ABSTRACT

AOUADA, DJAMILA. Geometric, Statistical, and Topological Modeling of Intrinsic Data Manifolds: Application to 3D Shapes. (Under the direction of Professor Hamid Krim).

The increasing size and complexity of data often invokes the extraction of information from their reduced representations while preserving their inherent structure. In this thesis, we explore the statistical, geometric and topological intrinsic information contained in high dimensional data. We focus on applications related to 3-dimensional objects, and model their 2-dimensional surfaces using compact curved-skeletal models that we refer to as "squigraphs". These models are multi-level representations that superpose global topological and local geometric 3D shape descriptors. Squigraphs are subsequently used for classification, and ensure a high discrimination between in-class 3-dimensional shapes.

The extraction of squigraphs starts by sampling the surface of an object for a resulting set of curves. This may be accomplished by defining an appropriate intrinsic characteristic function on the surface itself, referred to as a Morse function; which we use in a two-phase approach. To ensure the invariance of the final representation to isometric transforms, we choose the Morse function to be an intrinsic global geodesic function. The first phase is a coarse representation through a reduced topological Reeb graph. We use it for a meaningful decomposition of shapes into primitives. At the second phase, we add detailed geometric information by tracking the evolution of Morse function's level curves along each primitive. We then embed the manifold corresponding to this evolution of curves into $\mathbb{R}^3$, and obtain a simple space curve. We further define a Riemannian metric to quantitatively compare the geometry of shapes.

We point the flexibility of our techniques for other applications, namely, face recognition, behavioral modeling, and sensor network data analysis. While all these applications face the same curse of dimensionality, we show that they may be formalized under similar geometrical settings.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2009		**00-00-2009 to 00-00-2009**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Geometric, Statistical, and Topological Modeling of Intrinsic Data Manifolds: Application to 3D Shapes	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **North Carolina State University,Department of Electrical and Computer Engineering,Raleigh,NC,27695**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**136**	

Geometric, Statistical, and Topological Modeling of Intrinsic Data Manifolds: Application to 3D Shapes

by

Djamila Aouada

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fullfillment of the
requirements for the Degree of
Doctor of Philosophy

Electrical Engineering

Raleigh, North Carolina

2009

APPROVED BY:

Prof. Michael J. Escuti	Prof. Kazufumi Ito
Prof. Hamid Krim Chair of Advisory Committee	Prof. Alexandra Duel-Hallen

DEDICATION

To my mother

Anna Anatolievna Ismailova

And

To my father

Mohamed Aouada

BIOGRAPHY

Djamila Aouada was born on November 10, 1982 to Mohamed Aouada and Anna A. Ismailova, in Blida, Algeria. She received the State Engineering degree (Diplôme d'Ingénieur d'État) in electrical engineering (electronics) in June 2005, from the École Nationale Polytechnique (ENP), Algiers, Algeria. In August 2005, she joined the Vision, Information and Statistical Signal Theories and Applications group (VISSTA) at North Carolina State University (NCSU), Raleigh, NC, as a research assistant, where she began working toward a Ph.D. degree in signal processing and computer vision.

From June to August 2007, Djamila participated in the data sciences summer school at Los Alamos National Laboratory (LANL), Los Alamos, NM, as part of the Geometric Measure Theory (GMT) group led by Dr. Kevin R. Vixie. From July to September 2008, she worked as a consultant for Alcatel-Lucent Bell Laboratories, Murray Hill, NJ, in the department of Industrial Mathematics and Operations Research supervised by Dr. Yuliy Baryshnikov.

Djamila's research interests span the areas of signal and image processing, computer vision, pattern recognition and data modeling. Her current emphasis is on modeling 2D and 3D shapes with various applications ranging from target recognition to biomedical imaging.

Djamila is member of the Institute of Electrical and Electronics Engineers (IEEE) since 2005, a member of the Society of Women Engineers (SWE) since 2007, and a member of the Eta Kappa Nu electrical engineering honor society (HKN) since 2007.

ACKNOWLEDGMENTS

A Ph.D. degree is often a very complex experience that inescapably involves more than one's brains. I wish to leave some words to reflect on the human side of my three and a half year journey; and acknowledge all the people who contributed to the completion of this work.

