf{F} \in \mathbb{R}^{k \times k}$:

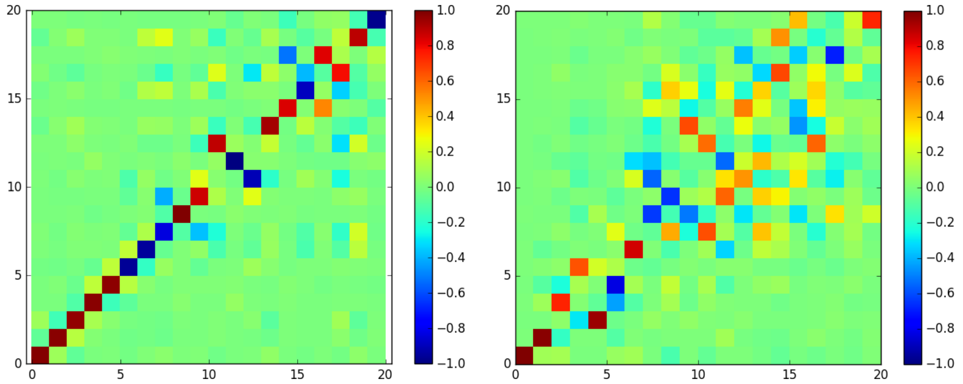
$$\mathbf{F}_{ij} = \langle \phi_i^{\mathcal{M}}, \phi_j^{\mathcal{N}} \circ T \rangle_{\mathcal{M}} \tag{2.53}$$

The resulting map and the cost to compute it depend on the choice of the basis:

- The basis should be small as it determines the size of the equation system to solve.
- L_2 distances on delta-distributions should approximate intrinsic distances.
- Functional constraints should be representable in the basis.
- The basis should span similar function spaces on two near-isometric shapes.

Furthermore, calculation of functional maps and the resulting point-wise map is most stable when working with smooth functions, i.e. functions of small Dirichlet energy. Smooth functions vary little when moving along the surface. Therefore, the image and preimage of smooth functions are similar when disturbing T by moving correspondences by small distances on the surface.

The eigenvectors of the Laplace operator are the most prominent choice for the function basis. They are one of the smallest basis to represent smooth functions and



isometric (cats) near-isometric (fourlegged)

Figure 2.11.: Matrix representations of two functional map.

L_2 distances of delta-distributions in the eigenbasis approximate intrinsic distances in a small neighborhood.

For the shapes of 2.7 the matrix representation of the functional map is given in Figure 2.11. The matrix has several properties:

- The first eigenvector is the constant function. Thus, $\mathbf{F}_{i,1} = 0$ for $i > 1$ as constant functions map onto the constant functions. Furthermore, $\mathbf{F}_{1,i}$ are the mean values of $\phi_i^{\mathcal{N}} \circ T$ on $\mathcal{M}$.
- For near-isometric maps the off-diagonal entries tend to be small due to commutativity of F with the Laplace operator.

The basis of the Laplace operator is stable against local changes in the diffusion weights and depends most on the global diffusion properties. Therefore, also generalizations of the Laplace operator to point clouds give similar good results, e.g. Eq. 2.21.

2.4.3. Convex relaxation

Solving for a functional map via Eq. (2.44) might seem like a silver bullet: despite intrinsic ambivalences in shape matching it solves for a functional map. It is no silver bullet and those ambivalences are shifted into the calculation of a point-wise map from a functional map, which we further illuminate.

The point-wise induced functional maps F_T are only a fraction of the functional maps and all point-wise induced functional maps fulfill the equation:

$$\forall p \in \mathcal{M}\ \exists q_p \in \mathcal{N}\colon F^T \delta_p^{\mathcal{M}} = \delta_{q_p}^{\mathcal{N}} \tag{2.54}$$

On the other hand, Eq. (2.54) defines a point-wise map T from the functional map F by mapping p onto q_p. And because (2.48) uniquely determines a functional map we have $F = F_T$:

Theorem 2. *Let F be a functional map then the following statements are equivalent:*

(i) F aligns dual delta-distributions by Eq. (2.54).
(ii) There is a point-wise map T such that $F = F_T$.

Therefore, solving for a point-wise induced functional map F_T from the functional constraints $(f_1, g_1), \ldots, (f_s, g_s) \in \mathcal{L}^2(\mathcal{N}) \times \mathcal{L}^2(\mathcal{M})$ and the commuting operators $\mathcal{O}_{\mathcal{M}}$ and $\mathcal{O}_{\mathcal{N}}$ amount to the non-convex optimization:

$$\begin{aligned}\text{(FCT-C)} \quad & \min_{F\colon \mathcal{L}^2(\mathcal{N})\to\mathcal{L}^2(\mathcal{M})} && \sum_i \|F(f_i) - g_i\|_2^2 \quad + \ \|F\Delta_{\mathcal{N}} - \Delta_{\mathcal{M}} F\|_F^2 \\ & && \text{s.t.} \quad \forall p \in \mathcal{M}\ \exists q_p \in \mathcal{N}\colon F^T \delta_p^{\mathcal{M}} = \delta_{q_p}^{\mathcal{N}}\end{aligned}$$

Theorem 3. *Least-squares solving for a function map by Eq. 2.44 is a convex relaxation of solving for a point-wise functional map (FCT-C).*

2.4.4. Equiareal, conformal and isometric maps

Further prior knowledge about the point-wise map T simplifies solving for its functional map F_T. For example if T is area-preserving, the functional map is *orthogonal* (Eq. (2.26)):

$$T \text{ area-preserving} \quad \Leftrightarrow \quad F_T^T = F_T^{-1} \tag{2.55}$$

This allows an important refinement process for functional maps by alignment of (dual-)delta-distributions with the *rigid ICP alignment* described in section 2.3.1. Furthermore, section 5 shows how to utilize that a conformal map T pulls the Laplace operator from $\mathcal{M}$ onto $\mathcal{N}$ (Eq. (2.33)):

$$T \text{ conformal} \quad \Leftrightarrow \quad \Delta_{\mathcal{N}} = F_T^T \Delta_{\mathcal{M}} F_T$$

Constructing isometries between isometric shapes is an important and common shape matching task. Isometries preserve the intrinsic structure and intrinsic quantities of the shape making the creation of such maps especially effective. The functional map F_T of an isometry is orthogonal, as T is area-preserving, and pulls Laplace operator from $\mathcal{M}$ onto $\mathcal{N}$, as T is conformal. Combined both properties result in the Laplace operator commuting with F_T: $\Delta_{\mathcal{M}} F_T = F_T \Delta_{\mathcal{N}}$. All together these properties are powerful aids for the construction of functional maps.

Even without aligning the shapes with functional maps the intrinsic metric can effectively limit the potential matches of any point to a small set by deriving and

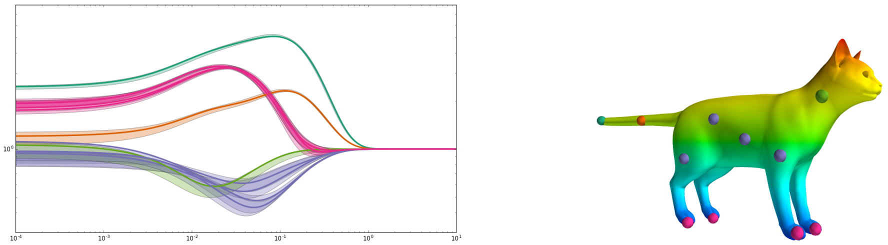


Figure 2.12.: Heat-kernel-signatures of highlighted points on the cat shape (thick-lines), as well as the minimum/maximum of the heat-kernel-signatures of corresponding points on all tosca cat models (colored bands).

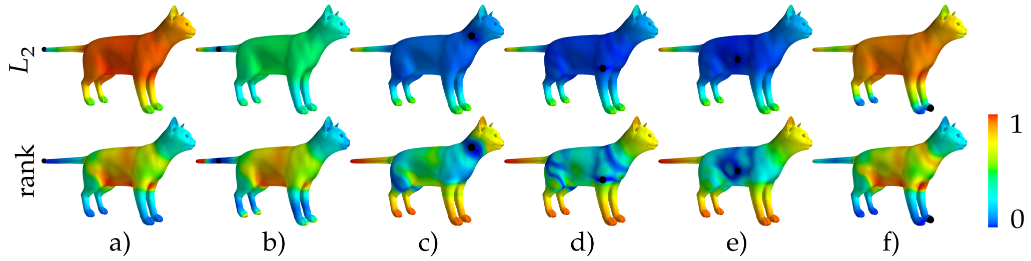


Figure 2.13.: L_2 distance of the heat-kernel-signatures from the selected points (top) and shapes colored by the fraction of area with smaller distances (bottom).

comparing point signatures based on diffusion processes. To this extend the heat-kernel-signature (hks) [SOG09] compares two points $p, q \in \mathcal{M}$ by the weighted L_2 distance of the diagonals of the heat-kernels $h_t(x, y)$ in a time interval $[a, b]$:

$$\mathrm{hks}_p \colon \mathbb{R}^+ \to \mathbb{R}, \quad t \mapsto \frac{h_t(p,p)}{\int_{\mathcal{M}} h_t(x,x)\; dx} = \frac{\sum_i \exp(-t\lambda_i)\phi_i^2(p)}{\sum_i \exp(-t\lambda_i)} \tag{2.56}$$

$$d^2_{\mathrm{hks},[a,b]}(p,q) := \int_0^1 \left(\mathrm{hks}_p(t) - \mathrm{hks}_q(t)\right)^2 \, d\tau \qquad \text{with } t = a^{1-\tau}b^{\tau} \tag{2.57}$$

Figure. 2.12 depicts the heat-kernel-signatures (hks) of several points on the cat. For large times the diffusion process converged to an equilibrium and all descriptors are equal. The number of eigenvalues used to calculate the signatures decides the smallest times that can be calculated before the signature converges to a constant value related to the Gaussian curvature.

The figure shows how similar descriptors of intrinsically equivalent regions are, such as the feet and body. Furthermore, it shows that the descriptors can reliably calculated over isometries as the colored bands around each descriptor is between the minimum and maximum of the signature of corresponding points on all tosca cat models.

Figure. 2.13 shows how to retrieve points by their signature. The top shows the L_2 signature distance of the marked point to all other points. The magnitude of L_2 distances varies strongly and more interesting than the distance might be how points are sorted by similarity as shown on the bottom. As the descriptors intrinsically similar regions have similar descriptors these queries by descriptor return intrinsic similar regions first, such as in f) where all feet are returned equally.

Visualizing the Laplace operator gives a better understanding how isometry and the commutativity with the Laplace operator is so helpful for the creation of maps. The Laplace operator can be visualized by the following ellipsoid (Δ is positive-semidefinite):

$$\{f \in \mathcal{L}^2(\mathcal{M}) \,|\, f^T \Delta f = 1\} = \left\{ \sum_i \alpha_i \phi_i \,\middle|\, \alpha_i \in \mathbb{R}, \sum_i \left(\frac{\alpha_i}{\sqrt{\lambda_i}} \right)^2 = 1 \right\} \tag{2.58}$$

Figure 2.14a shows instances of the cat with the above ellipsoid. The functional map of an isometry aligns the delta-distributions embeddings *as well as* the ellipsoid with a *rotation*. The interplay of the ellipse with the embeddings is used in the computation of heat-kernel-signatures. Especially for the first eigenvectors the alignment of the ellipses strongly restricts the possible functional maps. The rest of the figure shows how well such an alignment is possible for the tosca cats, which are mostly isometric apart from small local distortions due to bending. The sign of the eigenvectors in the eigendecomposition is arbitrary as shown in b) and corrected in c). As the relative difference of eigenvalues (and thus the length of the ellipsoid axis) decreases, the eigenspaces of larger eigenvalues are increasingly rotated into each other. How well the alignment with rotations is possible in d) is one reason for the good correspondences achievable by functional maps on isometric shapes.

Furthermore, the smallest Laplace eigenvectors correspond to the (orthogonal) directions in which the ellipsoid has the largest extents. The first eigenvectors on both shapes therefore span function spaces that correspond in an isometric map.

Typically, the computation of an isometry in the functional setting is as follows. Matching heat-kernel-signatures for equally spaced values τ are usually used as functional constraints. From these, Laplace operator commutativity, as well as additional functional constraints the initial functional map is computed. Note that neither the Laplace commutativity nor the functional constraints from the heat-kernel-signature can decide an intrinsic symmetry, such as the cat which can be mapped onto itself switching the left and the right. For those shapes usually additional functional constraints are required. The functional map obtained in this way is then refined into a point-wise map via rigid ICP alignment of the delta-distributions described in section 2.3.1.

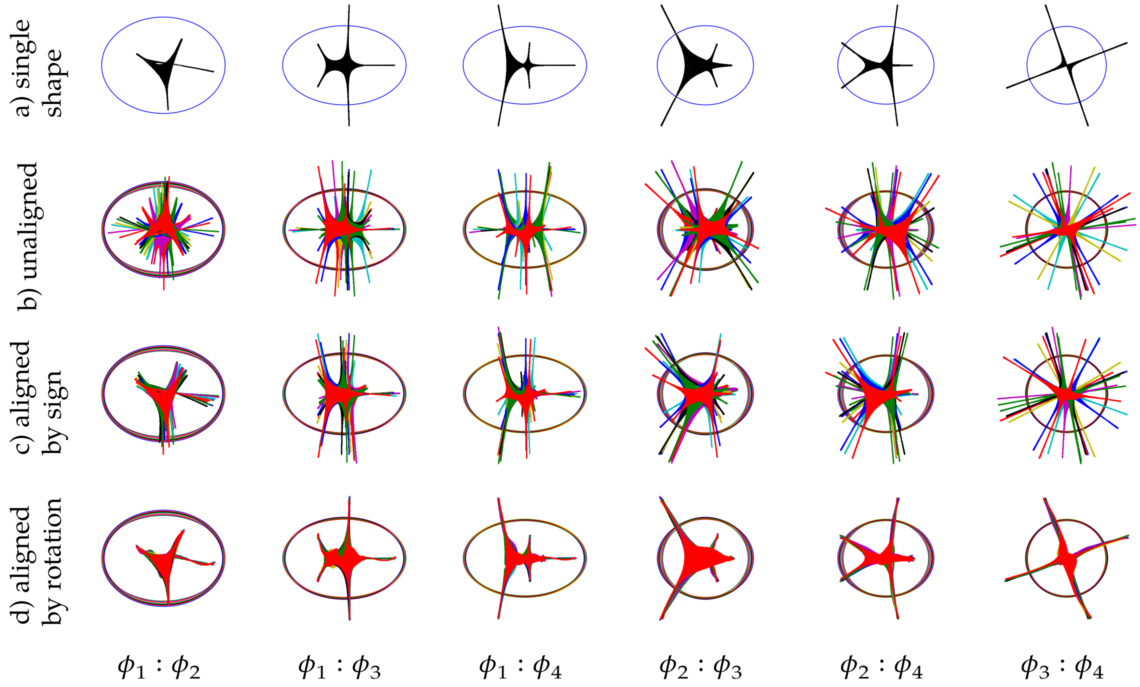


Figure 2.14.: Embedding the cats of Figure 2.3 with delta-distributions, either without alignment or after aligning with a rotation.

2.4.5. Summary

In summary functional maps allow alignment of non-rigid deformations with a linear map. This is done with a relaxation of the shape matching problem to the functional maps and deferring the extraction of the point-wise map. The linear map is found as the minimum of a quadratic energy solvable via the normal equations. The formulation is intrinsic independent of the articulation of the embeddings, as long as the used constraints are, and there is a rich theory on isometric maps. The eigenbasis of the Laplace operator allows a multi-scale representation, where the lower eigenvectors encode the global behavior of the maps. Smooth functions have a prominent role as they encapsulate the global behavior and are stable under small movements of the correspondences. On the other hand, there is *no* guarantee that the obtained functional map F is close to a point-wise induced functional map F_T and extraction of the point-wise correspondences might fail. Inclusion of differential properties of the map, such as area, angle or length distortion are difficult to include in the optimization.

2.5. Assignment problems

Previously presented methods compute point-wise maps by first aligning shapes (extrinsically or intrinsically) followed by mapping points with nearest neighbor searches (Eq. (2.36), (2.51), (2.52)). Mapping points *independently* from each other introduces artifacts, such as mapping multiple points onto one and not mapping onto a region of the target at all. Solving for a one-to-one assignment of two point sets, one on each shape, avoids these artifacts.

Here an underlying assumption is, that we can generate two point sets which are in one-to-one correspondence. Interestingly this is often feasible. For near-isometric shapes the correspondences map often is approximately area-preserving after a global scaling, such that sampling equally distributed points results in such sets. Otherwise, more involved heuristics based on feature point detection can be used.

2.5.1. Linear assignment problems

Minimizing a cost function over the one-to-one correspondences of two point sets is called an *assignment problem*. If furthermore the cost function summarizes terms depending only on single point-to-point assignments the problem is called a *linear assignment problems*. After writing S_n for the symmetric group, which contains all n-element permutations, and $\mathbf{D}_{i,j}$ with $\mathbf{D} \in \mathbb{R}^{n \times n}$ for the cost of assigning the i-th onto the j-th point, the linear assignment problem minimizes:

$$\text{(LAP-}\phi\text{)} \quad \min_{\phi \in S_n} \sum_i \mathbf{D}_{i,\phi(i)}$$

Every permutation ϕ can be represented by a permutation matrix $\mathbf{X}^\phi$, where $\mathbf{X}^\phi_{ij} = 1$ if $\phi(i) = j$ and 0 otherwise. We can therefore formulate (LAP-ϕ) with a linear objective over the permutation matrices:

$$\text{(LAP-OPT)} \quad \min_{\mathbf{X} \in \mathbb{R}^{n \times n}} \langle \mathbf{X}, \mathbf{D} \rangle \qquad \text{s.t.} \quad \mathbf{X} \in \{\mathbf{X}^\phi \mid \phi \in S_n\}$$

Interestingly, the minimal cost of a linear cost function under non-convex constraints can be found with a linear programs:

Theorem 4. *The minimal cost of a linear function f under non-convex constraints equals the minimal cost of a linear program over f, whose solutions are constrained to the convex hull of the original problem's feasibility set.*

Proof. Let x be the solution of the original problem and y be the solution of the linear program, which can be written as a convex combination of feasible solutions x_i of the original problem: $y = \sum_i \alpha_i x_i$. As any solution of the original problem is also a feasible solution of the linear program, we have $f(y) \leq f(x) \leq \min_i f(x_i)$. Due to the linearity of f we also have $f(y) = \sum_i \alpha_i f(x_i)$ and thus $\min_i f(x_i) \leq f(y)$. □

The convex hull of the permutation matrices $\{\mathbf{X}^\phi \mid \phi \in S_n\}$ is known as the *Birkhoff polytope* Π^n [AM14] and we can efficiently compute the minimum of (LAP-ϕ) with the linear program:

$$\text{(LAP-R)} \quad \min_{\mathbf{X} \in \mathbb{R}^{n \times n}} \langle \mathbf{X}, \mathbf{D} \rangle \qquad \text{s.t.} \quad \mathbf{X} \in \Pi^n$$

where

$$\Pi^n := \operatorname{conv}\big(\{\mathbf{X}^\phi \mid \phi \in S_n\}\big) = \{\mathbf{X} \in \mathbb{R}^{n \times n} \mid 0 \leq \mathbf{X},\ \mathbf{X}\mathbf{1} = \mathbf{1},\ \mathbf{X}^T\mathbf{1} = \mathbf{1}\}\,.$$

Figure 2.15 shows several pairs of aligned shapes, each sampled with a point set, and the correspondences computed using either nearest neighbor matching or linear assignment problems. If there is a good one-to-one assignment of the point sets (a,b), linear assignment problems usually yield much better correspondences. Figure 2.16 recalculates the extrinsic shape alignments of Figure 2.3.2 using linear assignment problems instead of nearest neighbor searches to match points.

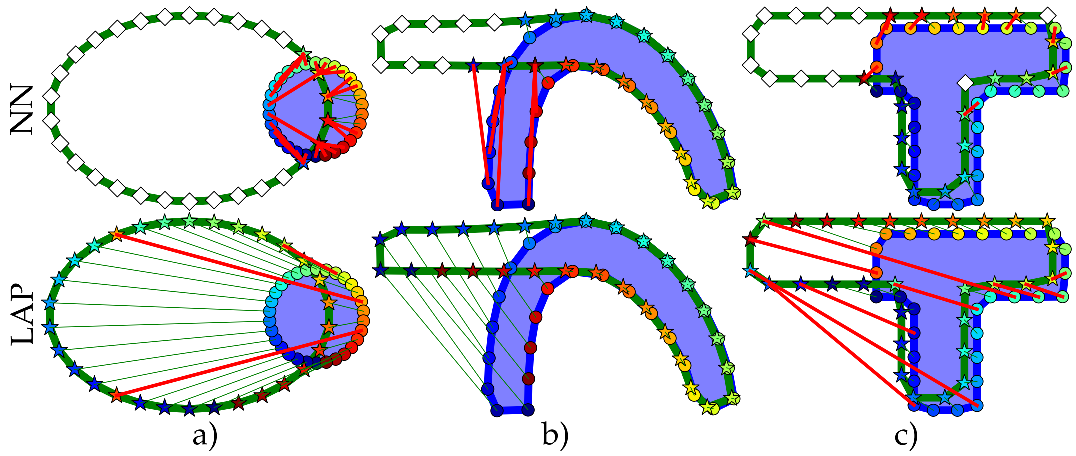


Figure 2.15.: Linear assignment problems (LAP) yields better correspondences than nearest-neighbor queries for relative area-preserving maps (a,b).

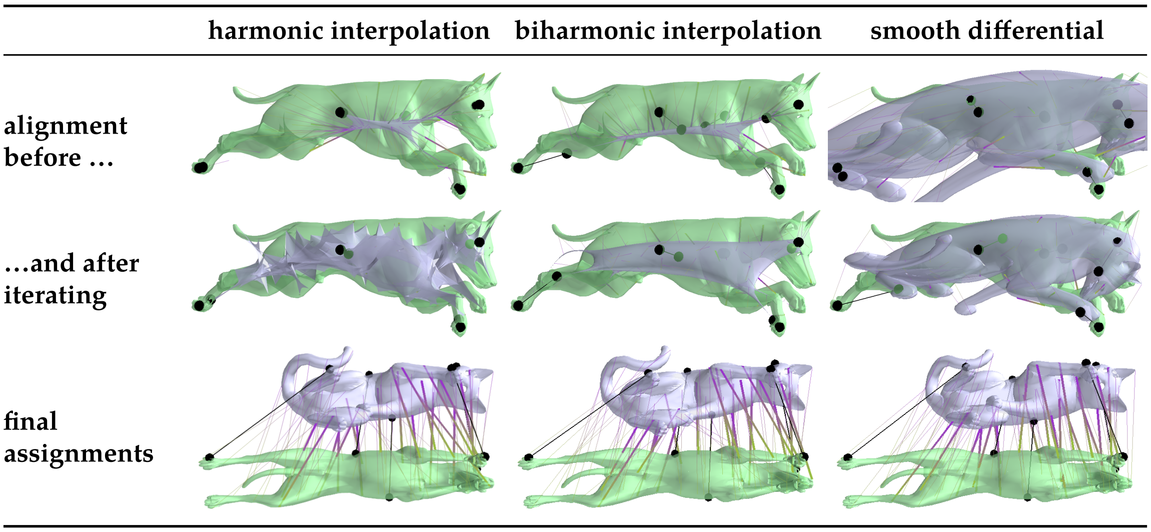


Figure 2.16.: A practical example showing the correspondences computed with the same deformation models as in Figure 2.3.2, but with linear programming instead of nearest neighbor search.

2.5.2. Quadratic assignment problems

Good correspondences induce little stretch, that is mapping an edge results in an edge of similar geodesic length. A cost function based on stretch therefore summarizes terms depending on the assignment of point-*pairs*. An assignment problem minimizing such a cost function is called a *quadratic assignment problem*. After writing $\mathbf{A}_{i,\phi(i),j,\phi(j)}$ for the cost of assigning the pair (i,j) onto $(\phi(i),\phi(j))$, an instance of the quadratic assignment problem can be then be written as:

$$\text{(QAP-}\phi\text{)} \quad \min_{\phi\in S_n} \sum_{ij} \mathbf{A}_{i,\phi(i),j,\phi(j)}$$

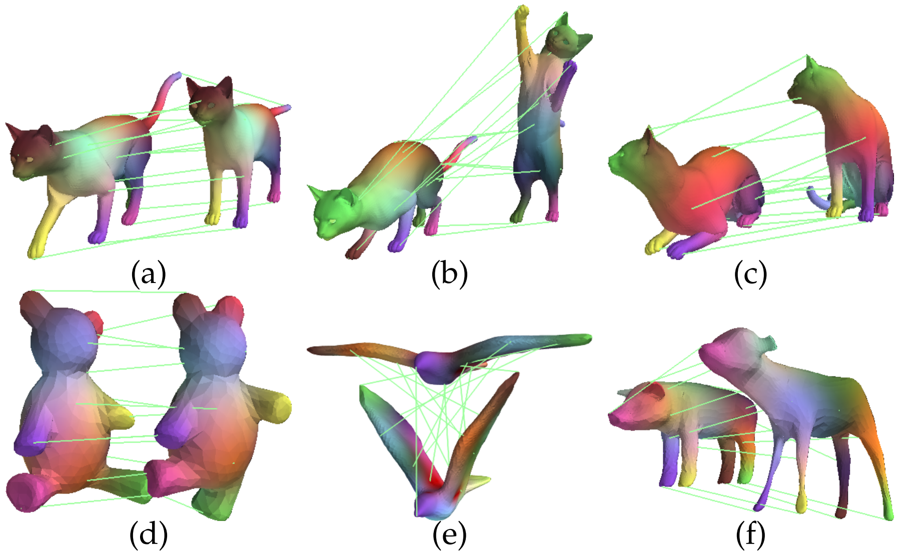
Various authors [BBM05; FS06; Kez+15b] modeled the computation of correspondences with quadratic assignment problems. For example, Kezurer et al. [Kez+15b] minimizing the following stretch based cost function on a few sampled feature points to determine correspondences ($d_{ij} \in \mathbb{R}$ and $d'_{ij} \in \mathbb{R}$ are pairwise geodesic distances of the feature points on both shapes and $\sigma \in \mathbb{R}$ is a parameter):

$$\min_{\phi\in S_n} \sum_{ij} -\exp\left(-\ (d_{ij} - d'_{\phi(i)\phi(j)})^2/\sigma^2\right) \tag{2.59}$$

Figure 2.17 depicts several computed correspondences.

Solving quadratic assignment problems is NP-hard with a computational complexity growing exponentially with the number of points to match. Much research [Bur+98] therefore went into approximating good solutions. To this end, the theory of convex relaxations, where non-convex constraints are approximated by convex ones, proved especially fruitful. Section 6 will give a detailed introduction into current state-of-the-art approximations of quadratic assignment problems via convex relaxations.

Despite its complexity, the major advantage of shape matching formulations using quadratic assignment problems is that they often yield globally good solutions. These methods therefore do not depend on any prior information, such as prior correspondences or prior functional constraints. This is an important difference to previously discussed alignment methods, which are local optimizations of *non-convex* cost functionals.



(a) (b) (c)

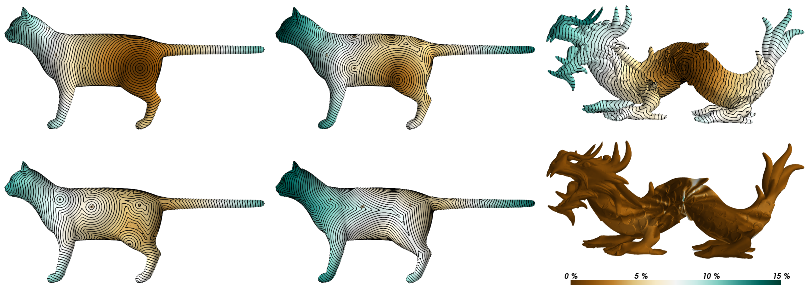
(d) (e) (f)

Figure 2.17.: Correspondences compute by solving the QAP defined by Eq. 2.59.

3 Simple, robust, constant-time bounds on geodesic distances using point landmarks

Abstract In this paper we exploit redundant information in geodesic distance fields for a quick approximation of all-pair distances. Starting with geodesic distance fields of equally distributed landmarks we analyze the lower and upper bound resulting from the triangle inequality and show that both bounds converge reasonably fast to the original distance field. The lower bound has itself a bounded relative error, fulfills the triangle equation and under mild conditions is a distance metric. While the absolute error of both bounds is smaller than the maximal landmark distances, the upper bound often exhibits smaller error close to the cut locus. Both the lower and upper bound are simple to implement and quickly to evaluate with a constant-time effort for point-to-point distances, which are often required by various algorithms.

This chapter corresponds to the paper[BK15b]: Oliver Burghard and Reinhard Klein. "Simple, Robust, Constant-Time Bounds on Surface Geodesic Distances using Point Landmarks". In: *Vision, Modeling & Visualization*. 2015.



a) lower bound
front & back
(30 landmarks)

b) upper bound
front & back
(30 landmarks)

d) lower bound
& relative error
(100 landmarks)

Figure 3.1.: (teaser) Lower and upper bounds on two shapes for 30 and 100 landmark points.

3.1. Introduction

Geodesic distances on surfaces are an important tool providing intrinsic information derived from the metric. Even though there has been much research in approximating geodesic distances their calculation can take a significant time in current processing.

To motivate our approximation notice that distance fields at different points typically share a lot of common information (see [XYH12] for a discussion and our later analysis). The triangle inequality estimates lower and upper distance bounds between two points based on distances to a third point. Starting with a reasonable set of landmark points we derive a lower and upper bound on all-pair geodesic distances. We analyze these bounds and show that they are accurate, simple to implement and efficient to compute.

The lower bound has itself a bounded relative error, so that it can be used as an approximation for geodesic distances. The upper bound often exhibits smaller absolute errors close to the cut locus, which makes them better suited for certain class of applications. And because the difference of lower and upper bound is limited by the maximal distance of landmark points, so is the absolute approximation error by both bounds.

Under the mild condition that no point has equal distances to all landmarks (which should not happen for more than 3 landmarks) the lower bound is a distance metric ($d_{min}(p,p) = 0$, triangle equation and $p \neq q \Rightarrow d_{min}(p,q) > 0$). As some efficient methods for calculating geodesic do not assure the triangle equation (e.g. [CWW13]), our lower bound might be used as an approximation instead.

Our distance fields are efficient to compute in the sense that there is a constant effort required for calculating bounds on the distance of two points. Often algorithms depend on pairwise distances only instead of global distance fields (e.g. Karcher means or Voronoi regions). Such algorithms will be typically much faster with our approach, then with global distance fields (see [CWW13; Xu+15]).

3.2. Related work

Calculating geodesic distance fields there are two different classes of algorithms, exact and approximate ones. Exact algorithms [MMP87; CH90; XW09; Xu+15] often utilize that single-source distances (on a piece-wise linear mesh) equal to a set of quadratic functions on an edge (called windows). Similar to Dijkstra on graphs, they distribute windows between triangles over adjacent edges. Best algorithms have a complexity of $O(n^2)$ [CH90; Xu+15], which is minimal [Xu+15] and thus their complexity is optimal.

Exact distances are often not critical for applications as long as errors are small. Indeed, frequently used piece-wise linear surfaces often are an approximation of a continuous surface themselves. Geodesic distances solve the Eikonal PDE: $\|\nabla d_p(x)\| = 1$, so that [KS98] approximate geodesics as solutions to this equation. Their approach hast a complexity of $O(n \log n))$. Predicting the geodesic gradient from heat diffusion [CWW13] speeds up approximation further. Ignoring a one-time matrix factorization, it reduces the complexity to $\mathcal{O}(n)$ per distance field, which is trivially optimal for an entire distance field.

Yet this complexity is not optimal when approximating point-to-point distances. Typically, applications require only few distances and not the whole distance field, e.g. calculating Karcher means [Kar77], intrinsic Voronoi regions [XW10] or non-rigid registration [Hua+08b]. After preprocessing a constant time approximation, i.e. not depending on the number of mesh vertices such as our algorithm, would be optimal.

A different set of algorithms has this constant complexity for point-to-point distances. They define intrinsic distances by embedding a manifold into some Euclidean space and back-projecting the distance metric [LRF10a; CL06; QH07]. This construction guarantees a distance metric with constant time point-to-point distances. Still, no embedding has been found so that distances are assured to approximate geodesic distances (indeed, exact preservation of geodesic distance is often impossible).

[XYH12] proposed a method for constant time geodesic distance approximation with a similar motivation such as our method. From equally spaced landmarks they construct a coarse intrinsic Delaunay triangulation on the surface with precomputed pair-wise distances on the landmarks. Mesh distances are then interpolated utilizing this triangulation by projecting quadrilaterals into the Euclidean plane, as side lengths and one diagonal are known.

Their method is similar to ours and delivers good approximations to the exact distances. Yet their approximations might be lead to less smooth, even non-continuous distance fields as they depend on the induced coarse triangulation (which changes non-continuously). They have no bounds on the approximation error and their approximated distance might not be a distance metric. See Sec. 3.4 for a comparison.

The chapter is structured as follows: First we introduce our lower and upper bounds on geodesic distances. Then we analysis their properties and error bounds. Finally, we show qualitative and quantitative evaluations.

3.3. Landmark induced distance bounds

To motivate our method notice that on a surface $\mathcal{M}$ there is much redundant information in the all-pair geodesic distances. For example, each distance field from p amounts to the information of all shortest paths starting in p. Therefore, with all-pair distances all geodesics, which shortest paths are subsets of, can be reconstructed and vice-versa.

A distance field is typically required or desired to adhere to the triangle inequality , that is the shortest path from p to q must get longer if we additionally require that it passes some other point r:

$$d(p,q) \leq d(p,r) + d(r,q) \tag{3.1}$$

Subtraction of $d(p,r)$ and changing q and r gives a reformulation with a *lower* and *upper* bound on distance $d(p,q)$ induced by a distance field from some landmark point r:

$$|d(p,r) - d(r,q)| \leq d(p,q) \leq d(p,r) + d(r,q) \tag{3.2}$$

Unifying bounds induced by a set of landmark points $R = \{r_1, \dots, r_k\}$ we gain the following bounds:

Definition 5. *The minimal and maximal induced distances of the landmarks R over the distance metric $d : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_0^+$ are:*

$$d_{min}(p,q) := \max_{r \in R} \ |d(r,p) - d(r,q)| \tag{3.3}$$

$$d_{max}(p,q) := \min_{r \in R} \ d(r,p) + d(r,q) \tag{3.4}$$

The actual landmark r that gave rise to a maximum of $d_{min}(p,q)$ or a minimum of $d_{max}(p,q)$ is called the inducing landmark. We add an upper r index to denote the distance w.r.t. a single landmark and will use this also on upcoming definitions: d^r_{min}, d^r_{max}.

For completeness, we quickly recapitulate properties defining a distance metric:

	pos	sym	ident	strict pos	tri. ineq.
d_{min}				mostly	
d_{max}			×	(mostly)	×

Figure 3.2.: Properties emerging from the definitions of d_{min} and d_{max} (see Theorem 7). Properties in brackets are valid for landmark distances that do not emerge from a proper distance metric (see Sec. 3.3.1).

Definition 6. *A pseudo distance metric is a map $d : \mathcal{M} \times \mathcal{M} \mapsto \mathbb{R}$ with $(\forall p, q, r \in \mathcal{M})$*

$$d(p,q) \geq 0 \qquad \text{(pos)} \tag{3.5}$$
$$d(p,q) = d(q,p) \qquad \text{(sym)} \tag{3.6}$$
$$d(p,p) = 0 \qquad \text{(id)} \tag{3.7}$$
$$d(p,q) \leq d(p,r) + d(r,q) \qquad \text{(tri. ineq.)} \tag{3.8}$$

A distance metric is a pseudo metric with $(\forall p, q \in \mathcal{M})$

$$d(p,q) > 0 \text{ if } p \neq q \qquad \text{(strict pos)} \tag{3.9}$$

As the next Theorem and Tab. 3.2 shows d_{min} and d_{max} share several desirable properties:

Theorem 7. *$d_{min}(p,q)$ is a pseudo distance metric and a proper distance metric iff there are no points with equal distances to all landmarks. d_{max} is symmetric and strictly positive. (Please find all proofs in the appendix)*

After defining the absolute approximation error as

$$e_{min}(p,q) := d(p,q) - d_{min}(p,q) > 0 \tag{3.10}$$
$$e_{max}(p,q) := d_{max}(p,q) - d(p,q) > 0 \tag{3.11}$$

there are the following relations:

$$d_{min}(r,\cdot) = d_{max}(r,\cdot) = d(r,\cdot) \tag{3.12}$$
$$e_{max}(p,q) + e_{min}(p,q) \leq \min_{r \in R, z \in \{p,q\}} d(z,r) \tag{3.13}$$
$$e_{min}(p,q) \leq e_{min}(p',q) + 2\,d(p,p') \tag{3.14}$$
$$e_{max}(p,q) \leq e_{max}(p',q) + 2\,d(p,p') \tag{3.15}$$

On the landmark points the approximation is exact (Eq. 3.12). Otherwise, it is bounded by the maximal inter landmark distance (Eq. 3.13), which assures that adding landmarks decreases the absolute errors. For example placing landmarks on

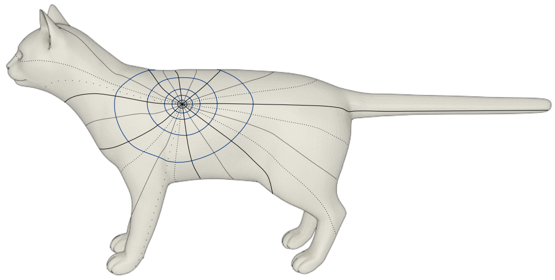


Figure 3.3.: Illustration of a few maximal shortest paths induced from a single landmark point.

a torus in a regular grid, n grid points will lead to a landmark distance in $\mathcal{O}(\sqrt{1/n})$. Thus, for twice the precision 4 times the landmarks are needed.

Eq. 3.14 and Eq. 3.15 allow limiting the error with distances to well approximated sets, which leads to much better convergence. Let $V_{min}(p) := \{q | \, d_{min}(p,q) = d(p,q)\}$ and $V_{max}(q) := \{q | \, d_{max}(p,q) = d(p,q)\}$ be the sets of points where d_{min} respective d_{max} are exact. Then the distance of either p or q bounds the absolute error:

$$e_{min}(p,q) \leq 2\, d(q, V_{min}(p)) \tag{3.16}$$

$$e_{max}(p,q) \leq 2\, d(q, V_{max}(p)) \tag{3.17}$$

Now for estimating the error one has to determine the set $V_{min}(p)$ and $V_{max}(p)$, i.e. all q where d_{min} or d_{max} are exact. Because $e_{min}(p,q) = \min_{r \in R} e^r_{min}(p,q)$ and analog $e_{max} = \min_{r \in R} e^r_{max}(p,q)$ determining $V^r_{min}(p)$ and $V^r_{max}(p)$ is sufficient (remember that upper indices are restrictions to single landmarks).

We assume that we have a smooth surface and for simplicity assume that there is a single shortest path between two points. For geodesic distances and some landmark r, the induced distance d^r_{min} is exact iff q is located on the shortest path of p and r, or p is located on the shortest path of q and r. d_{max} is exact iff r is located on the shortest path connecting p and q.

Shortest paths starting in r either intersect only in r or one is the subset of the other. This partial ordering gives rise to maximal shortest paths. A few of those maximal shortest paths are visualized in Fig. 3.3. Let R^+ denote the maximal shortest path starting in r including p, R^- be the opposite maximal shortest path located on the same geodesic. Similar let P be the union of maximal shortest path containing r and its opposite. Then:

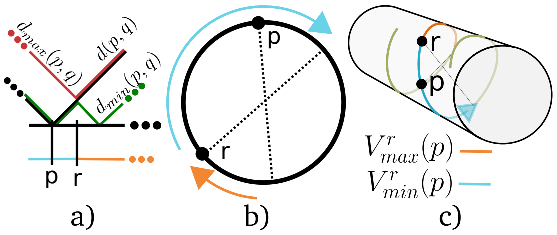


Figure 3.4.: a) The Euclidean line is split up by a single landmark between two regions - one where d_{min} is exact and one where d_{max} is exact. b) On the circle there due to topology influence there is a third region where none is exact. c) The geodesic spanned by r and p shown with same regions (V_{min}, V_{max}). As in (b) due to topology in parts of the geodesic neither is exact.

$$V^r_{min}(p) = R^+ \cap P = R^+ \qquad V_{min}(p) = \bigcup_r V^r_{min} \tag{3.18}$$

$$V^r_{max}(p) = R^- \cap P \qquad V_{max}(p) = \bigcup_r V^r_{max} \tag{3.19}$$

In Fig. 3.4 we see 3 different domains and fixed p and r. In the first example the Euclidean line is partitioned by V_{min} and V_{max} into two segments and for every point on the line, either d_{min} or d_{max} is exact. Any subset of a geodesic (the Euclidean line) is a shortest path. Top right we see a closed circle demonstrating the topological influence in comparison with the line. There is a region where neither d_{min} nor d_{max} is exact. Finally, bottom left we see a geodesic on a smooth surface showing topological and tangential error.

An illustration of V_{min} can be seen in Fig. 3.5. Equally, spread landmarks create curves quickly becoming dense everywhere. Errors get smaller, the closer p and q, which in our tests resulted in a bounded relative error of d_{min} as well. This dense field of lines leads to a decrease of absolute errors of d_{min} and d_{max} as well as the relative error of d_{min} ($e_{min}(p,q)/d(p,q)$).

Approximation errors can be classified into two categories. For a good approximation one needs a landmark r inducing maximal shortest paths P so that P is close to q. We call this first class of errors tangential errors. Additionally, r must be located on P in such a way that errors on P can be inferred and this second class of errors we call topological errors, as it does not appear in Euclidean domains. Moving a landmark r along P will change the topological error, moving r so that P changes, changes the tangential error.

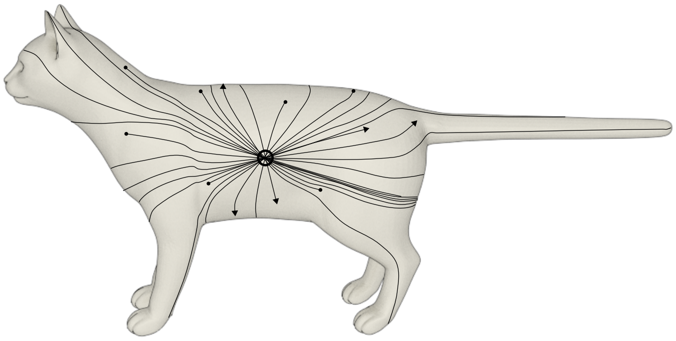


Figure 3.5.: Visualization of $V_{min}(p)$. For a single point p we collect the maximal induced shortest paths from all landmark points (exactly one curve for each landmark). The error $e_{min}(p,q)$ is bound by twice the distance of q to any of these curves. The curves of d_{min} have smaller lengths than the maximal shortest paths starting in p, what we described as the topological error (ends are marked with small arrows).