I thus would like to start by expressing my sincere recognition for all those who are behind the success of audiovisual technologies and services. You have made possible my daily trips across the Atlantic. I owe you, without a doubt, the happy ending of this chapter of my life.

My greatest thanks go to my parents who were there whenever I called. No matter the time difference, no matter the difficulties, you were always there to make the toughest times more bearable. It is extraordinary all those full nights you spent remotely looking after your sick, but grown-up kid. Your support, care, as well as the love for science and mathematics you both transmitted to me, have been keys in completing this work.

Papa, Mama, this thesis is your achievement !

The *Aouada dream team* would not be complete without my brothers, Saïd and Rostom. Thank you for your love and support. I am not sure that this degree is worth the years of separation; but at least, I hope that you are proud of your little sister.

I would also like to thank my dissertation advisor, Professor Hamid Krim, for recruiting me. The best thing I will keep from working in your group is by far your very first advice: "Research is like art; be creative; allow yourself to dream". Your unconventional trust in your students and their abilities allowed me to explore ideas and directions that would have otherwise been forbidden.

In addition, since my second semester in the Ph.D. program, and in the context of the ECE department interdisciplinary seminars, you have given me the opportunity to meet, almost every Friday, world renowned scientists. All the fascinating discussions I had with them and their challenging questions had an important impact in tuning this thesis. I would like to especially acknowledge, in chronological order, Dr. Ilya Pollak, Dr. David Dreisigmeyer, Dr. Nikos Paragios, Dr. Anuj Srivastava, Dr. Sheila Hemami, Dr. Sven Dickinson, Dr. Yuliy Baryshnikov, Dr. Yosi Keller, Dr. Robert Calderbank, and Dr. Ahmed Tewfik.

Many other professors have been part of this journey. I sincerely thank, in alphabetical order, Dr. Alexandra Duel-Hallen, Dr. Michael Escuti, Dr. Kazufumi Ito, Dr. Irina Kogan, and Dr. Demetrio Labate for serving, at a point or another, in my dissertation committee. Dr. Escuti and Dr. Duel-Hallen always found the time to sit down with me whenever I solicited their advice; moreover, Dr. Michael Escuti as well as Dr. Simon Morgan and Dr. Kevin Vixie, whom I first met at Los Alamos National Laboratory, are three professors who have shown an exceptional interest in my work and who went beyond their duties in encouraging me. Your genuine belief in my potential has kept me going with confidence.

I also thank Professor Betty Lise Anderson, my e-mentor, from Ohio State University. Requesting an e-mentor has shown to be a smart move thanks to your incredible devotion to transmitting your experience. Our relationship has surpassed my expectations. You are to me a model professor. I will always look up to you, and strive to give to others as much as you do.

I finally would like to thank all the nice people I met during my stay in the US. I am glad to realize that their number is far too high to possibly provide an exhaustive list. My thoughts go to all former and current graduate students in the ECE department, with a special thanks to Sheng Yi; all my former roommates, with a special thanks to Liping Li; all my class instructors, with a special thanks to Dr. Hyeokho Choi, may you rest in peace; and last, but not least, I thank all the unknown smiling faces, with a special thanks to H. A. N.; a young lady who approached me with a napkin from Panera bread before disappearing without uttering a word. I would like to express my gratitude to everyone by ending my acknowledgments with your napkin.

This thesis has been funded by the following agencies:

- Defense Threat Reduction Agency (DTRA). *September 2008 - February 2009.*
- US Office of Naval Research (ONR). *March 2007 - December 2007.*
- US Air Force Office of Scientific Research (AFOSR). *August 2005 - February 2007.*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

3D (2D)	3 (2) dimensional
aMRG	Augmented Multi-resolution Reeb Graph
AUC	Area Under the Curve
CCR	Classification by Characteristic Resolution
GGF	Global Geodesic Function
Isomap	Isometric feature mapping
JSD	Jensen Shannon Divergence
LLE	Locally Linear Embedding
LTMADS	Lower Triangular Mesh Adapted Direct Search algorithm
MDS	Multidimensional Scaling
PCA	Principal Component Analysis
PDF	Probability Density Function
ROC	Receiver Operating Characteristic
SVD	Singular Value Decomposition