3.3.1. Arbitrary landmark distance fields

The input of the d_{min} and d_{max} is strictly speaking not a distance metric, but consists of $|R|$ different distance fields ($d(r,\cdot) := d_r(\cdot)$). For the given input, there might not exist a distance metric reconstructing input distances. This might be due to numerical errors or might be because the input distances were not derived from a distance metric in the first place. In the following we reason about effects on the bounds.

There are various reasons why arbitrary distance fields might not be compatible with any distance metric: Distances might not be symmetric ($d_r(r') \neq d_{r'}(r)$), triangle equation might not hold between two distance fields, distances might not be 0 at the landmarks ($d_r(r) \neq 0$), distance might be 0 elsewhere ($d_r(p) = 0, p \neq r$) or distance might be negative ($d_r(p) < 0$). Interestingly d_{min} will still be a (pseudo) distance metric:

Theorem 8. *Given arbitrary distance fields $d_r : \mathcal{M} \to \mathbb{R}$ with $d_r(r) = 0 \forall r \in R$ as input and define d_{min} and d_{max} as:*

$$d_{min}(p,q) := \max_{r \in R} |d_r(p) - d_r(q)| \tag{3.20}$$

$$d_{max}(p,q) := \min_{r \in R} |d_r(p)| + |d_r(q)| \tag{3.21}$$

Then most results of Theorem7 stay valid: $d_{min}(p,q)$ is a pseudo distance metric and a proper distance metric iff there are no points with equal distances to all landmarks. d_{max} is symmetric and positive.

On landmark points there is:

$$d_{max}(r,\cdot) \leq d_r(\cdot) \leq d_{min}(r,\cdot) \tag{3.22}$$

and for two points $p, q \in \mathcal{M}$ the inequality $d_{max}(p,q) < d_{min}(p,q)$ holds if and only if there exists two landmarks r_1,r_2 where the triangle inequality can not be fulfilled for p, q, r_1, r_2 (not necessarily pairwise different).

The reason why the exactness of d_{min} depends only on the triangle equation is simply that violations of identity, positivity and symmetry lead to triangle equation violations.

Let $d_{min}[d_r]$ and $d_{max}[d_r]$ denote the distance fields emerging from the distances d_r at the fixed landmarks. Then $d_{min}[d_{min}[d_r]]$ equals $d_{min}[d_r]$, because $d_{min}[d_r]$ is a distance metric, that will be exactly reproduced (Eq. 3.12). This is generally not true for $d_{max}[d_{max}[d_r]] = d_{max}[d_r]$.

As a simple example we inspect a triangle with edge lengths $1, 2, 4$, that violates the triangle inequality. All three vertices should be landmark points. Then d_{min} and d_{max} are $2, 3, 4$ and $1, 2, 3$ respectively. As guaranteed d_{min} adheres to the triangle inequality, but in this simple case also d_{max}. Because the triangle equation was initially violated for all edges we have $d_{max} \leq d_{min}$ everywhere.

There could be 'better' distance metric approximations. For example, we could define the optimal least squares approximation with a least squares energy

$$\text{argmin}_{d \text{ is a distance metric}} \sum_{r \in R} \int (d(r,x) - d_r(x))^2 dx$$

which defines a quadratic program. d_{min} is not optimal, but for our example above $1\frac{1}{3}, 2\frac{1}{3}, 3\frac{2}{3}$ would be.

3.4. Evaluation

To get a first qualitative idea of the bounds, Fig. 3.6 contains plots of d_{min} and d_{max} in a Euclidean plane (top) and on the torus (bottom). There are up to 4 landmarks (black points) and we infer bounds for the distances from the origin o to the plane. First observation is that errors are indeed bound by twice the distance from the origin to the closest landmark ($2 \cdot \sqrt{2}$ in our case) which follows from Eq. 3.13. The error is bound for each point by twice its distance to the closest position without error ($V_{min}(o)$ and $V_{max}(o)$). In the Euclidean plane the distances to $V_{min}(o)$ and e_{min} decrease quickly, which is not true for d_{max}. The torus additionally exhibits topological error, which leads to worse lower bounds d_{min}, but affects d_{max} less.

One interesting insight from the Euclidean case is that relative errors of d_{min}, i.e. e_{min}/d are bound (if there is at least one landmark). Let α_r be the minimal angle

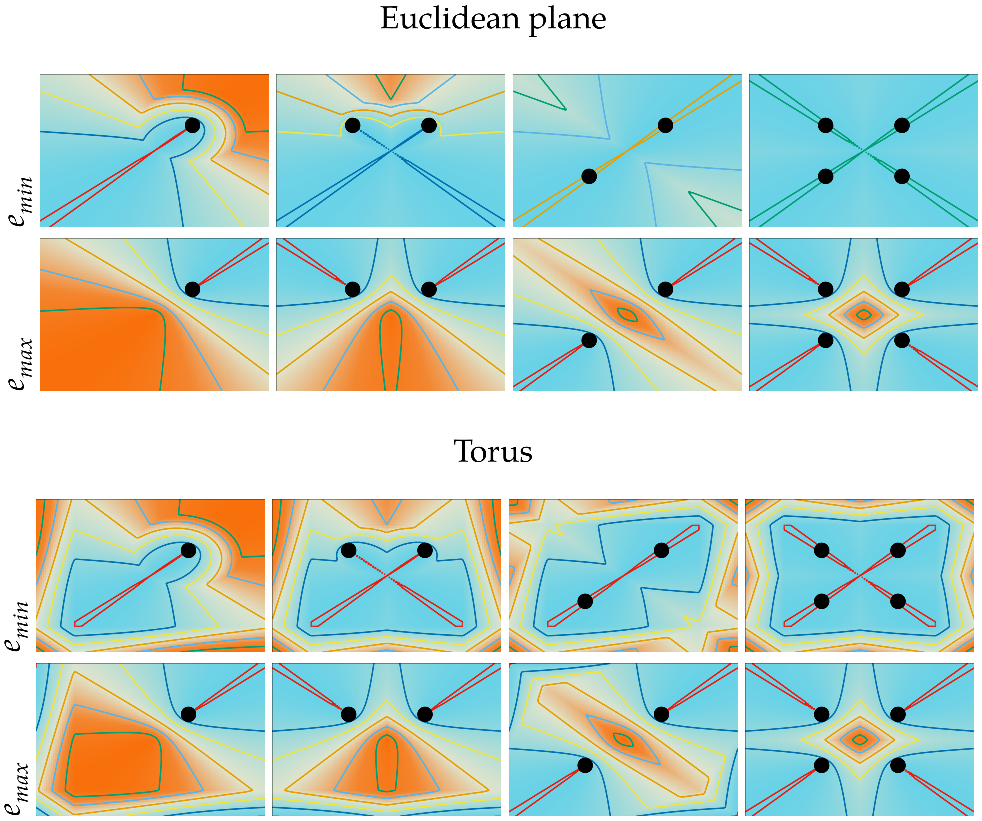


Figure 3.6.: e_{min} and e_{max} on different domains with up to 4 landmarks (black dots). Top rows depict an unbounded Euclidean space and the bottom rows a bounded torus (i.e. warping around left-to-right and top-to-bottom). Please see the text for further discussion. [Coordinates: -3 to 3, landmarks on a circle of radius $\sqrt{2}$, colors from 0 (light blue) to 3 (light brown) with 6 equal spaces contour lines].

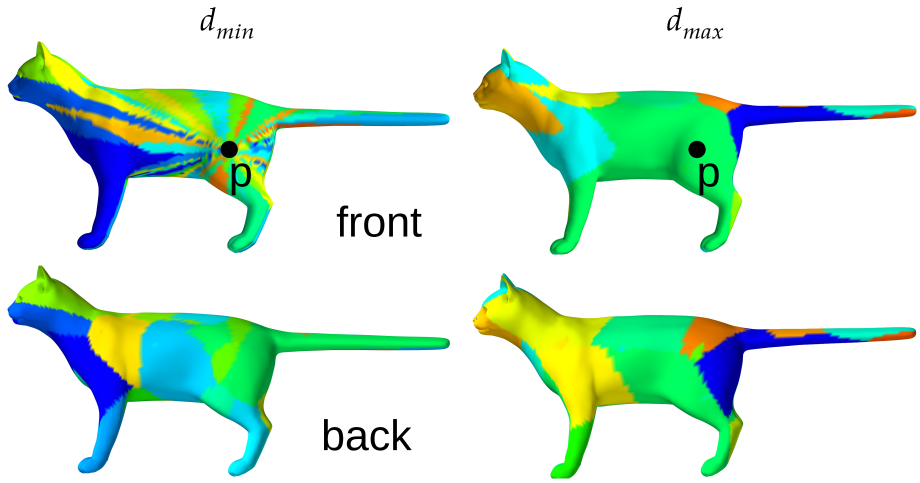


Figure 3.7.: Regions colored based on the landmark inducing distances to p. Regions should resemble Voronoi regions around lines of $V_{min}(p)$ and $V_{max}(p)$.

between the shortest path from p to q and some path in $V^r_{min}(p)$. Then

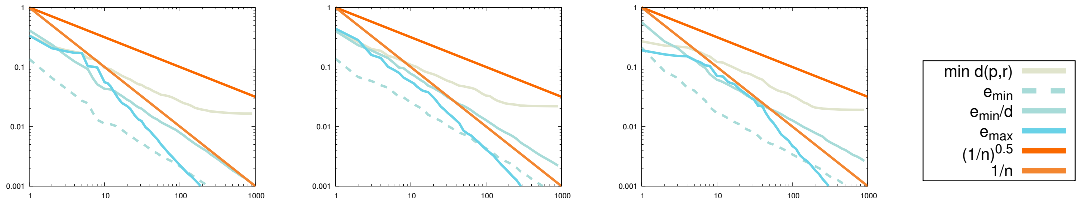
$$e_{min}(p,q)/d(p,q) \leq 1 - \cos\alpha_r$$

The same is true on a smooth manifold for some small neighborhood around p. But then the relative error is also globally bound. In our experiments the largest relative errors appeared locally, so that the finer the directions of the tangent space are sampled by shortest paths $V^r_{min}(p)$, the smaller is the maximal relative error of d_{min} (see Fig. 3.1d and Fig. 3.11).

For evaluation on real world data we need to decide on landmark points. In our tests we chose farthest point sampling, which worked quite well. We choose a random point first and then iteratively add the point with maximal distance to all previously chosen. Distance calculations were done with [CWW13] so that for k landmarks and n points the run-time is $\mathcal{O}(kn)$, excluding the once required matrix factorization (tests with exact geodesics [MMP87] led to similar results).

After having landmark points spread we can evaluate d_{min} and d_{max} on actual meshes. Fig. 3.7 shows a cat, where two points share the same color if distances to a fixed p are induced by the same landmark. The exactness of $d_{min}(p,q)$ depends on the minimal distance from q to one of the sets $V^r_{min}(p)$. Thus, we expect regions of same colors to resemble Voronoi regions of $V^r_{min}(p)$. Same is true for d_{max} and $V^r_{max}(p)$. In agreement with our previous writing they change frequently for d_{min}, less so for d_{max}.

A quantitative analysis of the bounds, their distances and absolute and relative errors were done on 3 different models in Fig. 3.8. First 1000 landmark points and their distances were calculated with farthest point sampling. Then the last 100



Tosca Cat Happy Buddha Stanford Dragon

Figure 3.8.: Absolute and relative errors in a logarithmic plot over the number of points. In agreement with our model errors decrease approximative with $\mathcal{O}(1/n)$ in contrast the distance to the closest landmark decreases only with order $\mathcal{O}(\sqrt{1/n})$. Straight lines represent $\mathcal{O}(1/n)$ and $\mathcal{O}(\sqrt{1/n})$ for reference.

were chosen as test points, on whose distance fields the bounds were compared to the exact distances. The graph e_{min} for example contains the mean value of e_{min} as measured from the test points to all others. The graph is twice logarithmic, so that exponential functions become straight lines whose slope is the exponent. Two guide lines were added showing the functions $\mathcal{O}(\sqrt{1/n})$ and $\mathcal{O}(1/n)$ to which the other plots can be set in relation. In agreement to our theoretical considerations, absolute errors e_{min} and e_{max} and the relative error e_{min}/d are decreasing similar to $\mathcal{O}(1/n)$, while the distance to the closest landmark point decreases only with order $\mathcal{O}(\sqrt{1/n})$, which is thus not the reason for good convergence. The bounds can be further visually inspected in Figures 3.9, 3.10 and 3.11.

From the related work the work of Xin et al. [XYH12] is most significant, as they approximate geodesic distances in constant time as well. For a fair comparison we chose for their algorithm the same landmark points as for our approach, from which they then build a coarse triangulation to infer distances. They deliver a good approximation, w.r.t. the absolute approximation error. Yet through the influence of the triangulation, their approximation is not continuous. Further their approximation might not result in a distance metric. For large Gaussian curvatures the real distances might deviate largely for their approximation, while our method always gives assured bounds. Finally, the implementation of our algorithm is of a remarkable simplicity.

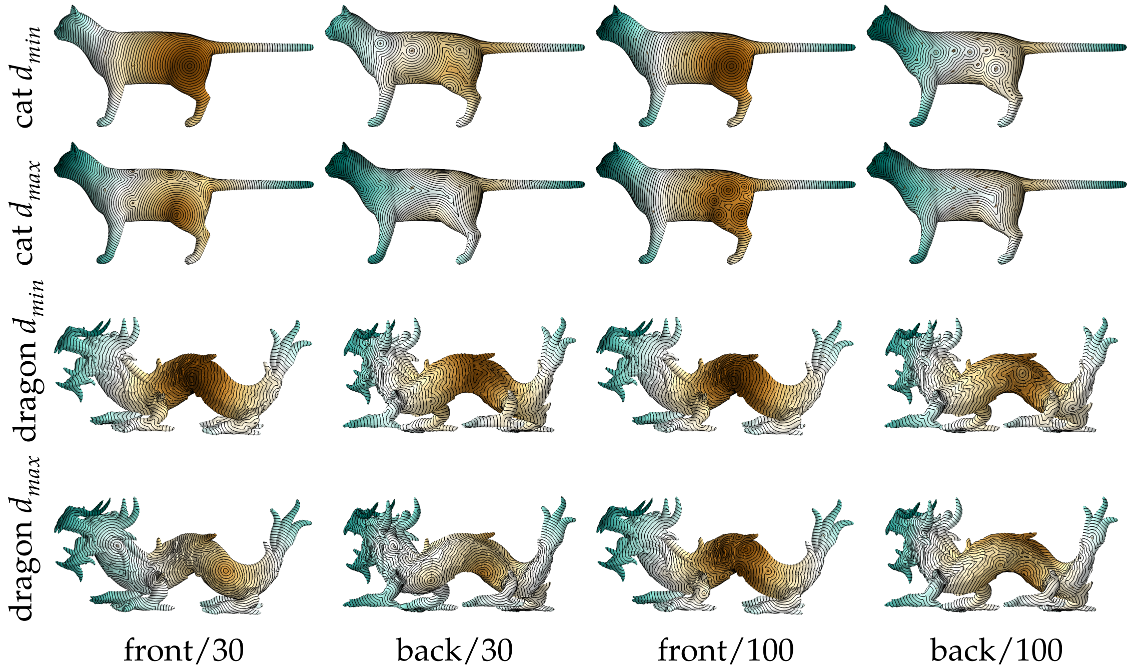


Figure 3.9.: Visualization of bounds on the Tosca Cat and Stanford Dragon for a single query point (30/100 landmarks).

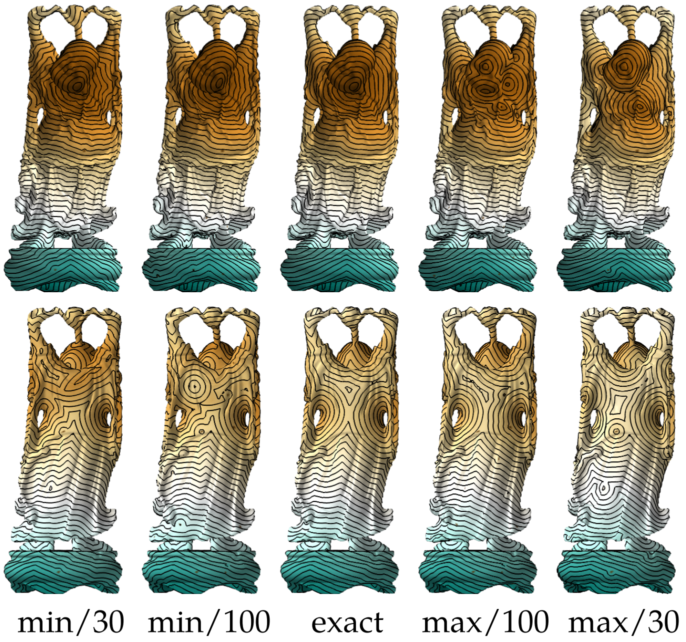
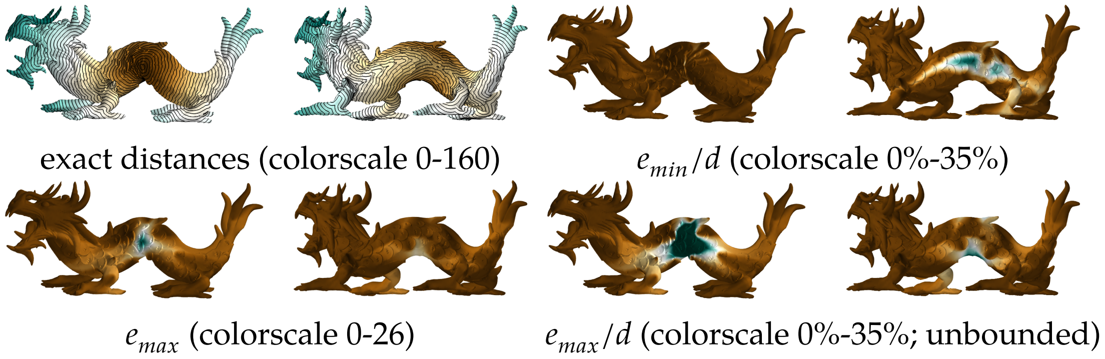


Figure 3.10.: Visualization of boundaries on the Happy Buddha.



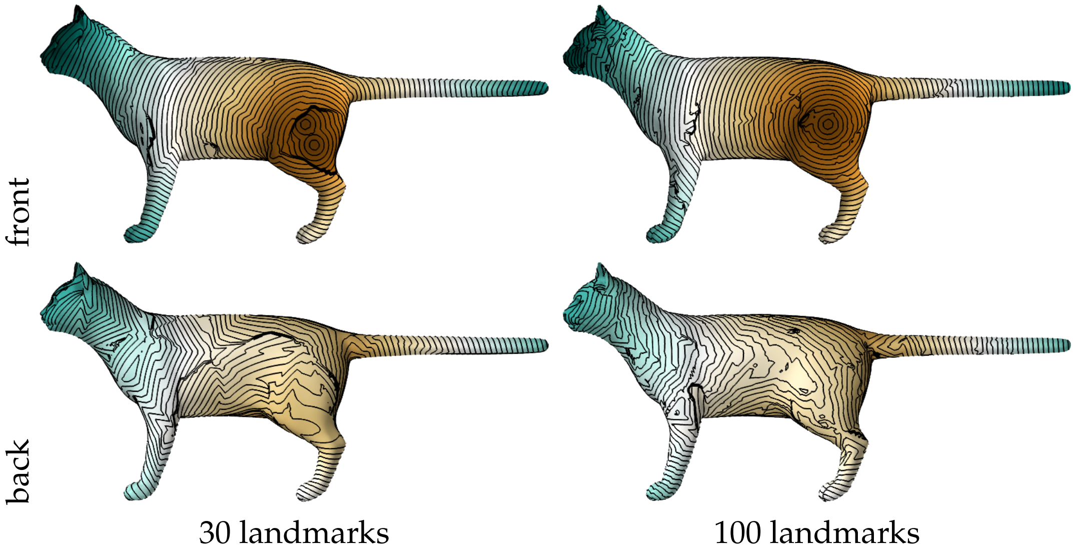
exact distances (colorscale 0-160)

e_{min}/d (colorscale 0%-35%)

e_{max} (colorscale 0-26)

e_{max}/d (colorscale 0%-35%; unbounded)

Figure 3.11.: Visualization of the errors for the Stanford dragon.



30 landmarks

100 landmarks

Figure 3.12.: Comparison to [XYH12]. They calculate constant time all-pairs distances as well. We utilize the same sample points as in our results (farthest point sampling). Note the discontinuities and see the text for discussions.

3.5. Future work

It could be well worth, investigating ideas for better landmark placing. We present the following simple Theorem, that might help to relate potential landmarks to the resulting approximation error.

Theorem 9. *Let $p, q, s \in \mathcal{M}$, further $d(p,s) \leq d(p,q)$ and $d(q,s) \leq d(p,q)$ and h_s be the shortest distance of s to any shortest path connecting p and q. Then*

$$e^{q}_{min}(p,s) = e^{p}_{min}(s,q) = e^{s}_{max}(p,q) \leq h_s \tag{3.23}$$

When solving for distance metrics (for example with linear/quadratic programs), it might be interesting to represent these over finite distance fields as discussed in Section 3.3.1.

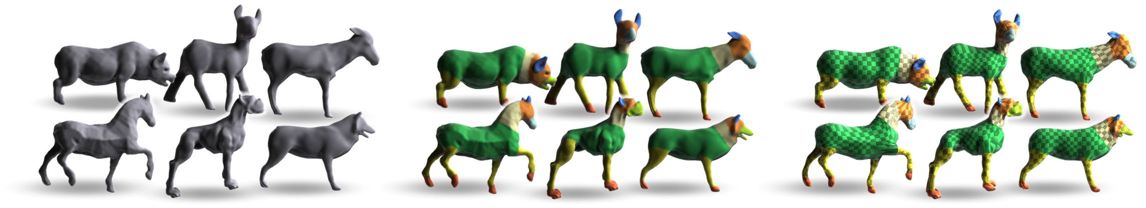
Additionally, it would be interesting to investigate the information that general pairwise distances [LRF10a; Sol+14] share and it would be interesting whether a generalization of our method is applicable in their setting.

4 Compact part-based shape spaces for dense correspondences

Abstract We consider the problem of establishing dense correspondences within a set of related shapes of strongly varying geometry. For such input, traditional shape matching approaches often produce unsatisfactory results. We propose an ensemble optimization method that improves given coarse correspondences to obtain dense correspondences. Following ideas from minimum description length approaches, it maximizes the compactness of the induced shape space to obtain high-quality correspondences. We make a number of improvements that are important for computer graphics applications: Our approach handles meshes of general topology and handles partial matching between input of varying topology. To this end we introduce a novel part-based generative statistical shape model. We develop a novel analysis algorithm that learns such models from training shapes of varying topology. We also provide a novel synthesis method that can generate new instances with varying part layouts and subject to generic variational constraints. In practical experiments, we obtain a substantial improvement in correspondence quality over state-of-the-art methods. As example application, we demonstrate a system that learns shape families as assemblies of deformable parts and permits real-time editing with continuous and discrete variability.

This chapter corresponds to the paper[Bur+13b]: Oliver Burghard et al. “Compact Part-Based Shape Spaces for Dense Correspondences”. In: *arXiv:1311.7535* (2013). An accompanying video is available online[1]. Further, sampling results can be seen in chapter A.2.

[1] https://www.youtube.com/watch?v=2m3TbG09Kls



(a) shape collection (b) coarse segmentation (c) dense correspondences

Figure 4.1.: (teaser) We compute high-quality dense correspondences between semantically models with strongly varying geometry, such as the four-legged animals in (a) (6 out of 12 input shapes shown). We require a coarse initialization as input (b) and refine them by maximizing the compactness of a part-wise Gaussian generative model (c).

4.1. Introduction

Computer graphics has reached impressively high standards in representation and rendering of 3D scenes, regularly achieving photo-realism. As a consequence, the problem of creating 3D models of matching quality has become a serious problem, making content creation a major bottleneck in practice.

Data-driven methods are a promising avenue towards addressing this problem. The reuse of existing content, such as models available in large online data-bases, might become a viable option for reducing the content creation costs in the future. In order to be useful as a creative tool, the goal is not to just copy existing models like clip-arts, but to be able to maneuver within the space spanned by the examples and synthesize new shapes of related structure.

An important low-level problem for building such navigable shape spaces is the correspondence problem: We need to determine which parts of objects are equivalent and which surface points have to be matched, establishing dense correspondences. Most statistical analysis techniques for building parameterized shape spaces require such a dense prior alignment as input [BV99a; ACP03b; Has+09b]. The results crucially depend on the quality of the correspondences: Inaccurate and drifting correspondences yield bad shape spaces. In such spaces, sampling and interpolation yields implausible results (see the accompanying [2]for a visualization). In other words, such models fail to *generalize* beyond the input data. The problem can be reduced by using large training sets to learn rather low-dimensional shape spaces. This averages out drift but also reduces the accuracy; only low frequency bands of the geometry are still predicted (see for example the results in [Has+09b] obtained from almost 2000 shapes).

Hence, good correspondences are a key requirement for building useful and

[2] `https://www.youtube.com/watch?v=2m3TbGO9Kls`

informative shape spaces. The correspondence problem comes in two flavors: *Global* and *local* matching. Local matching requires a rough initialization and refines it but is prone to getting stuck in local optima. Global methods are complementary: They aim at providing the initialization, but are usually unable to compute detailed and dense solutions. Our paper addresses the local problem: We want to find good dense correspondences given a coarse initialization. Specifically, we address this problem for the case of shapes that have strongly varying geometry, as typically required for learning shape spaces.

Computing correspondences among rather "similar" shapes is a problem that is by now already quite well understood. Variants of the ICP algorithm handle local alignment [RL01a] for both rigid and deformable models. Deformable ICP employs differential deformation priors such as elasticity [HTB03], isometry [BBK06a; Ovs+10], conformal maps [Lév+02; KLF11a], or thin-plate-splines [ACP03b; BR07]. These approaches model the behavior of infinitesimally small portions of the object: For example, elasticity penalizes local stretch and bending, and thin-plate splines optimize for smooth deformations.

The problem with differential deformation models is that their assumptions are often not justified when considering shape families with substantial geometric variability, such as a diverse collection of four-legged animals (Figure 4.1). Elastic models can capture pose changes of a single shape reasonably well [HTB03; LSP08], but matching objects of different proportions creates strong artifacts (see Figure 4.9d). Thin-plate splines [ACP03b; BR07] are more flexible but their bias towards affine mappings still causes very noticeable artifacts (Figure 4.9f). In both cases, reducing the weight of the regularization reduces bias but also increases noise and drift in the correspondences.

Isometry and conformal maps are by design already quite rigid: Both are already fixed by three point-to-point correspondences (for spherical topology), which is very valuable for solving the global matching problem efficiently [LF09a]. These models are again useful for modeling pose changes, but shape sets of largely varying geometry are very unlikely to fall into the prescribed, low-dimensional sub-manifold of matchable shapes. Blending between partial maps [KLF11a] can reduce the problem, but substantial bias persists (see Figure 4.9g).

Overall, computing dense correspondences among shapes of strongly varying geometry remains a problem that is mostly unsolved. The conceptual problem is that we need an effective notion of similarity that does not yet prescribe very specific geometric properties. Supervised machine learning from user annotated examples [KHS10; Kai+11; Sun+13] has shown promising results for establishing coarse correspondences. However, it cannot be easily extended to the dense case because it is very difficult if not impossible for a human to prescribe accurate dense correspondences for the training data.

This observation is a major motivation for our paper: We assume that coarse annotations are available. In addition to existing coarse matching methods [HKG11; Sid+11], we can always resort to manual human labeling. However, this is impossible for dense matches. We therefore develop a new method to get high-quality dense correspondences from a sparse and inaccurate initialization.

Towards this end we build upon another recent idea: Correspondence extraction from *shape collections*. By considering many shapes of a similar kind simultaneously, more information is available. Several recent papers employ the *cycle consistency* constraint to build correspondences in shape collections [Ngu+11a; Hua+12; Kim+12]: Correspondences are usually understood as a point-wise equivalence relation, being transitive over multiple shapes. Thus, unclosed loops indicate errors in pairwise matches that can be detected and removed. As pairwise regularizer, near-isometry [Ngu+11a; Hua+12] or (optionally) extrinsic shape similarity [Kim+12] are employed. However, this implicitly assumes that the shapes in the collection are dense samples of a continuous manifold of shapes, i.e., nearby samples are intrinsically very similar. This is not always the case in practical shape sets and thus introduces, as we will demonstrate experimentally, substantial artifacts.

In this paper, we therefore improve this model by explicitly regarding correspondence estimation as *optimization of shape spaces*, aiming at capturing the class of observed models well. This can be understood as a statistical learning problem: A good explanation for a phenomenon is one that not only fits the observed data tightly but that is also simple [DHS01]. It is trivial to fit a large number of observations with a highly flexible model with lots of parameters (overfitting). However, making accurate predictions with a small and concise model makes such a hypothesis statistically meaningful.

Matching shapes of widely varying geometry forces us to choose mappings from a very large and sufficiently flexible set. However, from this large set, we aim at picking the simplest, the most *compact representation*: The model should minimize the degrees of freedom utilized for representing the various shapes, rendering accidental matching unlikely: only natural correspondences will create simple shape spaces because they arise from a hidden, simple explanation for the observed geometric variability. Technically, this is formalized by minimizing the description length (MDL) of objects created by a Gaussian generative probabilistic model on a linear shape space. This approach has been originally developed in computer vision and medical imaging [KT98; Dav+02b].

In order to extend the applicability to a spectrum of typical computer graphics problems, we extend the original idea: First, we adapt the representation to handle meshes of generic topology. Second, we introduce a part-based representation that permits modeling correspondences across shapes of *varying* topology, interpreting each shape as an assembly of dockable, deformable parts. This allows us to learn a larger class of such composite models with both continuous (part deformation) and discrete (part assembly) variations. In particular, we introduce a novel algorithm to synthesize seamless and continuous models for assemblies of parts. Finally, the part-based approach yields high quality results: It decouples correlations between distant parts, which permits learning of expressive shape spaces with fewer examples, and with higher-quality correspondences.

In summary, we make the following main contributions: First, we introduce compact shape spaces for correspondence optimization to graphics, and demonstrate that this approach has a substantial impact on correspondence quality. Second, in order to make the method applicable to general meshes, we develop a new algorithm that can handle manifold meshes of generic topology while still maintaining meshing quality (uniform sampling and avoiding fold-overs). Third, we introduce a part-based formulation that represents shapes of variable topology; in particular, we describe new analysis and synthesis algorithms for composite shapes. We show that the part-based approach also improves the quality of the results over global optimization. Finally, as an example application, we demonstrate an interactive system for designing deformable shapes with continuous and discrete variability.

4.2. Related work

In this section, we discuss previous work on compactness of shape spaces, complementary to generic correspondence estimation methods already discussed above. The concept originates from studying point distribution models such as active shape/appearance models [Coo+95] that build generative Gaussian models of variability in images.

For model optimization, Hill et al. [HT94] have proposed compactness as criterion, and modeled this as the total variance of the shape distribution. Kotcheff and Taylor [KT98] employ normal-distribution entropy, which creates sparse representations.

Davies et al. [Dav+02b] refine this model by formulating the objective as minimum description length (MDL) approach [Ris78] that avoids inconsistencies and singularities.

Ericsson et al. [EÅ03] derive a gradient for the MDL energy, replacing the rather slow genetic algorithms and simplex methods by more efficient gradient descent [Hei+05].

The approach can be combined with surface parameterization [Dav+02a; Hei+05; Dav+10] to handle manifolds and guarantee bijectivity, however this restricts the topology to the spherical case.

Cates et al. [Cat+06] extend the approach to regularly sampled point-based representations of manifolds, handling the sampling uniformity by an elegant complementary entropy term. This approach also removes the topological restrictions but does not yield continuous, bijective mappings between meshes.

Our technique builds upon the entropy-based approach of Kotcheff and Taylor [KT98]. Unlike previous methods, we use a smooth implicit representation of input meshes and parameterize the correspondences over a single such a shape of general topology. We enforce regular and uniform meshing by a bi-Laplacian regularizer and dynamic resampling [BK04a]. Our representation automatically ensures cycle-consistent correspondences and permits handling of meshes of general topology while maintaining meshing quality (in practice, also effectively avoiding fold-overs). Further, the smoothness of the representation allows us to employ an efficient quasi-Newton method for optimization.

A problem of straightforward Gaussian MDL models is that they create bias towards linear representation of global shape rather than aligning surface features. Thodberg et al. [TO03] address this by adding a curvature-matching error. Our part-based approach can be seen as an alternative and complementary measure to limit such artifacts by providing localized adaptivity. In addition, it permits more flexibility in analyzing and representing composite shapes, which none of the previous methods provide.

A second, orthogonal problem is the global nature of the statistics. The model tends to overfit correlations between unrelated parts. For example, the poses of the arms in a human model are mostly independent, but excessive training data is required for a PCA model to recognize this. For this reason, many approaches have used part-based formulations [BV99a; Zha+04; FKY08; TDM11]. Our main contribution in this respect is that our analysis algorithm optimizes such models automatically. As a convenient by-product of the part-based correspondence optimization, our method optimizes the boundaries of the segmentation automatically given only a very coarse initialization. Further, our synthesis method works in the gradient-domain and thereby provides improved smoothness across boundaries in comparison to previous spatial domain methods [TDM11].

4.3. Creating compact shape spaces

In this section, we describe the basic method for optimizing shape correspondences with the objective of creating compact shape spaces. We here first discuss the case of each shape consisting of a single part only; composite, part-based shape spaces will be discussed later, in Section 4.4.

Input: In the following, let $\mathcal{S}_1, ..., \mathcal{S}_n \subset \mathbb{R}^3$ be a set of 3D shapes. We assume that these are smooth, compact 2-manifolds embedded in $\mathbb{R}^3$. The topology can be arbitrary but has to be fixed across all input shapes for now (by assembling multiple such parts, this requirement can be relaxed later). In practice, the shapes are discretized as triangle meshes. We denote the corresponding vertices by $\mathbf{S}_1, ..., \mathbf{S}_n$; each $\mathbf{S}_i = (\mathbf{s}_1, ..., \mathbf{s}_{n_i})$ is a matrix formed by the vector of the individual vertices. The set of triangles of each mesh are denoted by $\mathcal{T}_{\mathbf{S}_i}$.

4.3.1. Linear shape spaces

We first recap Gaussian generative shape models, as well known from literature [Coo+95; BV99a; ACP03b], and define our notation.

First, we define the generative process: Let $\mathcal{U}$ be a *urshape*, i.e., a base shape that has the same topology as each of the input shapes and that serves as parameterization domain for the shape space. This space is formed by the mappings:

$$f : \mathcal{U} \times \mathbb{R}^d \rightarrow \mathbb{R}^3. \tag{4.1}$$

For each vector $\boldsymbol{\lambda} = (\lambda_1, ...\lambda_d) \in \mathbb{R}^d$ and each $\mathbf{x} \in \mathcal{U}$, the function f returns a point on the generated shape. We assume the generative process to be linear. This shapes can be described by coordinates in an orthogonal basis. For a $\mathbf{x} \in \mathcal{U}, \boldsymbol{\lambda} \in \mathbb{R}^d$, we have:

$$f_{\boldsymbol{\lambda}}(\mathbf{x}) = f_{\lambda_1,...,\lambda_k}(\mathbf{x}) = b_0(\mathbf{x}) + \sum_{i=1}^{d} \lambda_i \cdot b_i(\mathbf{x}) \tag{4.2}$$

Where the function b_0 encodes the mean shape and $b_1, ..., b_d$ are orthogonal basis functions that describe the possible linear modes of variation. In our implementation, we use (as most others) the mean shape as urshape, i.e., $b_0 \equiv id$. In practice, $\mathcal{U}$ will be approximated by a triangle mesh of n vertices. We denote the $3 \times n$ matrix of the n vertices of the mesh by $\mathbf{U}$, and denote the created meshes by $f_{\boldsymbol{\lambda}}(\mathbf{U})$, and the continuous version by $f_{\boldsymbol{\lambda}}(\mathcal{U})$, respectively.

We equip the shape space $\{f_{\boldsymbol{\lambda}}(\mathcal{U}) | \boldsymbol{\lambda} \in \mathbb{R}^d\}$ with a Gaussian probability measure with an axis aligned neg-log likelihood

$$-\log \Pr(\boldsymbol{\lambda}) = \frac{1}{2} \sum_{i=1}^{d} \frac{\lambda_i^2}{\sigma_i^2} + \text{const}, \tag{4.3}$$

where $\boldsymbol{\sigma} = (\sigma_1, ..., \sigma_d)$ specifies the standard deviations along the main axes of the model.

Further, we will usually consider the space of shapes generated by f and then rigidly arranged in $\mathbb{R}^3$. Given $\mathbf{R} \in O(3)$ and $\mathbf{t} \in \mathbb{R}^3$, we denote a rigidly transformed shape (in slight abuse of notation) by:

$$\mathbf{R}\left(f_{\boldsymbol{\lambda}}(\mathcal{U})\right) + \mathbf{t} := \{\mathbf{R} \cdot f_{\boldsymbol{\lambda}}(\mathbf{x}) + \mathbf{t} \mid \mathbf{x} \in \mathcal{U}\}. \tag{4.4}$$

Building the model: Given a set of input shapes and correspondences between them, we can easily build Gaussian shape spaces using principal component analysis (PCA): Assume that we are given a set of consistently triangulated vertex meshes $\mathbf{S}_1^*, ..., \mathbf{S}_n^*$ that match the input $\mathcal{S}_1, ..., \mathcal{S}_n$ with vertex correspondence, i.e., corresponding vertices located at matching geometry (we use the star to denote known correspondences). We compute the mean $\overline{\mathbf{S}}$ by averaging the input shapes and determine the covariance matrix

$$\Sigma_{\mathbf{S}} = \frac{1}{n-1} \sum_{i=1}^{n} \left(\mathbf{S}_i^* - \overline{\mathbf{S}}\right) \left(\mathbf{S}_i^* - \overline{\mathbf{S}}\right)^{\mathrm{T}} \tag{4.5}$$

The mean and the eigenvectors of $\Sigma_{\mathbf{S}}$ yield the mean and basis meshes and the eigenvalues correspond to the standard deviations σ. Further, it is easy to see (for example, by applying a singular value decomposition and rearranging terms) that the Gram matrix

$$\mathbf{G}_{\mathbf{S}} = \sum_{i=1}^{n} \left(\mathbf{S}_i^* - \overline{\mathbf{S}}\right)^{\mathrm{T}} \left(\mathbf{S}_i^* - \overline{\mathbf{S}}\right) \tag{4.6}$$

has the same eigenvalue spectrum (up to the factor $n-1$). In the continuous case, the sum is replaced by an integral. Assume that we have homeomorphisms $s_i^* : \mathcal{U} \rightarrow \mathcal{S}_i$ that encode continuous correspondences to our input shapes $\mathcal{S}_1, ..., \mathcal{S}_n$. We then again form the mean function $\bar{s}$ by averaging and the $n \times n$ Gram matrix:

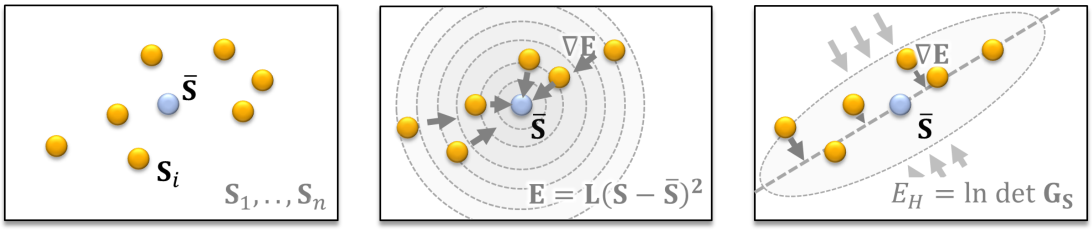
$$\mathbf{G}_{\mathcal{S}} = \int_{\mathcal{U}} \left(s_i^*(\mathbf{x}) - \bar{s}(\mathbf{x})\right)^{\mathrm{T}} \left(s_i^*(\mathbf{x}) - \bar{s}(\mathbf{x})\right) d\mathbf{x}. \tag{4.7}$$

The matrix has at most rank n; in the (typical) case of redundancy in the shape collection, the number of significant eigenvalues will typically be substantially smaller than n. Importantly, the spectrum does not just depend on the geometry of $\mathcal{S}_1, ..., \mathcal{S}_n$ but crucially on the correspondences encoded in the functions $s_1^*, ..., s_n^*$. While the variability of the shapes prescribes a lower bound on the rank of $\mathbf{G}_{\mathcal{S}}$, we can in general artificially inflate it up to full rank by just letting the correspondences drift randomly along the surface.

4.3.2. Compactness

We now discuss how to measure the compactness of the shape space and how to minimize it. We also recap ideas from [KT98; Cat+06; Dav+10] to keep the paper self-contained.

Spectral view: Let σ denote the vector of eigenvalues of $\mathbf{G}_{\mathcal{S}}$. If the correspondences $s_1^*, ..., s_n^*$ include unnecessary movements along the surfaces of the objects, the spectrum will spread out, creating more non-zero eigenvalues. In reverse, a compact shape space should have a compact spectrum. A simple way of modeling this is to penalize the square norm $\|\sigma\|_2^2 = \mathrm{tr}(\mathbf{G}_{\mathcal{S}})$ of the eigenvalues, as proposed by Hill and Taylor [HT94]. It is equivalent to trying to keep all surface points in deformed shapes close to the mean shape, independent of each other (therefore not transporting information globally). From a spectral perspective, it favors multiple small eigenvalues over a few large ones, which does not match the intuition of a low-dimensional generative process that we want to reconstruct. Rather than that,


$\bar{\mathbf{S}}$
$\mathbf{S}_i$
$\mathbf{S}_1,..,\mathbf{S}_n$
$\nabla\mathbf{E}$
$\bar{\mathbf{S}}$
$\mathbf{E} = \mathbf{L}(\mathbf{S} - \bar{\mathbf{S}})^2$
$\nabla\mathbf{E}$
$\bar{\mathbf{S}}$
$E_H = \ln\det \mathbf{G}_\mathbf{S}$

(a) untransformed (b) linear differential prior (c) entropy prior

Figure 4.2.: Geometric interpretation: Differential models such as (linearized) elasticity or thin-plate splines impose a quadratic energy on a linearly transformed shape space that attracts all target shapes to the source shape. Our method minimizes the entropy of the ensemble, moving correspondences such that the shapes align in a lower-dimensional subspace, creating less bias.

we aim at a sparse spectrum, as detailed next.

Probabilistic view: We can also look at the probability distribution the shapes are drawn from. The less variability it permits, without reducing the likelihood of the training examples, the more concisely it captures the shape space. In this view, we should measure the *entropy* of the Gaussian model:

$$H_{\Pr(\boldsymbol{\lambda})} = \frac{1}{2}\ln\prod_{i=1}^{n}\sigma_i^2 + const. \tag{4.8}$$

This approach suffers from singularities: If one of the eigenvalues becomes zero, the determinants of the covariance and Gram matrix become zero, leaving the entropy ill-defined. Further, driving even just the least eigenvalue close to zero would falsely indicate a near-perfect solutions, which leads to instability and inconsistency.

Information theoretic view: From the point of view of information theory, we can measure the capacity of the generative probabilistic model (Equations (4.2, 4.3)) to encode different models by considering the description length of a specific shape, given the knowledge of the generative model in terms of the probabilistic shape space. To transmit one shape, we need to encode the shape parameters $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_d)$. Given an independent Gaussian distribution along each axis b_i with variance σ_i^2, and assuming that a finite accuracy of $\Delta > 0$ is required in our application, encoding a single parameter requires roughly $\mathcal{O}(\log\frac{\sigma_i}{\Delta})$ bits [Tho03] (see Davies et al. [Dav+02b] for the accurate and more detailed derivation). For small variances $\sigma_i < \Delta$, no information needs to be encoded. This suggests the following energy [KT98; Cat+06] that approximates the information content of the shape space depending on correspondences $s_1, ..., s_n$:

$$E_H(s_1, ..., s_n) = \ln \prod_{i=1}^{n} (\sigma_i + \delta) = \ln \det (\mathbf{G}_{\mathcal{S}} + \delta \mathbf{I}) \tag{4.9}$$

$\delta > 0$ is a regularizer that determines the accuracy of the shape space: We assume that independent of the example data, there is always an isotropic Gaussian noise component of standard deviation δ in all dimensions of the space. This removes the singularity and makes the entropy usable as measure that encourages sparse PCA spectra during correspondence optimization [KT98]. This is an approximation to coding length [Dav+02b]; nonetheless, it already yields favorable results in practice.

Geometric view: We can also interpret these results as imposing a prior in a shape space. Figure 4.2 shows schematically a number of example shapes $\mathbf{S}_1, ..., \mathbf{S}_n$ as points in a high-dimensional shape space. Traditional regularizers such as thin-plate-splines or linearized elasticity impose a Gaussian prior, i.e., the neg-log-likelihood is a quadratic energy of the form

$$E(\mathbf{S}_1, ..., \mathbf{S}_n) = \sum_{i=1}^{n} \left(\mathbf{L}(\mathbf{S}_i - \overline{\mathbf{S}}) \right)^2, \tag{4.10}$$

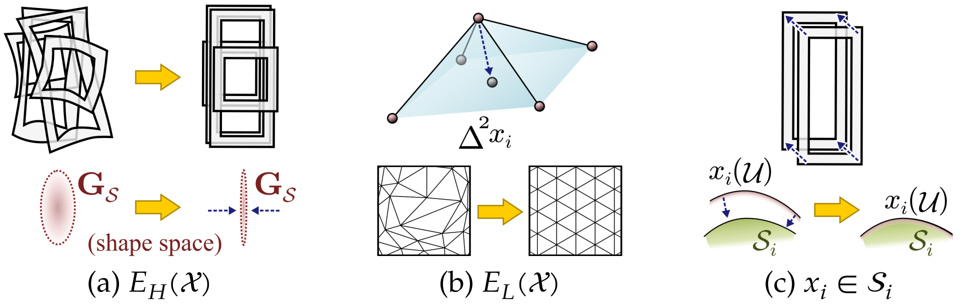
where $\mathbf{L}$ is a linear operator (a matrix) that acts on the vertex sets $\mathbf{S}_i$ interpreted as $(3 \cdot n_i)$-vectors. For example, in thin-plate-splines, $\mathbf{L}$ measures the bending by taking second derivatives. In other words, traditional (linear) differential priors can be seen as an isotropic attraction to a single point (the urshape) in a linearly transformed shape space (Figure 4.2b). Contrarily, minimizing the entropy encourages a tight fit of an ellipsoid to the data, minimizing its volume, and thereby encouraging all models to be located on a low-dimensional linear subspace (Figure 4.2c). This creates bias towards a linearly correlated representation rather than towards a single shape. It is not surprising that this yields significantly better results when the final objective is to describe a shape collection with exactly this representation rather than reconstructing it from pairs of biased, point-wise matches in shape space.

4.3.3. Shape optimization

Let $\mathbf{S}_1, ..., \mathbf{S}_n$ be a set of example models given as triangle meshes. We approximate these by smooth surfaces $\mathcal{S}_1, ..., \mathcal{S}_n$, as detailed later. Let $\mathbf{U}$ be an urshape of matching topology. We denote the vertices of $\mathbf{U}$ by $\mathbf{u}_1, ..., \mathbf{u}_m$. We now want to compute correspondences

$$x_i \colon \{\mathbf{u}_1, ..., \mathbf{u}_m\} \to \mathcal{S}_i. \tag{4.11}$$

We denote the set of all correspondences by $\mathcal{X} = \{x_1, .., x_n\}$. All of these are



(a) $E_H(\mathcal{X})$ (b) $E_L(\mathcal{X})$ (c) $x_i \in \mathcal{S}_i$

Figure 4.3.: Shape space optimization. Our energy consists of three terms (a-c) and a hard constraint (d). (a) The term E_H minimizes the description length of the model by aiming at minimizing the entropy (volume) of the Gram matrix of the model collection. (b) E_L encourages uniform meshing using a Laplacian energy. (c) Matching the data surfaces $\mathcal{S}_i$ is a hard constraint.

hard-constrained to be located on the (smoothed) input surfaces. We optimize the correspondences by minimizing the following energy, subject to the constraint of moving only along the surface (as illustrated in Figure 4.3):

$$E(\mathcal{X}) = E_H(\mathcal{X}) + \mu_L E_L(\mathcal{X}) \tag{4.12}$$

The term E_H approximates the description length as discussed above and E_L is a bi-Laplacian regularizer.

We set its weight μ_L to the ratio of the number of triangles divided by the surface area squared (to make the overall weight mesh-independent), multiplied by a relative weight of $0.25 \cdot 10^{-5}$.

The overall energy is minimized using l-BFGS, a nonlinear quasi-Newton solver. Further, we factor out rigid motions according to Equation (4.4): We compute a least-squares optimal translation, rotation, and reflection from the initial correspondences. The rigid motion is updated during the optimization by including the rotation as variable in the optimization (parameterization the small rotational update as Euler angles with respect to the initial least-squares fit).

Compactness

For creating compact shape spaces, we use the energy E_H from Equation (4.9). We compute the Gram matrix by integrating over the deformed triangle meshes according to Equation (4.7). Because of additional regularization (described next), it is sufficient to approximate the integrals by an unweighted sum over vertex positions (Equation (4.6)). We compute the derivative of the energy using the explicit formula derived in [KT98].

Regularization

The regularization term E_L is a prior on the graph Laplacian of the deformed meshes $x_i(\mathbf{U})$. With N_i denoting the set of indices of vertices sharing an edge with vertex $\mathbf{u}_i$ in the mesh $\mathbf{U}$, we obtain:

$$E_L(\mathcal{X}) = \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{1}{|N_j|} \left(\sum_{k \in N_j} (x_i(\mathbf{u}_j) - x_i(\mathbf{u}_k)) \right)^2 \tag{4.13}$$

This term encourages the graph Laplacian of the triangle mesh to be zero, which is the case if every vertex is located in the center of its 1-ring neighborhood, corresponding to a uniform triangulation [BK04a]. Although adding this least-squares energy does not guarantee bijectivity of the mapping, it also effectively avoids fold-overs in practice.

Data Modeling

We model the hard-constraint that correspondences must remain on the input surfaces by a level-set approach. As our input is only discrete, C^0 mesh approximation of a shape, we first build a smooth surface S_i that tightly approximates $\mathbf{S}_i$ so that we can slide along the surface smoothly during optimization. We first sample the input mesh with a dense, uniform point cloud $\mathbf{S}'_i = \{\mathbf{s}'_1, ..., \mathbf{s}'_{n_i}\}$ representing the input mesh $\mathbf{S}_i$ with (given) oriented normals $\{\mathbf{n}_1, ..., \mathbf{n}_{n_i}\}$. We fit a signed distance function $d : \mathbb{R}^3 \to \mathbb{R}$ to $\mathbf{S}_i$ by minimizing the following energy [CT11]:

$$\begin{aligned} E(d) = \mu_z \underbrace{\sum_{i=1}^{n_i} d(\mathbf{s}'_i)^2}_{\text{zero crossing}} + \mu_g \underbrace{\sum_{i=1}^{n_i} \|\nabla d(\mathbf{s}'_i) - \mathbf{n}_i\|^2}_{\text{gradients}} \\ + \mu_F \underbrace{\int_\Omega \|H_d(\mathbf{x})\|_F^2 d\mathbf{x}}_{\text{smoothness}} \end{aligned} \tag{4.14}$$

The first term assures that the zero crossing of d is at the data points. The second term aligns the gradients with the normals, creating a smooth result and removes the trivial solution ($d = 0$). The last term integrates the squared Frobenius norm of the Hessian of d over a bounding volume Ω, acting at a regularizer that propagates function values linearly and encourages smoothness. We set the weight $\mu_z = 1/n_i$, $\mu_g = 0.1/n_i$ and $\mu_H = 10^{-4}/|\Omega|$, which is sufficient to smooth very sharp corners a bit.

We optimize this quadratic energy by solving the linear system resulting of a finite difference discretization with spacing h set to $h = 1.5\%$ of the bounding box of the object. Continuous values for $d(\mathbf{x})$ and $\nabla d(\mathbf{x})$ are obtained by interpolation with radial basis functions at each grid point; we employ Wendland kernels $\max(0, (1 - \|\mathbf{y} - \mathbf{x}\|^2/h^2)^3)$. H_d is approximated by finite differences over the grid. The domain Ω is obtained by including all grid cells within a distance of $4h$ to data points. Triangle meshes sampled with spacing $h/4$ to obtain $\mathbf{S}'_i$. We refer to the zero-level set of the result as $\mathcal{S}_i$.

Using the implicit function: The $\mathcal{S}_i$ serve as constraint manifolds for correspondences during optimization: First, any initial solution is projected to $\mathcal{S}_i$ by simple gradient descent. During numerical optimization, the quasi-Newton l-BFGS solver attempts to update the correspondences positions: $x_i(\mathbf{u}_j) \rightarrow x'_i(\mathbf{u}_j)$ by first finding a new direction and then the distance by a line search. In each iteration, we project $x'_i(\mathbf{u}_j)$ back onto the surface (using the exponential map in x_i). The rational is that small step sizes turn the smooth constraint into a sequence of linear subspace constraints that can are handled by the quadratic (low-rank) optimizations performed in the inner loop of l-BFGS.

Motivation: In experiments with various formulations, the implicit function formulation with hard constraints to the zero level-set turned out to be most reliable and crucial for good results. Other options did not give satisfactory results: Least-squares soft-constraints are unreliable: weak constraints have trouble with thin structures and sharp creases, and strongly weighted soft constraints yield a numerically ill-conditioned energy, preventing convergence. The option of just using the input triangle meshes was not satisfactory either: Using such a C^0 surface lead to spurious local optima in our experiments. Experiments with a projection to dynamically computed MLS-approximation of the surface have also turned out to be slow and unreliable for general surfaces with small feature size.

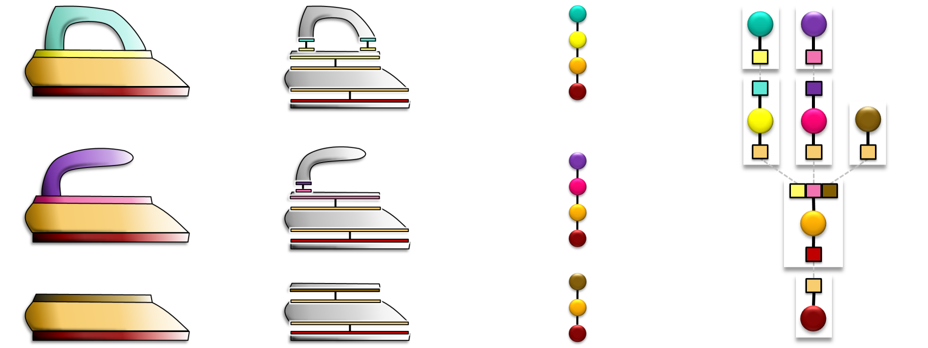
4.4. Extended model

We now extend our approach by introducing composite, *part-based* models that capture correspondences among objects of varying topology.

4.4.1. Part-based modeling

The method as discussed so far, as well as previous proposals in literature, is restricted to shapes that have global correspondences and form a single, global shape space. In practice, this is often a strong restriction. Many man made shapes consist of composite parts (for example, the irons in Figure 4.7b have been assembled from different parts), forming shape spaces of varying topology that cannot be captured by a single shape space.

We therefore propose a model that decomposes complex shapes into a set of *parts* that have individual shape spaces. First, we modify the analysis algorithm to optimize both the shape and the decomposition of the surface. Second, develop a



(a) input shapes (b) docking sites (c) part graphs (d) docking rules

Figure 4.4.: Decomposing objects into morphable parts (iron example; c.f. Figure 4.7b). (a) Objects are segmented into different parts, indicated by different colors.(b) Boundaries permit docking of shapes according to the observed variations. (c) Each discrete assembly corresponds to a graph of parts. (d) This yields a set of rules for possible part arrangements.

synthesis algorithm that can build seamless models consisting of deformed parts in different poses. The synthesis can handle general arbitrary constraints (changing the discrete composition of the parts, handles for free-form deformation, subspace constraints).

Analysis

Input: We again assume that we are given a set of example shapes $\mathbf{S}_1, ..., \mathbf{S}_n$ as triangle meshes (Figure 4.4 shows an example, reflecting the actual result demonstrated in Figure 4.7b). We further assume that the shapes are segmented into parts, i.e., every triangle is tagged with a part type $p \in \{1..K\}$, where K is the number of different part types (Figure 4.4a shows the types as different colors). Each part of the same type must have the same topology (The irons example use a adapter pieces (yellow/pink/brown) to attach handles of different topology to the body). Each discrete configuration corresponds to a different graph of parts (Figure 4.4c). In addition, each part has continuous parameters (not shown) that permit deformation according to the shape space learned from all parts of the same type (same color in our figures).

The initialization of the part boundaries does not need to be precise; only the topology and coarse geometry needs to match. We will improve the segmentation geometry automatically.

Part docking: Parts will share common boundaries, and possibly in different combinations. We learn the way parts can be discretely assembled from the input

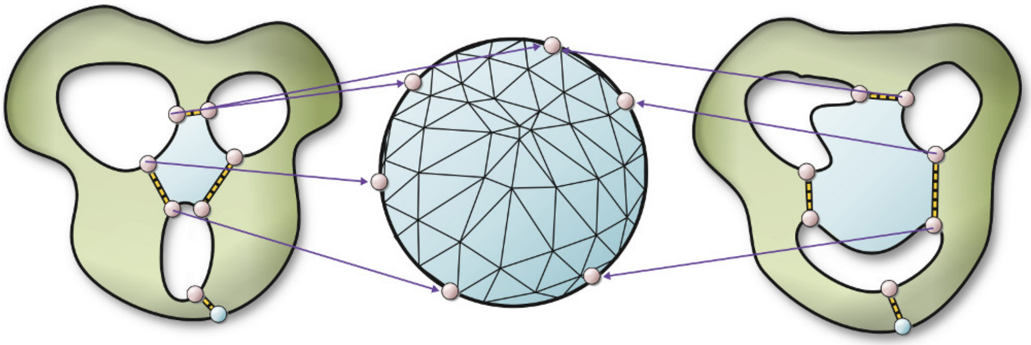


Figure 4.5.: In order to obtain initial correspondences, we cut the parts into pieces of disc topology. Afterwards, a cross parameterization ensures a bijective initialization.

by just reading of the observed adjacency relations from the input.

Figure 4.4d,e show the rules that have been deduced from the input. Boundaries between that connect two types of parts across a common docking site are always in fixed correspondence; i.e., the dense correspondences of the parts themselves are enforced at the boundary, too.

In the following we discuss our analysis algorithm that creates part graphs and shape spaces for each part automatically given a coarse user segmentation and possibly a few additional landmark matches.

Part parameterization: The first step is to compute initial dense correspondences. We need bijective correspondences without fold-overs. For this, we use cross--parameterization [KS04a]: We first cut the parts further into topological discs, and then compute a cross-parameterization of the discs to obtain initial correspondences (see Figure 4.5). For cutting, we first detect the interior boundaries within all parts. We connect each resulting boundary curve to its closest neighbor (see Figure 4.5) and then cut along a geodesic path between the corresponding closest point. Cutting is iterated until only topological discs remain, and the process is done in all parts simultaneously. This initialization is presented to the user, who can move the initial landmark correspondences along the boundaries (red dots in Figure 4.5). The resulting sub-parts are set into correspondences by a least-squares conformal map to a unit circle; the boundaries of the circle are set into correspondence by comparing the relative arc length (normalized to $[0, 2\pi]$), using the cutting points as starting point. For the outer boundary $\partial \mathbf{P}_j^{(p)}$ of the initial segmentation, the user has to specify this starting point manually.

The result of this step are dense correspondences between all parts, with topology consistent to the user-defined segmentation. The correspondences are guaranteed to be bijective, but the quality is usually very bad, showing strong drift and distortions across the shape.

Optimization: We now perform the optimization from Section 4.3 to improve this initial guess. For each part type (same color in our figures), we setup a separate energy $E^{(p)}$ according to Equation (4.12). We use the cross-parameterization result as initial correspondences $x_i^{(p)}$, and the resulting mean shape as urshape $\mathbf{U}^{(p)}$. We first run the optimization separately, constraining the boundaries of the domain to fit the boundary curve by a point-to-line energy that snaps the closest vertex to its the boundary of the parts. We add the boundary energy

$$E_B = \sum_{p=1}^{K} \sum_{j=1}^{n_p} \int_{\partial\mathbf{U}^{(p)}} \text{dist}\left(x_j^{(p)}(\mathbf{x}), \partial\mathbf{P}_j^{(p)}\right) d\mathbf{x} \tag{4.15}$$

that measures the deviation of boundary vertices from the boundaries of the input parts.

Boundary optimization: After the energy has converged, we remove the constraint of Equation (4.15) and start optimizing the boundary location. We need to make sure that parts still meet at the boundaries, and this should happen in a consistent way. As consistency condition, we maintain fixed correspondences along all matching part boundaries of the same type. We impose this consistency as a soft constraint in an alternating two-stage optimization:

In stage one, we find all pairs of closest points between boundaries of matching type: For points $\mathbf{x} \in b_l^{(p)}$ we compute all instances in the data, and for each instance, the closest point in the adjacent instance of type $b_l^{(p)}$. We average over all of these matches and set a soft constraint penalizes the quadratic distances for all these pairs. In stage two, we run the optimization of $E^{(p)}$ according to Equation (4.12) with the additional constraint energy added. We obtain improved correspondences, which are again used to refine the correspondences.

Conceptually, this could be interpreted as a variant of iterative closest points (ICP), performed simultaneously along multiple boundary curves while keeping their correspondences consistent. The alternating estimation of boundary correspondences is combined with the estimation of global rigid motions for each part, as already introduced in Section 4.3.3.

Further details: In our implementation, a few extra steps are performed to improve the efficiency of the method. First, before computing the cross-parameterization, we use the dynamic remeshing algorithm of Botsch et al. [BK04a] in order to improve the mesh quality of each part (which might already have been bad in the input meshes). The method iteratively minimizes the sum of the squared graph Laplacians and performs edge contractions / vertex splits in order to create a uniformly sampled mesh. We use the average edge length in all part instances as length criterion. Second, after parameterization, we might end up with very uneven sampling; the conformal map can have large scale factors that lead to a uneven distribution of triangles. Using vertex splits, again according to the same criterion, we refine regions that are undersampled and project the resulting newly inserted points onto the implicit surface that models the data. These steps could in principle be omitted, but then a very dense initial mesh is required to obtain results of good quality. Another improvement is to perform a final optimization pass of the correspondence energy $E(\mathcal{X})$ (Equation 4.12) where the Laplacian regularizer is evaluated for each composite input shape $\mathbf{S}_i$ instead of its parts separately, which makes sure that the boundaries between parts are only determined by the compactness criterion and not by the mesh regularization (we obtain a slight improvement here).

Overall, the result of the preceding is an optimized composite shape space in which (i) the correspondences within each part, (ii) the position of the boundaries on the example shapes, and (iii) the correspondences among matching boundaries have been optimized with the goal of compactness and mesh quality. In the following, we discuss how we can utilize the result to create new shapes.

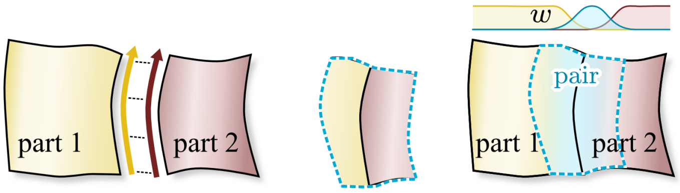
Synthesis

The model that we have obtained in the previous step describes a shape by a set of parts that are connected along their boundary lines. A key feature of this extended model is that we can instantiate composite models consisting of multiple parts, potentially rearranged by attaching the parts differently across compatible boundaries. We therefore need to devise a generative process by which we can instantiate such composite shapes, governed by multiple local shape spaces. We aim at maximum flexibility: Given an arbitrary arrangement of parts and arbitrary user constraints on geometry and shape parameters for each part, we want to find a global geometry that fits all of these constraints best.

Variational Part Reconstruction

The first step is to formulate the problem of reconstructing the part shapes as a variational problem. In order to facilitate a smooth reconstruction later on, we formulate the whole process in the gradient domain [Sum+05a; SA07]. We first consider a single part. Assume that we are given a part shape space by its urshape $\mathbf{U}$, its mean and variation modes $b_0, ..., b_d$, and the standard deviations $\boldsymbol{\sigma} = (\sigma_1, ..., \sigma_d)$. Our objective is to reconstruct an instance $\mathbf{V} = (\mathbf{v}_1, ..., \mathbf{v}_n)$ of this shape with shape parameters $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_d)$. Because of the formulation as an optimization problem, multiple parts can be coupled along boundaries, thereby implicitly constraining the reconstruction and finding the best embedding of the part graph in a least-squares sense.

We model the similarity by comparing each vertex $\mathbf{v}_i$ with a reconstructed vertex $b_0(\mathbf{u}_i) + \sum_{k=1}^{d} \lambda_k b_k(\mathbf{u}_i)$. The residual is minimized in a least squares sense. In order to get a smooth transition between multiple parts later, we follow [Sum+05a] and formulate the optimization in a gradient domain. We do not compare absolute



(a) mesh boundaries

(b) pair geometry

(c) docking with weights

Figure 4.6.: Attaching two part instances: (a) Boundaries are joined continuously by sharing variables. (b) We extract geometry adjacent to the boundary and (c) blend between part and pair shape spaces for smoother transitions.

coordinates but edge vectors in the mesh. Finally, we also include an orthogonal transformation $\mathbf{R}$ to be invariant to rigid motion (translational invariance is automatically obtained by working on edge differences).

Formally, we get the following energy:

$$E_{rec}(\mathbf{V}, \boldsymbol{\lambda}, \mathbf{R}) = \sum_{i=1}^{n} \sum_{j \in N_i} \omega_{i,j} \Big((\mathbf{v}_i - \mathbf{v}_j) - \mathbf{R}(b_0(\mathbf{u}_i) - b_0(\mathbf{u}_j)) - \sum_{k=1}^{d} \lambda_k \mathbf{R} \left(b_k(\mathbf{u}_i) - b_k(\mathbf{u}_j) \right) \Big)^2 + \sum_{i=1}^{d} \frac{\lambda_i^2}{2\sigma^2} \tag{4.16}$$

Again, N_i denotes the set of neighboring vertices of $\mathbf{u}_i$ in the mesh $\mathbf{U}$. $\omega_{i,j}$ is the cotangent weight of the edge $(\mathbf{u}_i, \mathbf{u}_j)$ in the mean shape. The variable $\mathbf{R}$ is a rotation variable, to be optimized along with the variables $\mathbf{V}$ and $\boldsymbol{\lambda}$.

This formulation is an extension of the as-rigid-as-possible shape deformation of [SA07], encouraging the result to be as-close-as-possible to a linear subspace of models (ignoring rigid differences as well). We use the same optimization method: The linear system is solved alternatingly with an update of the rotation matrix $\mathbf{R}$ (which, in our approach, is global for the whole part); see Sorkine and Alexa's paper [SA07] for details.

Reconstructing Part Graphs

In order to reconstruct shapes consisting of multiple parts, we add up the energies E_{rec} for all parts. Along the boundaries, the analysis stage gives us fixed and consistent correspondences. Therefore, we can remesh the urshapes of the parts such that they share common points along the boundaries. We then use the same variables in order to enforce a C^0 continuous solution (see Figure 4.6a).

Improved smoothness: Although the shape of the boundary curve transports information between pairs of parts, it only captures limited information on the correlation between the part shapes. We therefore learn a more expressive model from the input data: We form an extended region by gathering the geometry within a fixed distance to the boundary between the pair of parts, called *pair geometry* (Figure 4.6b). As the parts are in dense correspondence, we have dense correspondences between all pair geometry that connects the same part type through the same pair of boundaries. We build the probabilistic shape space for the pair geometry by a simple PCA analysis. We add the additional energy to the overall energy for all docked part pairs.

To avoid discontinuities, we use smooth weights for all singleton part and pairwise constraints (Figure 4.6c): The attraction to the shape spaces of the parts fades continuously to zero when approaching the boundaries. Contrarily, the attraction to the pair geometry model grows when moving towards the boundary of the parts. We weight each vertex by $\exp(-d^2/\sigma_{bdr}^2)$, where d is the distance to the boundary. For seamless results, we set σ_{bdr} to to blend within about one third of the part diameter.

4.5. Results

We have implemented the method in C++ and tested the implementation on a dual socket PC (Intel Core i7 with 2.6Ghz and 6 cores per processor). The results are shown in Figures 4.1 and 4.7-4.9. We strongly encourage the reader to watch the accompanying ,[3] which shows interpolation and sampling results from the constructed shape spaces; these make the improvements due to our method much clearer.

Dense correspondences from coarse co-segmentation: We use the painting interface discussed in Section 4.4.1 to annotate a number of models from the SHREC 2007 model collection. The user has to mark the colored regions shown in Figure 4.1b,4.9b by painting on the surface. Additionally, point-to-point correspondences have to be set if the initialization is not clear. For example, for the birds (Figure 4.7d), the tip of the wings needed one more such point match per wing. Additional constraints are not always necessary; for example, the animals data set has been build from the user segmentation only. After such initialization, we run the optimization. The user has to chose the parameter μ_L as well as the level of resolution for the remeshing step (after initial cross parameterization). The first parameter is critical for the results, the second trades-off run-time and accuracy. Finding an appropriate annotation and parameterization that works for a whole shape ensemble requires multiple iterations of interaction and optimization. Here, computing correspondences, minimizing Eq. 4.12), took on average about 20min for

[3] `https://www.youtube.com/watch?v=2m3TbGO9Kls`

a shape set. Given the additional steps (parameterization, remeshing etc.), the net computation time adds up to to roughly 1h per model. In our examples, interaction and computation amounted to up to 6h for our example models, depending on the complexity (e.g., animals were more difficult than teddies). It would probably be possible to automate the procedure by using recent fully automatic co-analysis [HKG11; Sid+11; Hua+12; Kim+12] for initialization, but this is still subject to future work.

Results: The resulting correspondences within several subsets of this collection are shown in Figure 4.7: We use a checker-board texture projected to one instance of the input and transfer it to all other models to visualize correspondences. Additionally, differently colored regions depict parts. The resulting correspondences capture salient features of the models and there is not unwarranted drift. This is very well visible in our video, where we obtain good interpolations for within all of the shape spaces: Intermediate shapes due to morphing as well as due to random sampling from the underlying Gaussian at the learned standard deviations (sampling at $\sigma = 1$) are plausible (Figure 4.7f). We should highlight that the model is able to handle structures with fine details, such as the legs of the animals in Figure 4.7a. As discussed in Section 4.4.1, we have tried various alternative approaches, all of which failed at this data set.

Comparison to pairwise local registration: We compare to a number of base-line methods first. In all cases, matching is done by first sampling 43 points uniformly from our solution to be used as initialization and then switching to deformable ICP [ACP03b]. We have also tried alignment without landmarks, as well as deactivating landmarks; the show result (keeping the landmarks during ICP) yielded the best results.

Figure 4.9 shows the results of pairwise matching between a pig and a young deer. We examine as-rigid-as-possible (ARAP) deformation model ([SA07], Figure 4.9d) and a closely related variant, using a smooth subspace deformation model ([Ada+08], Figure 4.9e): The subspace model uses a volumetric low-frequency basis (12^3 grid in our case), which leads to smoother results than ARAP. Nonetheless, both cases suffer from artifacts such as wrinkles and drifting correspondences. The video illustrates the disastrous effect on the obtained shape spaces.

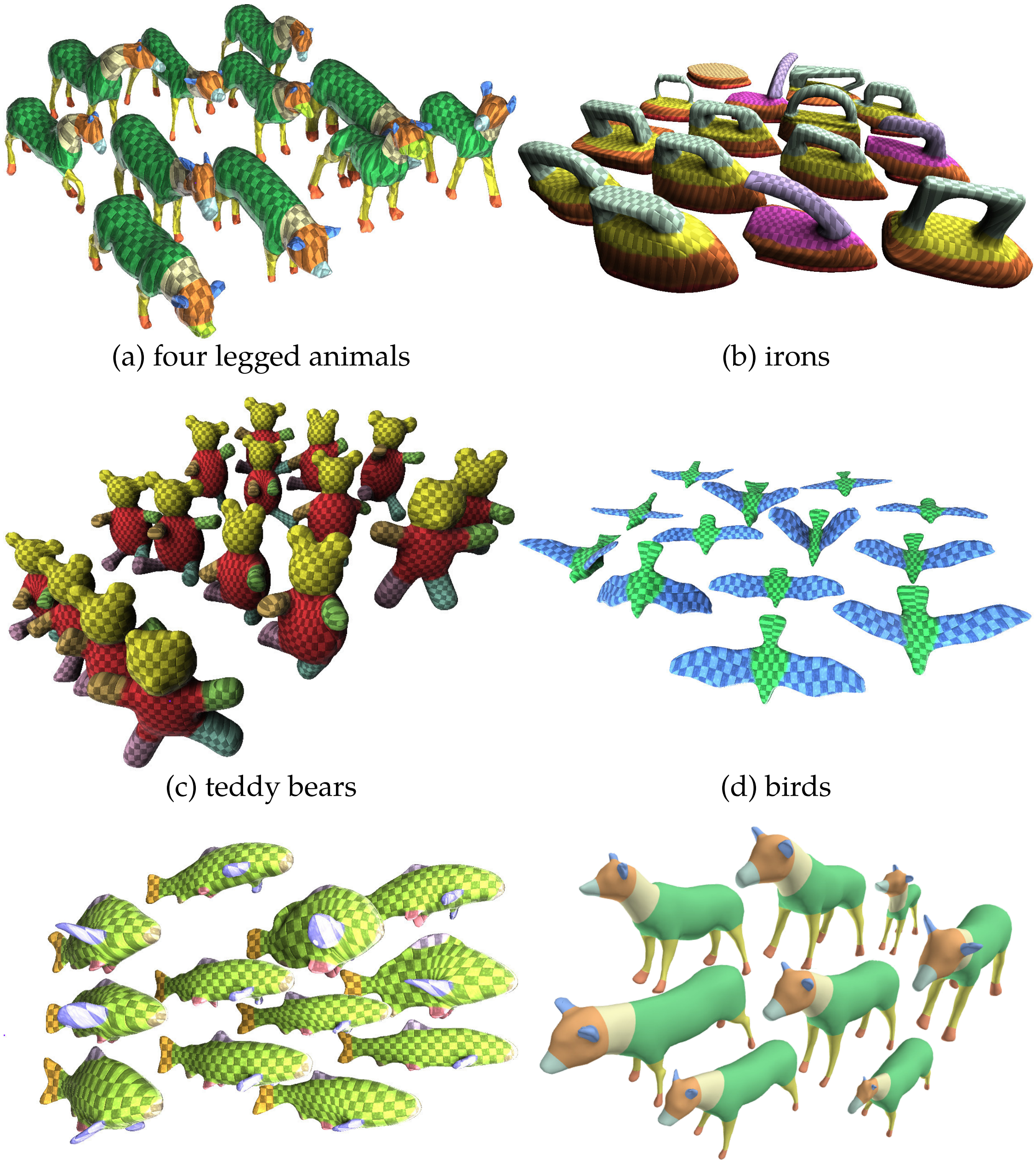
Thin-plate splines (TPS) are substantially better (Figure 4.9f), which was to be expected as this is the current standard solution for this type of matching problems [ACP03b; Has+09b]. Nonetheless, the TPS model still creates wrinkles and unwarranted drift. Our video shows various artifacts in the resulting shape spaces.

Intrinsic matching methods: In order to compare to recent state-of-the-art methods, we have also performed pairwise matches with blended intrinsic maps (BIM) [KLF11a]. As shown in Figure 4.9g, the pairwise partial isometries cannot capture the variation in this challenging data set well (please note though, that BIM is a global correspondence method; it solves a more difficult problem than our paper). Subsequently, we use the hub-and-spoke ensemble matcher of Huang et al. [Hua+12] (also a global matcher) that takes these results as input and performs a selection of best partial isometries in order to create consistent equivalence relations. The method improves the quality of the correspondences (Figure 4.9h), but substantial misalignment persists, which is seen best in the morphs shown in our video (remark: the output of their method is not as dense as the original vertices; we use interpolation to visualize the results; the same artifacts are also visible in the sparser output alone). In comparison, our method (Figure 4.9h) has a very good feature alignment and virtually no drift (see the video).

Modeling with deformable parts: The reconstruction from part graphs is formulated as an optimization problem. A variational approach permits us to easily include additional constraints. For example, any of the points can be fixed. We can for example use an energy $(\mathbf{v}_i - \mathbf{y})^2$ to implement handles that the user can attach to the shape for editing. Further, shape parameters can be prescribed. We use energies of the form $(\lambda_i - y)^2$ to control the shape of individual parts. We can also couple parameters of different parts (energies of the form $(\lambda_i - \lambda_j)^2$), for example, to keep shapes of the same type symmetric. The accompanying video shows some morphs between shapes with random shape space parameters, as well as an interactive editing session. A result of interactive editing is also shown in Figure 4.8c.

Impact of the part-based model: Using a part-based approach has a number of advantages: First, as shown in Figure 4.7a,b and in the video, we can capture discrete, topological variations in addition to continuous shape parameters: The irons consist of parts that can be assembled in different variations; the four-legged animals also distinguish open and closed mouths. Learning these shape families would be very challenging with global approaches. Despite the part-wise approach, our gradient-domain synthesis algorithm yields perfectly smooth boundaries in all cases (deactivating for example the smoothed connections illustrated in Figure 4.6c degrades the quality significantly). Further, our learning method benefits from symmetry within a shape; for example, all four legs in each animal share the same shape space, similarly the wings of the birds. We also obtain additional benefits: As illustrated in Figure 4.8a,b we can learn more compact shape spaces using well-chosen parts. A global models yield bad correspondences for strong entropy penalties (i.e., low values of μ_L). Reducing these improves the results, but the model then learns global correlations that are often unwanted. For example, in the case of the teddy bear in Figure 4.8c, global pose correlations are captured, which tilt the object against the position constraints on the chest; this that make harder and yields worse results (see also the video for an animated visualization). In summary, parts give us a more flexible model that allows us to integrate topologically diverse shapes and to learn shape spaces from fewer examples, avoiding overfitting.

Limitations: The most important limitation of our method is that it is a local optimization technique, thus requiring quite some user interaction as well as parameter



(a) four legged animals

(b) irons

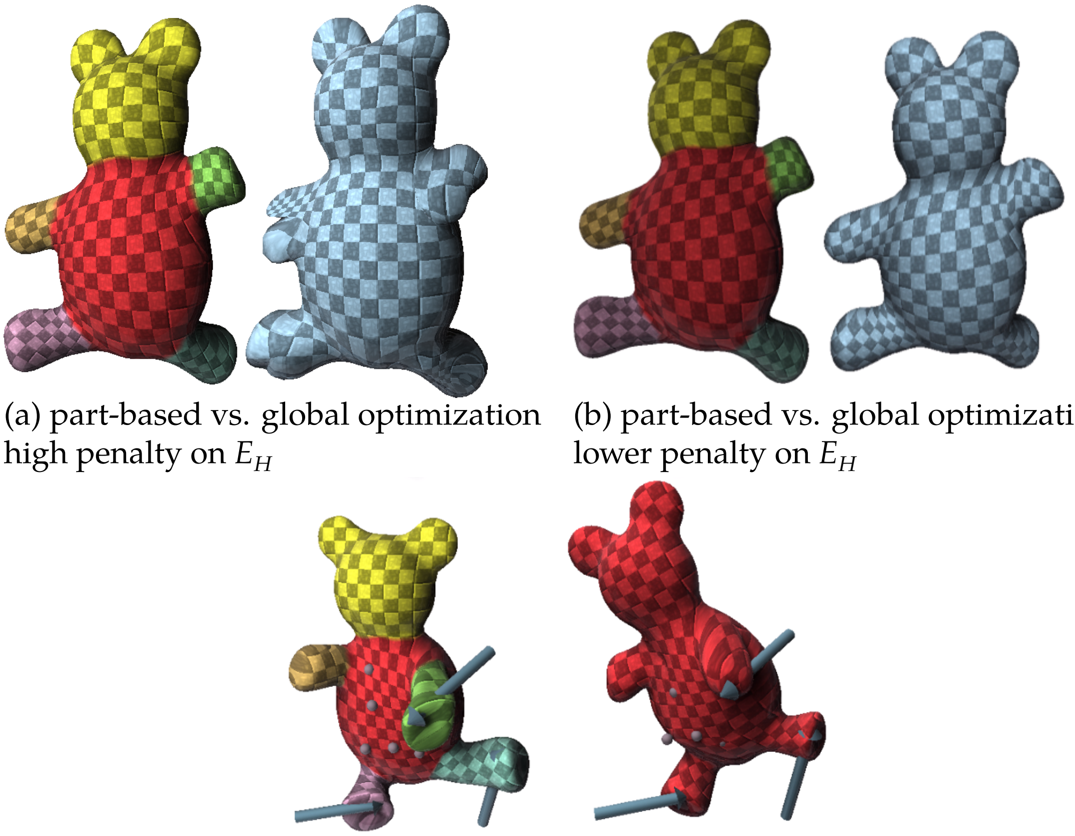
(c) teddy bears

(d) birds

(e) fish

(f) samples from the shape space of (a)

Figure 4.7.: Correspondences obtained with our method for different test data sets taken from the SHREC 2007 benchmark.

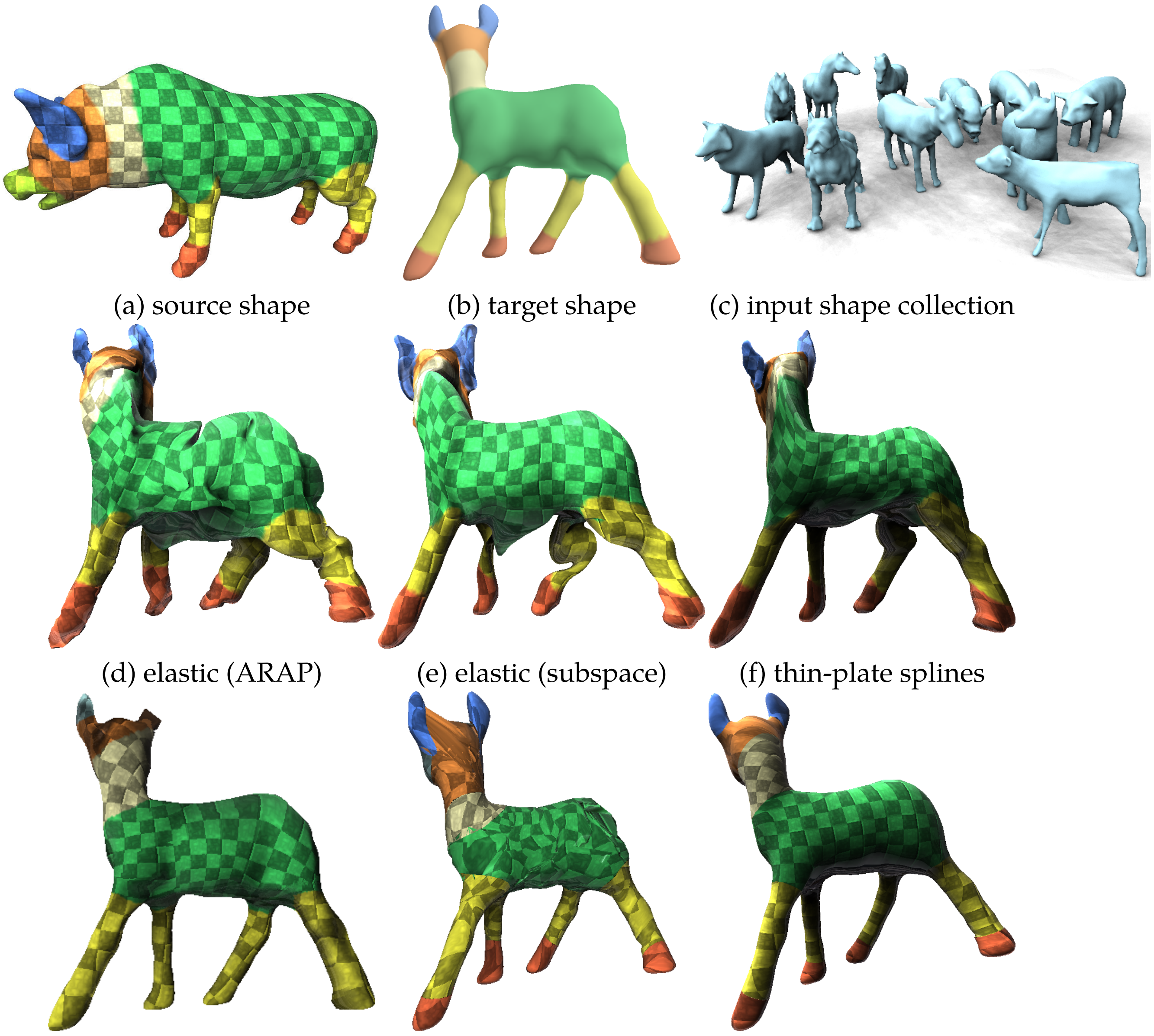


(a) part-based vs. global optimization high penalty on E_H

(b) part-based vs. global optimization lower penalty on E_H

(c) editing: part-based (left) vs. global (right)

Figure 4.8.: Impact of parts: (a) For high entropy penalties, a one-part model cannot capture the pose variations, while our method still yields good results. Using a lower weight (higher μ_L; factor 10) resolves the problem, albeit with higher entropy for the global method. (c) shows interactive editing with constraints. Here, the global model has overfit to global pose correlations while parts avoid the effect.