Chapter 1

Introduction

The last decade has seen an important growth in applied problems that are tightly related to the explosive increase in data sizes and complexities. The nature of these applications calls for a parsimonious representation of the data. The most challenging constraint associated with this task is the preservation of the inherent data structure in the course of analysis and modeling.
In this thesis, we explore the statistical, geometric and topological intrinsic information contained in high dimensional data structures. Our objective is to define simpler models with significantly reduced complexities, which will subsequently facilitate the usual computationally demanding applications such as data classification and data retrieval. Reducing the dimension of a data set is often based on information that is characteristic and restricted to a small portion of the whole data space to be fully expressed or summarized. Finding and correctly manipulating the necessary and sufficient features constitute the central goal of our work.

1.1 Motivations and overview

In the present thesis, we focus on modeling the 2-dimensional surface of a 3D object. Our interest in this kind of data is motivated by a variety of applications ranging from target recognition and face identification to shape retrieval and behavior analysis.
While a number of available 3D shape modeling techniques [1, 2, 3, 4, 5, 6] yield satisfactory object classification results, many applications require a more refined and efficient identi-

fication/recognition of objects in a given class. To that end, we invoke Morse theory in a two-phase approach. To ensure the invariance of a 3D object final representation to isometric transforms, we choose the Morse function to be a simple and intrinsic global geodesic function defined on its surface. At first, a coarse representation through a reduced topological Reeb graph is obtained. This is in turn used in a meaningful decomposition of shapes into primitives. Following this step, we add detailed geometric information by tracking the evolution of Morse function's level curves along each primitive. We subsequently embed the manifold corresponding to the evolution of these curves into $\mathbb{R}^3$, and obtain a simple space curve. By combining phases one and two, we construct new graphs rich in topological and geometric information that we refer to as squigraphs.

Our proposed analysis framework essentially exploits curves in space (which are representative of 3D objects). As we further elaborate later, spatial curves may indeed be exploited in various configurations. They may, for instance, be extracted as contours of landmark surfaces [7], as level curves of a Morse function [8], or also as elements of curved skeletons [9, 10]. All these techniques, despite their differences, commonly rely on curves' properties in solving computer vision problems. This approach is conceptually motivated by the fact that curves in 3D are fairly well known geometric entities; moreover, under some conditions, they can accurately describe the overall geometry of an object in 3D space [7]. Translating the constraints of 3D shape representation techniques to curves, reduces the level of analytical difficulty associated with the 3D representation problem and makes it more tractable. To this end, we focus our efforts on defining an effective modeling framework for curves in space along with the corresponding Riemannian metric.

While 3D objects form the core of the applications of interest in this thesis, we also discuss the exploration of other types of complex data (*e.g.,* video frames, sensor network data). Our techniques are thus readily adaptable to various applications as long as the measured data may be correctly assumed to lie on a smooth manifold. Indeed, we show how our proposed Whitney modeling of curves can be used to represent the evolution of video frames, where a frame is equated to a segmented silhouette. The evolution of a set of silhouettes that is ordered in time becomes a 1-dimensional manifold that can be embedded in 3D.

A less visually intuitive problem is that of analyzing sensor network data. Our goal is to develop mathematical models which will lead to a clear understanding of the interde-

pendencies between different nodes within one network, as well as of the interdependencies between different networks. Such models will be crucial in guiding the implementation of network infrastructures/protocols which are robust and resilient to intrusions. Our approach starts by abstracting networks from their physical layered description. We view a network as a set of features which best characterize it. The same feature space may house multiple distinct networks. Our hypothesis is that each network is characterized by a smooth topologically homogeneous manifold embedded in the network feature space; therefore, we propose to carry out our work by relying on notions from topological and geometric manifold learning. The interdependencies between networks may thus be reflected by statistical correlations between the different space elements (*e.g.,* manifolds and points), where data points on a manifold are representative of nodes in a network. In addition, an attack or any intrusion is interpreted as having the effect of destroying the initial topological structure of a manifold. Our work is hence geared toward detecting and determining any topological changes.