(a) source shape (b) target shape (c) input shape collection

(d) elastic (ARAP) (e) elastic (subspace) (f) thin-plate splines

(g) blended intrinsic maps (h) hub and spoke (i) our result

Figure 4.9.: Comparison to previous work. We map source shape (a) to target (b). We use deformable ICP based on as-rigid-as-possible (d) and subspace (e) deformation, as well as (f) thin-plate splines, all initialized with 15 landmarks. We have also employed (g) blended intrinsic maps, (e) ensemble optimization by Huang et al. [2012]. Our result is shown in (h); (g) and (h) are ensemble matches, optimized over 19 and 12 animals, respectively, as shown in (i). Please watch the accompanying video to see the impact on the resulting shape spaces; the differences are much more visible in morphing and sampling.

choices. Although we require only a coarse initialization, a too coarse annotation causes the algorithm to get stuck in a local optimum. A further, theoretical limitation is that we cannot formally guarantee bijectivity of correspondences, but we have not observed problems in practice. Finally, the hard-constraints for the surface constraints in the optimization limits the applicability to manifold input. Noisy and, in particular, incomplete data from 3D scans currently cannot be handled.

4.6. Conclusions

We have presented a new method for refining correspondences in families of shapes. By taking the compactness of the shape space into account as an optimization criterion, we obtain high-quality dense correspondences well-suited for the creation of shape spaces among shapes of considerable variability. In direct comparison, previous methods show substantial artifacts in such situations that we can avoid. Even difficult situations such as strong deformations and widely varying geometry yield good results. Our method handles objects of general topology, it handles challenging meshes with small feature sizes reliably, and is able to learn from objects of varying part composition, which can be used to synthesize new shapes with variable part configuration and continuous variability that adapts automatically to the designed part layout. Further, the part-based approach yields higher quality correspondences and is a useful tool to avoid overfitting.

In future work, we would like to extend the method towards fully automatic global matching, avoiding tedious manual initialization. Recent progress in co-segmentation would provide a starting point here, but a fully automatic method would require making our method robust to slight variations in part topology and outlier mismatches. In the long term, the question of how to build compact explanations from observed data is of fundamental importance. An ultimate modeling system with deformable parts would decompose shape collections automatically to obtain a shape grammar and various deformable, dockable shape spaces of parts, both optimized for compactness of encoding. While our model can in principle already handle such scenarios in terms of representation and synthesis, the automated analysis is the key challenge.

5 Embedding shapes with Green's functions for global shape matching

Abstract We present a novel approach for the calculation of dense correspondences between non-isometric shapes. Our work builds on the well-known functional map framework and investigates a novel embedding for the alignment of shapes. We therefore identify points with their Green's functions of the Laplace-Beltrami operator, and hence, embed shapes into their own function space. In our embedding the L_2 distances are known as the biharmonic distances, so that our embedding preserves the intrinsic distances on the shape. In the novel embedding each point-to-point map between two shapes becomes and can be represented as an affine map. Functional constraints and novel conformal constraints can be used to guide the matching process.

This chapter corresponds to the paper[BDK17b]: Oliver Burghard, Alexander Dieckmann, and Reinhard Klein. "Embedding shapes with Green's functions for global shape matching". In: *Computers & Graphics* (2017).

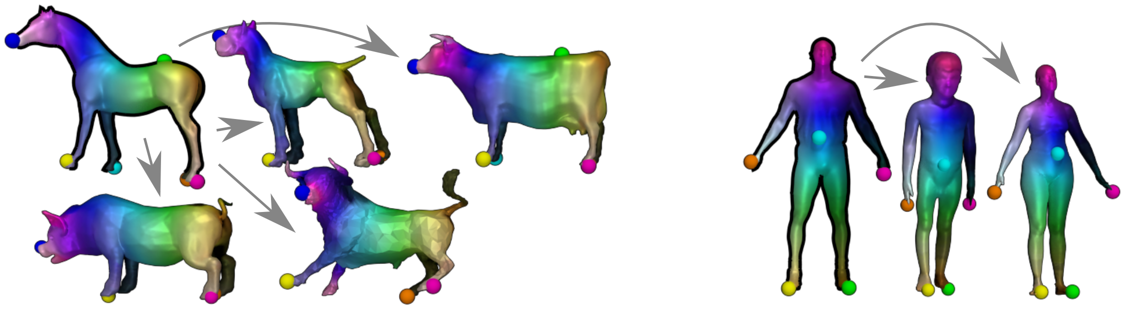


Figure 5.1.: A variety of maps generated by aligning the Green's embeddings. The color from the source shape (black contour) is mapped onto the target shapes using the sparse correspondences depicted by small spheres.

5.1. Introduction

Finding correspondences between two or more shapes is an important sub-task for a variety of applications, in which information has to be transferred or correlated between shapes. For example, local deformations can be transferred for shape editing [SP04; Sum+05b; KMP07], and correlations between corresponding regions can be exploited to compress dynamic meshes [SSK05; Váš+14] and to create generative shape models [BV99b; ACP03a; Has+09a].

Finding correspondences is especially interesting between non-isometric shapes, see Figure 5.1 for some examples. Many previous approaches tailored to register isometric shapes fail in this case. Extrinsic non-rigid ICP [ARV07; ACP03a] and variants [SP04; Yeh+11; Bur+13b; Yos+14] suffer from unreliable correspondences on extrinsic distances and from difficulties in solving the non-linear deformation models. The Blended Intrinsic Maps method [KLF11b; LF09b] replaces the extrinsic metric by an intrinsic one and delivers good registration results by assuming the resulting maps to be locally conformal. A problem of BIM is that it is not clear how to incorporate a priori constraints which might be necessary to guide the method to the correct map out of the multiple reasonable ones (Figure 5.2). Furthermore, the stitching of local maps leads to inconsistencies at their boundaries. Another group of approaches embeds shapes into a high dimensional space, where L_2 distances approximate intrinsic distances. Although most of them allow incorporating additional constraints, many share the major drawback that their embedding requires a *non-linear* alignment.

Functional maps [Ovs+12] overcome this problem by constructing an embedding, in which shapes can be aligned with a *linear* deformation. Unfortunately L_2 distances of delta-distributions, that are typically used to embed points, only approximate intrinsic distances between intrinsically close points, see Figure 5.4. This is especially important when only a few implicit constraints are available, such as when matching non-isometric shapes.

In contrast to functional maps [Ovs+12] we identify points with their Green's functions of the Laplace-Beltrami operator. In this embedding the L_2 distances are the well-known biharmonic distances [LRF10b], which are an intrinsic distance metric on the shape. They are invariant to isometric shape deformations so that pose deformations have little influence on the matching process. We calculate correspondences by aligning these embeddings with an *affine* deformation, which can be computed reliably and efficiently. There is a linear relation between the Green's alignment and the (pullback) functional map [Ovs+12], so that we can incorporate functional constraints and operator commutativity into our setting. Last but not least, we can include additional constraints on the alignment, which require the resulting map to be close to conformal.

The main contributions of our paper are (a) a novel embedding of shapes in the functional map framework by identifying points with their Green's functions, (b) combining our novel embedding with functional constraints and (c) including conformality constraints into functional shape matching. The paper is organized as follows:

Section 5.2 discussed the related work. Section 5.3 introduces the alignment of shapes with Green's functions and relates it to the functional maps framework. We further motivate the novel embedding by a comparing Green's functions and delta-distributions in Section 5.4. Then Section 5.5 shows how to utilize functional constraints and conformality in the matching process. After discussing the discretization in Section 5.6 we describe a shape matching algorithm in Section 5.7, which is evaluated in Section 5.8.

5.2. Related work

Estimating correspondences between different shapes is a challenging task that has been addressed intensively in literature. In this section, we only provide a brief overview on directly related works and kindly refer the interested reader to the recent surveys [Van+11; Tam+13].

ICP. Initially the problem of shape matching appeared in the context of registering sequential point cloud scans of a static scene. This led to the development of rigid ICP algorithms [RL01b], which alternate between detecting corresponding points and rigidly aligning shapes. Due to the local optimization, these techniques depend strongly on the initial correspondences and on heuristics to prune novel correspondences. A variety of methods extend the original ICP metaphor to match deformed shapes by allowing non-rigid deformations in $\mathbb{R}^3$ for the alignment [ACP03a; SP04; ARV07; Hua+08a; Yeh+11; Bur+13b; Yos+14]. A common shortcoming of these methods is the detection of corresponding points based on extrinsic instead of intrinsic distances. For deformable shapes extrinsic distances can be small even for intrinsically distant points. As a consequence these methods typically require numerous point-to-point constraints to begin with and utilize sophisticated heuristics to prune novel correspondences.

The Blended Intrinsic Maps method [KLF11b] obtains good results by concatenating and blending multiple conformal maps into a single global map. However, it cannot incorporate user constraints, which are sometimes necessary to solve ambiguities (e.g. Figure 5.2). Furthermore, at the boundaries of the local maps the results often exhibit discontinuities. Additionally, the method is difficult to generalize to point-clouds or shapes of genus other than zero.

Other methods map shapes by parameterizing them on a common domain and then aligning their parameterizations so that either an intrinsic measure of stretch

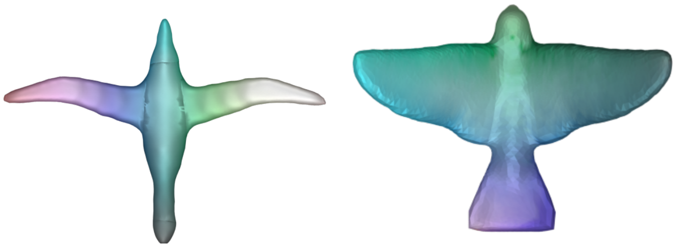


Figure 5.2.: Failure case of Blended Intrinsic Maps, that are difficult to resolve without predefined constraints.

from source to target becomes minimal [KS04b; APL14; APL15; AL15; AL16] or so that the integrated stretch along a sequence of deformations from the source onto the target [Kur+13; Kur+12; Lag+16] becomes minimal. These methods deliver continuous maps of high quality, but are often computational demanding and their application on non-simple topologies is non-trivial.

Yet another class of methods uses an ICP-like alignment *after* embedding shapes into a high dimensional space where L_2 distances approximate intrinsic ones. Shapes have been embedded with the eigenvectors of an affinity matrix [JZ06; SY12], with an embedding approximating geodesic distances [AK13], with an embedding based on electrostatic repulsion [BGB14] and with delta-functions [RMC15]. All of these methods use non-linear maps to align the embeddings. Slightly different are the methods [BBK06b; ADK13], where the alignment of shapes is avoided by directly embedding one shape into the other by minimizing a non-linear functional.

Functional maps. The remarkably successful functional maps framework was introduced in [Ovs+12]. To the best of our knowledge this paper was the first to fully exploit the fact that a *linear* alignment of a functional embedding is sufficient to represent arbitrary non-linear alignments in $\mathbb{R}^3$. This significantly simplifies the matching process. Additionally, the authors demonstrate the usefulness of functional constraints, such as matching labeled regions. Particularly for isometric shapes, where many functional constraints are available [SOG09; ASC11] and the alignment is a rotation commuting with the Laplacian, superior results have been achieved. Furthermore, the resulting maps can be optimized with an ICP-like alignment algorithm after embedding the shapes with delta-distributions.

Typically, when matching non-isometric shapes there are few a priori constraints available and the ICP-like alignment becomes especially important. In this case a drawback of their embedding emerges, namely the L_2 distances on delta-distributions, which are a critical ingredient for a ICP-like method, are not well-defined. As we describe in Section 5.4, the distances are only defined after projecting the delta--

distributions onto the first eigenvectors of the Laplace operator and the distance metric depends strongly on the number of eigenvectors that were used. The more eigenvectors are used the more localized these distances become. In our work we therefore provide a novel embedding for shape matching that respects intrinsic distances.

Functional maps initiated a series of publications, such as improving the extraction of correspondences [Ngu+11b; COC15; RMC15] utilizing low-rank assumptions on the functional map [Kov+13; Kov+15] (which still requires multiple initial constraints typically not available for non-isometric shapes) or investigating the matching of shape collections [Ngu+11b; HWG14].

Previous work represented points as Green's functions[LRF10b] and as the related Global Point Signatures [Rus07]. The invariance of the Dirichlet energy to conformal deformations was prominently explored in [Rus+13a]. Yet to the best of our knowledge, we are the first to compute correspondences by aligning shapes represented with Green's functions and the first to use functional correspondences to approximate the functional map representation of a conformal map.

5.3. Embedding with Green's functions

We start our exposition by describing the embedding of a shape $\mathcal{M}$ using Green's functions. Let $\mathcal{L}^2(\mathcal{M}) = \{f\colon \mathcal{M} \to \mathbb{R} \mid \int_{\mathcal{M}} f^2(x)\,\mathrm{d}x < \infty\}$ be the set of square integrable, real-valued functions on $\mathcal{M}$ and let δ_p be the delta-distribution at a point p, i.e. $\langle \delta_p, f\rangle = f(p)\ \forall f \in \mathcal{L}^2(\mathcal{M})$. Let further $\Delta\colon \mathcal{L}^2(\mathcal{M}) \to \mathcal{L}^2(\mathcal{M})$ be the Laplace-Beltrami operator and its spectral decomposition have the eigenvectors $\phi_i \in \mathcal{L}^2(\mathcal{M})$ and the eigenvalues $\lambda_i \in \mathbb{R}$ $(\lambda_1 \leq \lambda_2 \leq \dots)$. Then the Green's function g_p of the Laplace operator Δ at point p is the solution of the equation:

$$\Delta g_p = \delta_p \tag{5.1}$$

This equation has a solution if and only if δ_p is orthogonal to the null-space of Δ. For a simple exposition, we assume that $0 = \lambda_1 < \lambda_2 \neq 0$, which is the case for a compact, simply-connected shape. Hence, the null-space of Δ is the subspace of constant functions, which is spanned by the first eigenvector ϕ_1. We further write Π for the orthogonal projection on the complement of this null-space, i.e. $\Pi(f) := f - \phi_1\langle\phi_1, f\rangle$. Therefore, in our context we define the Green's function of the Laplace operator of point $p \in \mathcal{M}$ as the solution to the slightly different equation

$$\Delta g_p = \Pi\left(\delta_p\right) = \delta_p - \phi_1\langle\phi_1, \delta_p\rangle \qquad\qquad \langle g_p, \phi_1\rangle = 0\,, \tag{5.2}$$

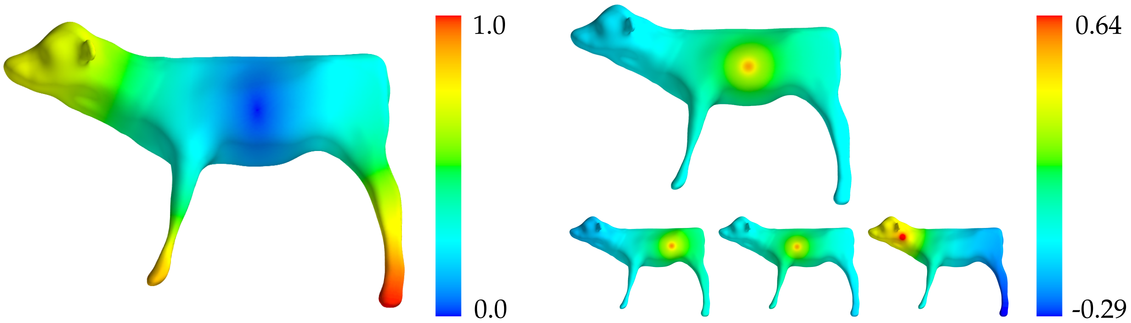


Figure 5.3.: (left) Biharmonic distances $d_b(p, x)$ from a fixed point p and (right) its Green's function g_p and three other Green's functions.

which now always has a solution that can be written with the pseudo-inverse Δ^+ as

$$g_p = \Delta^+ \delta_p = \sum_{i=2}^{\infty} \phi_i \frac{\phi_i(p)}{\lambda_i} \, . \tag{5.3}$$

We state a few properties, for the upcoming discussion (proof see appendix):

Theorem 10. *(a) The mapping from a surface $\mathcal{M}$ onto its Green's functions $g_p^{\mathcal{M}}$ is injective, thus an embedding. (b) The functions $\{g_p^{\mathcal{M}} + \phi_1 \mid p \in \mathcal{M}\}$ form a basis of $\mathcal{L}^2(\mathcal{M})$. (c) The functions $\{g_p^{\mathcal{M}} \mid p \in \mathcal{M}\}$ are linearly dependent and span a subspace of $\mathcal{L}^2(\mathcal{M})$ of co-dimension 1 (see the Appendix).*

Several Green's functions can be seen in Figure 5.3. The further points are located on the surface, the more their Green's functions differ. L_2 distances on the Green's functions are called biharmonic distances [LRF10b] and the left of Figure 5.3 shows the distance fields of several points.

Matching shapes. In the following we utilize the properties of Green's functions for the construction of a mapping $T \colon \mathcal{M} \to \mathcal{N}$ between the shapes $\mathcal{M}$ and $\mathcal{N}$. As Green's functions define a distance field on the surface, we can recover a point from its Green's function simply by determining the surface point whose Green's function is most similar. Therefore, Green's functions represent the *intrinsic* location of a point on the shape. We can therefore solve for a map T by solving for a map $G \colon \mathcal{L}^2(\mathcal{M}) \to \mathcal{L}^2(\mathcal{N})$ which aligns the Green's functions of $\mathcal{M}$ onto the Green's functions of $\mathcal{N}$ and only later calculate T from G. For clarification, we add superscripts to the involved quantities on $\mathcal{M}$ and $\mathcal{N}$. Then such an alignment G has to fulfill:

$$G\left(g_p^{\mathcal{M}}\right) = g_{T(p)}^{\mathcal{N}} \;\; \forall p \in \mathcal{M} \qquad\qquad G\left(\phi_1^{\mathcal{M}}\right) = 0 \tag{5.4}$$

G is well-defined as due to Prop. (10a) different points have different Green's functions. Actually there is one *unique* affine map G which fulfills Eq. (5.4). It can be written as $G(f) = B(f) + t$ where $B\colon \mathcal{L}^2(\mathcal{M}) \to \mathcal{L}^2(\mathcal{N})$ is a linear map (i.e. $B(f + \lambda g) = B(f) + \lambda B(g)\ \forall \lambda \in \mathbb{R} \forall f, g \in \mathcal{L}^2(\mathcal{M})$) and $t \in \mathcal{L}^2(\mathcal{N})$. Choosing G as an affine map simplifies solving for G and allows the inclusion of least squares constraints. To see why choosing G as an affine map is possible, note that due to Prop. (10b) there is a unique linear map $\tilde{G}$ with

$$\tilde{G}\left(g_p^{\mathcal{M}} + \phi_1^{\mathcal{M}}\right) = g_{T(p)}^{\mathcal{N}} + \phi_1^{\mathcal{N}}\ \forall p \in \mathcal{M}\,. \tag{5.5}$$

Using $\tilde{G}$ one can write $G(f) = B(f) + t$ as $t = \tilde{G}\left(\phi_1^{\mathcal{M}}\right) - \phi_1^{\mathcal{N}}$ and $B(f) = \tilde{G}(f) - \langle f, \phi_1^{\mathcal{M}}\rangle(\phi_1 + 2t)$. G is unique, because it can be used to write $\tilde{G}$ using $\tilde{G}(f) = B(f) + \langle f, \phi_1^{\mathcal{M}}\rangle(\phi_1 + 2t)$ and from $\tilde{G}$ one can infer T.

Yet not every map G is induced by some point-wise map T. If G is induced by some point-wise map T, then it aligns the Green's functions:

$$\forall p \in \mathcal{M}\ \exists q_p \in \mathcal{N}\colon G\left(g_p^{\mathcal{M}}\right) = g_{q_p}^{\mathcal{N}} \tag{5.6}$$

On the other hand, if G fulfills the last equation, we can reconstruct T via $T\colon \mathcal{M} \to \mathcal{N}, p \mapsto q_p$. The above results are summarized in the following proposition:

Theorem 11. *For each point-wise map T, Eq. (5.4) defines a unique affine map G, that maps the Green's functions of $\mathcal{M}$ onto the Green's functions on $\mathcal{N}$. Such an alignment of Green's functions supports Eq. (5.6), which can be used to restore T from G. Therefore, there is an one-to-one relation of point-wise maps T and the Green's alignment G supporting Eq. (5.6).*

Pullback Functional maps. The affine map G is related to the pullback functional maps F_T introduced in section 2.4:

$$F_T\colon \mathcal{L}^2(\mathcal{N}) \to \mathcal{L}^2(\mathcal{M}) \qquad\qquad f \mapsto f \circ T \tag{5.7}$$

Both F and G map functions between $\mathcal{M}$ and $\mathcal{N}$ and are thus "functional maps". For the sake of a clear notation we refer to T as a *point-wise* map, to F as the corresponding *pullback functional map* and to G as the corresponding *alignment of Green's functions*. Because F preserves function values at corresponding points, it aligns (dual) delta-distributions ($\cdot^T$ denotes transposed):

$$\begin{aligned} \langle \delta_p^{\mathcal{M}}, F(f)\rangle_{\mathcal{M}} = F(f)(p) = f(T(p)) = \langle \delta_{T(p)}^{\mathcal{N}}, f\rangle_{\mathcal{N}} \\ \Rightarrow \quad F^T \delta_p^{\mathcal{M}} = \delta_{T(p)}^{\mathcal{N}} \quad \forall p \in \mathcal{M} \end{aligned} \tag{5.8}$$

Note: The original publication on functional maps [Ovs+12](Section 6.1) assumes that functional maps defined by Eq. (5.7) align delta-distributions:

$$\delta_p^{\mathcal{M}} = F\delta_{T(p)}^{\mathcal{N}} \ \forall p \in \mathcal{M} \tag{5.9}$$

This is true *only for area-preserving* maps, which fulfills $F^{-1} = F^T$. Interestingly the difference of Eq. (5.8) and Eq. (5.9) seems to have had limited effect in previous works. One reason might be, that for any functional map F defined by Eq. (5.7) there is another map $F' = F^{-T}$ that aligns delta-distributions. The difference of F and F' depends on the change in the area-form and is often small.

Theorem 12. *The alignment of Green's functions G can be written in terms of the pullback functional map F ($A_{\mathcal{M}}$ is the area of $\mathcal{M}$, see the appendix) as*

$$G(f) = \Delta_{\mathcal{N}}^{+} F_T^T \Delta_{\mathcal{M}}(f) + \frac{1}{\sqrt{A_{\mathcal{M}}}} \Delta_{\mathcal{N}}^{+} F_T^T \phi_1^{\mathcal{M}} \tag{5.10}$$

and F can be written in terms of $G(f) = B(f) + t$ as

$$F = \Delta_{\mathcal{M}}^{+} B^T \Delta_{\mathcal{N}} + \sqrt{A_{\mathcal{M}}} \phi_1^{\mathcal{M}} t^T \Delta_{\mathcal{N}} + \sqrt{\frac{A_{\mathcal{M}}}{A_{\mathcal{N}}}} \phi_1^{\mathcal{M}} \phi_1^{\mathcal{N},T} . \tag{5.11}$$

Each constraint on F has an equivalent constraint on G and vice versa. The final correspondences do not depend on whether one optimizes for F or G. They depend on the involved constraints to solve for F and G and on the method to extract correspondences from F and G.

The point-wise maps T induce only a fraction of all possible functional maps. Not only does any induced functional map align dual delta-distributions (Eq. (5.8)), but also any functional map F which does so according to Eq. (5.12) is induced by a point-wise map T. Thus, the alignment of dual delta-functions by F and the alignment of Green's functions by G are equivalent conditions.

$$\forall p \in \mathcal{M} \ \exists q_p \in \mathcal{N}\colon \ F^T \delta_p^{\mathcal{M}} = \delta_{q_p}^{\mathcal{N}} \tag{5.12}$$

Theorem 13. *Let F be a pullback functional map and G an alignment of Green's functions which are related by Eq. (5.10) or (5.11), then the following statements are equivalent:*

(i) G aligns Green's functions by Eq. (5.6)
(ii) F aligns dual delta-distributions by Eq. (5.12)
(iii) There is a point-wise map T inducing F by Eq. (5.7) or G by Eq. (5.4)

5.4. Green's functions vs delta-distributions

Next we discuss the differences between our novel embedding and the original embedding with delta-distributions and why it matters for non-isometric shape matching. Aligning shapes with an ICP-like algorithm is a minimization of the non-linear ICP energy functional:

$$E[G;T] = \int_{\mathcal{M}} \left\| G\left(g_p^{\mathcal{M}}\right) - g_{T(p)}^{\mathcal{N}} \right\|_2^2 \,\mathrm{d}p \tag{5.13}$$

This functional depends on the L_2 distances in the embedding. For Green's functions these distances are known as biharmonic distances - a well-defined intrinsic distance metric [LRF10b]. In contrast, delta-distributions are not square integrable ($\delta_p^{\mathcal{M}} \notin \mathcal{L}^2(\mathcal{M})$) and their L_2 distances are therefore not well-defined. To replace the Green's functions in Eq. (5.13) with delta-distributions they first have to be projected onto a finite subspace. For example, the authors of [Ovs+12] (in Section 6.1) embed the shapes with delta-distributions projected onto the first k eigenvectors of the Laplace operator:

$$p \in \mathcal{M} \mapsto \delta_p^{(k)} = \sum_{i=1}^{k} \phi_i \phi_i(x) \in \mathcal{L}^2(\mathcal{M}) \tag{5.14}$$

Unfortunately, the emerging distances approximate intrinsic distances only for a few eigenvectors (e.g. $k < 10$), while for large k the L_2 distances do no more approximate intrinsic distances, but only discriminate points:

$$\left\| \delta_p^{(k)} - \delta_q^{(k)} \right\|^2 = \sum_{i=1}^{k} \left(\phi_i(x) - \phi_i(y)\right)^2 \xrightarrow{k\to\infty} \begin{cases} 0 & x = y \\ \infty & \text{else} \end{cases} \tag{5.15}$$

Eventually also the Green's functions have to be approximated by a finite basis. In contrast to L_2 distances on delta-distributions the biharmonic distances are well approximated with the first few eigenvectors of the Laplace operator. Approximations of both distance fields with different numbers of eigenvectors are shown in Figure 5.4. For 1000 eigenvectors delta-distributions only discriminate the query point in agreement with Eq. (5.15). The effect of the different embeddings on the alignment process is shown in Figure 5.5, where two shapes are matched with a ICP-like method that will be described in Section 5.7 (no conformality, $\alpha = 0$). Using a few eigenvectors ($k = 5$) the results using delta-distributions and Green's functions are similar, while for 20 eigenvectors the alignment with delta-distributions becomes discontinuous, even after iterating. This is especially the case, when only few a priori correspondences are known, which is typical for non-isometric matching.

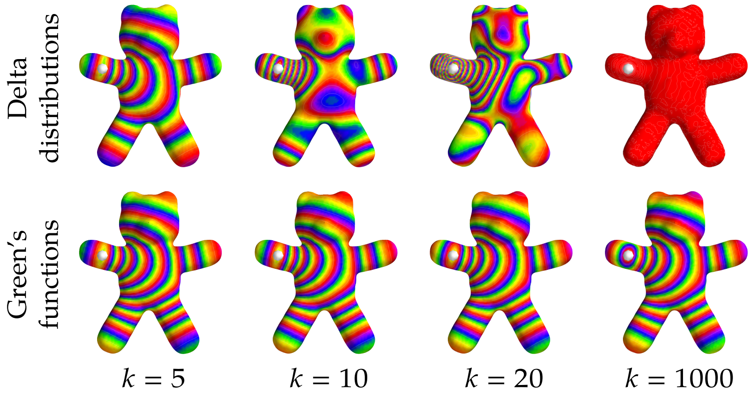


Figure 5.4.: Biharmonic distances and L_2 distances on delta-distributions for different numbers of eigenvectors.

5.5. Functional constraints and conformality

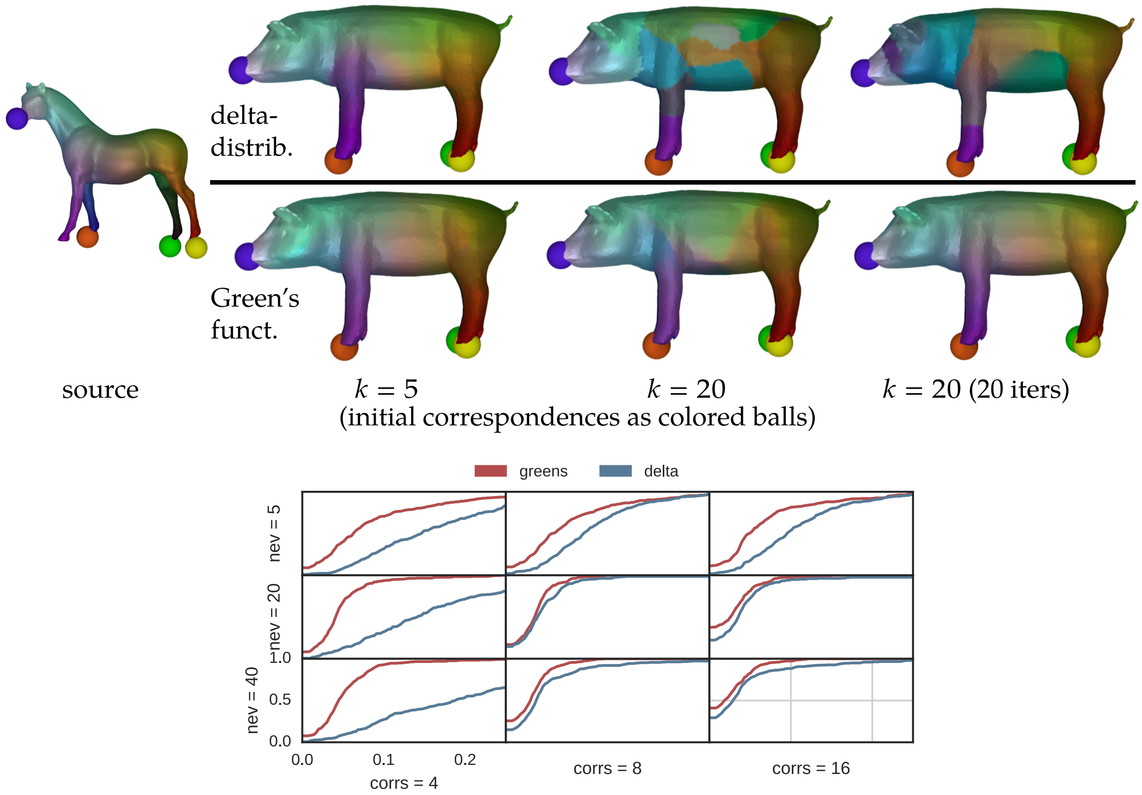
Next we utilize the linear relation between F and G to transfer functional constraints and operator commutativity into our setting. Afterwards we propose novel functional constraints for conformal maps.

Functional constraints. Instead of localizing correspondences at single points, it is often more appropriate to determine corresponding regions, or equivalently to require indicator functions to match. This is an instance of a functional constraint, where the map T is known to pull a function $f_{\mathcal{N}} \in \mathcal{L}^2(\mathcal{N})$ back onto another function $h_{\mathcal{M}} \in \mathcal{L}^2(\mathcal{M})$:

$$Ff_{\mathcal{N}} = h_{\mathcal{M}} \tag{5.16}$$

Functional constraints emerge in other applications as well. For example isometric shapes have the same heat- and wave-kernel-signatures [SOG09; ASC11]. As functional constraints are linear constraints in F, they can be written as a linear constraints in G using Eq. (5.11).

Operator commutativity. If we have a functional operator on each shape and these operators have the same effect on equivalent functions, then these operators commute with the functional map F. In this case applying the first operator followed by a projection onto the other shape has the same effect as first projecting onto the other shape followed by applying the second operator there. A typical example for operator commutativity are intrinsic symmetries or the Laplace-Beltrami operator



Cumulative distribution of L_2 distances.

Figure 5.5.: Matching two shapes from the "fourlegged" dataset from point-to-point constraints using either delta-distributions or Green's functions.

for isometric shapes where we have:

$$\Delta_{\mathcal{M}} F = F \Delta_{\mathcal{N}} \tag{5.17}$$

Using Eq. (5.11) this linear constraint on F becomes a linear constraint on G.

Conformal maps. Additionally we introduce novel constraints on F and G by assuming that T approximates a conformal map. A conformal map is a locally angle preserving, continuous map. The class of conformal maps includes the isometric maps and is general enough to match typical shapes of equivalent topology. At the same time a conformal map is already determined by a few known correspondences [LF09b]. Furthermore, conformal maps were successfully used in previous work to match near-isometric shapes [LF09b; KLF11b].

Conformality does not only restrict the map T, but also the functional map F and the Green's alignment G. For two functions $f, h \in \mathcal{L}^2(\mathcal{N})$ the Dirichlet energy is defined by

$$E_D[f,h] = \langle \nabla_{\mathcal{N}} f, \nabla_{\mathcal{N}} h \rangle_{\mathcal{N}} = \langle \Delta_{\mathcal{N}} f, h \rangle_{\mathcal{N}} \,.$$

and measures how much their gradients agree. A point-wise map is conformal if and only if its functional representation preserves the Dirichlet energy [Rus+13a]:

$$\langle \Delta_{\mathcal{M}} F(f), F(h) \rangle_{\mathcal{M}} = \langle \Delta_{\mathcal{N}} f, h \rangle_{\mathcal{N}}$$

or equivalently

$$F^T \Delta_{\mathcal{M}} = \Delta_{\mathcal{N}} F^{-1} \,. \tag{5.18}$$

Eq. (5.18) is an instance of a non-linear Procrustes problem, namely that $\Delta_{\mathcal{M}}^{1/2} F \Delta_{\mathcal{N}}^{-1/2}$ is orthogonal [GD04; Vik06]. We propose to use the current functional and point-to-point constraints to transfer Eq. (5.18) into linear constraints instead.

Each functional constraint $Ff_{\mathcal{N}} = h_{\mathcal{M}}$ is equivalent to $f_{\mathcal{N}} = F^{-1} h_{\mathcal{M}}$, which is combined with Eq. (5.18) into

$$(\Delta_{\mathcal{N}}^{+} F^T \Delta_{\mathcal{M}})\,(f_{\mathcal{M}}) = \Pi_{\mathcal{N}}(f_{\mathcal{N}}) \,, \tag{5.19}$$

and each point-to-point constraint $G\left(g_p^{\mathcal{M}}\right) = g_q^{\mathcal{N}}$ is equivalent to $g_p^{\mathcal{M}} = G^{+}\left(g_{T(p)}^{\mathcal{N}}\right)$, which we combined with Eq. (5.18) into

$$(\Delta_{\mathcal{M}}^{+} B^T \Delta_{\mathcal{N}})\left(g_{T(p)}^{\mathcal{N}}\right) + \frac{1}{\sqrt{A_{\mathcal{N}}}} \Delta_{\mathcal{M}}^{+} F^{-T} \phi_1^{\mathcal{N}} = g_p^{\mathcal{M}} \tag{5.20}$$

where $G^{+}(f) = \Delta_{\mathcal{M}}^{+} F^{-T} \Delta_{\mathcal{N}}(f) + \frac{1}{\sqrt{A_{\mathcal{N}}}} \Delta_{\mathcal{M}}^{+} F^{-T} \phi_1^{\mathcal{N}}$ is the inverted Green's alignment.

Theorem 14. *Let $T \colon \mathcal{M} \to \mathcal{N}$ be a conformal, point-wise map with the corresponding functional map F and the corresponding Green's alignment G, then F adheres to the Eq. (5.18). If F fulfills a functional constraint $Ff_{\mathcal{N}} = h_{\mathcal{M}}$, then it also fulfills Eq. (5.19) and if G fulfills a point-to-point correspondence $G\left(g_p^{\mathcal{M}}\right) = g_q^{\mathcal{N}}$, then it also fulfills Eq. (5.20).*

Area preserving maps ($F^T = F^{-1}$) have similar constraints, whose investigation we leave for future work.

Area preserving maps ($F^T = F^{-1}$) have similar constraints, whose investigation we leave for future work.

Conformal maps and Green's functions. It is interesting how Green's functions change under a conformal map. Let $T \colon \mathcal{M} \to \mathcal{N}$ be a conformal map with the corresponding functional map F and the corresponding Green's alignment G,

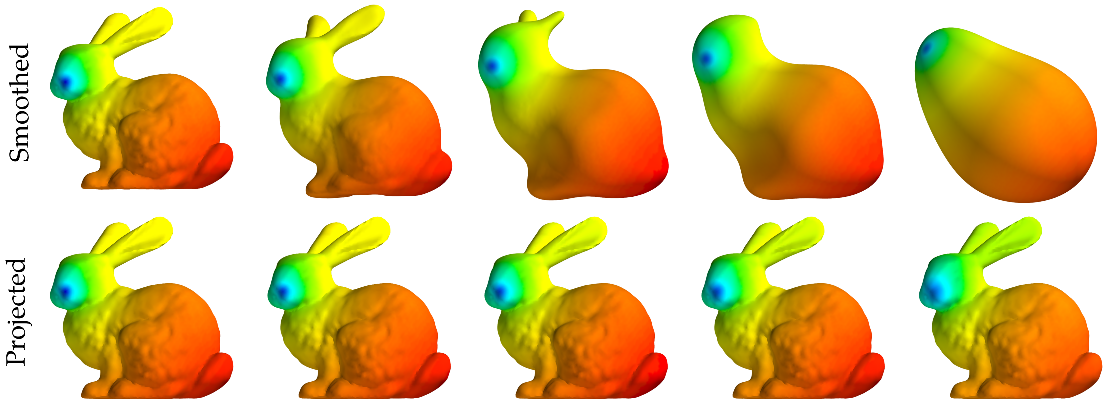


Figure 5.6.: Smoothed versions of the Stanford bunny colored by the Green's function of a vertex at the nose. To illustrate the preservation of Green's functions, the Green's functions from the deformed meshes were mapped onto the original mesh in the lower row.

then the Green's function of a point $p \in \mathcal{M}$ and the Green's function of the mapped point image $T(p) \in \mathcal{N}$ differ by (see the appendix):

$$F^{-1} g_p^{\mathcal{M}} - g_{T(p)}^{\mathcal{N}} = \phi_1^{\mathcal{N}} \langle \phi_1^{\mathcal{N}}, F^{-1} g_p^{\mathcal{M}} \rangle + \Delta_{\mathcal{N}}^{+} F^T \phi_1^{\mathcal{M}} \frac{1}{\sqrt{A_{\mathcal{M}}}} \tag{5.21}$$

$$= \phi_1^{\mathcal{N}} \sum_{i=2}^{\infty} \frac{\left(F^{-1}\right)_{1i}}{\lambda_i^{\mathcal{M}}} + \frac{1}{\sqrt{A_{\mathcal{M}}}} \sum_{i=2}^{\infty} \phi_i^{\mathcal{M}} \frac{(F)_{1i}}{\lambda_i^{\mathcal{N}}} \tag{5.22}$$

The values F_{1i} and $\left(F^{-1}\right)_{1i}$ represent the mean of the mapped eigenvectors on $\mathcal{N}$, which typically are small if the area form changed little, in which case we expect little difference between the original and the mapped Green's function. Figure 5.6 shows the Stanford bunny and multiple increasingly smoothed versions of it. During the smoothing a conformal map between the meshes was obtained [CPS11]. Projecting the Green's functions from the smoothed meshes onto the original mesh shows that Green's functions of a single point indeed changed little. Apart from the observation above our work does not exploit the connection of conformal maps and Green's functions further, but leaves this interesting issue open for future work.

5.6. Discretization

To represent functions and with them the Green's alignment G a basis of the function spaces $\mathcal{L}^2(\mathcal{M})$ and $\mathcal{L}^2(\mathcal{N})$ is required. We use the first k eigenvectors of the Laplace operator $[\phi_1, \phi_2, \dots, \phi_k]$, which is a common choice in literature [Ovs+12].

Practical evaluation shows that biharmonic distances can be well approximated with as little as 5 eigenvectors (Figure 5.4).

Spectral decomposition. Next we discuss the calculation of the eigenvalues λ_i and eigenvectors ϕ_i of a shape's Laplace-Beltrami operator Δ. Let functions on the surface be represented by vectors, whose coefficients are the function values at the mesh vertices or point-cloud points. Let W be the matrix representation of the scalar product on functions and L be the matrix representation of the Laplace-Beltrami operator, which is typically written as $L = W^{-1}C$, where C is the Dirichlet energy on functions. The eigenvectors and eigenvalues of L are defined by the following generalized eigenvalue problem:

$$C\phi_i = \lambda_i W\phi_i \text{ so that } \phi_i^T W\phi_j = \delta_{ij} \text{ and } \lambda_1 \leq \lambda_2 \leq \ldots$$

For triangle meshes we utilize the well-known Cotan Laplacian [PP93; Mey+03]. We estimate the vertex areas as a third of the sum of the adjacent triangle areas and set W to be a diagonal matrix of the estimated vertex areas. Furthermore,

$$(Cu)_i = \sum_{j\in\mathcal{N}(i)} \frac{1}{2}\,(\cot\alpha_{ij} + \cot\beta_{ij})(u_i - u_j) \tag{5.23}$$

defines the Dirichlet energy, where $\mathcal{N}(i)$ is the 1-neighborhood of the vertex i and α_{ij}, β_{ij} are the two angles opposing the edge from vertex i to vertex j.

For point-clouds (e.g. Figure 5.11a) we use a variant of the Laplace operator from Belkin and Liu [BSW09; LPG12]. They define L so that for a small time t heat-diffusion matches the Euclidean one. With the point positions $p_i \in \mathbb{R}^3$ the Dirichlet energy is:

$$(Cu)_i = \frac{1}{4\pi t^2}\sum_j \exp\left(-\frac{\|p_i - p_j\|^2}{4t}\right)(u_i - u_j) \tag{5.24}$$

W is again the diagonal matrix of the estimated point areas. We estimate the area of a point as $\pi r^2/3$, where r is the average Euclidean distance to its 6 nearest points. This simple heuristic worked well in our experiments and allows us to avoid the more complicated area estimations of Belkin and Liu [BSW09; LPG12].