1.2 Contributions

We summarize our main contributions bellow.

- We have investigated the representation, classification and recognition of 3D objects. We defined and implemented a complete machinery for a precise identification and recognition of 3D objects from multiple classes and within one class. In light of the fact that shapes can be very complex, all our proposed techniques promote the necessity to distinguish between global and local features, or between the coarse and fine representations. For each of our proposed applications, we first determine the group of transforms to which the corresponding representations are to be invariant. Our focus has been on isometries, as they gather rigid and non-rigid shapes and cover a wide range of applications from biomedical imaging to computer animations. To that end, we make an extensive use of Morse theory throughout this work. Specifically, we choose to define a fully intrinsic global geodesic Morse function on the surface of 3D objects. We exploit the information contained in this function in a two-phase approach. The first phase is a coarse representation through a reduced topological Reeb graph, which is instrumental in decomposing a complex shape into primitives.

In a second phase, the objective is to provide a more fine representation. Taking advantage of the same Morse function, we proceed to analyze the evolution of its level curves along each primitive and further reduce its representation to a simple curve in 3D by using the Whitney embedding theorem. In a very ludic way, we propose to combine phases one and two to construct "squigraphs" [10]. Experiments show that squigraphs are more general than existing techniques [11]; they achieve very competitive classification rates in comparison to those achieved by classical shape descriptors. Their performance, however, becomes clearly superior when finer classification and identification operations are desired.

- In the same spirit of intrinsic modeling, we introduce a Riemannian metric to compare curves in space. The classification/recognition of 3D objects by way of squigraphs, forms the primary motivation of developing a quantitative comparison framework for space curves. Our additional objective is to limit the invariance of curves' shapes to rigid transforms. Inspired by results in visual cognition and human perception, we propose a technique that simultaneously verifies all the following properties: high discrimination level, independence of parametrization and independence of any reference point, uniqueness property, as well as a good preservation of the correspondence between curves to effectively tackle partial matching problems [12, 13].

- In a different context yet with the same view of geometry-driven analysis, we seek to understand network data through statistical manifold learning. The key idea in this problem is to consider data samples as lying on one or more manifolds. From there, one may start studying the topo-geometric and statistical properties of the data by calling upon well-established theorems. The main contribution of this work is to harness space and time information with the correlation between networks' nodes.

1.3 Outline

The present thesis is organized as follows:

- We cover in Chapter 2 the necessary background to understand the developments in subsequent chapters. We start by describing the common discrete representation of 3D objects, *i.e.*, the triangulated meshes; and by providing a brief introduction

to Morse theory along with a Global Geodesic Function (GGF) as a Morse function which we repeatedly use throughout this thesis.

- Chapter 3 presents a statistical analysis of the GGF and the new resulting object classification algorithm. We refer to this algorithm as Classification using object Characteristic Resolution (CCR).

- Chapter 4 describes the partitioning of 3D objects using their Reeb graphs. The extraction of Reeb graphs is herein based on level curves of a Morse function instead of its intervals.

- Chapter 5 defines squigraphs and the Whitney embedding for the geometric modeling of shapes.

- In Chapter 6, we define a new similarity invariant signature that provides a Riemannian metric for an intrinsic comparison of shapes via their squigraphs.

- Chapter 7 demonstrates, through extensive experiments, the effectiveness of squigraphs when combined with the distance metric defined in Chapter 6.

- Chapter 8 illustrates the problem of sensor network analysis with the same spirit of intrinsic modeling.

- We provide conclusions and possible future research directions in Chapter 9.

Chapter 2

Background

Research interest in 3D object analysis has witnessed an explosive growth over the last few years. While this may in part be explained by an equally impressive growth in computing power, the availability of 3D data acquisition systems and ready access to it at relatively low cost, have been key in addressing the numerous problems which arise in applications. Laser scanners, ranging cameras and others have indeed made solutions to multimedia applications, biometrics, and computer graphics more realistic and affordable. The wide distribution of 3D data over the internet at no-cost is also testimony to the high level of interest in the area [14, 15, 16, 17, 18].
In what follows, we describe the discrete representation that is often used to present 3D objects. We also describe the continuous surface of these objects that we use in our theoretical formulations.

2.1 Digital and continuous 3D objects

3D objects are commonly represented as polygonal or triangulated meshes. Two matrices $\mathcal{F}$ and $\mathcal{V}$ are the usual digital representation of a 3D object, and are often of prohibitive size. This in turn, unveils the computational challenge often encountered in practice and particularly in 3D shape classification and recognition applications. A triangulated mesh M is thus denoted by $M = (\mathcal{V}, \mathcal{F})$, where $\mathcal{V} = [\mathbf{v}_1, ..., \mathbf{v}_n]$ is the matrix of vertices, such that $\mathbf{v}_i$ contains the Euclidean coordinates of the vertex labeled i, *i.e.*, $\mathbf{v}_i = [x_i, y_i, z_i]^T$ and $\mathcal{F} = [\mathbf{f}_1, ..., \mathbf{f}_m]$ is the matrix of faces, such that $\mathbf{f}_j$ contains the labels

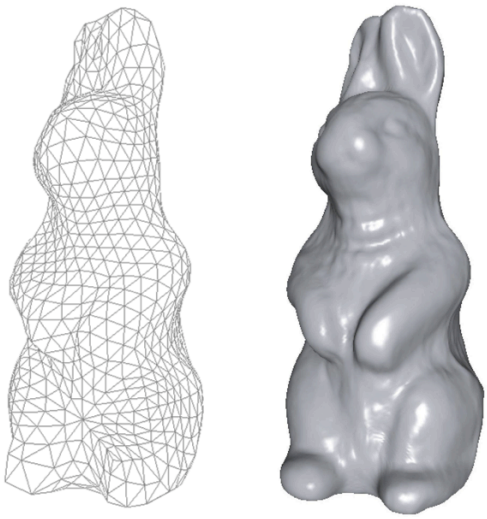


Figure 2.1: Digital and continuous representations of a 3D object.

of all the vertices of the face under the label j, $j = 1, \cdots, m$. We refer to the number n as the resolution for a digital representation of a 3D object. Mathematically, however, we view 3D objets as smooth and compact surfaces $\mathcal{S}$ embedded in $\mathbb{R}^3$. We show in Figure 2.1 an example of a triangulated mesh, and the corresponding continuous surface.

2.2 Morse theory

In order to efficiently represent 3D objects and extract their topological and geometric features, we make an extensive use of Morse theory [19, 37]. Morse theory states that it is possible to define a particular smooth function $f(\cdot)$ on a smooth surface $\mathcal{S}$ (or more generally a manifold $\mathcal{S}$), and track its critical points in order to study the topology of $\mathcal{S}$. Such a function $f(\cdot)$ is called a Morse function and is defined as follows:

Definition 1 (Morse function) *A smooth function $f : \mathcal{S} \to \mathbb{R}$ on a smooth manifold $\mathcal{S}$ is called* Morse *if all of its critical points are non-degenerate.*

A standard Morse function is a *height* function [37, 21, 22]. The image of a point $\mathbf{p}$ on $\mathcal{S}$ via the height function is reduced to its z coordinate as shown in Figure 2.2 (b). Any representation that is based on the height function is clearly varying, as critical points of a