We choose t so that $\exp(-\bar{d}^2/4/t) = 1/10$, where $\bar{d}$ is the average Euclidean distance of all the points to their 10 closest neighbors. Next we sparsify C by removing small elements as follows. First we mark all coefficients of C larger than $1/(10 \cdot 4\pi t^2)$. Then in each row we mark the ten largest coefficients (excluding the diagonal). For symmetry, we mark C_{ij} if C_{ji} is marked. Lastly we set all unmarked coefficients to 0 and update the diagonal entries of C so that $C(1,\ldots,1)^T = (0,\ldots,0)^T$.

5.7. Alignment algorithm

Next we propose a concrete method to calculate a Green's alignment of two shapes and therefore a point-wise map T from a few known point-to-point or functional constraints.

Alignment energy. We use the variables $B \in \mathbb{R}^{(k-1)\times(k-1)}$; $t, \tilde{t} \in \mathbb{R}^{k-1}$ to describe the affine alignment and we define a quadratic energy, where we incorporate point-to-point and functional constraints in a least squares sense. The parameter $\alpha \in [0,1]$ encodes whether we assume conformality. For the point-to-point correspondences $\mathcal{C} = ((p_1, q_1), \dots, (p_n, q_n)) \in (\mathcal{M} \times \mathcal{N})^n$ the matching energy $E^p_{\mathcal{C}}[B; t; \tilde{t}]$ is

$$\frac{1}{n}\sum_{i=1}^{n}\left(\left\|Bg^{\mathcal{M}}_{p_i} + t - g^{\mathcal{N}}_{q_i}\right\|_2^2 + \alpha\left\|\Pi_{\mathcal{M}}Fg^{\mathcal{N}}_{q_i} + \tilde{t} - g^{\mathcal{M}}_{p_i}\right\|_2^2\right) \tag{5.25}$$

and for the functional constraints $\mathcal{D} = ((f_1, h_1), \dots, (f_m, h_m)) \in \left(\mathcal{L}^2(\mathcal{M}) \times \mathcal{L}^2(\mathcal{N})\right)^m$ the matching energy $E^f_{\mathcal{D}}[B; t]$ is

$$\frac{1}{m}\sum_{i=1}^{m}\left(\left\|Fh^{\mathcal{N}}_i - f^{\mathcal{M}}_i\right\|_2^2 + \alpha\left\|\Delta^+_{\mathcal{N}}F^T\Delta_{\mathcal{M}}f^{\mathcal{M}}_i - \Pi_{\mathcal{N}}h^{\mathcal{N}}_i\right\|_2^2\right) \tag{5.26}$$

where F, B and t are related by Eqs. (5.10) and (5.11).

Initial solving. Typically, there are so few a priori point-to-point and functional constraints, that in the first iteration the energy is under-constrained. We therefore add further regularization constraints. Translation depends on the area scale and is typically small, so that we assume $t = \tilde{t} = 0$. Furthermore, the distortion of G should be as little as possible, so that we add the regularizer $\epsilon\|B\|_F^2$ ($\epsilon = 10^{-6}$). In our experiments this simple choice lead to consistently good results and clearly outperformed other possible terms, such as $\|F\|_2^2$ or $\|\Delta^{\mathcal{M}}F - F\Delta^{\mathcal{N}}\|_2^2$. In conclusion the initial alignment G for the a priori point-to-point $\mathcal{C}_0$ and functional constraints $\mathcal{D}_0$ is the unique minimum of:

$$E^p_{\mathcal{C}_0}[B, 0, 0] + E^f_{\mathcal{D}_0}[B, 0, 0] + \epsilon\|B\|_F^2\,; \qquad\qquad t = \tilde{t} = 0 \tag{5.27}$$

Updating correspondences. Once we have an alignment G we map the Green's functions of $\mathcal{M}$ onto the Green's functions of $\mathcal{N}$ and determine novel point-to-point

Algorithm 1 Matching two shapes

```
Input: shapes $\mathcal{M}, \mathcal{N}$; initial point-to-point $\mathcal{C}_0$ and
        functional $\mathcal{D}_0$ constraints; $\alpha \in [0,1]$
$S_{\mathcal{M}}/S_{\mathcal{N}} \leftarrow$ FARTHEST_POINT_SAMPLING($\mathcal{M}/\mathcal{N}, 200$)
$B, t \leftarrow$ solve Eq. (5.27) for an initial alignment
        using $\mathcal{C}_0, \mathcal{D}_0, \alpha, \epsilon = 10^{-6}$
for i=1…20 do
    $\mathcal{C}_1 \leftarrow$ MATCHES($B, t, S_{\mathcal{M}}, S_{\mathcal{N}}$)
    $B, t \leftarrow$ solve Eq. (5.29) for a novel alignment
            using $\mathcal{C}_0, \mathcal{D}_0, \mathcal{C}_1, \alpha$
return MATCHES($B, t, \mathcal{M}, \mathcal{N}$)
                                                ▷ Sub routines:
function MATCHES($B, t, M, N$):                 ▷ Eq. (5.28)
        return $\left\{(p,q) \mid p \in M, q = \arg\min_{y \in N} \left\| B\, g_p^{\mathcal{M}} + t - g_y^{\mathcal{N}} \right\|_2 \right\}$
            $\cup \left\{(p,q) \mid q \in N, p = \arg\min_{x \in M} \left\| B\, g_x^{\mathcal{M}} + t - g_q^{\mathcal{N}} \right\|_2 \right\}$
function FARTHEST_POINT_SAMPLING($\mathcal{M}, n$)
    $S \leftarrow \{p\}$ with a random point $p$ on $\mathcal{M}$
    for $i = 2 \ldots n$ do                     ▷ $d_b$ biharmonic distances
        $S \leftarrow S \cup \left\{\arg\max_{x \in \mathcal{M}} \min_{y \in S} d_b(x,y)\right\}$
    return $S$
```

correspondences with a nearest neighbor search:

$$\begin{aligned} \rho_{\mathcal{M}}\colon \mathcal{M} \to \mathcal{N}, \qquad & p \mapsto \operatorname*{arg\,min}_{x \in \mathcal{N}} \|B\, g_p^{\mathcal{M}} + t - g_x^{\mathcal{N}}\|_2 \\ \rho_{\mathcal{N}}\colon \mathcal{N} \to \mathcal{M}, \qquad & q \mapsto \operatorname*{arg\,min}_{y \in \mathcal{M}} \|B\, g_y^{\mathcal{M}} + t - g_q^{\mathcal{N}}\|_2 \end{aligned}$$

It is sufficient to calculate correspondences from and onto a subset of both shapes $S_{\mathcal{M}}$ and $S_{\mathcal{N}}$, which were initially calculated using farthest point sampling in the Green's embedding, i.e. using biharmonic distances. For efficient nearest neighbor queries we use k-d trees with the "sliding mid-point rule" [MM99], which adapts to the low intrinsic dimensionality of the data. In summary, the current point-to-point correspondences $\mathcal{C}_1$ are inferred from G by:

$$\mathcal{C}_1 = \left\{(p, \rho_{\mathcal{M}}(p)) \mid p \in S_{\mathcal{M}}\right\} \cup \left\{(\rho_{\mathcal{N}}(q), q) \mid q \in S_{\mathcal{N}}\right\} \tag{5.28}$$

Iterative alignment. We refine the alignment by alternating between solving for the alignment G using the current point-to-point constraints and calculating novel point-to-point constraints $\mathcal{C}_1$ from the current alignment. From the second iteration on solving for G is over-constraint and no further regularization is required. G is then defined as the minimum of:

$$E^{p}_{\mathcal{C}_0}[B, t, \tilde{t}] + E^{f}_{\mathcal{D}_0}[B, t] + E^{p}_{\mathcal{C}_1}[B, t, \tilde{t}] \tag{5.29}$$

In our experiments we use 20 iterations, which was enough to converge. The effect of iterating is shown in Figure 5.7 and the entire alignment procedure is shown in Algorithm 1.

5.8. Evaluation

An important application of our method is matching non-isometric shapes, which we evaluate on the Shrec dataset [GBP07]. The dataset contains several shape classes, in which we create maps and evaluate their quality using the dense interclass correspondences from [Bur+13b] as ground truth.

Initial results. The first results in Figures 5.1, 5.5 and 5.7 show that good results can be obtained from few sparse point-to-point correspondences. As discussed in Section 5.4 and shown in Figure 5.5, the well-defined distances on the Green's functions result in a better alignment than distances on delta-distributions. The novel embedding gives smoother maps already in the first iteration, which are further smoothed by the iterative ICP-like alignment. Figures 5.5 and 5.7 show the

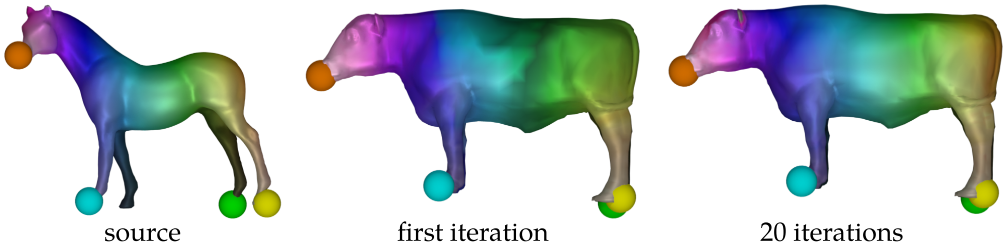
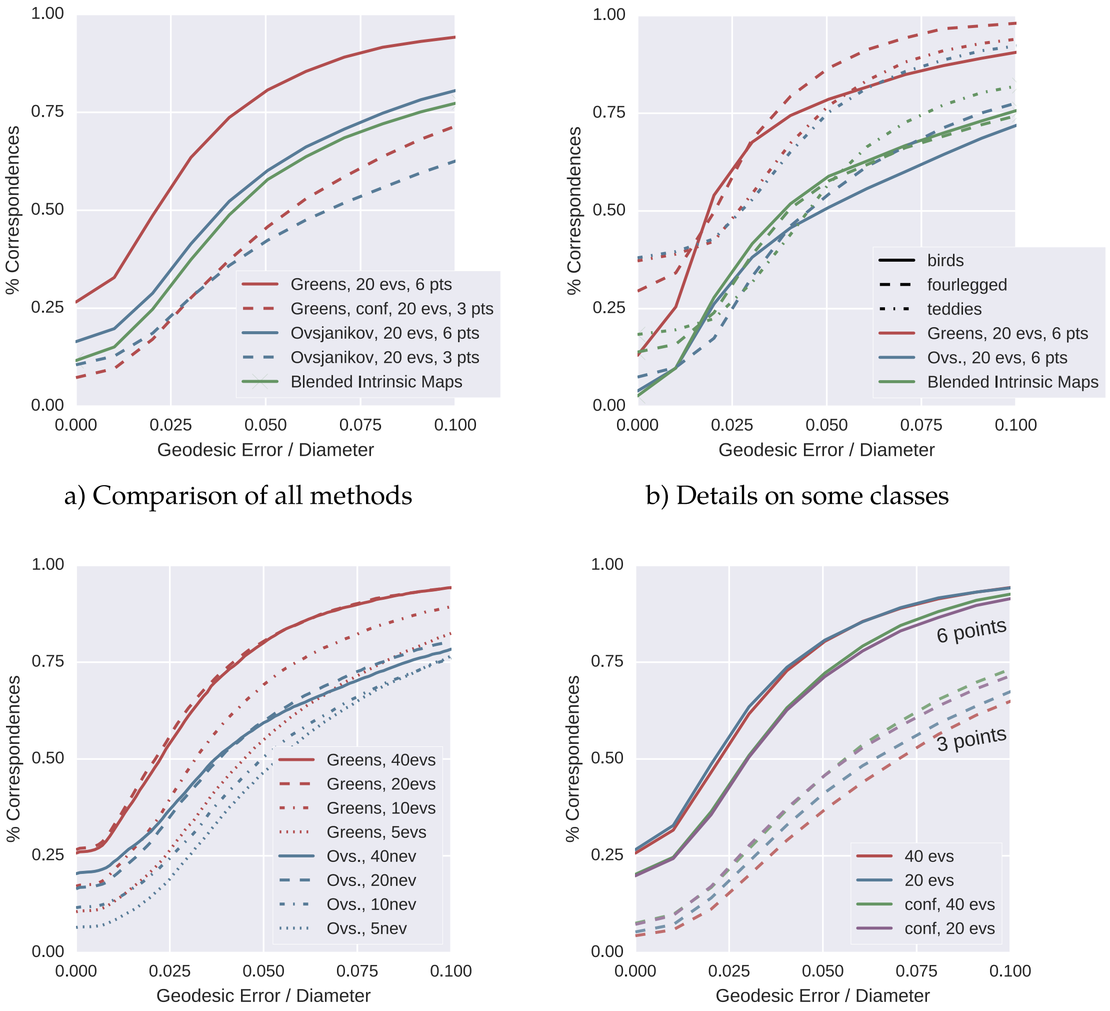


Figure 5.7.: Correspondences before and after iterating (20 eigenvectors, $\alpha = 0$).

effect of iterative alignment depicting maps without iterating and after 20 iterations.

Solving with point-to-point constraints. We compare our method to functional maps (FM), which aligns delta-distributions, and Blended Intrinsic Maps (BIM), which is a state-of-the-art method for automatic shape matching and does not use predefined constraints. More precisely, when matching with functional maps, we use Eq. (2.49) to iteratively align the delta-distributions and extract correspondences via nearest neighbor search. We omit the additional rigid ICP alignment described in [Ovs+12] as it requires isometric shapes and in principle can be applied to our method as well. In each class of the Shrec dataset we build between 30 random maps. Then we build the point-to-point constraints by geodesic farthest point sampling on the source shape and mapping these points onto the target with the ground truth map. To evaluate the maps we equally distributed 200 points on the source, mapped them onto the target and measured the deviation from the ground-truth. Figure 5.8a shows the results of the three methods with either three or six point-to-point correspondences and Figure 5.8b gives details for three classes. Independent of the number of constraints our method outperforms the functional maps method.

The number of eigenvectors used for the calculations influences the results. Figure 5.8c shows a steady improvement as the number of eigenvectors is increased (six point-to-point constraints, $\alpha = 0$). In principle maps are not restricted when represented with Green's functions. For every map T there is a Green's alignment G and for a finite set of constraints there is an infinite number of possible maps. Yet in our case there are two additional requirements for the solution, namely the alignment must be represented with k eigenvectors and for $\alpha > 0$ the solution must fulfill the conformality conditions. These two requirements can be seen as the degrees-of-freedom of the optimization, as without these requirements any (continuous) map is a valid solution. The degrees-of-freedom are reduced by either decreasing the number of eigenvectors k or by increasing the weight α of the conformality constraints. The influence of the degrees-of-freedom on the optimization is shown in Figure 5.8d. For three initial point-to-point constraints results improve when degrees-of-freedom are reduced, while for six initial point-to-point constraints the



a) Comparison of all methods

b) Details on some classes

c) Varying number of eigenvectors (6 pts)

d) Varying degrees-of-freedom

Figure 5.8.: Quantitative comparison of approximating point-to-point correspondences on the Shrec dataset [GBP07] using ground truth correspondences of [Bur+13b] (20 iterations, no functional constraints, no conformality $\alpha = 0$, sub-sampling 2000 points).

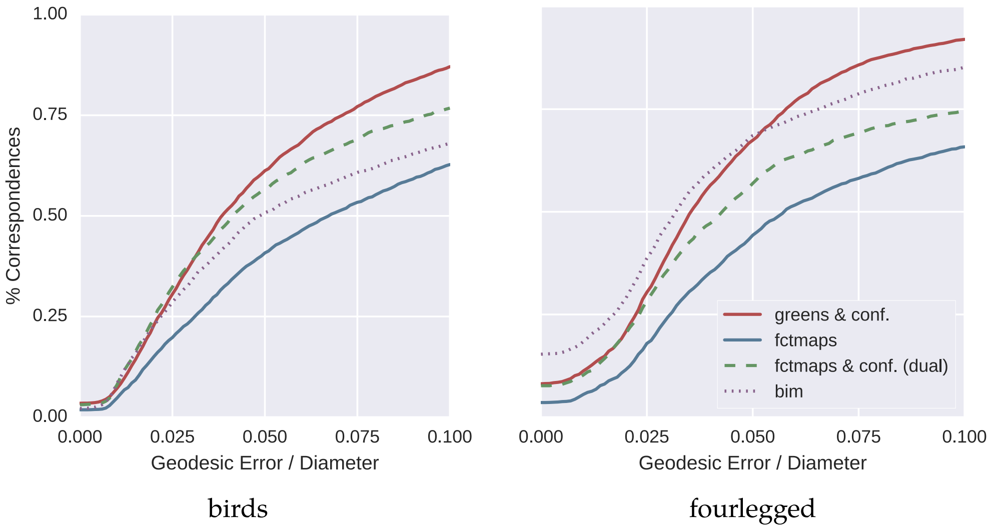


Figure 5.9.: Quantitative results of matching with functional correspondences (parameters as in Figure 5.10).

results improve when degrees-of-freedom are increased.

Solving with functional constraints. We labeled three regions on a representative shape of the classes 'birds' and 'fourlegged' and transfer these labels onto all shapes of the same class using the ground truth correspondences [Bur+13b]. Each label consists of four colors, so that we can build four functional constraints from the indicator functions. Figure 5.10 shows the labels on the bird class as well as the matching results. Here we intentionally have chosen an example where BIM fails to demonstrate the usefulness of predefined constraints. Figure 5.9 depicts a quantitative evaluation of the matching process and further results are depicted in the additional material.

Our method differs from the FM method in two aspects, namely in the novel embedding and in the addition of conformality constraints. We also considered a third method, that uses *dual* delta-distributions (to constrain and extract point-to-point correspondences) and conformal functional constraints. This third method differs from each of the other two formulations in only one aspect. The results show that both the novel embedding and the conformality constraints improve the results.

While a variety of different combinations of embeddings and constraints are possible only few achieved good results in our experiments. For example, the embedding with dual-delta-distributions ($F^T\delta_p = \delta_q$) does not work well with functional constraints without conformality ($Ff = g$). The reasons might be that the first is a constraint on F, while the second is a constraint on F^T and mixing

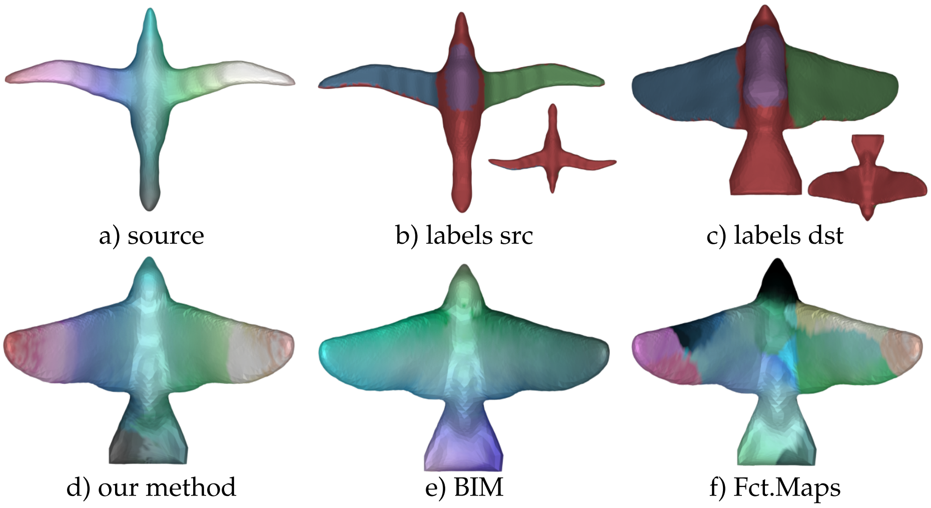


Figure 5.10.: Labels on the source (b) and target (c) were used to map colors from the source (a) with the various methods (d-e). We use the four functional constraints depicted here and in more detail in the additional material. No point-to-point constraints were used (20 eigenvectors, with conformality $\alpha = 1$, sub-sampling 2000 points).

constraints on F and F^T does not work well:

Constraints on F and G^T	Functional constraints; embedding with primal delta-distributions
Constraints on F^T and G	Conformal functional constraints; embedding with dual delta-distributions; embedding with Green's functions.

Point clouds and topological changes. Due to its underlying simplicity, our method is rather general. It works with point clouds (Figure 5.11a) and complicated topologies (Figure 5.11b). Figure 5.11c shows an example of matching in the presence of severe topological differences, where meshes were sewed together at self-intersections. Apparently our method can be seen as a global intrinsic alignment. Where the matching succeeded, most errors were localized around areas where the topology has changed.

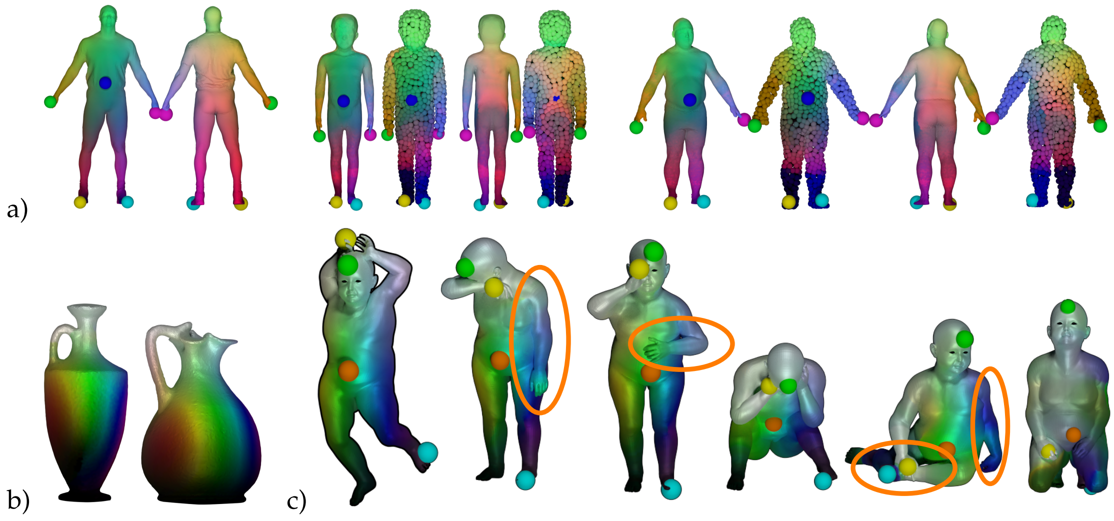


Figure 5.11.: Demonstrating the generality of the approach: matching point-clouds (a), shapes of higher genus (b) and shapes with severe differences in their topology and inner metric (c).

Isometric Matching. The embedding with Green's functions is most useful when there are few known constraints. If the functional constraints already determine the functional map, there is little difference between embedding with Green's functions and delta-distributions. For example, when matching isometric shapes the heat-kernel-signatures and wave-kernel-signatures[SOG09; ASC11] provide a magnitude of functional constraints. Figure 5.12 shows a quantitative evaluation of matching several classes from the TOSCA[BBK08] dataset using heat-kernel-signatures and a single point-to-point constraint due to the intrinsic symmetry (see also additional material). Here the results of our novel embedding are indeed similar to the original functional maps. Note the results of both algorithms can be further improved by applying rigid ICP on the delta-distributions as described in [Ovs+12]. This was not done here as our focus is on near-isometries and as it improves both methods in the same way.

Blending shapes and limitations. A good demonstration of the quality of the correspondences is their utilization to linearly blend the source triangulation onto the target shape, which results a novel triangulation of the target shape. This new triangulation depends on the correspondences and shows their quality. Typically, the following errors occur: (1) If the correspondence map is not surjective, i.e. not all target vertices have a corresponding source vertex, then these vertices are not contained in the novel triangulation at all (e.g. see missing vertices on the teddy ear). (2) If neighboring source vertices are mapped differently, then this leads to intersecting edges, outstanding long edges, and edges that are not located on the

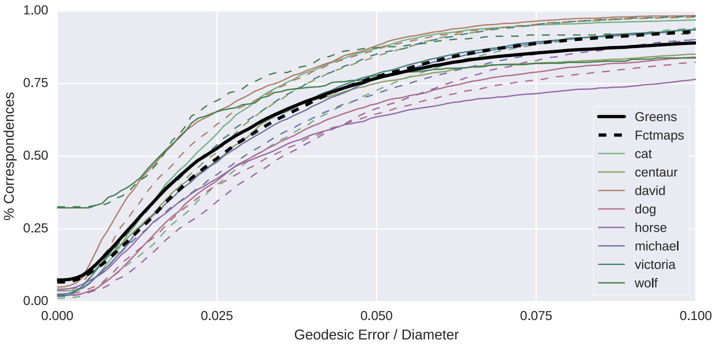


Figure 5.12.: Quantitative results of matching isometric shapes with heat-kernel-signatures. As described in the text, the alignment of Green's functions and delta-distributions yield similar results if there are sufficient constraints, as it is the case for isometric shapes when using heat-kernel-signatures (20 iterations, 20 eigenvectors, 1 point-to-point correspondence, 40 HKS functional constraints, conformality $\alpha = 1$, sub-sampling 2000 points).

target shape.

Figure 5.13 shows such a blending using our results. While in principle our results are of good quality, there are at least two reasons for the occurring misalignments. One is that Green's functions are very similar in proximity of thin extrusions (e.g. horse legs) and another is that only a local but not a global optimization was used (e.g. teddy arm). Our results can be further improved using a method for affine point cloud alignment such as Coherent Point Drift[M+07], which is also shown in Figure 5.13 and in much more detail in the additional material.

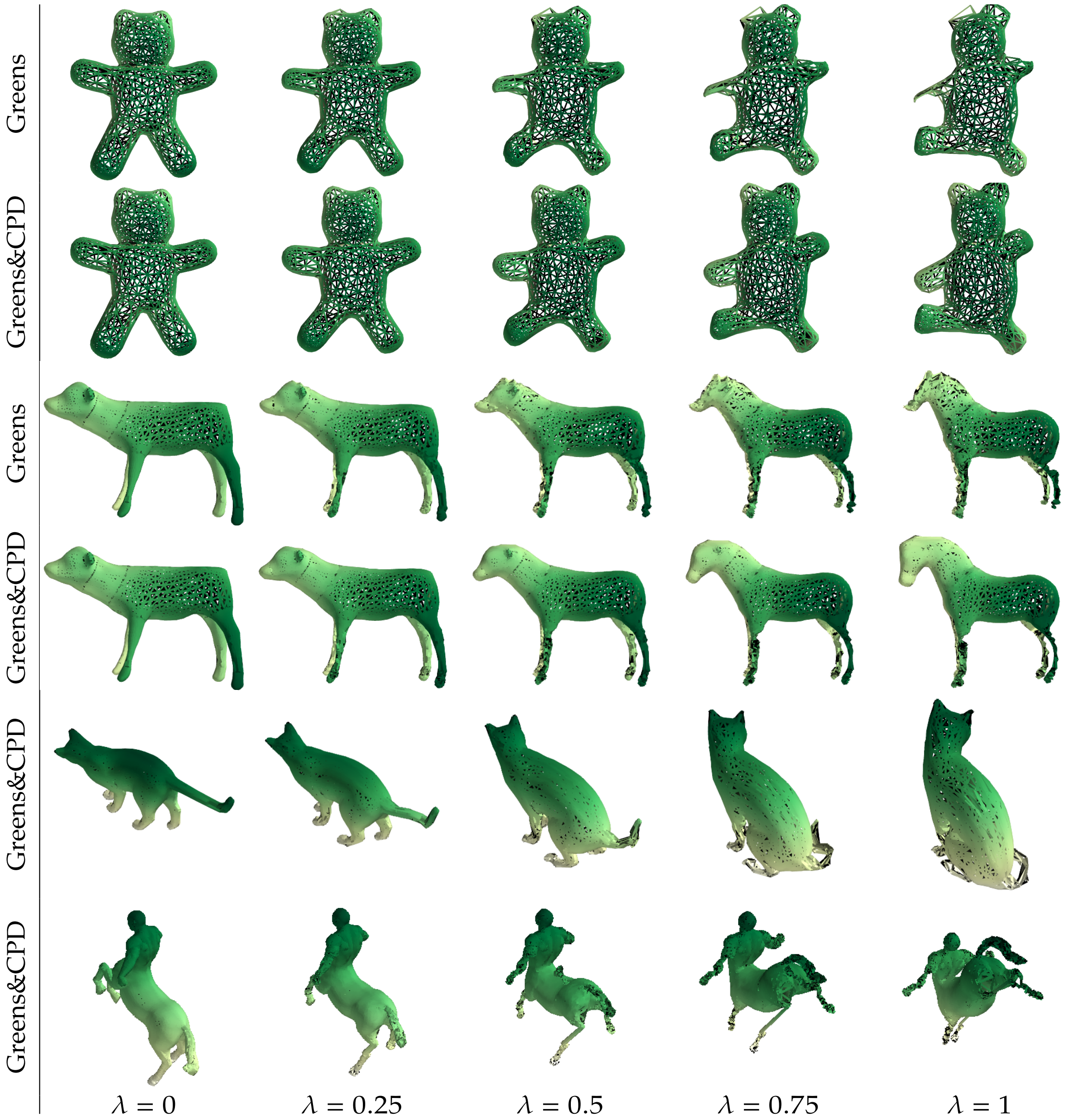


Figure 5.13.: Visualization of the correspondences by a linear interpolation of the source triangulation ($\lambda = 0$) onto the target shape ($\lambda = 1$). Please see the additional material for further results. (20 iterations, 6 point-to-point constraints, no functional constraints, no conformality $\alpha = 0$, sub-sampling 2000 points)

5.9. Concluding remarks

We expanded on the understanding of the intrinsic alignment of shapes by reformulating the alignment in a Euclidean space with a well-defined metric. Our novel embedding preserves the important advantages of the functional framework, namely that it can be aligned with an *linear* map, that it is invariant to shape deformations preserving the intrinsic metric and that it can incorporate functional constraints. Using the Green's embedding has proven to be especially useful for matching non-isometric shapes, where typically only a few correspondences are known. Additionally, to the best of our knowledge, we are the first to include conformality as functional constraints into the matching process. Our evaluation shows that to match non-isometric shapes the novel embedding is superior to the previously utilized delta-distributions. Due to its simplicity and generality our method works on shapes of higher genus, on point clouds and to some degree even in the presence of severe changes in the topology and the inner metric. It therefore might as well serve as a basis for further development of techniques for shape matching.

Appendix

Prop. 10a): Injectivity of $g_{\cdot}^{\mathcal{M}} : \mathcal{M} \to \mathcal{L}^2(\mathcal{M}), p \mapsto g_p^{\mathcal{M}}$: Let $p, q \in \mathcal{M}$ with $g_p^{\mathcal{M}} = g_q^{\mathcal{M}}$ and $f \in \mathcal{L}^2(\mathcal{M})$, then $0 = \langle \Delta f, (g_p^{\mathcal{M}} - g_q^{\mathcal{M}}) \rangle = \langle f, \delta_p^{\mathcal{M}} - \delta_q^{\mathcal{M}} \rangle = f(p) - f(q) \Rightarrow p = q$

Prop. 10b): Let $\beta \in \mathcal{L}^2(\mathcal{M})$ be the coefficients of a linear combination $h(y) = \int_q \beta(q)(\phi_1 + g_q(y))\,\mathrm{d}q \in \mathcal{L}^2(\mathcal{M})$, then

$$\langle \phi_i, h \rangle = \langle \phi_i, \beta \rangle \begin{cases} \sqrt{A} & i = 1 \\ 1/\lambda_i & i \geq 1 \end{cases} :$$

$$\langle \phi_1, h \rangle = \int_q \beta(q) \langle \phi_1, \phi_1 + g_q \rangle \,\mathrm{d}q = \int_q \beta(q)\,\mathrm{d}q = \sqrt{A} \langle \phi_1, \beta \rangle$$

$$\text{“}i \geq 2\text{”}: \quad \lambda_i \langle \phi_i, h \rangle = \langle \Delta \phi_i, h \rangle = \langle \phi_i, \Delta h \rangle =$$
$$\left\langle \phi_i, \int_q \beta(q) \Delta(\phi_1 + g_q)\,\mathrm{d}q \right\rangle = \left\langle \phi_i, \int_q \beta(q) \Pi \delta_q \,\mathrm{d}q \right\rangle =$$
$$\left\langle \phi_i, \Pi\left(\int_q \beta(q) \delta_q \,\mathrm{d}q \right) \right\rangle = \langle \phi_i, \Pi(\beta) \rangle = \langle \phi_i, \beta \rangle$$

“Independent”: Let h be a linear combination as defined above with $h(y) = 0$, then also $\beta = 0$.

“Spans $\mathcal{L}^2(\mathcal{M})$”: Because the ϕ_i form a basis of $\mathcal{L}^2(\mathcal{M})$ it is sufficient to represent the ϕ_i with linear combinations of the $g_p^{\mathcal{M}}$. Choosing $\beta = \phi_i$ results in $h = \mu\phi_i$ for some $\mu \neq 0$.

Prop. 10c): Let $h(y) = \int_q \beta(q) g_q(y)\,\mathrm{d}q \in \mathcal{L}^2(\mathcal{M})$ then analog to above $\langle \phi_1, h\rangle = 0$ and $\langle \phi_i, h\rangle = 1/\lambda_i \langle \phi_i, \beta\rangle$ for $i \geq 2$. Therefore, the set is not independent ($\int_q g_q(y)\,\mathrm{d}q = 0$) and has co-dimension 1 because ϕ_1 is not contained in the span.

Prop. 12, Eq. 5.11): $F^T \delta_p^{\mathcal{M}} = \phi_1^{\mathcal{N}}(T(p)) + \Delta_{\mathcal{N}} B \Delta_{\mathcal{M}}^+ \delta_p^{\mathcal{M}} + \Delta_{\mathcal{N}} t = \phi_1^{\mathcal{N}}(T(p)) + \Delta_{\mathcal{N}} G(\Delta_{\mathcal{M}}^+ \delta_p^{\mathcal{M}})$ $= \phi_1^{\mathcal{N}}(T(p)) + \Pi_{\mathcal{N}} \delta_{T(p)}^{\mathcal{N}} = \delta_{T(p)}^{\mathcal{N}}$

Eq. 5.22): $F^{-1} g_p^{\mathcal{M}} = \phi_1^{\mathcal{N}} \langle \phi_1^{\mathcal{N}}, F^{-1} g_p^{\mathcal{M}} \rangle + \Pi_{\mathcal{N}} F^{-1} g_p^{\mathcal{M}}$ and

$$\Pi_{\mathcal{N}} F^{-1} g_p^{\mathcal{M}} = \Delta_{\mathcal{N}}^+ \Delta_{\mathcal{N}} F^{-1} \Delta_{\mathcal{M}}^+ \delta_p^{\mathcal{M}} = \Delta_{\mathcal{N}}^+ F^T \Delta_{\mathcal{M}} \Delta_{\mathcal{M}}^+ \delta_p^{\mathcal{M}} =$$

$$\Delta_{\mathcal{N}}^+ F^T [Id + (\Pi - Id)] \delta_p^{\mathcal{M}} = \Delta_{\mathcal{N}}^+ \delta_{T(p)}^{\mathcal{N}} + \Delta_{\mathcal{N}}^+ F^T \phi_1^{\mathcal{M}} \frac{1}{\sqrt{A_{\mathcal{M}}}}$$

6 Efficient lifted relaxations of quadratic assignment problems

Abstract Quadratic assignment problems (QAPs) and quadratic assignment matchings (QAMs) recently gained a lot of interest in computer graphics and vision, e.g. for shape and graph matching. Literature describes several convex relaxations to approximate solutions of the NP-hard QAPs in polynomial time. We compare the convex relaxations recently introduced in computer graphics and vision to established approaches in discrete optimization. Building upon a unified constraint formulation we theoretically analyze their solution spaces and their approximation quality. Experiments on a standard benchmark as well as on instances of the shape matching problems support our analysis. It turns out that often the bounds of a tight linear relaxation are competitive with the bounds of semidefinite programming (SDP) relaxations, while the linear relaxation is often much faster to calculate. Indeed, for many instances the bounds of the linear relaxation are only slightly worse than the SDP relaxation of Zhao [Zha+98; PR09], which itself is at least as accurate as the relaxations currently used in computer graphics and vision. Solving the SDP relaxations can often be accelerated considerably from hours to minutes using the recently introduced approximation method for trace bound SDPs [Wan+16], but nonetheless calculating linear relaxations is faster in most cases. For the shape matching problem all relaxations generate the optimal solution, only that the linear relaxation does so faster. Our results generalize as well to QAMs for which we deliver new relaxations. Furthermore, by interpreting the Product Manifold Filter [Ves+17] in the context of QAPs we show how to automatically calculate correspondences between shapes of several hundred points.

This chapter is based on [BK17b]: Oliver Burghard and Reinhard Klein. "Efficient Lifted Relaxations of the Quadratic Assignment Problem". In: *Vision, Modeling & Visualization*. 2017.

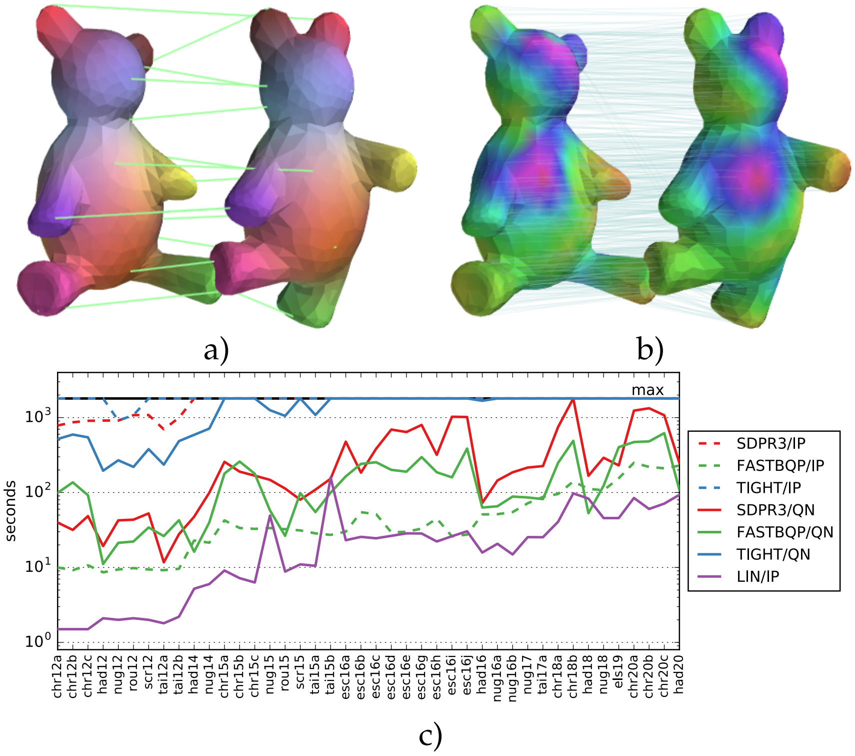


Figure 6.1.: QAPs can be used to model point assignment problems such as in (a), which are then further refined using linear assignment problems (b). We show that solving QAPs with linear relaxations is often sufficient and expectedly much faster, even when compared to fast approximative solvers for the SDPs (c) [Wan+16].

6.1. Introduction

Assigning two point sets, which were sampled on two different surfaces, onto each other is a discretization of the shape matching problem. We initially assume that the points have a one-to-one correspondence, which can then be represented by an n-element permutation contained in the symmetric group S_n. Good point-to-point assignments have small isometric distortion[BBM05; FS06; Kez+15b] defined over the pairwise geodesic distances $d_{ij} \in \mathbb{R}$ and $d'_{ij} \in \mathbb{R}$, and some parameter σ:

$$\min_{\phi \in S_n} \sum_{ij} \exp\left(-\ (d_{ij} - d'_{\phi(i)\phi(j)})^2/\sigma^2\right) \tag{6.1}$$

The minimization of the simple functional already yields good point-to-point assignments as shown in Figure 6.1a. The minimization is an instance of a quadratic assignment problem (QAP) - a difficult discrete optimization problem whose solution is our main interest. For two cost matrices $\mathbf{A} \in \mathbb{R}^{n^2 \times n^2}, \mathbf{B} \in \mathbb{R}^{n \times n}$ and noting $\mathbf{A}_{pq,rs}$ for $\mathbf{A}_{pn+q,rn+s}$ the quadratic assignment problem (QAP) minimizes:

$$\text{(QAP-}\phi\text{)} \quad \min_{\phi \in S_n} \sum_{ij} \mathbf{A}_{i\,\phi(i),j\,\phi(j)} + 2 \sum_{i} \mathbf{B}_{i\,\phi(i)} \tag{6.2}$$

Setting $\mathbf{A}_{i\,k,j\,l} = \exp\left(-\ (d_{ij} - d'_{kl})^2/\sigma^2\right)$ transforms Eq. (6.1) in this new form. $\mathbf{B}$ can be removed from the formulation as the diagonal of $\mathbf{A}$ has the same effect.

Several practically relevant hard matching problems can be modeled as a QAP, such as matching shapes as above or matching feature points between images [BMP02; SRS07; Cae+09; EKG13]. Sadly approximating QAPs to any precision is already NP-hard[SG76] and solving instances with as little as 30 points is typically not considered any more practical.

Convex relaxations estimate a lower bound on the cost and can be used to approximate solutions with small costs. The idea is to drop non-convex constraints from the problem formulation such that the optimization becomes a convex one. Because a solution of the original problem is still a feasible solution of the relaxation, the efficiently computable minimal cost of the relaxation are a lower bound for the original costs. Furthermore, as the solution of the relaxation fulfills all but the dropped constraints, projecting it onto the feasibility set of the original problem estimates a solution. The cost difference from the estimated to the minimal solution is smaller than the cost difference from the estimated solution to the lower bound of the relaxation, which is therefore a measure of the quality of the relaxation and is called the optimality gap.

Recent methods in computer graphics and vision relax QAPs into semidefinite programs [Kez+15b; Wan+16], which we compare to already established convex relaxations of discrete optimization[AJ94; Zha+98; PR09]. Interestingly on shape matching problems a carefully built linear programming relaxations is only slightly inferior to the best SDP relaxations, but solving it is often much faster. We furthermore investigate the approximation of the SDP relaxations by quasi-Newton minimization of the Lagrange dual[Wan+16] and which for certain relaxation is an order of magnitude faster than calculating their solution with interior-point methods.

We show theoretically that the SDP relaxation of Zhao [Zha+98] yields lower bounds at least as large as the SDP relaxations currently used in computer graphics and vision, and this claim is supported by our practical evaluation of the QAPLIB [BKR97] benchmark and on typical shape matching problems. Interestingly the shape matching problems result in QAPs which are solved exactly with all investigated relaxations, only the linear methods are typically faster.

Furthermore, we show how to use these relaxations to solve the related quadratic assignment matching (QAM) for which we present novel relaxations. Last but not least we show, that we can interpret the product manifold filter [Ves+17] as an iterative minimization of a QAP. This insight allows us to calculate correspondences of several hundred points *without* predefined correspondences.

6.2. Related work

Koopmans[KB57] introduced a first restricted version of the QAP to locate economic activities. The more general formulation of Eq. (6.2) was presented by Lawler[Law63] soon after. Since then a variety of discrete optimization problems have been reformulated as QAPs, which therefore themselves became an important research topic. There is much related work on solving QAPs, reviewed in several good surveys[Bur+98; Loi+07; Cel13], and in the following we only present the most relevant developments for our work.

Solving arbitrary QAPs is NP-hard as for example the traveling salesman problem can be modeled as a QAP. Despite extensive research, solving QAP instances with $n \geq 30$ is still not considered to be practical. Relaxations provide lower bounds for the minimal cost and estimate a solution. If the estimated solution is insufficient then the exact solution can be determined by Branch and Bound methods[Gil62; Law63; Ans03]. Hereby the solution space is traversed and subspaces, whose lower bound is larger than the cost of the best solution found so far, are discarded. The Gilmore-Lawler bound[Gil62; Law63] is one of the earliest lower bounds and it is still used, due to its fast calculation. There are many other relaxations such as spectral relaxations[LH05; ADK13] (which were among the first used in computer graphics and vision), linear programs[HG98; Kar+99], mixed linear integer programs [KB78; FY83; AJ94], quadratic constraint quadratic programs[Luo+10] and semidefinite programs [GW95; LS91; Kar95; Zha+98; PR09].

There are relaxations of varying sizes, for example using $\mathcal{O}(n^2)$ [Pen+15; KST15] or

$\mathcal{O}(n^4)$ [FY83; AJ94; Zha+98] variables. Our interest is in linear and SDP relaxations over the lifted permutation matrices using $\mathcal{O}(n^4)$ variables (as will be introduced below), which are known to deliver tight bounds. They depend on the second-order Birkhoff polytope [JK96a; JK96b; JK01; AM14], whose affine subspace is known but not all of its facets.

Recently there is growing interest in relaxations as copositive programs[PVZ15; Bur12; PR09; BMP12], which minimize a linear objective under linear constraints over the convex set of copositive matrices, i.e. matrices with $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$ for all $\mathbf{x} \geq 0$. Although solving copositive programs is NP-hard, as several NP-hard problems can be modeled as copositive programs, there are SDP approximations of copositive programs to any accuracy.

Solving large SDPs with current interior point solvers is costly, especially the semidefinite programming relaxations of QAP of size $\mathcal{O}(n^4)$, and several methods specialize in solving or approximating QAP relaxations and the related binary integer programs. For example by reformulating the convex program with a separable cost function[BV06], by approximating the relaxation with the bundle method[RS06] and recently with a fast approximative semidefinite program solver of trace bound SDPs[WSV13; Wan+16]. We quickly present and evaluate the latter as for some relaxations it is a magnitude faster than current interior point solvers.

6.3. Solving quadratic assignment problems

A permutation ϕ can be represented by a matrix $\mathbf{X}^\phi$ where $\mathbf{X}^\phi_{ij} = 1$ if $\phi(i) = j$ and 0 otherwise. Then the following is an equivalent formulation of (QAP-ϕ):

$$\begin{aligned} \text{(QAP)} \quad & \min_{X} \ [\mathbf{X}]^T\mathbf{A}[\mathbf{X}] + 2\,\langle \mathbf{B}, \mathbf{X}\rangle \\ \text{s.t.} \quad & \mathbf{X} \in \mathbb{R}^{n\times n},\ \mathbf{X}_{pq}^2 = \mathbf{X}_{pq}\ \forall p, q \\ & 0 \leq \mathbf{X},\ \mathbf{X}\mathbf{1} = \mathbf{1},\ \mathbf{X}^T\mathbf{1} = \mathbf{1} \end{aligned} \tag{6.3}$$

The convex hull of the permutation matrices $\{\mathbf{X}^\phi \mid \phi \in S_n\}$ is called the *Birkhoff polytope* Π^n [AM14]:

$$\begin{aligned} \Pi^n := & \operatorname{conv}\big(\{\mathbf{X}^\phi \mid \phi \in S_n\}\big) \\ = & \{\mathbf{X} \in \mathbb{R}^{n\times n} \mid 0 \leq \mathbf{X},\ \mathbf{X}\mathbf{1} = \mathbf{1},\ \mathbf{X}^T\mathbf{1} = \mathbf{1}\} \end{aligned}$$

6.3.1. Lifted variables

We reformulate (QAP) by replacing the quadratic factors $\mathbf{X}_{pq}\mathbf{X}_{rs}$ with the lifted variables $\mathbf{Y}_{pq.rs}$. Each permutation ϕ induces a lifted feasible solution of the form $(\mathbf{X}^\phi, \mathbf{Y}^\phi) := (\mathbf{X}^\phi, [\mathbf{X}^\phi][\mathbf{X}^\phi]^T)$, which is called a second-order permutation. The convex hull of the second-order permutations is called the *second-order Birkhoff polytope* Π_2^n [AM14]:

$$\Pi_2^n := \operatorname{conv}\left(\left\{\left(\mathbf{X}^\phi, \mathbf{Y}^\phi\right) \mid \phi \in S_n\right\}\right) \tag{6.4}$$

Several authors explored the second-order Birkhoff polytope and the isomorphic QAP-polytope[JK96a] (which describes the lifted permutations with a graph structure):

Theorem 15. *[JK96a; JK01] The affine hull of* Π_2^n *is:*

$$\begin{aligned} \mathbf{X1} = \mathbf{X}^T\mathbf{1} &= \mathbf{1} && && \text{(6.5a)}\\ \mathbf{Y} = \mathbf{Y}^T,\ \operatorname{diag}(\mathbf{Y}) &= [\mathbf{X}] && && \text{(6.5b)}\\ \mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} &= 0 && \forall k \forall q \neq s && \text{(6.5c)}\\ \textstyle\sum_k \mathbf{Y}_{pq,ks} = \sum_k \mathbf{Y}_{pq,sk} &= \mathbf{X}_{pq} && \forall p,q,s && \text{(6.5d)} \end{aligned}$$

Theorem 16. *[JK96a; AM14] Some facets of* Π_2^n *are given by:*

$$0 \leq \mathbf{Y} \tag{6.6}$$

For $3 \leq m \leq n-3$ *and* $(i_0, \ldots, i_m), (j_0, \ldots, j_m) \in [1:n]^{m+1}$ *pairwise disjunct other facets are:*

$$\sum\nolimits_r \ \mathbf{Y}_{i_r j_r, i_0 j_0} + \mathbf{Y}_{i_r j_r, i_r j_r} \ \leq \ \mathbf{X}_{i_0 j_0} + \sum\nolimits_r \sum\nolimits_s \ \mathbf{Y}_{i_r j_r, i_s r_s} \tag{6.7}$$

And there are additional currently unknown facets.

Every second-order permutation fulfills the following semidefinite constraints, whose equivalence follows from the Schur complement:

$$\mathbf{Y} \succcurlyeq [\mathbf{X}][\mathbf{X}]^T \qquad \left(\Leftrightarrow \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} \succcurlyeq 0\right) \tag{6.8}$$

Because Eq. (6.8) is a convex constraint fulfilled on all second-order permutations it is fulfilled on their convex hull as well:

Proposition 17. *Eq.* (6.8) *holds for all second-order permutations as well as for any tuple in* Π_2^n.

6.3.2. Convex relaxations

In the following we define several convex relaxations of (QAP). Over a fixed representation of the solutions with variables, the convex relaxation with the largest lower bounds is the one with the smallest convex solution space that still contains all valid solutions, i.e. the convex hull of the valid solutions. Over the lifted permutations (LIN-OPT) is the convex relaxation with the largest lower bound as it restricts the solutions to the second-order Birkhoff polytope Π_2^n:

$$\begin{aligned} \text{(LIN-OPT)} \quad & \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle \\ \text{s.t.} \quad & (\mathbf{X}, \mathbf{Y}) \in \Pi_2^n \end{aligned}$$

Despite being a linear program, solving (LIN-OPT) is difficult. Not only are not all facets of Π_2^n known, but it is also NP-hard:

Theorem 18. *(LIN-OPT) and (QAP) have the same minimal cost. Because the decision problem of QAP ("is there a solution of cost less than x") is NP-complete, so (LIN-OPT) is NP-hard as well.*

Proof. Every minimal solution of (LIN-OPT) can be expressed as a convex combination of second-order permutations $\phi_1, \dots, \phi_k \in S_n$: $(\mathbf{X}, \mathbf{Y}) = \sum_i \alpha_i (\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i})$ with $\alpha_i > 0, \sum_i \alpha_i = 1$. As the cost $f(\mathbf{X}, \mathbf{Y}) := \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle$ is *linear* in $\mathbf{X}$ and $\mathbf{Y}$, we have $f(\mathbf{X}, \mathbf{Y}) = \sum_i \alpha_i f(\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i})$. Because $f(\mathbf{X}, \mathbf{Y})$ is minimal, the summands of $0 = \sum_i \alpha_i (f(\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i}) - f(\mathbf{X}, \mathbf{Y}))$ are not negative and therefore all 0. Thus, $f(\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i}) = f(\mathbf{X}, \mathbf{Y})$ for all i. □

(LIN)[FY83; AJ94] is an efficiently solvable linear programming approximation of (LIN-OPT). It replaces Π_2^n with the approximation of Theorems 15 and 16, dropping the exponential number of facets from Eq. (6.7). It has the same affine subspace as Π_2^n, as Eq. (6.5c) follows from Eqs. (6.10b) and (6.10c):

$$\begin{aligned} \text{(LIN)} \quad & \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle \\ \text{s.t.} \quad & \mathbf{X1} = \mathbf{X}^T\mathbf{1} = \mathbf{1} && (6.10a) \\ & 0 \le \mathbf{Y},\ \operatorname{diag}(\mathbf{Y}) = [\mathbf{X}] && (6.10b) \\ & \textstyle\sum_k \mathbf{Y}_{pq,ks} = \sum_k \mathbf{Y}_{pq,sk} = \mathbf{X}_{pq} \quad \forall p,q,s && (6.10c) \\ \text{with} \quad & \mathbf{X} \in \mathbb{R}^{n \times n}; \mathbf{Y} \in \mathcal{S}^{n^2} \end{aligned}$$

A tighter relaxation (SDP-R3) follows by adding the semidefinite constraint from Eq. (6.8) to (LIN). It was introduced in [Zha+98] and reformulated as presented here in [PR09]:

$$\begin{aligned} \text{(SDP-R3)} \quad & \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle \\ \text{s.t.} \quad & \text{as in (LIN) and} \\ & 0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} && (6.11a) \end{aligned}$$

Adding one of the following two equivalent non-convex constraints

$$\operatorname{rank}\begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} = 1 \quad \Leftrightarrow \quad Y = [\mathbf{X}][\mathbf{X}]^T$$

to (SDP-R3) makes it equivalent to (QAP), as $\mathbf{X}_{pq} \in \{0, 1\}$ follows from $Y = [\mathbf{X}][\mathbf{X}]^T$ and $\mathbf{X}_{pq}^2 = \mathbf{Y}_{pq,pq} = \mathbf{X}_{pq}$.

We compare the above relaxation against two state-of-the-art relaxations of the recent computer graphics [Kez+15b] and vision literature [Wan+16]. Both are relaxations of a variation of the QAP, which minimizes over the partial permutations, i.e. it assigns subsets of the nodes onto each other. Here we present their relaxations adapted to (QAP) and postpone a discussion of the differences to Section 6.3.5:

$$\begin{aligned}
\text{(TIGHT)} \quad & \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle \\
\text{s.t.} \quad & 0 \leq \mathbf{X},\ \mathbf{X}\mathbf{1} = \mathbf{X}^T\mathbf{1} = \mathbf{1} && \text{(6.12a)} \\
& 0 \leq \mathbf{Y},\ \operatorname{tr}(\mathbf{Y}) = n,\ \mathbf{1}^T\mathbf{Y}\mathbf{1} = n^2 && \text{(6.12b)} \\
& \mathbf{Y}_{pq,ks} \leq \mathbf{X}_{pq} \qquad \forall p, q, k, s && \text{(6.12c)} \\
& \mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} = 0 \qquad \forall k \forall q \forall s \neq q && \text{(6.12d)} \\
& 0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} && \text{(6.12e)} \\
\text{with} \quad & \mathbf{X} \in \mathbb{R}^{n\times n}; \mathbf{Y} \in \mathcal{S}^{n^2}
\end{aligned}$$

$$\begin{aligned}
\text{(FASTBQP)} \quad & \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle \\
\text{s.t.} \quad & \mathbf{X}\mathbf{1} = \mathbf{X}^T\mathbf{1} = \mathbf{1} && \text{(6.13a)} \\
& \operatorname{diag}(\mathbf{Y}) = [\mathbf{X}] && \text{(6.13b)} \\
& \mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} = 0 \qquad \forall k \forall q \forall s \neq q && \text{(6.13c)} \\
& 0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} && \text{(6.13d)} \\
\text{with} \quad & \mathbf{X} \in \mathbb{R}^{n\times n}; \mathbf{Y} \in \mathcal{S}^{n^2}
\end{aligned}$$

We say that a relaxation (A) *dominates* another relaxation (B) and write (A) ≥ (B) if they have the same cost function and if the feasibility set of (A) is contained in the feasibility set of (B). In this case the minimum of (B) is a *lower bound* for the minimum of (A).

Theorem 19. *Using this notation the following relations hold:*

$$\begin{aligned}
\text{(LIN-OPT)} &\geq \text{(SDP-R3)} \\
\text{(SDP-R3)} &\geq \text{(LIN)} \\
\text{(SDP-R3)} &\geq \text{(TIGHT)} \\
\text{(SDP-R3)} &\geq \text{(FASTBQP)}
\end{aligned}$$

All relaxations have $\mathcal{O}(n^4)$ variables and $\mathcal{O}(n^4)$ constraints, except for (FASTBQP) which has only $\mathcal{O}(n^3)$ constraints.

Proof. We show all relations by showing that the feasibility sets are subsets of each other. Due to Proposition 17 the feasible set of (LIN-OPT) is a subset of the feasibility set of (SDP-R3). The feasibility sets of (LIN) + Eq. (6.11a), (TIGHT) + Eq. (6.10b) + Eq. (6.10c) and (FASTBQP) + Eq. (6.10b) + Eq. (6.10c) are equivalent to the feasible set of (SDP-R3). □

We conclude that, the lower bound of (SDP-R3) is at least as large as the other lower bounds, except for the lower bound of (LIN-OPT) which cannot be efficiently computed. Typically, the larger the lower bound, the less projection onto the original feasible set changes the solution, so that (SDP-R3) is as well likely to have one of the tightest duality gaps. Despite the bounds of (LIN) being weaker than the bounds of (SDP-R3), solving a linear program is often faster than solving a semidefinite program, so that (LIN) offers a trade-off between performance and the quality of the bounds.

6.3.3. Fast approximation of semidefinite programs

While (LIN) can be solved efficiently with an interior-point solver, solving the SDP relaxations (SDP-R3), (TIGHT) and (FASTBQP) with interior-point solvers is often slow. Recently the authors of [Wan+16] proposed a fast approximation method for SDPs, whose solution has a fixed trace, which all previous SDP relaxation have ($\operatorname{tr}(Y) = n$). For completeness, we quickly rephrase the necessary steps.

After subsituting $\mathbf{X}$ and $\mathbf{Y}$ by the new variable

$$\mathbf{Z} = \mathbf{Z}[\mathbf{X}, \mathbf{Y}] := \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} .$$

we express (SDP-R3), (TIGHT) and (FASTBQP) in the form:

$$\begin{aligned}
\text{(SDP-F)} \quad & \min_{\mathbf{Z} \succcurlyeq 0} \langle \mathbf{A}_0, \mathbf{Z} \rangle \\
\text{s.t.} \quad & \langle \mathbf{B}_i, \mathbf{Z} \rangle = b_i \quad i \in 1, \dots, j \\
& \langle \mathbf{B}_i, \mathbf{Z} \rangle \leq b_i \quad i \in j+1, \dots, J
\end{aligned}$$

We approximate (SDP-F) with another convex program (SDP-A):

$$\begin{aligned} \text{(SDP-A)} \quad & \min_{\mathbf{Z} \succcurlyeq 0} \langle \mathbf{A}_0, \mathbf{Z} \rangle + \frac{1}{2\gamma} \|\mathbf{Z}\|_F^2 \\ \text{s.t.} \quad & \text{as in (SDP-F)} \end{aligned}$$

The difference of the minimal costs of (SDP-F) and (SDP-A) depends on $\|\mathbf{Z}\|_F$ and γ. Let $\lambda_i \geq 0$ be the eigenvalues of $\mathbf{Z}$, then

$$\|\mathbf{Z}\|_F^2 = \sum \lambda_i^2 \leq \left(\sum \lambda_i \right)^2 = \operatorname{tr}(\mathbf{Z})^2 = (n+1)^2 \tag{6.14}$$

and the minimal costs of (SDP-A) and (SDP-F) are related by

$$c_{\text{SDP-A}} - \frac{(n+1)^2}{2\gamma} < c_{\text{SDP-F}} \leq c_{\text{SDP-A}} \,. \tag{6.15}$$

For γ large enough the solutions of (SDP-F) and (SDP-A) are arbitrarily close and solving variants of (SDP-A) yields arbitrary precise upper and lower bounds on (SDP-F).

Let $\Pi(\mathbf{C}) = \sum_i \max(0, \lambda_i)\ \phi_i \phi_i^T$ be the projection of a matrix $\mathbf{C}$ with the eigendecomposition $\mathbf{C} = \sum_i \lambda_i \phi_i \phi_i^T$ onto the positive semidefinite cone $\{\mathbf{X} \in \mathcal{S}_n \mid \mathbf{X} \succcurlyeq 0\}$. Then (SDP-A) can be minimized with a quasi-Newton method on the dual problem:

Theorem 20 ([WSV13; Wan+16])**.** *If (SDP-F) is feasible, so is (SDP-A) for which then strong duality holds. Instead of minimizing (SDP-A) we can maximize its dual:*

$$\max_{\mathbf{u} \in \hat{\mathbf{U}}} d_\gamma(\mathbf{u}) = -\mathbf{b}^T \mathbf{u} - \frac{\gamma}{2} \|\Pi(\mathbf{C}(\mathbf{u}))\|_F^2 \tag{6.16}$$

where

$$\hat{\boldsymbol{U}} = \left\{ \boldsymbol{u} \in \mathbb{R}^J \mid \boldsymbol{u}_i \geq 0 \quad \forall i \in j+1, \dots, J \right\} \tag{6.17}$$

$$\mathbf{C}(\mathbf{u}) = -\mathbf{A}_0 - \sum_i \mathbf{u}_i \mathbf{B}_i \,. \tag{6.18}$$

The dual is once but not twice differentiable and its gradient is

$$\left(\nabla d_\gamma \right)_i = +\langle \mathbf{B}_i, \Pi(\mathbf{C}(\mathbf{u})) \rangle - b_i \,. \tag{6.19}$$

6.3.4. Determining a solution

The above convex relaxations solve for $\mathbf{X}$ and $\mathbf{Y}$, yet what is required is a permutation $\phi \in S_n$. If $\mathbf{Z}[\mathbf{X}, \mathbf{Y}] \succcurlyeq 0$ and $\operatorname{rank}(\mathbf{Z}[\mathbf{X}, \mathbf{Y}]) = 1$ then $\mathbf{X}$ is indeed a solution of (QAP), as noted in the comments of (SDP-R3). But often the rank of $\mathbf{Z}[\mathbf{X}, \mathbf{Y}]$ is larger than 1.

In this case we can project $\mathbf{X}$ onto the possible permutation matrices Π^n by solving the Linear Assignment Problem:

$$\min_{\phi \in S_n} \left\| \mathbf{X}^{\phi} - \frac{n}{[\mathbf{X}]^T \mathbf{1}} \mathbf{X} \right\|_1 \tag{6.20}$$

Often better results are obtained using the randomized approach described in [Luo+10]. Let $\xi \in \mathbb{R}^{n^2}$ be a vector sampled from a centered multidimensional normal distribution $\mathcal{N}(0, \mathbf{Z})$ with covariance $\mathbf{Z}$. For any matrix $\mathbf{A}_0$ the expected value of $\xi^T \mathbf{A}_0 \xi$ is

$$\mathrm{E}_{\xi \sim \mathcal{N}(0,\mathbf{Z})}[\xi^T \mathbf{A}_0 \xi] = \langle \mathbf{Z}, \mathbf{A}_0 \rangle \ .$$

Thus (SDP-F) solves for a matrix $\mathbf{Z}$ such that sampling from $\mathcal{N}(0, \mathbf{Z})$ fulfills and minimizes (SDP-F) *in expectation*. It is therefore reasonable to sample several solutions $\xi_i \sim \mathcal{N}(0, \mathbf{Y})$, project each onto the permutations and choose the one with minimal cost:

$$\begin{aligned} \phi_i &= \min_{\phi \in S_n} \left\| [\mathbf{X}^{\phi}] - \frac{n}{1^T \xi_i} \xi_i \right\|_1 \\ \phi &= \min_i \ [\mathbf{X}^{\phi}]^T \mathbf{A}_0 [\mathbf{X}^{\phi}] + 2 \, \langle B, \mathbf{X}^{\phi} \rangle \end{aligned}$$

6.3.5. Quadratic assignment matching

A partial permutation is a bijection from k of n elements onto k of m elements. Quadratic Assignment Matching (QAM) [Kez+15b] is a generalization of QAP to partial permutations, used for example for partial graph matching. QAM minimizes the following cost defined with the matrices $\mathbf{A} \in \mathbb{R}^{nm \times nm}$ and $\mathbf{B} \in \mathbb{R}^{n \times m}$ over the partial permutations:

$$\begin{aligned} \text{(QAM)} \quad & \min \ \sum_{ij} \mathbf{A}_{\phi(i)\psi(i),\phi(j)\psi(j)} + \sum_i \mathbf{B}_{\phi(i)\psi(i)} \\ \text{s.t.} \quad & \phi\colon \{1, \dots, k\} \to \{1, \dots, n\}, \ \phi \text{ injective} \\ & \psi\colon \{1, \dots, k\} \to \{1, \dots, m\}, \ \psi \text{ injective} \end{aligned} \tag{6.21}$$

We can model (QAM) as a QAP of size $\hat{n} = n + m - k$ by adding extra nodes on both shapes (loosely following an idea from [MD58]). With $\mathbf{A}' \in \mathbb{R}^{\hat{n}^2 \times \hat{n}^2}$, $\mathbf{B}' \in \mathbb{R}^{\hat{n}^2}$

as follows

$$\mathbf{A}'_{pq,rs} = \begin{cases} \mathbf{A}_{pq,rs} & p,r \leq n \text{ and } q,s \leq m \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{B}'_{pq} = \begin{cases} \mathbf{B}_{pq} & p \leq n \text{ and } q \leq m \\ 0 & \text{otherwise} \end{cases}$$

we define the following problem:

$$\begin{array}{rl} \text{(QAM-QAP)} & \text{Minimize the QAP defined by } \mathbf{A}', \mathbf{B}' \\ \text{s.t.} & \mathbf{X}_{pq} = 0 \quad \forall p > n \; \forall q > m \end{array}$$

To show the equality of (QAM) and (QAM-QAP) let ϕ be a solution of (QAM-QAP). Then $\mathbf{X} = \mathbf{X}^{\phi}$ it is of the form:

$$\mathbf{X} = \begin{pmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & 0 \end{pmatrix} \text{ with } \mathbf{E} \in \mathbb{R}^{n \times m}, \mathbf{F} \in \mathbb{R}^{n \times (n-k)}, \mathbf{G} \in \mathbb{R}^{(m-k) \times m}$$

$\mathbf{X}$ has exactly one 1 in each row and column and in total $n + m - k$ ones. Thus $\mathbf{F}$ has $n - k$, $\mathbf{G}$ has $m - k$ and $\mathbf{E}$ has k ones. Let $(\phi(1), \psi(1)), \ldots, (\phi(k), \psi(k))$ be the indices of the ones of $\mathbf{E}$ then ϕ, ψ defines a solution of QAM. On the other hand if ϕ and ψ is a solution of QAM then we can define a solution of (QAM-QAP) of equal cost by setting $\mathbf{E}_{\phi(i)\psi(i)} = 1$ and 0 otherwise, and filling $\mathbf{F}$ and $\mathbf{G}$ such that the constraints are met. Due to the definition of $\mathbf{A}'$ the costs of both solutions are equivalent and one is optimal if and only if the other is, thus:

Theorem 21. *The problems (QAM) and (QAM-QAP) are equivalent and a solution of one leads to the solution of the other.*

We can therefore utilize the previous relaxations (SDP-R3) and (LIN) to solve (QAM), which we name (QAM-SDP-R3) and (QAM-LIN).

But we can also build smaller, less tight relaxations by dropping F and/or G

altogether, turning equality into inequality constraints where necessary:

$$
\begin{aligned}
\text{(QAM-LD)} \qquad & \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle \\
\text{s.t.} \quad & 0 \le \mathbf{X},\ \mathbf{X1} \le \mathbf{1},\ \mathbf{X}^T\mathbf{1} \le \mathbf{1},\ \mathbf{1}^T\mathbf{X1} = k && \text{(6.22a)} \\
& 0 \le \mathbf{Y},\ \operatorname{diag}(\mathbf{Y}) = [\mathbf{X}],\ \mathbf{1}^T\mathbf{Y1} = k^2 && \text{(6.22b)} \\
& \max(\textstyle\sum_k \mathbf{Y}_{pq,ks}, \sum_k \mathbf{Y}_{pq,sk}) \le \mathbf{X}_{pq} \qquad \forall p,q,s && \text{(6.22c)} \\
& \mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} = 0 \qquad \forall k \forall q \forall s \neq q && \text{(6.22d)} \\
& 0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} && \text{(6.22e)} \\
\text{with} \quad & \mathbf{X} \in \mathbb{R}^{n \times m}; \mathbf{Y} \in \mathcal{S}^{nm}
\end{aligned}
$$

This is similar to the QAM relaxation of [Kez+15b], which we add for completeness:

$$
\begin{aligned}
\text{(QAM-TIGHT)} \quad & \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle \\
\text{s.t.} \quad & 0 \le \mathbf{Y},\ \operatorname{tr}(\mathbf{Y}) = k,\ \mathbf{1}^T\mathbf{Y1} = k^2 && \text{(6.23a)} \\
& \mathbf{Y}_{pq,ks} \le \mathbf{X}_{pq} \qquad \forall p,q,k,s && \text{(6.23b)} \\
& \text{and Eqs. 6.22a, 6.22d, 6.22e and } \mathbf{X}, \mathbf{Y} \text{ as above}
\end{aligned}
$$

Theorem 22. *For the relaxations the following relations hold:*

$$
\begin{aligned}
&\text{(QAM-SDP-R3)} \ge \text{(QAM-LIN)} \\
&\text{(QAM-SDP-R3)} \ge \text{(QAM-LD)} \ge \text{(QAM-TIGHT)}
\end{aligned}
$$

(QAM-SDP-R3) and (QAM-LIN) use $(n + m - k + 1)^4$ variables and (QAM-LD) and (QAM-TIGHT) use $(nm + 1)^2$ variables.

Proof. The first relation follows from Theorem 19. "(QAM-LD) ≥ (QAM-TIGHT)": The feasibility set of (QAM-TIGHT) + Eq. (6.22b) + Eq. (6.22c) is equivalent to the feasibility set of (QAM-LD). "(QAM-SDP-R3) ≥ (QAM-LD)": Let **X**, **Y** be a solution of (QAM-SDP-R3). Then $\mathbf{X}' = \mathbf{E}$ with **E** as above and $\mathbf{Y}' \in \mathcal{S}^{nm}$ with $\mathbf{Y}'_{pq,rs} = \mathbf{Y}_{pq,rs} \forall p,q,r,s$ are a valid solution of (QAM-LD) and by construction of $\mathbf{A}'$ of equal cost. □

In conclusion modelling (QAM) as (QAP-SDP-R3) gives lower bounds at least as large as the other relaxations. For k large solving (QAM) via (QAM-LIN) is likely fastest and still gives good results. If k is small (QAM-QAP) is significantly larger than the relaxation (QAM-LD), which are therefore fastest. The lower bound of (QAM-LD) is always larger than the lower bound of (QAM-TIGHT).

6.4. Evaluation and applications

We compare solving QAPs by the following methods:

- solving the linear programming relaxation (LIN) with the Mosek[Mos10] state-of-the-art interior-point solver ("LIN/IP"),
- solving one of the semidefinite programming relaxations with the Mosek interior-point solver (".../SDP")
- approximating the semidefinite programming relaxations via maximization of the dual of (SDP-A) with the L-BFGS[Noc80] quasi-Newton method (".../QN").

For a relaxation of a QAP instance let c^- be the lower bound, c^+ be the minimal cost of 100 projected solutions as discussed in Section 6.3.4 and c^* be the minimal cost of the instance. The relative error $\frac{c^+-c^*}{|c^*|}$ measures the quality of the solution. It is bounded by the relative optimality gap $\frac{c^+-c^-}{|c^*|}$, whose calculation is independent of the minimal cost c^*.

6.4.1. QAPLIB

QAPLIB[BKR97] is a collection of QAP instances of various authors. Despite being released in 1997 it is commonly used to benchmark QAP solvers, as the difficulty of QAPs increases quickly when their dimensions grow. Our first evaluation is on the QAPLIB instances of dimension 20 and less. Figure 6.2 shows the relative optimality gap and the relative error, and Figure 6.1c depicts the times the different algorithms used. The table in Figure 6.3 depicts the fraction of the instances, in which one method had smaller optimality gap/error than the other.

Solving problem instances depends on the convex relaxation as well as on the solver and this dependency on the solver makes a reliable practical evaluation of the relaxations difficult. For convex programs with strong duality, remedy comes from primal-dual solvers. They not only minimize the objective, but delimit it with lower and upper bounds. Once the bounds are sufficiently close the global optimum has reliably been found. The Mosek[Mos10] interior-point solver is such a primal-dual solver, which we therefore strive to use in our evaluation when possible.

On the downside interior-point solvers can be slow, so that not all instances can be solved with an interior-point solver in a reasonable time. For the relaxations (SDP-R3) and (TIGHT) solving with the Mosek interior-point solver even the smaller instances took hours, which we therefore did only on the instances "chr12a" to "chr15c".

When solving the SDP approximation (SDP-F) we limit the approximation introduced error to 1% of the (known) minimal cost by choosing γ accordingly. The

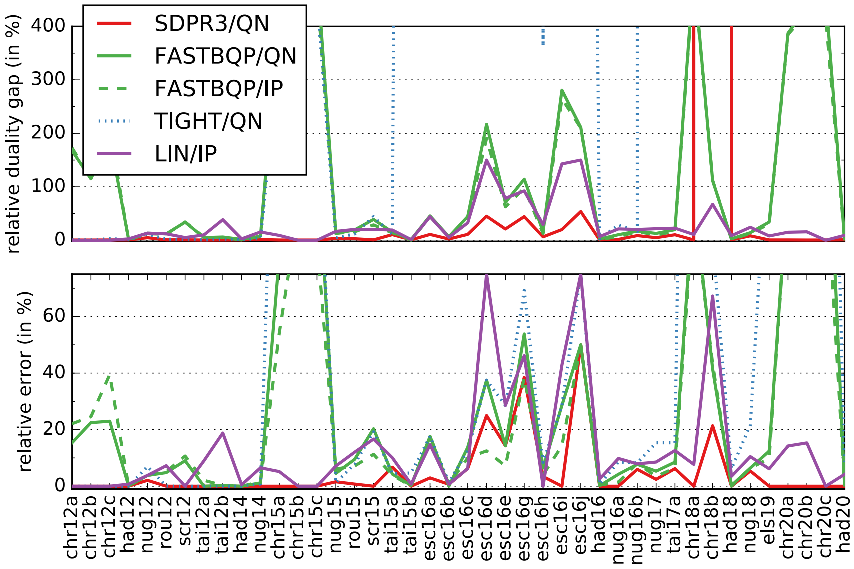


Figure 6.2.: Optimality gap and relative error on the QAPLIB instances.

reliability of the results then *only* depends on the minimization of a convex function on a convex domain with a quasi-Newton method. From our experiments we draw the following conclusions:

The quasi-Newton method and the interior-point solver yield similar results for the (FASTBQP) relaxation as well as for the (SDP-R3) relaxation on the instances "chr12a" to "chr15c" and for the (TIGHT) relaxation on the instances "chr12a" to "nug14". Thus in most cases the quasi-Newton approximation gave reliable estimations of the lower bound. Only when solving (TIGHT) on the instances "chr15a" to "chr15c" did the quasi-Newton method fail to converge and delivered a much smaller lower bound then the interior-point solver, whose lower bounds were much more comparable to the results of (SDP-R3). This is especially important as research hints that the relaxations (TIGHT) and (SDP-R3) are indeed *equal* ([DML17], proof of lemma (2) in the appendix).

After this note of caution we proceed with interpreting the results. (SDP-R3) (and possibly (TIGHT)) provides in nearly all cases the smallest relative error and the tightest bounds as expected from Theorem 19. Where this is not the case it might be due to missing convergence of the quasi-Newton solver and due to the randomization of the upper bound. (LIN) and (FASTBQP) have similar relative errors and optimality gaps, although in a few examples (FASTBQP) has a very large relative error and optimality gap.

(SDP-R3) is not only known for its good results but also for the long time interior-point solvers require solving it. For most instances (LIN) and (FASTBQP) are the fastest methods, which is clearly demonstrated in Figure 6.1c showing the solver

	Optimality gap			Error		
	SDPR3	LIN	FAST	SDPR3	LIN	FAST
SDPR3	-	0.81	0.98	-	0.67	0.86
LIN	0.19	-	0.67	0.10	-	0.43
FAST	0.02	0.33	-	0.05	0.48	-

Figure 6.3.: Fraction of experiments, where methods have smaller optimality gap/error (FAST is FASTBQP, TIGHT was left out due to uncertainties of the quasi-Newton solver).

time relative to (LIN/IP). Approximating (SDP-R3) and (TIGHT) by maximizing the dual of (SDP-A) with a quasi-Newton method is much faster than solving (SDP-F) with an interior-point solver. Indeed, in nearly all cases the interior-point solver requires much more than 30 minutes. This is a striking difference, from seconds (LIN) to minutes (SDP-R3/QN) to hours (SDP-R3/IP) (note the logarithmic scale in Figure 6.1c).

The various SDP relaxations differ not only in their solutions but also in their timings. Typically, adding constraints to a relaxation restricts the solution space and decreases the number of iterations, but increases the time for each iteration. (TIGHT) has the most constraints and a feasible set at least as large as (SDP-R3) and accordingly takes the most time to solve. (FASTBQP) on the other hand has got only $\mathcal{O}(n^3)$ constraints and takes the least time to solve.

6.4.2. Shape matching

Graph and shape matching is an application of the QAP in computer graphics and vision [SRS07; Kez+15b; Ves+17]. In the following we evaluate the performance of our relaxations and solvers to match shapes from the Tosca[BBK08] and the Shrec[GBP07] datasets.

On both surfaces, that are to be matched, we choose all points of extremal average geodesic distance in a geodesic neighborhood of 1/5 the geodesic diameter, which are usually located at semantically meaningful locations. Then we iteratively add the geodesically farthest point until we have n points. Let d_{ij} and d'_{ij} be the pairwise geodesic distances on both shapes, let σ be the mean of the distances from each point to its closest neighbor. Then we define the geodesic distortion as:

$$\mathbf{A}^{\mathrm{ISO}}_{pq,rs} = \exp\left(- \frac{(d_{pr} - d'_{qs})^2}{\sigma^2} \right) \tag{6.24}$$

Good assignments have low distortions and the minimizer of the QAP defined by $\mathbf{A}^{\mathrm{ISO}}$ is often a good assignment of the points[LH05; Kez+15b].

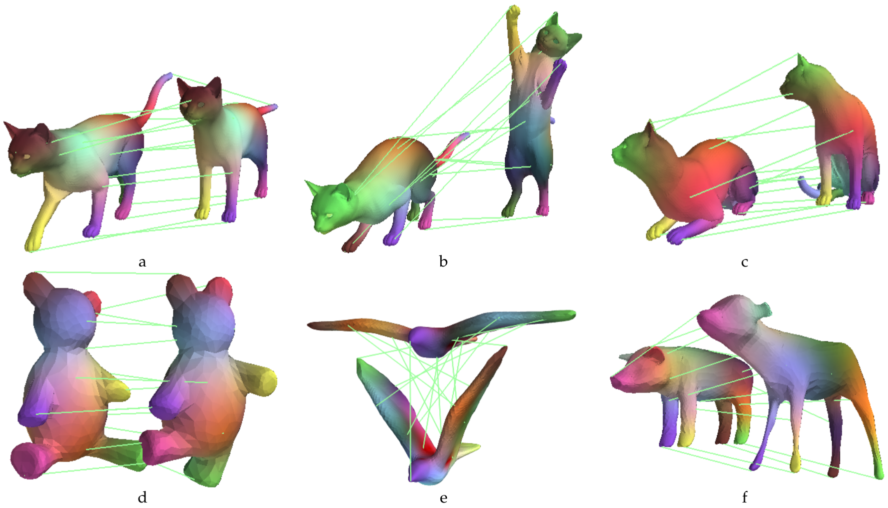


Figure 6.4.: Solving A^{ISO} on pairs of shapes sampled at fifteen points. Corresponding points have random colors, which are then diffused over the shape. (b,e) match the intrinsic symmetry and is not a failure case. Points are sampled as described in the text and do not necessarily agree exactly, e.g. head in (b,d).

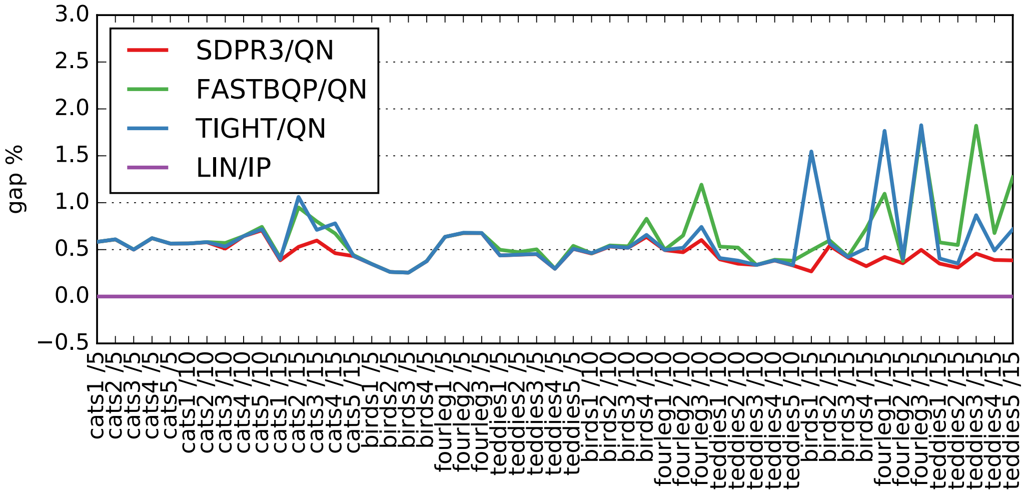


Figure 6.5.: Optimality gap after solving the QAP of Eq. (6.24) with several methods.

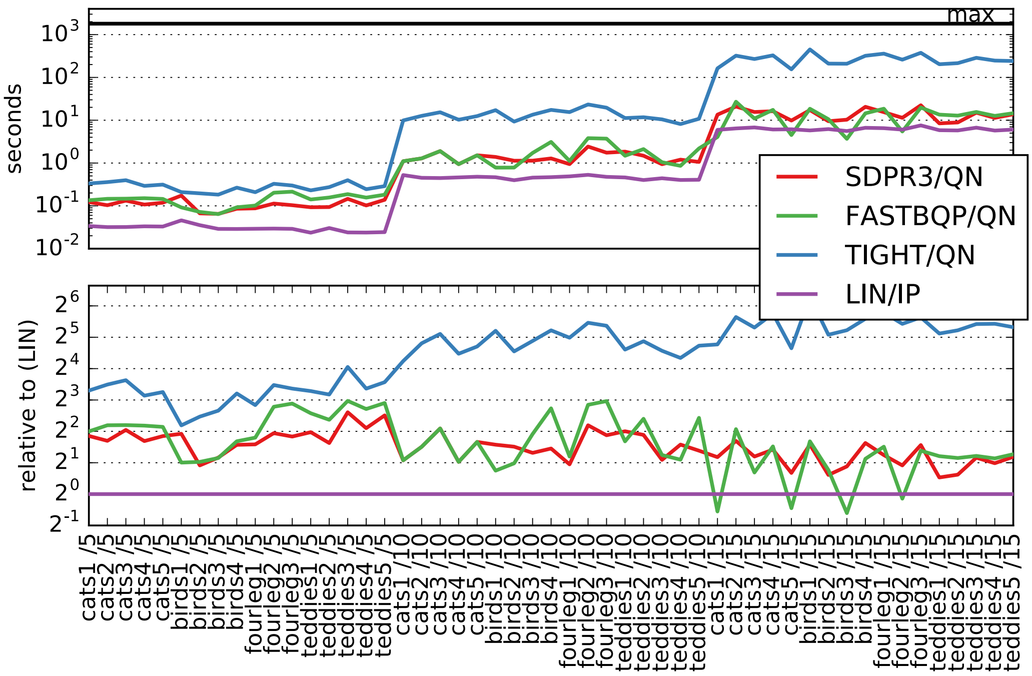


Figure 6.6.: Solver times for the different methods to solve the QAP of Eq. (6.24).

Figure 6.4 shows several example shapes with sampled and assigned points on isometric and near-isometric shapes. Figure 6.5 and Figure 6.6 depict the relative optimality gaps and the times required to solve a series of test cases such as the examples in Figure 6.4 sampled with 5, 10 or 15 points.

In the previous evaluation of QAPLIB the instances led to varying optimality gaps and the optimality gaps can be seen as a measure of hardness of the QAP instance. Interestingly, the QAP instances from Eq. (6.24) lead to a very tight relaxation, i.e. a small optimality gap. (LIN) has an optimality gap of 0 in all cases and the optimality gap of the SDP relaxations stems from the approximation with (SDP-A) and can be further reduced by increasing γ. Therefore, with the correct γ *all* relaxations result in the optimal solution.

Yet the methods differ greatly in the time required to solve. Solving (LIN) and possibly (FASTBQP) with an interior-point solver is the fastest method although fast approximations of (SDP-R3) and (FASTQAP) with (SDP-A) are only by a factor 2-4 slower. Approximating (TIGHT) with (SDP-A) or solving the SDP relaxation of (SDP-R3) with an interior-point solver is still one order of magnitude slower.