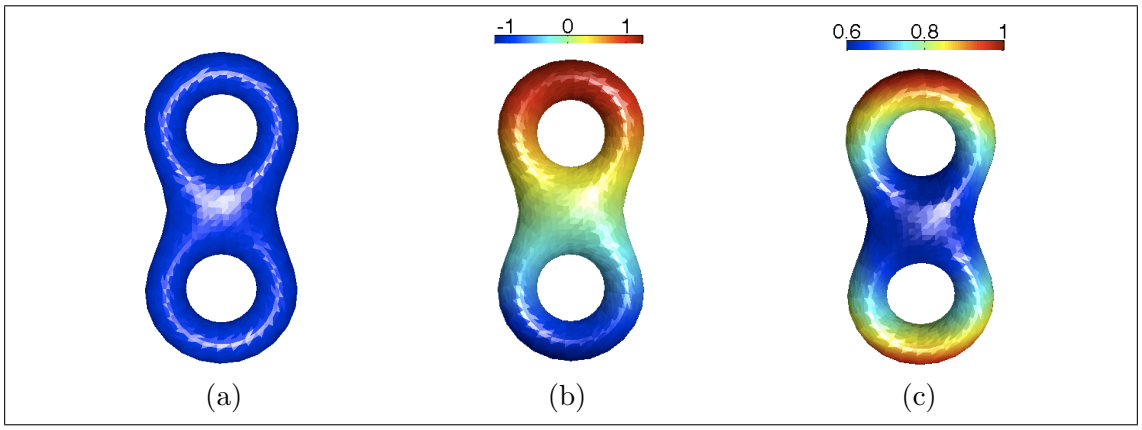


Figure 2.2: Illustration of different Morse functions: (a) Initial object. (b) height function on the surface of a double torus. (c) GGF function on the surface. (Best visualized in color)

surface in $\mathbb{R}^3$ will change as a result of its mere rotation. Since shape descriptors are, for a large number of applications, required to be invariant to similarity transforms[1], a choice of an appropriate Morse function is critical. In [1], Hilaga *et al.* define a Morse function for generic metrics on surfaces, that is invariant to isometric transforms. Specifically, this function is defined at every point $\mathbf{v}$ on $\mathcal{S}$ as the integral of the geodesic distance $\mathrm{d}(\mathbf{v}, \mathbf{p})$ from $\mathbf{v}$ to all other points $\mathbf{p}$ on $\mathcal{S}$,

$$f(\mathbf{v}) = \int_{\mathbf{p}\in\mathcal{S}} \mathrm{d}(\mathbf{v}, \mathbf{p}) \mathrm{d}\mathcal{S}. \tag{2.1}$$

2.3 Global Geodesic Function *GGF*

Hilaga *et al.*'s discrete approximation of Eq. (2.1), that we herein refer to as the first approximation $f_{appr}(\cdot)$, is defined as follows:

$$f_{appr}(\mathbf{v}) = \sum_i \mathrm{d}(\mathbf{v}, \mathbf{b}_i) \cdot \mathrm{area}(\mathbf{b}_i), \tag{2.2}$$

where $\{\mathbf{b}_i\}_{i=0,1,\ldots}$ is a finitely countable set of base vertices scattered on $\mathcal{S}$ and $\mathrm{area}(\mathbf{b}_i)$ is the area that $\mathbf{b}_i$ occupies, such that, $\sum_i \mathrm{area}(\mathbf{b}_i) \approx \mathrm{area}(\mathcal{S})$. Also, for an accurate result, Hilaga *et al.* emphasize the need for a mesh preparation through two operations: generation of short-cut edges based on a manually chosen threshold and a subdivision of the mesh. Let's consider the object "droplet" of Figure 2.3. and compute its integrated

[1]Similarity transforms include: translation, rotation, and scaling, or any combination thereof.