Product Manifold Filter

The product manifold filter (PMF) [Ves+17] uses a few predefined correspondences to infer improved ones. Typically, the method is applied iteratively until convergence. Let the current correspondences be encoded in a matrix $\mathbf{X}_i \in \mathbb{R}^{n\times n}$ where $(\mathbf{X}_i)_{pq} = 1$ if points p and q are assigned and 0 otherwise. Let σ be as above and let r and r' be measures of locality on both shapes. Then we calculate new correspondences ϕ_{i+1} by solving the Linear Assignment Problem:

$$\text{(PMF-LP)} \quad \phi_{i+1} = \underset{\phi\in S_n}{\arg\min}\ [\mathbf{X}^{\phi}]^T\ \mathbf{A}^{\text{PMF}}\ [\mathbf{X}_i]$$

$$\text{with} \quad \mathbf{A}^{\text{PMF}}_{pq,rs} = \begin{cases} 0 & d_{pr} < r \wedge d'_{qs} > 2r' \\ 0 & d'_{qs} < r' \wedge d_{pr} > 2r \\ \exp\left(-\frac{\left(d_{pr}+d'_{qs}\right)^2}{\sigma^2}\right) & \text{otherwise} \end{cases}$$

Requiring only a few initial correspondences, this method was shown to compute correspondences of several hundred points. We show next that the above formulation can be interpreted as an iterative minimization of the following QAP:

$$\text{(PMF-QAP)} \qquad \phi^* = \underset{\phi\in S_n}{\arg\min}\ \frac{1}{2}[\mathbf{X}^{\phi}]^T\ \mathbf{A}^{\text{PMF}}\ [\mathbf{X}^{\phi}] \tag{6.25}$$

First we relax both formulations onto the Birkhoff polytope Π^n:

$$\text{(PMF-LP')} \qquad \mathbf{X}_{i+1} = \underset{\mathbf{X}\in\Pi^n}{\arg\min}\ [\mathbf{X}]^T\ \mathbf{A}^{\text{PMF}}\ [\mathbf{X}_i] \tag{6.26}$$

$$\text{(PMF-QAP')} \qquad \mathbf{X}^* = \underset{\mathbf{X}\in\Pi^n}{\arg\min}\ \frac{1}{2}[\mathbf{X}]^T\ \mathbf{A}^{\text{PMF}}\ [\mathbf{X}] \tag{6.27}$$

Theorem 23. *The stationary points of (PMF-LP') are the local minima of (PMF-QAP'). If $\mathbf{A}^{PMF}$ is not positive definite, (PM-QAP') might have several local minima.*

Proof. $\mathbf{X}^*$ is a local minimum of $f(\mathbf{X}) = \frac{1}{2}[\mathbf{X}]^T\mathbf{A}^{\text{PMF}}[\mathbf{X}]$ if and only if $f(\mathbf{X}^*)$ grows in any direction d which does not leave the convex set Π^n. Such directions can be parameterized by $\mathbf{X} \in \Pi^n$ with $d = \mathbf{X} - \mathbf{X}^*$:

$$\begin{aligned} & \mathbf{X}^* \text{ is a local minimum of } f(\mathbf{X}) \\ \Leftrightarrow\ & \left((\nabla_{\mathbf{X}} f)|_{\mathbf{X}=\mathbf{X}^*}\right)^T [\mathbf{X} - \mathbf{X}^*] \geq 0 \quad \forall \mathbf{X} \in \Pi^n \\ \Leftrightarrow\ & [\mathbf{X}^*]^T\mathbf{A}^{\text{PMF}}[\mathbf{X}] \geq [\mathbf{X}^*]^T\mathbf{A}^{\text{PMF}}[\mathbf{X}^*] \quad \forall \mathbf{X} \in \Pi^n \\ \Leftrightarrow\ & \mathbf{X}^* = \underset{\mathbf{X}}{\arg\min}[\mathbf{X}^*]^T\mathbf{A}^{\text{PMF}}[\mathbf{X}] \end{aligned}$$

□

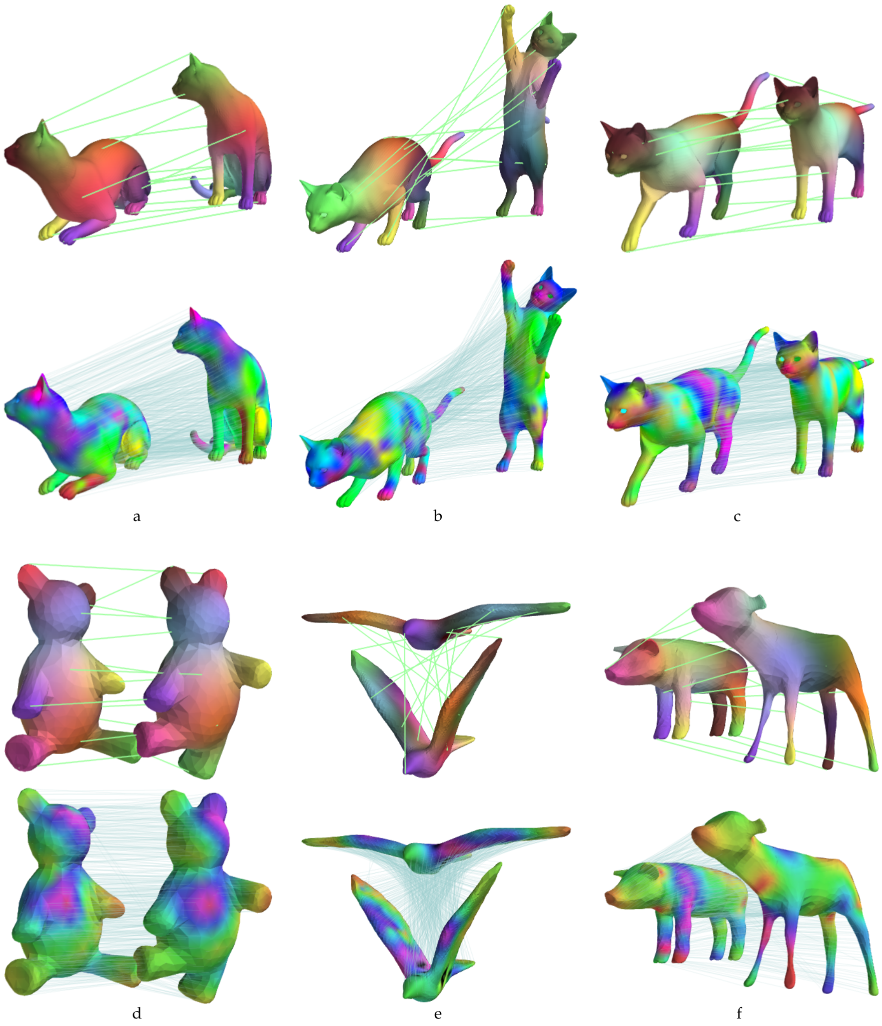


Figure 6.7.: Matching 15 points with (PMF-QAP) and refining the result with (PMF-LP). Maps are shown by mapping a random color signal. (a,b,d,e) matched correctly, although (b,e) match the intrinsic symmetry which happens in little less than half of the times; Failures: QAP twisted the feet in (c), finding the k best solutions with Branch&Bound could help; (PMF-LP) had problems converging on the shoulder of (f).

By interpreting PMF as a local minimization of a QAP energy we can remove the requirement of predefined correspondences. We first solve (PMF-QAP) over a small set of points ($n \approx 15$) and then refine the solution using (PMF-LP), which has only $\mathcal{O}(n^2)$ instead of $\mathcal{O}(n^4)$ variables and scales better. Some example applications can be seen in Figure 6.7, which shows the correspondences after (PMF-QAP) and after (PMF-LP).

6.5. Conclusions

We compared several methods to solve the quadratic assignment problem with a focus on their application to shape matching. Our results show that the formulation as a linear program (LIN) is often preferable to the more complicated SDP relaxations. At least for $n \leq 15$ it is nearly always faster. Its bounds are often tighter than the bounds of (FASTBQP). If required, (SDP-R3) yields lower bounds at least as large as the others. Furthermore, approximating (SDP-R3) or (TIGHT) using (SDP-A) [Wan+16] is always much faster than using state-of-the-art interior-points solvers. We showed how to use these insights to solve quadratic matching problems and how to utilize the results for shape matching. Furthermore our interpretation of (PMF) as a (QAP) allows to remove the requirement of predefined correspondences. Investigating the possibilities to iterate several low-cost solutions with a Branch and Bound approach might be interesting possibilities for future work.

7 Summary, conclusions and outlook

Today there is an opportunity to exploit the information contained in the steadily growing shape databases and in shapes obtained from large-scale 3d scanning. Often the most valuable information is not within single shapes but in their relation. To utilize this information dense correspondences between the shapes are required. Despite several years of intense research, computation of dense correspondence is still a major obstacle to exploit shape databases as well as other more advanced applications.

7.1. Contributions and summary

This thesis presents several contributions to near-isometric shape matching published by the author in the recent years. After an introduction to shape matching in chapter 1, the chapter summarizes our contributions and illustrates how they relate to the related work. The next chapters presented the publications the author published in the field in the recent years. Finally, after this chapter the mathematical foundations are presented in appendix 2. The discussion there will cover extrinsic and intrinsic shape alignment, functional maps and sparse correspondence generation with assignment problems. The contributions can be summarized as:

- Geodesic distances are an important tool in geometric shape processing. Chapter 3 presents novel lower and upper bounds to approximate geodesic distances. The bounds are simple in theory, they can be computed quickly, they are continuous and they have small absolute error. Additionally, the lower bound has a small relative error.

- Chapter 4 generalizes the description length approach for ensemble optimization [KT98] to the morphable-part model by Berner er al. [Ber+11]. Given sufficient prior correspondences the method delivers high quality correspondences especially well suited for the learning of shape spaces. In contrast to the original method our part-based approach can match shapes of different topology, has much fewer artefacts from part-wise rotations and the loose coupling of the part-wise shape spaces generalizes better from fewer examples. The bi-Laplacian regularizer additionally smooths the correspondences and works as a correction for the sampling bias caused by the entropy minimization. Using a quasi-Newton optimization allows the optimization of much larger models.

- As noted in chapter 5 and section 2.4.3 solving for a functional representation of a point-wise map is equivalent to computing an alignment of *dual*-delta--distributions. This requires pairwise distances on dual-delta-distributions, which are not well-defined. Previous methods therefore project dual-delta-distributions onto the first k eigenvectors of the Laplace operator, which makes distances well-defined but dependent on k.

  Chapter 5 explores the representation of point-wise maps by their alignment of the Green's functions of the Laplace operator. This alignment is well-defined in the continuous case. Approximating the alignment on a basis of Laplace eigenvectors converges quickly to the continuous result as the number of eigenvectors increases. Additional the chapter explores connections between Green's functions and conformal maps.

- In computer graphics there is a growing use of the NP-hard quadratic assignment problems to model shape matching problems. Chapter 6 compares convex relaxations over the lifted permutations of the quadratic assignment problem recently used in computer graphics to the established methods in discrete optimization. It shows that the best semidefinite relaxations provide only slightly tighter bounds than linear relaxations, which are generally faster to compute, despite approximating the semidefinite programs with the recently introduced method from Wang et al. [Wan+16], which is itself often a magnitude faster than the MOSEK [Mos10] interior-point solver. Additionally, we show that quadratic assignment problems emerging from shape matching are "simple" instances, as all the examined relaxations optimally solve all shape matching instances.

  Furthermore, we show that the Product Manifold Filter [Ves+17] can be seen as a heuristic to solve a quadratic assignment problem. This allows the automatic computation of correspondences between shapes of several hundred points.

Our contributions can be combined to compute correspondences on near-isometric shapes as follows: After approximating geodesic distances as described in chapter 3 we use the theory of chapter 6 to obtain up to fifteen sparse correspondences by creating and solving a quadratic assignment problem (e.g. Figure 6.7). These sparse correspondence are then used to compute a Green's alignment as described in chapter 5, from which we then extract dense correspondences. Typically such an alignment is successfully computed from as little as six correspondences, so that the fifteen computed above should be more than enough. Furthermore, as described in sections 2.2 and 2.4 this becomes notably simpler for isometric shapes. On a shape ensemble we subsequently apply the optimization of chapter 4 to obtain correspondences especially well-suited for the creation of shape spaces.

Occasionally user intervention might be required. Sometimes the best solution of the QAP mislabels few correspondences or maps to an intrinsic symmetry, which might be unwanted. The correct assignment is usually contained in the first few best solutions. Furthermore the dense correspondences extracted from the Green's alignment and computed by entropy optimization, might partially be non-continuous or non-surjective. Manually adding a few correspondences is usually sufficient for their correction.

7.2. Future work

Despite intense research computation of dense correspondences on near-isometric shapes often yields non-bijective maps with unnecessary large stretch. There are several interesting directions to further improve the current state-of-the-art.

If we define the quality of a correspondence map by the induced metric stretch, i.e. how far neighboring points are mapped to, then all but the simplest shapes have *several* locally optimal configurations, where no small change in the correspondence map yields lower stretch. For example, changing front and back as well as left and right of a fourlegged animal results in four different configurations of locally minimal stretch. This underlying non-convexity is one inherent reason for the complexity of shape matching. A formulation of shape matching is either non-convex, such as iterative closest points, or approximative, such as functional maps.

The theory on convex relaxations and on the quadratic assignment problem shows, that tight bounds for non-convex problems can be found and exploited to determine the global optimum and even enumerate local optima. On the downside assignment problems are too strict to model near-isometric shape matching due to the relative area-preservation. Remedy could be replacing bijectivity with a term assuring surjectivity, i.e. that everywhere on the target shape is mapped to. How to include surjectivity such that the formulation allows tight convex relaxations, similar to the convex relaxations that bijectivity allows (see chapter 6), is an open question.

Once there is a formulation of surjectivity, another problem of current approaches might become apparent: On near-isometric shapes the functionals $\mathbf{A}^{\mathrm{ISO}}$ and $\mathbf{A}^{\mathrm{PMF}}$ result in non-continuous maps as correspondences have global influence. Minimizing the target lengths of the source triangulation could be a viable alternative[1]. Furthermore, the current stretch energies are not triangulation invariant.

Moreover, assignment problems are currently limited to small problem sizes. On commodity hardware current methods solve linear assignment problems up to several thousand points and quadratic assignment problems of less than twenty points. While general instances of both problems have been thoroughly researched, there is evidence that applications in shape matching might allow larger problem sizes. Chapter 6 showed that shape matching formulations lead to "simple" quadratic assignment problems, which have even been approximated by iterative applications of linear assignment problems (chapter 6 and [Ves+17]). Understanding what makes shape matching instances simple and why approximation works so well might lead to faster solutions of shape matching instances. On a more practical note, after substituting surjectivity for bijectivity the maximal stretch is limited and correspondences in linear assignment problems and correspondence pairs in quadratic assignment problems can be pruned, which might be worth exploration.

There are many connections between the theory on functional maps, assignment problems, convex relaxations and quadratically constraint quadratic programs[Luo+10]. Any permutation matrix and indeed any matrix of the Birkhoff polytope (the feasible set of linear assignment problems) is a functional map represented in the nodal/hat basis. Linear assignment problems could be used to compute an assignment induced functional map under functional constraints. Under least squares functional constraints this could be modeled by a linearly constraint positive-semidefinite quadratic program. Especially interesting are formulations as quadratically constraint positive semi-definite quadratic programs, which were successfully approximated with semi-definite programs [Luo+10]. Recently there were used to compute an orthogonal functional map and a corresponding assignment between isometric shapes [Mar+16b].

The best correspondences maps between near-isometric shapes are not relative area preserving and can not be obtained from assignment problems. Solving for a functional map is a convex relaxation of solving for a point-wise functional map as noted in section 2.4.3. Instead of dropping the alignment constraints in (2.54) it would be preferable to constrain the images of delta-distributions under the adjoint of the functional map to be either within (linear constraints) or on (quadratic constraints) the convex hull of the delta-distributions of the other shape. Constraints on the inverse were shown to improve a functional map [Eyn+16] and can be formulated as quadratic constraints. Combining those ideas might result in a fast automatic computation of functional maps on near-isometries via quadratically

[1] Such a functional was evaluated for the publication of chapter 6, but possibly due to relative area-preservation it gave similar results to $\mathbf{A}^{\mathrm{ISO}}$ and $\mathbf{A}^{\mathrm{PMF}}$.

constraint quadratic programs possible.

Our ensemble optimization in chapter 4 becomes more reliable when formulated over a joint parameterization, or when points are embedded such that extrinsic match intrinsic distances (e.g. with delta-distributions over few eigenvectors or Green's functions). It would be interesting to extend such embeddings to optimize extrinsic functionals, such as Laplace smoothness and ensemble entropy.

Furthermore, it would be interesting to exploit (5.22) to build an efficient and simple method for the computation of conformal maps on arbitrary topologies.

A Additional evaluations

A.1. Further comparision of distance estimations (chapter 3)

This chapter compares our distance approximations in chapter 3 to the geodesic distance approximating Euclidean embedding of Aflalo et al. [AK13], which we quickly describe.

The smoother a distance field d_p is, i.e. the smaller their Dirichlet energy, the better it can be represented by the eigenvectors of the Laplace operator:

$$d_p = \left[\phi_1 \dots \phi_n\right]^T \left[\alpha_1 \dots \alpha_n\right] \tag{A.1}$$

Thus the symmetric distance operator $\tilde{D}\colon \mathcal{M} \times \mathcal{M} \to \mathbb{R}$, which maps two points onto their geodesic distance, can be expressed in the eigenbasis as well:

$$d_p(q) = \delta_p^T \, \tilde{D} \, \delta_q = \delta_p^T \left[\phi_1 \dots \phi_n\right] D \left[\phi_1 \dots \phi_n\right]^T \delta_q \qquad D \in \mathbb{R}^{n \times n} \tag{A.2}$$

In a first step the Aflalo et al. solve for D from a few distance fields in a least squares sense. The approximates all pairs distances, but these distances are not assured to be a proper distance metric, for example they can be negative.

In a next step the authors generalizing multidimensional scaling to the above distance representation and determine a Euclidean embedding, whose Euclidean distances approximate the (already approximated) geodesic distances. A Euclidean embedding defines a proper distance metric, but is actually much more. For example, it defines angles, allows point interpolation and rigid alignment.

Their method depends on the smoothness of geodesic distances, but except for the simplest domains[1] geodesic distances are not differentiable. For two or more dimensional manifolds geodesic distances are not differentiable at the source point and on compact surfaces they are not differentiable at the cusp regions, where they are locally maximal. Interestingly on compact surfaces the Dirichlet energy of a

[1] On the Euclidean line geodesic distances are indeed linear function

geodesic distance field is constant due to the Eikonal equation (A is the surface area):

$$E_{\text{Dirichlet}}[d_p] = \frac{1}{2}\int_{\mathcal{M}} \|(\nabla d_p)(x)\|_2^2 \, dx = \frac{1}{2}\int_{\mathcal{M}} 1 \, dx = \frac{A}{2} \tag{A.3}$$

Thus, for any number of eigenvectors n and the equation A.1 and A.2 are only approximations. Furthermore, there is no Euclidean embedding, such that the Euclidean distances are exactly the geodesic distances. Figures A.1 and A.2 shows a distance field, our bounds, the above method as well as both combined.

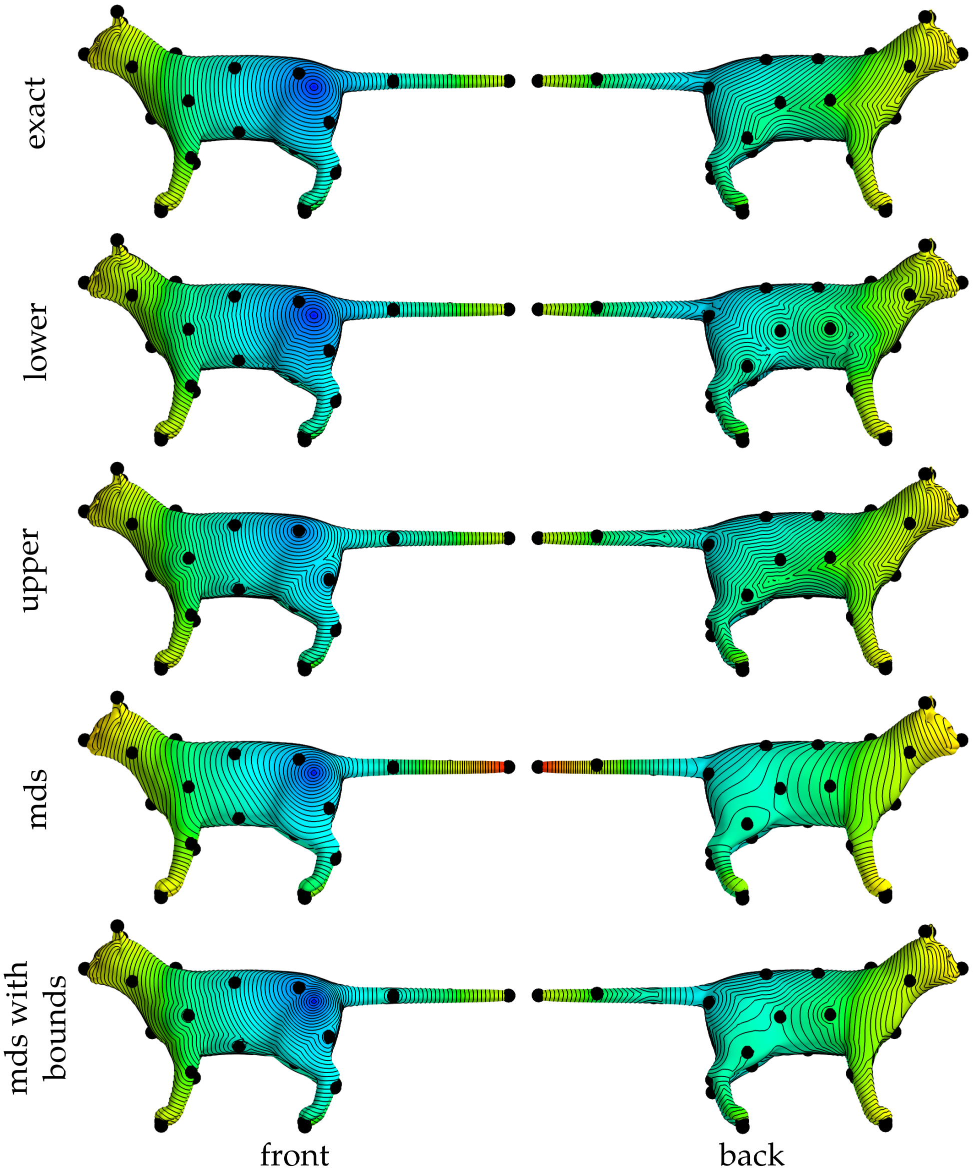


Figure A.1.: Approximating distances using 30 landmarks.

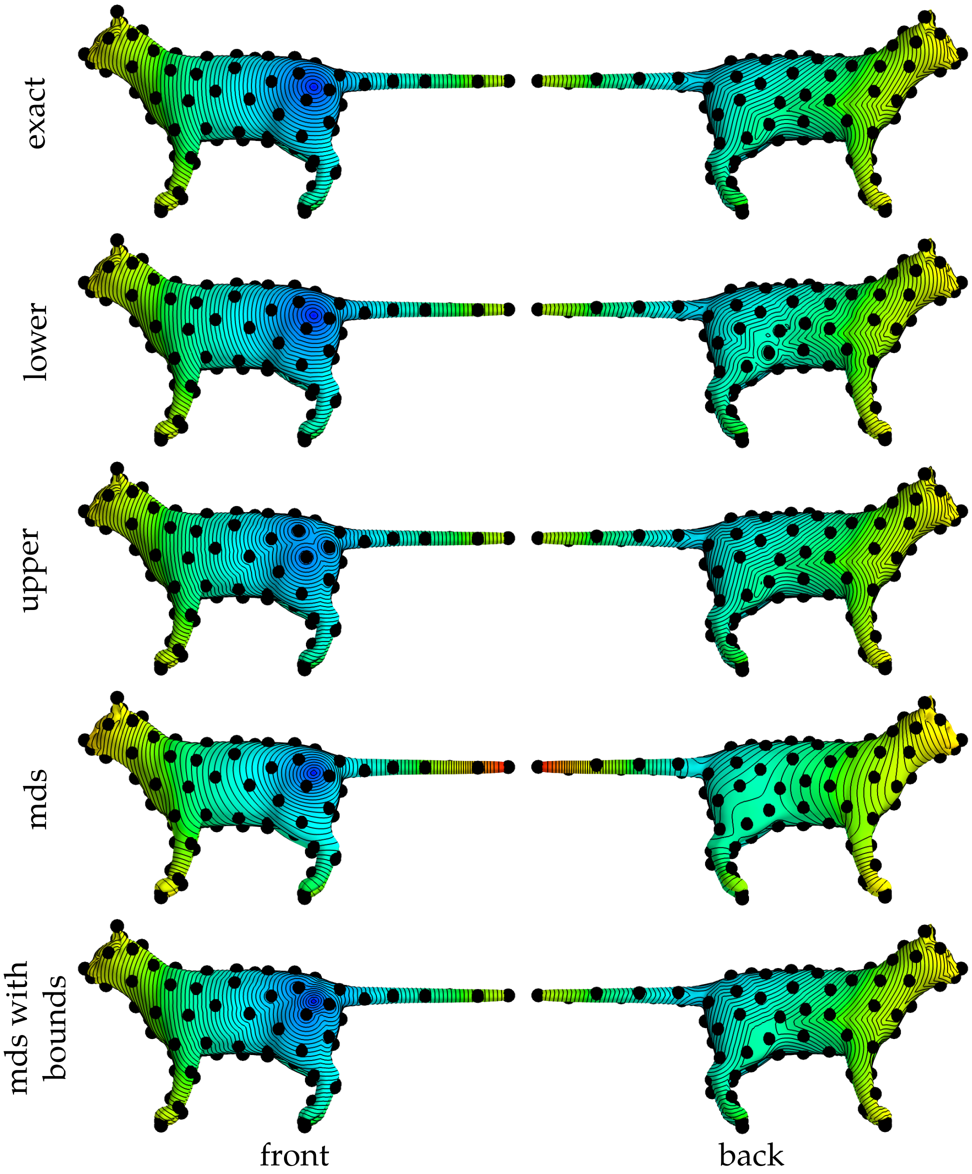


Figure A.2.: Approximating distances using 100 landmarks.

A.2. Further shape sampling results (chapter 4)

Once correspondences have been computed, they can be used to compute a shape space. A simple method to compute a shape space is to interpret the shapes as points in a high-dimensional space and fit a multivariate normal distribution. Therefore we sample the shapes at n corresponding points and represent each shape by a vector in $\mathbb{R}^{3\times n}$. Then the maximum-likelihood estimated normal distribution is computed by a principle component analysis. We use a method, that is similar in principle, but adapted to our part-based model[Ber+11].

Such a shape space is said to generalize if it contains plausible shapes. We now evaluate the correspondences computed in chapter 4, shown in Figure A.3,by inspecting the shape spaces they yield, i.e. inspecting if they generalize. Usually only high-quality correspondences, which exhibit little drift, lead to shape spaces that generalize. Figure A.4 and A.5 show sampled shapes taken from the accompanying video[2]. These shapes are not contained in the input examples, yet look very similar to the input examples. This shows how well the correspondences computed in chapter 4 are suited for the creation of shape spaces, as well as, how useful these correspondences and such shape spaces are.

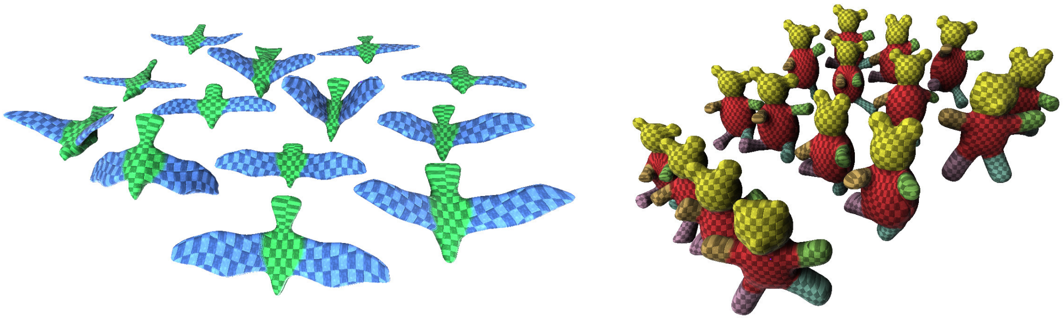


Figure A.3.: The correspondences computed in chapter 4.

[2] https://www.youtube.com/watch?v=2m3TbGO9Kls

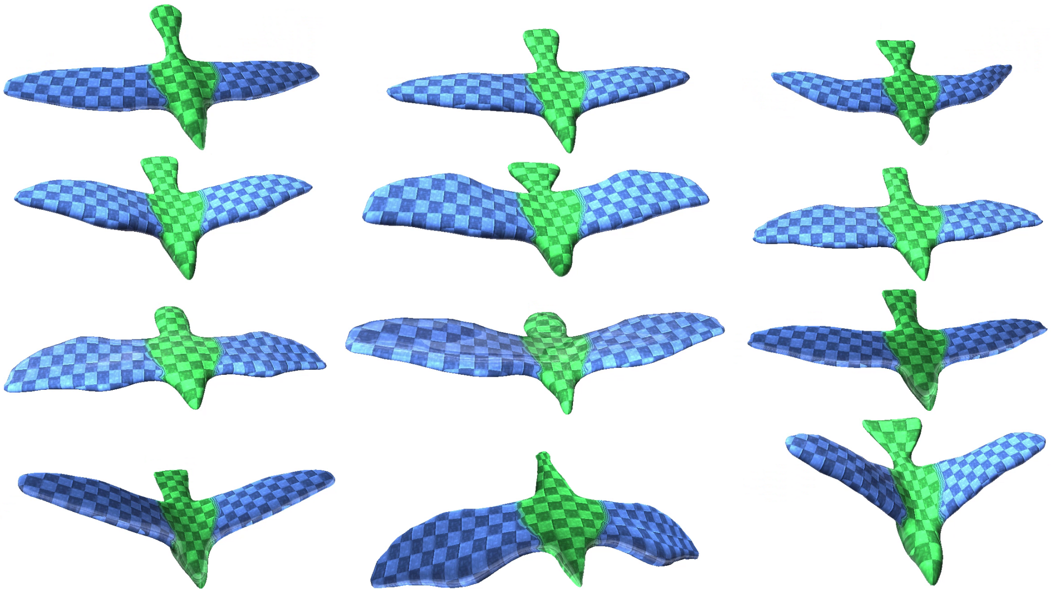


Figure A.4.: Sampling shapes from the bird ensemble.

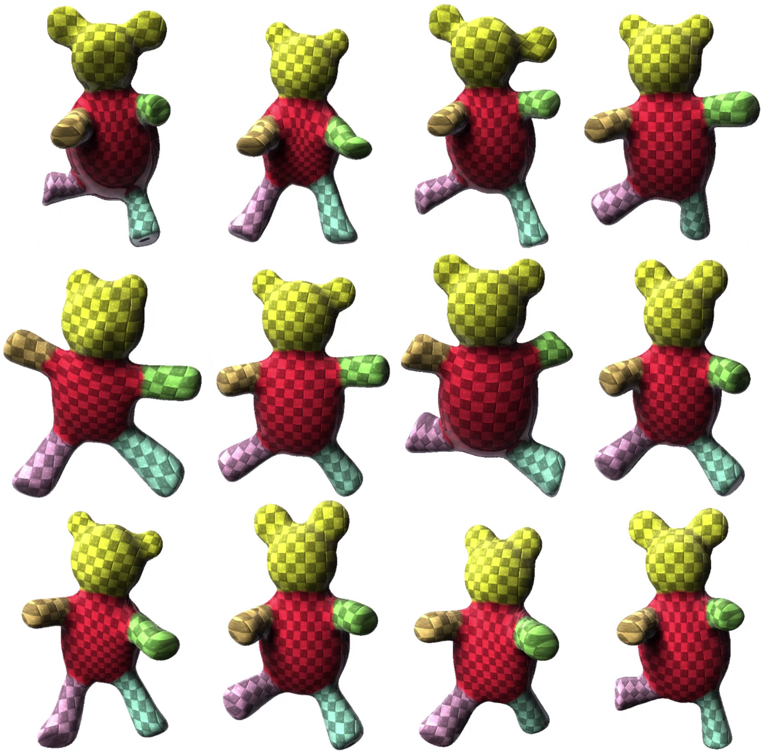


Figure A.5.: Sampling shapes from the teddy ensemble.

A.3. Evaluating the influence of the Laplace regularizer (chapter 4)

Chapter 4, as well as the previous section, made clear that the optimization of chapter 4 yields high-quality correspondences well-suited for the creation of shape spaces. As described in the Eqs. (4.12) and (4.16), the optimization moves points along the surface to minimize two energies: the compactness of the resulting shape space and the regularizer penalty. It was not yet evaluated which of these two terms is responsible for the quality of the correspondences, which we will in the following on the simple synthetic example shown in Figure A.6.

If the shape $\mathcal{M}$ is embedded by the three coordinate functions $x, y, z\colon \mathcal{M} \to \mathbb{R}$ then the regularizer minimizes $\|\Delta x\|^2 + \|\Delta y\|^2 + \|\Delta z\|^2$, i.e. a variant of the biharmonic energy from section 2.1.3. Hereby the Laplace operator is the graph Laplacian from section 2.1.6, which derives its metric from the connectivity. It was indeed shown, that such a regularizer alone is able to yield smooth correspondences [Yeh+11].

To show that the compactness assumption is a critical ingredient of the optimization of chapter 4, our goal is to discriminate the influence of the regularizer from the influence of the compactness assumption. A difficulty thereby is that the compactness term *requires* the regularizer to ensure that the surfaces are sampled everywhere. We therefore compare the correspondences computed with either the regularizer alone or with both, the regularizer and the compactness assumption.

The example consists of five shapes that differ only at a continuously extruding feature in the middle. We distinguish two regions on the shapes depicted on the right of the Figure A.7. The green region consists of the body and the tip of the extrusion and is constant during the deformation. In contrast, the brown part is stretched *anisotropically*, i.e. in only one direction, through out the deformation. We therefore expect the correct correspondences to map the colored regions onto

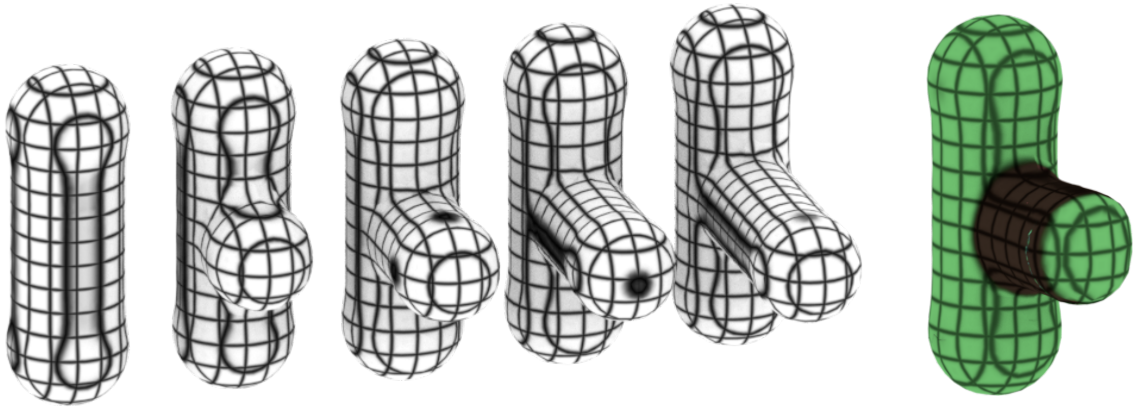


Figure A.6.: Source shapes for the evaluation and the colored regions.

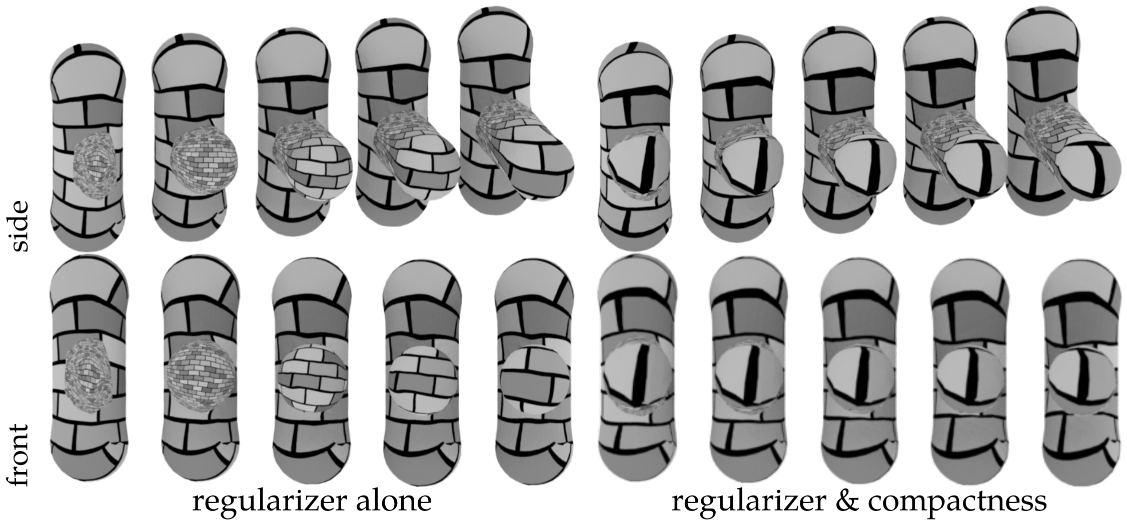


Figure A.7.: Correspondences computed with/without the compactness assumption.

themselves, such that the green regions are not distorted at all, while the brown regions are distorted anisotropically.

Figure A.7 depicts the correspondences computed with both variants. It shows that the compactness assumption not only has a major influence on the result, but that it yields exactly the correspondences we described above. It does not introduce stretch in the green regions, which is best seen in the nearly constant front facing views in the lower row, and thus necessarily distorts the brown region anisotropically. In contrast the regularizer alone leads to an almost conformal map, which is a minimizer of the biharmonic energy. It introduces stretch in all regions, especially at the tip of the extrusion, which is strongly distorted in the front facing views.

The correspondences directly affect the derived shape space, which we investigate next. Figure A.8 shows the principle directions of both shape spaces as well as the standard deviations into these directions. The regularizer alone results in a shape space with slowly decreasing standard deviations in the principal directions, where no direction exactly matches the extrusion process. The shape space computed with the compactness assumption has negligible variations in all directions but in one, which exactly matches the extrusion process. It is no surprise that it also yields a more compact shape space with a smaller entropy, which is computed as the logarithm of the product of the squared standard deviations.

As we discussed before, compactness is a machine learning principle to overcome the bias-variance dilemma. Applied to shapes, it claims that reproducing shapes with shape spaces tends to generalizes better the more compact they are. Figure A.9 evaluates the ability of the learned shape spaces to generalize by sampling from

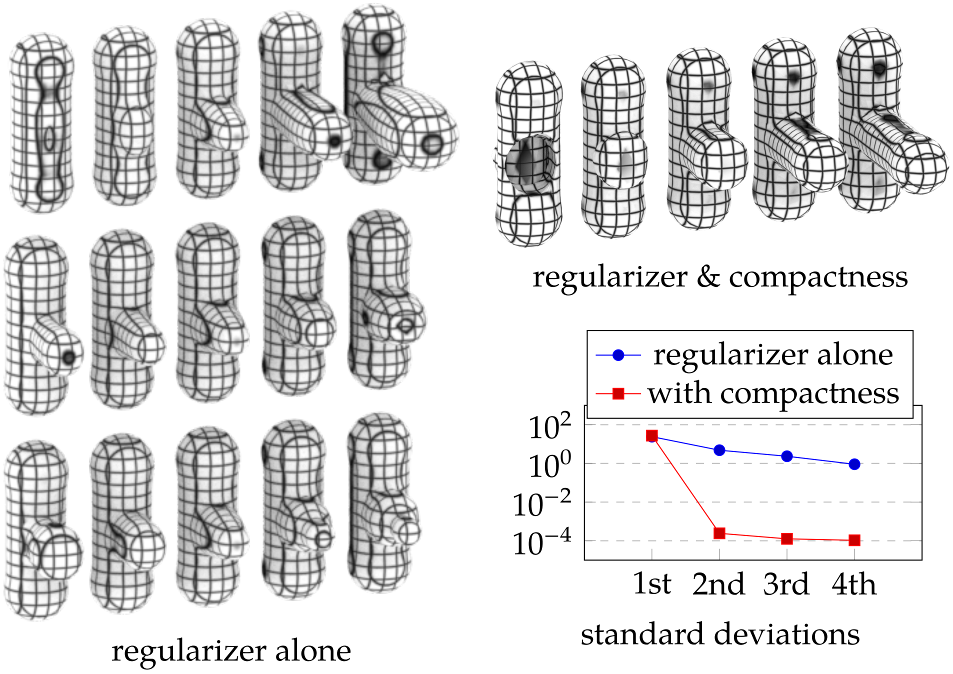
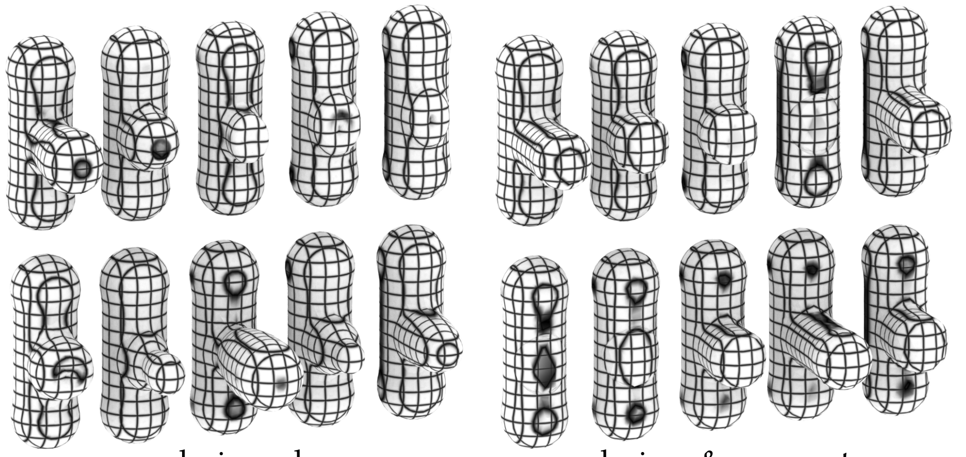


Figure A.8.: Principle directions of the shape spaces and their standard deviations.

their distribution as discussed in the previous section. And again, while the shape space from the regularizer alone contains shapes differing in unseen ways from the input shapes, the shape space computed under the compactness assumption reproduce exactly the variation seen in the input shapes. We can therefore conclude that the compactness assumption is a critical ingredient of the optimization.



regularizer alone regularizer & compactness

Figure A.9.: Sampling from the shape spaces.