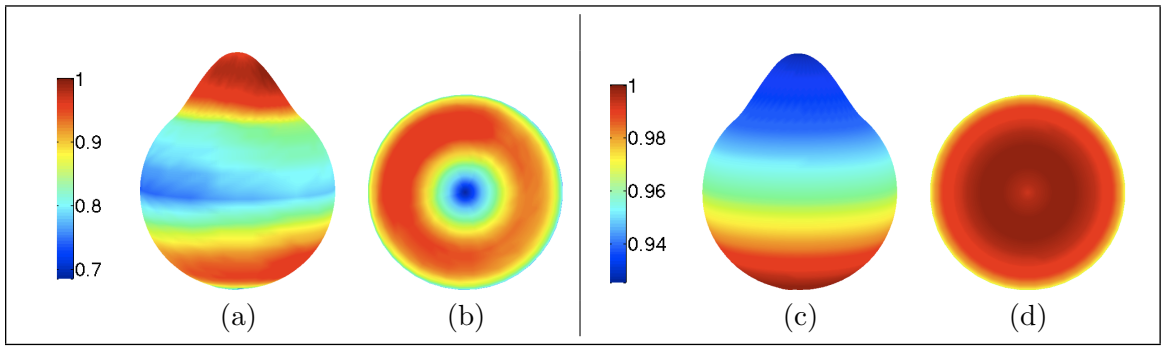


Figure 2.3: Comparison of the two approximations of the integrated geodesic Morse function. (a) and (b) are, respectively, the front and bottom views of the distribution of $f_{appr}(\cdot)$ on the object "droplet". (c) and (d) are, respectively, the front and bottom views of the distribution of $g(\cdot)$ on the object "droplet". (Best visualized in color)

geodesic function. Because of the perfect symmetry of this 3D shape with respect to the z axis, one should expect the function $f(\cdot)$ defined in Eq. (2.1) to have all its level sets exactly parallel to the xy plane, *i.e.*, horizontally constant. We find, however, that failure in achieving an appropriate prepossessing of the mesh, may drastically affect the distribution of $f_{appr}(\cdot)$, the first approximation of $f(\cdot)$ as defined in Eq. (2.2). In Figure 2.3, we see that the colors on (a) are not uniform as they should be and are on (c), hence our motivation in adopting a different approximation of $f(\cdot)$ as our Morse function, all the while keeping similar attractive properties; This includes its full invariance to isometric transformations, and a further improvement of the robustness to surface meshing and noise. We hence use the approximation $g(\cdot)$, that we first defined in [23], and referred to as the *Global Geodesic Function* (GGF), such that:

$$g(\mathbf{v}) = \frac{\sum_{\mathbf{p}\in\mathcal{S}} \mathrm{d}(\mathbf{v},\mathbf{p})}{\max_{\mathbf{q}\in\mathcal{S}} \left(\sum_{\mathbf{p}\in\mathcal{S}} \mathrm{d}(\mathbf{q},\mathbf{p})\right)}. \tag{2.3}$$

Unless specified otherwise, this is the GGF used throughout this thesis, and illustrated in Figure 2.4, and in Figure 2.2 (c), where the surface $\mathcal{S}$ is the previously seen double torus illustrated in Figure 2.2 (a). In addition to its independence of any reference point, an important property of this Morse function is its invariance to isometric transformations (because it is intrinsic), thus yielding a consistent and unique characterization of an object surface as illustrated in Figure 2.5. The independence of reference points is achieved by the geodesic integration procedure at each point. The normalization of the functional shown in

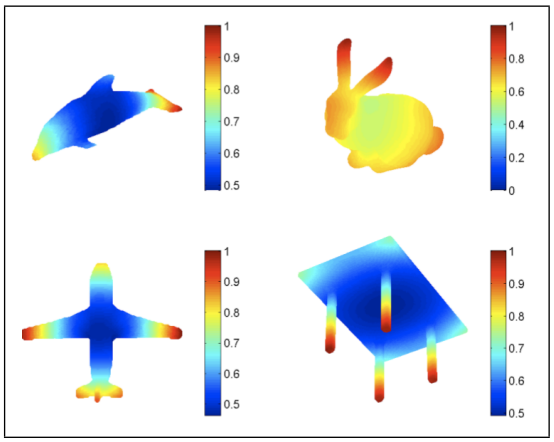


Figure 2.4: Examples of the global geodesic function on different 3D objects.

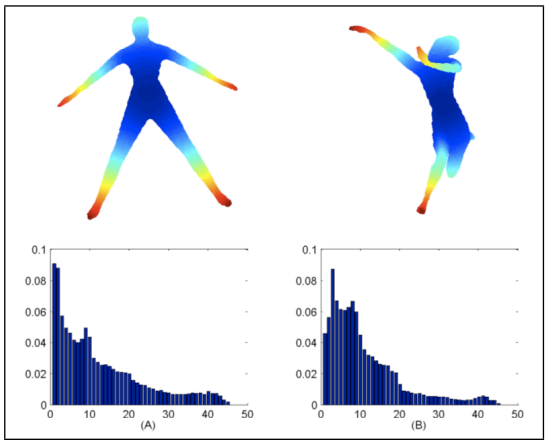


Figure 2.5: Invariance to isometric transformations: (A) Original pose and a corresponding PDF of its GGF (B) Same subject after deformation and noise addition. At the bottom the PDF of its GGF.