List of figures

Index

Bibliography

[ACP03a] Brett Allen, Brian Curless, and Zoran Popović. "The space of human body shapes: reconstruction and parameterization from range scans". In: *ACM Transactions on Graphics (TOG)*. Vol. 22. 3. ACM. 2003, pp. 587–594.

[ACP03b] Brett Allen, Brian Curless, and Zoran Popović. "The space of human body shapes: reconstruction and parameterization from range scans". In: *ACM transactions on graphics (TOG)*. Vol. 22. 3. ACM. 2003, pp. 587–594.

[Ada+08] Bart Adams et al. "Meshless modeling of deformable shapes and their motion". In: *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Eurographics Association. 2008, pp. 77–86.

[ADK13] Yonathan Aflalo, Anastasia Dubrovina, and Ron Kimmel. "Spectral Generalized Multi-Dimensional Scaling". In: *CoRR* abs/1311.2187 (2013).

[AJ94] WARREN P Adams and Terri A Johnson. "Improved linear programming-based lower bounds for the quadratic assignment problem". In: *DIMACS series in discrete mathematics and theoretical computer science* 16 (1994), pp. 43–75.

[AK13] Yonathan Aflalo and Ron Kimmel. "Spectral multidimensional scaling". In: *Proceedings of the National Academy of Sciences* 110.45 (2013), pp. 18052–18057.

[AL15] Noam Aigerman and Yaron Lipman. "Orbifold Tutte embeddings." In: *ACM Trans. Graph.* 34.6 (2015), p. 190.

[AL16] Noam Aigerman and Yaron Lipman. "Hyperbolic orbifold tutte embeddings". In: *ACM Transactions on Graphics (TOG)* 35.6 (2016), p. 217.

[AM08] Nima Asgharbeygi and Arian Maleki. "Geodesic k-means clustering". In: *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE. 2008, pp. 1–4.

[AM14] Pawan Kumar Aurora and Shashank K Mehta. "New Facets of the QAP-Polytope". In: *arXiv preprint arXiv:1409.0667* (2014).

[Ans03] Kurt M Anstreicher. "Recent advances in the solution of quadratic assignment problems". In: *Mathematical Programming* 97.1 (2003), pp. 27–42.

[APL14] Noam Aigerman, Roi Poranne, and Yaron Lipman. "Lifted bijections for low distortion surface mappings". In: *ACM Transactions on Graphics (TOG)* 33.4 (2014), p. 69.

[APL15] Noam Aigerman, Roi Poranne, and Yaron Lipman. "Seamless surface mappings". In: *ACM Transactions on Graphics (TOG)* 34.4 (2015), p. 72.

[ARV07] Brian Amberg, Sami Romdhani, and Thomas Vetter. "Optimal step nonrigid icp algorithms for surface registration". In: *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE. 2007, pp. 1–8.

[ASC11] Mathieu Aubry, Ulrich Schlickewei, and Daniel Cremers. "The wave kernel signature: A quantum mechanical approach to shape analysis". In: *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE. 2011, pp. 1626–1633.

[AW11] Marc Alexa and Max Wardetzky. "Discrete Laplacians on general polygonal meshes". In: *ACM SIGGRAPH 2011 papers on - SIGGRAPH '11* (2011). DOI: 10.1145/1964921.1964997.

[BBK06a] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. "Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching". In: *Proceedings of the National Academy of Sciences* 103.5 (Jan. 2006), pp. 1168–1172. ISSN: 1091-6490. DOI: 10.1073/pnas.0508601103.

[BBK06b] Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. "Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching". In: *Proceedings of the National Academy of Sciences* 103.5 (2006), pp. 1168–1172.

[BBK08] Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. *Numerical geometry of non-rigid shapes*. Springer Science & Business Media, 2008.

[BBM05] Alexander C Berg, Tamara L Berg, and Jitendra Malik. "Shape matching and object recognition using low distortion correspondences". In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1. IEEE. 2005, pp. 26–33.

[BDK17a] Oliver Burghard, Alexander Dieckmann, and Reinhard Klein. "Embedding shapes with Green's functions for global shape matching". In: *Computers & Graphics* (2017).

[BDK17b] Oliver Burghard, Alexander Dieckmann, and Reinhard Klein. "Embedding shapes with Green's functions for global shape matching". In: *Computers & Graphics* 68 (2017), pp. 1–10. ISSN: 0097-8493. DOI: `10.1016/j.cag.2017.06.004`.

[Ber+11] Alexander Berner et al. *A Morphable Part Model for Shape Manipulation*. eng. Research Report MPI-I-2011-4-005. Saarbrücken: Max-Planck-Institut für Informatik, 2011.

[BGB14] Davide Boscaini, Ramūunas Girdziušas, and Michael M Bronstein. "Coulomb shapes: using electrostatic forces for deformation-invariant shape representation". In: *Proceedings of the 7th Eurographics Workshop on 3D Object Retrieval*. Eurographics Association. 2014, pp. 9–15.

[BK04a] Mario Botsch and Leif Kobbelt. "A remeshing approach to multiresolution modeling". In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing - SGP '04* (2004). ISSN: 1727-8384. DOI: `10.1145/1057432.1057457`.

[BK04b] Mario Botsch and Leif Kobbelt. "An intuitive framework for real-time freeform modeling". In: *ACM SIGGRAPH 2004 Papers on - SIGGRAPH '04* (2004). DOI: `10.1145/1186562.1015772`.

[BK15a] Oliver Burghard and Reinhard Klein. "Simple, Robust, Constant-Time Bounds on Surface Geodesic Distances using Point Landmarks". In: *Vision, Modeling & Visualization*. 2015.

[BK15b] Oliver Burghard and Reinhard Klein. "Simple, Robust, Constant-Time Bounds on Surface Geodesic Distances using Point Landmarks". In: *Vision, Modeling & Visualization, VMV 2015, Aachen, Germany, October 7-10, 2015.* 2015, pp. 17–24. DOI: `10.2312/vmv.20151253`.

[BK17a] Oliver Burghard and Reinhard Klein. "Efficient Lifted Relaxations of the Quadratic Assignment Problem". In: *Vision, Modeling & Visualization*. 2017.

[BK17b] Oliver Burghard and Reinhard Klein. "Efficient Lifted Relaxations of the Quadratic Assignment Problem". In: *Vision, Modeling & Visualization*. Ed. by Matthias Hullin et al. The Eurographics Association, 2017. ISBN: 978-3-03868-049-9. DOI: `10.2312/vmv.20171267`.

[BKR97] Rainer E Burkard, Stefan E Karisch, and Franz Rendl. "QAPLIB–a quadratic assignment problem library". In: *Journal of Global optimization* 10.4 (1997), pp. 391–403.

[BMP02] Serge Belongie, Jitendra Malik, and Jan Puzicha. "Shape matching and object recognition using shape contexts". In: *IEEE transactions on pattern analysis and machine intelligence* 24.4 (2002), pp. 509–522.

[BMP12] L Bai, JE Mitchell, and J Pang. *On QPCCs, QCQPs and copositive programs*. Tech. rep. Technical report, Rensselaer Polytechnic Institute, 2012.

[BR07] Benedict J. Brown and Szymon Rusinkiewicz. "Global non-rigid alignment of 3-D scans". In: *ACM Transactions on Graphics* 26.3 (July 2007), p. 21. ISSN: 0730-0301. DOI: 10.1145/1276377.1276404.

[BSW09] Mikhail Belkin, Jian Sun, and Yusu Wang. "Constructing Laplace operator from point clouds in R d". In: *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics. 2009, pp. 1031–1040.

[Bur+13a] Oliver Burghard et al. "Compact Part-Based Shape Spaces for Dense Correspondences". In: *arXiv:1311.7535* (2013).

[Bur+13b] Oliver Burghard et al. "Compact Part-Based Shape Spaces for Dense Correspondences". In: *arXiv preprint arXiv:1311.7535* (2013).

[Bur+98] Rainer E Burkard et al. "The quadratic assignment problem". In: *Handbook of combinatorial optimization*. Springer, 1998, pp. 1713–1809.

[Bur12] Samuel Burer. "Copositive programming". In: *Handbook on semidefinite, conic and polynomial optimization*. Springer, 2012, pp. 201–218.

[BV06] Samuel Burer and Dieter Vandenbussche. "Solving Lift-and-Project Relaxations of Binary Integer Programs". In: *SIAM Journal on Optimization* 16.3 (Jan. 2006), pp. 726–750. ISSN: 1095-7189. DOI: 10.1137/040609574.

[BV99a] Volker Blanz and Thomas Vetter. "A Morphable Model for the Synthesis of 3D Faces". In: *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1999, Los Angeles, CA, USA, August 8-13, 1999*. 1999, pp. 187–194. DOI: 10.1145/311535.311556.

[BV99b] Volker Blanz and Thomas Vetter. "A morphable model for the synthesis of 3D faces". In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co. 1999, pp. 187–194.

[Cae+09] Tibério S Caetano et al. "Learning graph matching". In: *IEEE transactions on pattern analysis and machine intelligence* 31.6 (2009), pp. 1048–1058.

[Car16] M.P. do Carmo. *Differential Geometry of Curves and Surfaces: Revised and Updated Second Edition*. Dover Books on Mathematics. Dover Publications, 2016. ISBN: 9780486806990.

[Cat+06] Joshua Cates et al. "Entropy-based particle systems for shape correspondence". In: *1st MICCAI workshop on mathematical foundations of computational anatomy: geometrical, statistical and registration methods for modeling biological shape variability*. 2006, pp. 90–99.

[Cel13] Eranda Cela. *The quadratic assignment problem: theory and algorithms*. Vol. 1. Springer Science & Business Media, 2013.

[CH90] Jindong Chen and Yijie Han. "Shortest Paths on a Polyhedron". In: *Proceedings of the Sixth Annual Symposium on Computational Geometry*. SCG '90. Berkley, California, USA: ACM, 1990, pp. 360–369. ISBN: 0-89791-362-0. DOI: `10.1145/98524.98601`.

[CL06] Ronald R. Coifman and Stéphane Lafon. "Diffusion maps". In: *Applied and Computational Harmonic Analysis* 21.1 (2006). Special Issue: Diffusion Maps and Wavelets, pp. 5–30. ISSN: 1063-5203. DOI: `10.1016/j.acha.2006.04.006`.

[COC15] Etienne Corman, Maks Ovsjanikov, and Antonin Chambolle. "Continuous Matching via Vector Field Flow". In: *Computer Graphics Forum*. Vol. 34. 5. Wiley Online Library. 2015, pp. 129–139.

[Coo+95] T.F. Cootes et al. "Active Shape Models-Their Training and Application". In: *Computer Vision and Image Understanding* 61.1 (Jan. 1995), pp. 38–59. ISSN: 1077-3142. DOI: `10.1006/cviu.1995.1004`.

[CP16] Dylan Campbell and Lars Petersson. "GOGMA: Globally-Optimal Gaussian Mixture Alignment". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016). DOI: `10.1109/cvpr.2016.613`.

[CPS11] Keenan Crane, Ulrich Pinkall, and Peter Schröder. "Spin transformations of discrete surfaces". In: *ACM Transactions on Graphics (TOG)*. Vol. 30. 4. ACM. 2011, p. 104.

[CT11] F. Calakli and G. Taubin. "SSD: Smooth Signed Distance Surface Reconstruction". In: *Computer Graphics Forum* 30.7 (Sept. 2011), pp. 1993–2002. ISSN: 0167-7055. DOI: `10.1111/j.1467-8659.2011.02058.x`.

[CWW13] Keenan Crane, Clarisse Weischedel, and Max Wardetzky. "Geodesics in Heat: A New Approach to Computing Distance Based on Heat Flow". In: *ACM Trans. Graph.* 32.5 (Oct. 2013), 152:1–152:11. ISSN: 0730-0301. DOI: `10.1145/2516971.2516977`.

[Dav+02a] Rhodri H. Davies et al. "3D Statistical Shape Models Using Direct Optimisation of Description Length". In: *Lecture Notes in Computer Science* (2002), pp. 3–20. ISSN: 0302-9743. DOI: 10.1007/3-540-47977-5_1.

[Dav+02b] R.H. Davies et al. "A minimum description length approach to statistical shape modeling". In: *IEEE Transactions on Medical Imaging* 21.5 (May 2002), pp. 525–537. ISSN: 0278-0062. DOI: 10.1109/tmi.2002.1009388.

[Dav+10] R.H. Davies et al. "Building 3-D Statistical Shape Models by Direct Optimization". In: *IEEE Transactions on Medical Imaging* 29.4 (Apr. 2010), pp. 961–981. ISSN: 1558-254X. DOI: 10.1109/tmi.2009.2035048.

[DHS01] Richard O Duda, Peter E Hart, and David G Stork. "Pattern classification. 2nd". In: *Edition. New York* (2001), p. 55.

[DML17] Nadav Dym, Haggai Maron, and Yaron Lipman. "DS++: A flexible, scalable and provably tight relaxation for matching problems". In: *CoRR* abs/1705.06148 (2017).

[EÅ03] A. Ericsson and K. Åström. "Minimizing the description length using steepest descent". In: *Procedings of the British Machine Vision Conference 2003* (2003). DOI: 10.5244/c.17.10.

[EKG13] Amir Egozi, Yosi Keller, and Hugo Guterman. "A probabilistic approach to spectral graph matching". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.1 (2013), pp. 18–27.

[EOR17] EORA. *Eora 3d*. 2017. URL: https://eora3d.com/ (visited on 08/30/2017).

[Eyn+16] Davide Eynard et al. "Coupled Functional Maps". In: *2016 Fourth International Conference on 3D Vision (3DV)* (Oct. 2016). DOI: 10.1109/3dv.2016.49.

[FA07] Balazs Feil and Janos Abonyi. "Geodesic distance based fuzzy clustering". In: *Soft Computing in Industrial Applications* (2007), pp. 50–59.

[FC13] F. Flaherty and M.P. do Carmo. *Riemannian Geometry*. Mathematics: Theory & Applications. Birkhäuser Boston, 2013. ISBN: 9780817634902.

[Fit03] Andrew W Fitzgibbon. "Robust registration of 2D and 3D point sets". In: *Image and Vision Computing* 21.13-14 (Dec. 2003), pp. 1145–1153. ISSN: 0262-8856. DOI: 10.1016/j.imavis.2003.09.004.

[FKY08] Wei-Wen Feng, Byung-Uck Kim, and Yizhou Yu. "Real-time data driven deformation using kernel canonical correlation analysis". In: *ACM SIGGRAPH 2008 papers on - SIGGRAPH '08* (2008). DOI: 10.1145/1399504.1360690.

[FS06] Thomas Funkhouser and Philip Shilane. "Partial matching of 3D shapes with priority-driven search". In: *ACM International Conference Proceeding Series*. Vol. 256. 2006, pp. 131–142.

[FY83] AM Frieze and J Yadegar. "On the quadratic assignment problem". In: *Discrete applied mathematics* 5.1 (1983), pp. 89–98.

[GBP07] Daniela Giorgi, Silvia Biasotti, and Laura Paraboschi. "Shrec: shape retrieval contest: Watertight models track". In: (2007).

[GD04] John C Gower and Garmt B Dijksterhuis. *Procrustes problems*. Vol. 3. Oxford University Press Oxford, 2004.

[Gil62] Paul C Gilmore. "Optimal and suboptimal algorithms for the quadratic assignment problem". In: *Journal of the society for industrial and applied mathematics* 10.2 (1962), pp. 305–313.

[Gol+98] Steven Gold et al. "New algorithms for 2D and 3D point matching". In: *Pattern Recognition* 31.8 (Aug. 1998), pp. 1019–1031. ISSN: 0031-3203. DOI: 10.1016/s0031-3203(98)80010-1.

[GW95] Michel X Goemans and David P Williamson. "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming". In: *Journal of the ACM (JACM)* 42.6 (1995), pp. 1115–1145.

[Has+09a] Nils Hasler et al. "A statistical model of human pose and body shape". In: *Computer Graphics Forum*. Vol. 28. 2. Wiley Online Library. 2009, pp. 337–346.

[Has+09b] N. Hasler et al. "A Statistical Model of Human Pose and Body Shape". In: *Computer Graphics Forum* 28.2 (Apr. 2009), pp. 337–346. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2009.01373.x.

[Hei+05] Tobias Heimann et al. "3D Active Shape Models Using Gradient Descent Optimization of Description Length". In: *Information Processing in Medical Imaging* (2005), pp. 566–577. ISSN: 1611-3349. DOI: 10.1007/11505730_47.

[HG98] Peter Hahn and Thomas Grant. "Lower bounds for the quadratic assignment problem based upon a dual formulation". In: *Operations Research* 46.6 (1998), pp. 912–922.

[Hir03] Anil N Hirani. "Discrete exterior calculus". PhD thesis. Citeseer, 2003.

[HKG11] Qixing Huang, Vladlen Koltun, and Leonidas Guibas. "Joint shape segmentation with linear programming". In: *Proceedings of the 2011 SIGGRAPH Asia Conference on - SA '11* (2011). DOI: 10.1145/2024156.2024159.

[HT94] A. Hill and C. J. Taylor. "Automatic Landmark Generation for Point Distribution Models". In: *Procedings of the British Machine Vision Conference 1994* (1994). DOI: 10.5244/c.8.42.

[HTB03] Dirk Haehnel, Sebastian Thrun, and Wolfram Burgard. "An extension of the ICP algorithm for modeling nonrigid objects with mobile robots". In: *IJCAI*. Vol. 3. 2003, pp. 915–920.

[Hua+08a] Qi-Xing Huang et al. "Non-Rigid Registration Under Isometric Deformations". In: *Computer Graphics Forum* 27.5 (2008), pp. 1449–1457. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2008.01285.x.

[Hua+08b] Qi-Xing Huang et al. "Non-Rigid Registration Under Isometric Deformations". In: *Computer Graphics Forum* 27.5 (2008), pp. 1449–1457. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2008.01285.x.

[Hua+12] Qi-Xing Huang et al. "An optimization approach for extracting and encoding consistent maps in a shape collection". In: *ACM Transactions on Graphics* 31.6 (Nov. 2012), p. 1. ISSN: 0730-0301. DOI: 10.1145/2366145.2366186.

[Hub92] Peter J. Huber. "Robust Estimation of a Location Parameter". In: *Breakthroughs in Statistics* (1992), pp. 492–518. ISSN: 0172-7397. DOI: 10.1007/978-1-4612-4380-9_35.

[HWG14] Qixing Huang, Fan Wang, and Leonidas Guibas. "Functional map networks for analyzing and exploring large shape collections". In: *ACM Transactions on Graphics (TOG)* 33.4 (2014), p. 36.

[JK01] Michael Jünger and Volker Kaibel. "The QAP-polytope and the star transformation". In: *Discrete Applied Mathematics* 111.3 (2001), pp. 283–306.

[JK96a] Michael Jünger and Volker Kaibel. "A basic study of the qap-polytope". In: *Institut Für Informatik, Universität zu Köln*. 1996.

[JK96b] Michael Jünger and Volker Kaibel. "On the SQAP-polytope". In: *SIAM Journal of Optimization*. 1996.

[JZ06] Varun Jain and Hao Zhang. "Robust 3D shape correspondence in the spectral domain". In: *Shape Modeling and Applications, 2006. SMI 2006. IEEE International Conference on*. IEEE. 2006, pp. 19–19.

[Kai+11] Oliver van Kaick et al. "Prior Knowledge for Part Correspondence". In: *Computer Graphics Forum* 30.2 (Apr. 2011), pp. 553–562. ISSN: 0167-7055. DOI: 10.1111/j.1467-8659.2011.01893.x.

[Kar+99] Stefan E Karisch et al. "A dual framework for lower bounds of the quadratic assignment problem based on linearization". In: *Computing* 63.4 (1999), pp. 351–403.

[Kar77] H. Karcher. "Riemannian center of mass and mollifier smoothing". In: *Communications on Pure and Applied Mathematics* 30.5 (1977), pp. 509–541. ISSN: 1097-0312. DOI: 10.1002/cpa.3160300502.

[Kar95] Stefan E Karisch. *Nonlinear approaches for quadratic assignment and graph partition problems*. Inst. für Mathematik, Techn. Univ. Graz, 1995.

[KB57] Tjalling C Koopmans and Martin Beckmann. "Assignment problems and the location of economic activities". In: *Econometrica: journal of the Econometric Society* (1957), pp. 53–76.

[KB78] L Kaufman and Fernand Broeckx. "An algorithm for the quadratic assignment problem using Bender's decomposition". In: *European Journal of Operational Research* 2.3 (1978), pp. 207–211.

[Kez+15a] Itay Kezurer et al. "Tight relaxation of quadratic matching". In: *Computer Graphics Forum*. Vol. 34. 5. Wiley Online Library. 2015, pp. 115–128.

[Kez+15b] Itay Kezurer et al. "Tight relaxation of quadratic matching". In: *Computer Graphics Forum*. Vol. 34. 5. Wiley Online Library. 2015, pp. 115–128.

[KHS10] Evangelos Kalogerakis, Aaron Hertzmann, and Karan Singh. "Learning 3D mesh segmentation and labeling". In: *ACM Transactions on Graphics* 29.4 (July 2010), p. 1. ISSN: 0730-0301. DOI: 10.1145/1778765.1778839.

[Kim+12] Vladimir G. Kim et al. "Exploring collections of 3D models using fuzzy correspondences". In: *ACM Transactions on Graphics* 31.4 (July 2012), pp. 1–11. ISSN: 0730-0301. DOI: 10.1145/2185520.2185550.

[KLF11a] Vladimir G. Kim, Yaron Lipman, and Thomas Funkhouser. "Blended intrinsic maps". In: *ACM SIGGRAPH 2011 papers on - SIGGRAPH '11* (2011). DOI: 10.1145/1964921.1964974.

[KLF11b] Vladimir G Kim, Yaron Lipman, and Thomas Funkhouser. "Blended intrinsic maps". In: *ACM Transactions on Graphics (TOG)*. Vol. 30. 4. ACM. 2011, p. 79.

[KMP07] Martin Kilian, Niloy J Mitra, and Helmut Pottmann. "Geometric modeling in shape space". In: *ACM Transactions on Graphics (TOG)* 26.3 (2007), p. 64.

[Kov+13] Artiom Kovnatsky et al. "Coupled quasi-harmonic bases". In: *Computer Graphics Forum*. Vol. 32. 2pt4. Wiley Online Library. 2013, pp. 439–448.

[Kov+15] Artiom Kovnatsky et al. “Functional correspondence by matrix completion”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 905–914.

[KS04a] Vladislav Kraevoy and Alla Sheffer. “Cross-parameterization and compatible remeshing of 3D models”. In: *ACM SIGGRAPH 2004 Papers on - SIGGRAPH '04* (2004). DOI: 10.1145/1186562.1015811.

[KS04b] Vladislav Kraevoy and Alla Sheffer. “Cross-parameterization and compatible remeshing of 3D models”. In: *ACM Transactions on Graphics (TOG)*. Vol. 23. 3. ACM. 2004, pp. 861–869.

[KS13] Mathieu Desbrun Keenan Crane Fernando de Goes and Peter Schröder. “Digital Geometry Processing with Discrete Exterior Calculus”. In: *ACM SIGGRAPH 2013 courses*. SIGGRAPH '13. Anaheim, California: ACM, 2013.

[KS98] R. Kimmel and J. A. Sethian. “Computing geodesic paths on manifolds”. In: *Proceedings of the National Academy of Sciences* 95.15 (1998), pp. 8431–8435.

[KSC07] Jaehwan Kim, Kwang-Hyun Shim, and Seungjin Choi. “Soft geodesic kernel k-means”. In: *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*. Vol. 2. IEEE. 2007, pp. II–429.

[KST15] E de Klerk, R Sotirov, and U Truetsch. “A new semidefinite programming relaxation for the quadratic assignment problem and its computational perspectives”. In: *INFORMS Journal on Computing* 27.2 (2015), pp. 378–391.

[KT98] Aaron C.W. Kotcheff and Chris J. Taylor. “Automatic construction of eigenshape models by direct optimization”. In: *Medical Image Analysis* 2.4 (Dec. 1998), pp. 303–314. ISSN: 1361-8415. DOI: 10.1016/s1361-8415(98)80012-1.

[Kur+12] Sebastian Kurtek et al. “Elastic geodesic paths in shape space of parameterized surfaces”. In: *IEEE transactions on pattern analysis and machine intelligence* 34.9 (2012), pp. 1717–1730.

[Kur+13] Sebastian Kurtek et al. “Landmark-Guided Elastic Shape Analysis of Spherically-Parameterized Surfaces”. In: *Computer graphics forum (Eurographics)*. Vol. 32. 2pt4. Blackwell Publishing Ltd. 2013, pp. 429–438.

[Lag+16] Hamid Laga et al. “Numerical Inversion of SRNF Maps for Elastic Shape Analysis of Genus-Zero Surfaces”. In: *arXiv preprint arXiv:1610.04531* (2016).

[Läh+16] Z Lähner et al. “SHREC’16: Matching of deformable shapes with topological noise”. In: *Proc. 3DOR* 2.8 (2016), p. 11.

[Law63] Eugene L Lawler. “The quadratic assignment problem”. In: *Management science* 9.4 (1963), pp. 586–599.

[Lév+02] Bruno Lévy et al. “Least squares conformal maps for automatic texture atlas generation”. In: *Proceedings of the 29th annual conference on Computer graphics and interactive techniques - SIGGRAPH ’02* (2002). ISSN: 0730-0301. DOI: 10.1145/566570.566590.

[LF09a] Yaron Lipman and Thomas Funkhouser. “Möbius voting for surface correspondence”. In: *ACM SIGGRAPH 2009 papers on - SIGGRAPH ’09* (2009). DOI: 10.1145/1576246.1531378.

[LF09b] Yaron Lipman and Thomas Funkhouser. “Möbius voting for surface correspondence”. In: *ACM Transactions on Graphics (TOG)*. Vol. 28. 3. ACM. 2009, p. 72.

[LH05] Marius Leordeanu and Martial Hebert. “A spectral technique for correspondence problems using pairwise constraints”. In: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. Vol. 2. IEEE. 2005, pp. 1482–1489.

[Loi+07] Eliane Maria Loiola et al. “A survey for the quadratic assignment problem”. In: *European journal of operational research* 176.2 (2007), pp. 657–690.

[LPG12] Yang Liu, Balakrishnan Prabhakaran, and Xiaohu Guo. “Point-based manifold harmonics”. In: *IEEE Transactions on Visualization and Computer Graphics* 18.10 (2012), pp. 1693–1703.

[LRF10a] Yaron Lipman, Raif M. Rustamov, and Thomas A. Funkhouser. “Biharmonic Distance”. In: *ACM Trans. Graph.* 29.3 (July 2010), 27:1–27:11. ISSN: 0730-0301. DOI: 10.1145/1805964.1805971.

[LRF10b] Yaron Lipman, Raif M Rustamov, and Thomas A Funkhouser. “Biharmonic distance”. In: *ACM Transactions on Graphics (TOG)* 29.3 (2010), p. 27.

[LS91] László Lovász and Alexander Schrijver. “Cones of matrices and set-functions and 0–1 optimization”. In: *SIAM Journal on Optimization* 1.2 (1991), pp. 166–190.

[LSP08] Hao Li, Robert W. Sumner, and Mark Pauly. “Global Correspondence Optimization for Non-Rigid Registration of Depth Scans”. In: *Computer Graphics Forum* 27.5 (July 2008), pp. 1421–1430. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2008.01282.x.

[Luo+10] Zhi-Quan Luo et al. “Semidefinite relaxation of quadratic optimization problems”. In: *IEEE Signal Processing Magazine* 27.3 (2010), p. 20.

[M+07] Andriy Myronenko, Xubo Song, Miguel A Carreira-Perpinán, et al. “Non-rigid point set registration: Coherent point drift”. In: *Advances in Neural Information Processing Systems* 19 (2007), p. 1009.

[Mar+16a] Haggai Maron et al. “Point registration via efficient convex relaxation”. In: *ACM Transactions on Graphics* 35.4 (July 2016), pp. 1–12. ISSN: 0730-0301. DOI: 10.1145/2897824.2925913.

[Mar+16b] Haggai Maron et al. “Point registration via efficient convex relaxation”. In: *ACM Transactions on Graphics (TOG)* 35.4 (2016), p. 73.

[MD58] NS Mendelsohn and AL Dulmage. “The Convex Hull of Sub-Permutation Matrices”. In: *Rn* 1 (1958), R2.

[Mey+03] Mark Meyer et al. “Discrete differential-geometry operators for triangulated 2-manifolds”. In: *Visualization and mathematics III*. Springer, 2003, pp. 35–57.

[MF17] Matter and Form. *Bevel*. 2017. URL: https://bevel3d.net/store (visited on 08/30/2017).

[MM99] Songrit Maneewongvatana and David M Mount. “It’s okay to be skinny, if your friends are fat”. In: *Center for Geometric Computing 4th Annual Workshop on Computational Geometry*. Vol. 2. 1999, pp. 1–8.

[MMP87] Joseph S. B. Mitchell, David M. Mount, and Christos H. Papadimitriou. “The Discrete Geodesic Problem”. In: *SIAM J. Comput.* 16.4 (Aug. 1987), pp. 647–668. ISSN: 0097-5397. DOI: 10.1137/0216045.

[Mos10] APS Mosek. “The MOSEK optimization software”. In: 54 (2010), pp. 2–1.

[MS10] Andriy Myronenko and Xubo Song. “Point Set Registration: Coherent Point Drift”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.12 (Dec. 2010), pp. 2262–2275. ISSN: 0162-8828. DOI: 10.1109/tpami.2010.46.

[NFS15] Richard A. Newcombe, Dieter Fox, and Steven M. Seitz. “DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2015). DOI: 10.1109/cvpr.2015.7298631.

[Ngu+11a] Andy Nguyen et al. “An Optimization Approach to Improving Collections of Shape Maps”. In: *Computer Graphics Forum* 30.5 (Aug. 2011), pp. 1481–1491. ISSN: 0167-7055. DOI: 10.1111/j.1467-8659.2011.02022.x.

[Ngu+11b] Andy Nguyen et al. “An optimization approach to improving collections of shape maps”. In: *Computer Graphics Forum*. Vol. 30. 5. Wiley Online Library. 2011, pp. 1481–1491.

[Noc80] Jorge Nocedal. “Updating quasi-Newton matrices with limited storage”. In: *Mathematics of computation* 35.151 (1980), pp. 773–782.

[Ovs+10] Maks Ovsjanikov et al. “One Point Isometric Matching with the Heat Kernel”. In: *Computer Graphics Forum* 29.5 (Sept. 2010), pp. 1555–1564. ISSN: 0167-7055. DOI: 10.1111/j.1467-8659.2010.01764.x.

[Ovs+12] Maks Ovsjanikov et al. “Functional maps: a flexible representation of maps between shapes”. In: *ACM Transactions on Graphics (TOG)* 31.4 (2012), p. 30.

[Ovs+13] M. Ovsjanikov et al. “Analysis and Visualization of Maps Between Shapes”. In: *Computer Graphics Forum* 32.6 (May 2013), pp. 135–145. ISSN: 0167-7055. DOI: 10.1111/cgf.12076.

[Pan+13] Daniele Panozzo et al. “Weighted averages on surfaces”. In: *ACM Transactions on Graphics (TOG)* 32.4 (2013), p. 60.

[Pen+15] Jiming Peng et al. “Semi-definite programming relaxation of quadratic assignment problems based on nonredundant matrix splitting”. In: *Computational Optimization and Applications* 60.1 (2015), pp. 171–198.

[PP93] Ulrich Pinkall and Konrad Polthier. “Computing discrete minimal surfaces and their conjugates”. In: *Experimental mathematics* 2.1 (1993), pp. 15–36.

[PR09] Janez Povh and Franz Rendl. “Copositive and semidefinite relaxations of the quadratic assignment problem”. In: *Discrete Optimization* 6.3 (2009), pp. 231–241.

[PS13] C.H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Dover Books on Computer Science. Dover Publications, 2013. ISBN: 9780486320137.

[PVZ15] Javier Pena, Juan C Vera, and Luis F Zuluaga. “Completely positive reformulations for polynomial optimization”. In: *Mathematical Programming* 151.2 (2015), pp. 405–431.

[QH07] Huaijun Qiu and E.R. Hancock. “Clustering and Embedding Using Commute Times”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 29.11 (Nov. 2007), pp. 1873–1890. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2007.1103.

[Ris78] J. Rissanen. “Modeling by shortest data description”. In: *Automatica* 14.5 (Sept. 1978), pp. 465–471. ISSN: 0005-1098. DOI: 10.1016/0005-1098(78)90005-5.

[RL01a] S. Rusinkiewicz and M. Levoy. "Efficient variants of the ICP algorithm". In: *Proceedings Third International Conference on 3-D Digital Imaging and Modeling* (2001). DOI: 10.1109/im.2001.924423.

[RL01b] Szymon Rusinkiewicz and Marc Levoy. "Efficient variants of the ICP algorithm". In: *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*. IEEE. 2001, pp. 145–152.

[RMC15] Emanuele Rodolà, Michael Moeller, and Daniel Cremers. "Point-wise Map Recovery and Refinement from Functional Correspondence". In: *arXiv preprint arXiv:1506.05603* (2015).

[Rod+15] Emanuele Rodolà et al. "Partial functional correspondence". In: *arXiv preprint arXiv:1506.05274* (2015).

[RS06] Franz Rendl and Renata Sotirov. "Bounds for the quadratic assignment problem using the bundle method". In: *Mathematical Programming* 109.2-3 (Sept. 2006), pp. 505–524. ISSN: 1436-4646. DOI: 10.1007/s10107-006-0038-8.

[Rus+13a] Raif M Rustamov et al. "Map-based exploration of intrinsic shape differences and variability". In: *ACM Transactions on Graphics (TOG)* 32.4 (2013), p. 72.

[Rus+13b] Raif M Rustamov et al. "Map-based exploration of intrinsic shape differences and variability". In: *ACM Transactions on Graphics (TOG)* 32.4 (2013), p. 72.

[Rus07] Raif M Rustamov. "Laplace-Beltrami eigenfunctions for deformation invariant shape representation". In: *Proceedings of the fifth Eurographics symposium on Geometry processing*. Eurographics Association. 2007, pp. 225–233.

[RWP06] Martin Reuter, Franz-Erich Wolter, and Niklas Peinecke. "Laplace-Beltrami spectra as 'Shape-DNA'of surfaces and solids". In: *Computer-Aided Design* 38.4 (2006), pp. 342–366.

[SA07] Olga Sorkine and Marc Alexa. "As-rigid-as-possible surface modeling". In: *Symposium on Geometry processing*. Vol. 4. 2007.

[SG76] Sartaj Sahni and Teofilo Gonzalez. "P-complete approximation problems". In: *Journal of the ACM (JACM)* 23.3 (1976), pp. 555–565.

[Sid+11] Oana Sidi et al. "Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering". In: *Proceedings of the 2011 SIGGRAPH Asia Conference on - SA '11* (2011). DOI: 10.1145/2024156.2024160.

[SOG09] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. "A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion". In: *Computer graphics forum*. Vol. 28. 5. Wiley Online Library. 2009, pp. 1383–1392.

[Sol+14] Justin Solomon et al. "Earth Mover's Distances on Discrete Surfaces". In: *ACM Trans. Graph.* 33.4 (July 2014), 67:1–67:12. ISSN: 0730-0301. DOI: 10.1145/2601097.2601175.

[SP04] Robert W Sumner and Jovan Popović. "Deformation transfer for triangle meshes". In: *ACM Transactions on Graphics (TOG)*. Vol. 23. 3. ACM. 2004, pp. 399–405.

[SRS07] Christian Schellewald, Stefan Roth, and Christoph Schnörr. "Evaluation of a convex relaxation to a quadratic assignment matching approach for relational object views". In: *Image and Vision Computing* 25.8 (2007), pp. 1301–1314.

[SSK05] Mirko Sattler, Ralf Sarlette, and Reinhard Klein. "Simple and efficient compression of animation sequences". In: *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. ACM. 2005, pp. 209–217.

[SSP07] Robert W. Sumner, Johannes Schmid, and Mark Pauly. "Embedded deformation for shape manipulation". In: *ACM Transactions on Graphics* 26.3 (July 2007), p. 80. ISSN: 0730-0301. DOI: 10.1145/1276377.1276478.

[Sum+05a] Robert W. Sumner et al. "Mesh-based inverse kinematics". In: *ACM SIGGRAPH 2005 Papers on - SIGGRAPH '05* (2005). DOI: 10.1145/1186822.1073218.

[Sum+05b] Robert W Sumner et al. "Mesh-based inverse kinematics". In: *ACM Transactions on Graphics (TOG)*. Vol. 24. 3. ACM. 2005, pp. 488–495.

[Sun+13] M. Sunkel et al. "A Correlated Parts Model for Object Detection in Large 3D Scans". In: *Computer Graphics Forum* 32.2pt2 (May 2013), pp. 205–214. ISSN: 0167-7055. DOI: 10.1111/cgf.12040.

[SY12] Yusuf Sahillioglu and Yücel Yemez. "Minimum-distortion isometric shape correspondence using EM algorithm". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 34.11 (2012), pp. 2203–2215.

[Tam+13] Gary KL Tam et al. "Registration of 3D point clouds and meshes: a survey from rigid to nonrigid". In: *Visualization and Computer Graphics, IEEE Transactions on* 19.7 (2013), pp. 1199–1217.

[TDM11] J. Rafael Tena, Fernando De la Torre, and Iain Matthews. "Interactive region-based linear 3D face models". In: *ACM SIGGRAPH 2011 papers on - SIGGRAPH '11* (2011). DOI: 10.1145/1964921.1964971.

[Tho03] Hans Henrik Thodberg. "Minimum Description Length Shape and Appearance Models". In: *Information Processing in Medical Imaging* (2003), pp. 51–62. ISSN: 1611-3349. DOI: 10.1007/978-3-540-45087-0_5.

[TK04] Yanghai Tsin and Takeo Kanade. "A Correlation-Based Approach to Robust Point Set Registration". In: *Computer Vision - ECCV 2004* (2004), pp. 558–569. ISSN: 1611-3349. DOI: 10.1007/978-3-540-24672-5_44.

[TO03] H. H. Thodberg and H. Olafsdottir. "Adding Curvature to Minimum Description Length Shape Models". In: *Procedings of the British Machine Vision Conference 2003* (2003). DOI: 10.5244/c.17.26.

[Van+11] Oliver Van Kaick et al. "A survey on shape correspondence". In: *Computer Graphics Forum*. Vol. 30. 6. Wiley Online Library. 2011, pp. 1681–1707.

[Váš+14] L Váša et al. "Compressing dynamic meshes with geometric laplacians". In: *Computer Graphics Forum*. Vol. 33. 2. Wiley Online Library. 2014, pp. 145–154.

[Ves+17] Matthias Vestner et al. "Product Manifold Filter: Non-Rigid Shape Correspondence via Kernel Density Estimation in the Product Space". In: *arXiv preprint arXiv:1701.00669* (2017).

[Vik06] Thomas Viklands. "Algorithms for the weighted orthogonal Procrustes problem and other least squares problems". PhD thesis. Umea University, Sweden, 2006.

[Wan+16] Peng Wang et al. "Large-scale binary quadratic optimization using semidefinite relaxation and applications". In: *IEEE transactions on pattern analysis and machine intelligence* (2016).

[War+07] Max Wardetzky et al. "Discrete laplace operators: no free lunch". In: *Proceedings of the Fifth Eurographics Symposium on Geometry Processing, Barcelona, Spain, July 4-6, 2007*. 2007, pp. 33–37. DOI: 10.2312/SGP/SGP07/033-037.

[WSV13] Peng Wang, Chunhua Shen, and Anton Van Den Hengel. "A fast semidefinite approach to solving binary quadratic problems". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2013, pp. 1312–1319.

[Xu+15] C. Xu et al. "Fast Wavefront Propagation (FWP) for Computing Exact Geodesic Distances on Meshes". In: *Visualization and Computer Graphics, IEEE Transactions on* 21.7 (July 2015), pp. 822–834. ISSN: 1077-2626. DOI: 10.1109/TVCG.2015.2407404.

[Xu04] Guoliang Xu. “Discrete Laplace–Beltrami operators and their convergence”. In: *Computer Aided Geometric Design* 21.8 (Oct. 2004), pp. 767–784. ISSN: 0167-8396. DOI: 10.1016/j.cagd.2004.07.007.

[XW09] Shi-Qing Xin and Guo-Jin Wang. “Improving Chen and Han’s Algorithm on the Discrete Geodesic Problem”. In: *ACM Trans. Graph.* 28.4 (Sept. 2009), 104:1–104:8. ISSN: 0730-0301. DOI: 10.1145/1559755.1559761.

[XW10] Shi-Qing Xin and Guo-Jin Wang. “Applying the Improved Chen and Han’s Algorithm to Different Versions of Shortest Path Problems on a Polyhedral Surface”. In: *Comput. Aided Des.* 42.10 (Oct. 2010), pp. 942–951. ISSN: 0010-4485. DOI: 10.1016/j.cad.2010.05.009.

[XYH12] Shi-Qing Xin, Xiang Ying, and Ying He. “Constant-time All-pairs Geodesic Distance Query on Triangle Meshes”. In: *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*. I3D ’12. Costa Mesa, California: ACM, 2012, pp. 31–38. ISBN: 978-1-4503-1194-6. DOI: 10.1145/2159616.2159622.

[Yan+16] Jiaolong Yang et al. “Go-ICP: A Globally Optimal Solution to 3D ICP Point-Set Registration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.11 (Nov. 2016), pp. 2241–2254. ISSN: 2160-9292. DOI: 10.1109/tpami.2015.2513405.

[Yeh+11] I Yeh et al. “Template-based 3d model fitting using dual-domain relaxation”. In: *Visualization and Computer Graphics, IEEE Transactions on* 17.8 (2011), pp. 1178–1190.

[Yos+14] Yusuke Yoshiyasu et al. “As-Conformal-As-Possible Surface Registration”. In: *Computer Graphics Forum*. Vol. 33. 5. Wiley Online Library. 2014, pp. 257–267.

[Zha+04] Li Zhang et al. “Spacetime Faces: High-Resolution Capture for Modeling and Animation”. In: *Data-Driven 3D Facial Animation* (2004), pp. 248–276. DOI: 10.1007/978-1-84628-907-1_13.

[Zha+98] Qing Zhao et al. “Semidefinite programming relaxations for the quadratic assignment problem”. In: *Journal of Combinatorial Optimization* 2.1 (1998), pp. 71–109.

[ZPK16] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. “Fast Global Registration”. In: *Lecture Notes in Computer Science* (2016), pp. 766–782. ISSN: 1611-3349. DOI: 10.1007/978-3-319-46475-6_47.