Eq. (2.3) ensures an invariance to scaling by making the range of $g(\cdot)$ coincide with $[g_0, 1]$, with $g_0 = \frac{\min_{\mathbf{v}\in\mathcal{S}} \sum_{\mathbf{p}\in\mathcal{S}} \mathbf{d}(\mathbf{v},\mathbf{p})}{\max_{\mathbf{v}\in\mathcal{S}} \sum_{\mathbf{p}\in\mathcal{S}} \mathbf{d}(\mathbf{v},\mathbf{p})}$ and $g_0 > 0$. In Figure 2.2 (c), we show the color coding map for $g(\cdot)$ used on all our objects. In practice, the GGF for each vertex is obtained by computing the latter's geodesic distance to all other vertices. This normalization also obviates the explicit computation of the surface element. This is efficiently realized with the well known *Dijkstra* algorithm whose complexity is $O(N^2 \log N)$ [24].

Chapter 3

Statistical Analysis

In this Chapter, we present an economical and fast classification algorithm for 3D objects. Classifying 3D shapes independently of any rotation, translation, scaling transformation and/or non-elastic deformation is of great importance in many applications including multimedia, communications, computer graphics and biometrics. While many successful methods exist, they are of relatively high complexity for the task of putting objects into groups. We propose a classification strategy that is far less complex and still very robust to pose and articulation changes. Our method is based on the GGF defined in Section 2.3. Each object class is defined by only four parameters obtained during a learning stage. We use these parameters in the decision of a class membership. Experimental results demonstrate low cost, efficiency, and robustness to resolution and data benchmarks of the proposed approach.

3.1 Introduction

We propose to define a 3D object classification technique where we characterize each class with four parameters obtained from the global geodesic shape function. The first three parameters consist of a variability measure and two characteristic resolutions. The characteristic resolutions indicate how to maximally and effectively reduce the overall computational complexity with little or no loss of information about a given object. These parameters are generally class specific and may, hence, be used as a first order classifier. A refined level of discrimination is achieved by introducing a threshold measure as a fourth

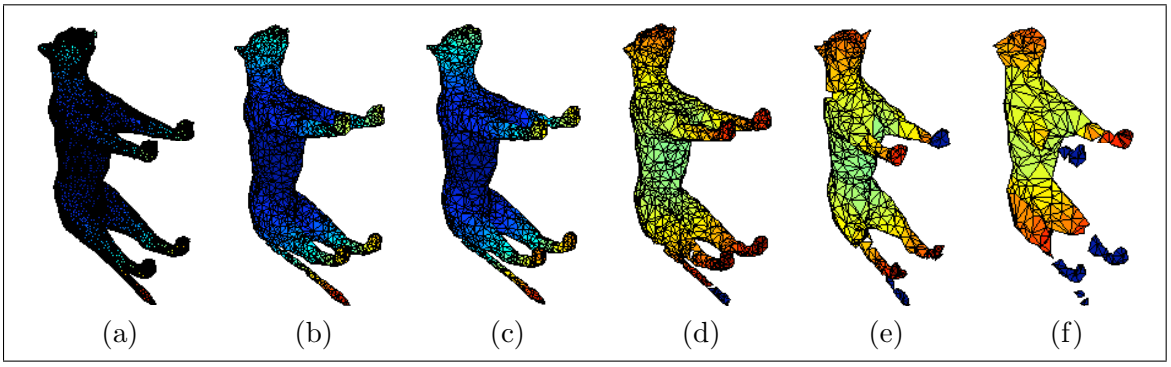


Figure 3.1: Illustration of the visual effect of the resolution reduction on the GGF of a 3D object. Resolution is decreasing from case (a) to (f). An abrupt change in the GGF occurs at (d).

parameter. To further improve on the discrimination power of these parameters, we propose a subsequent *zooming* comparison procedure to focus on distinctive features among objects, and, hence, provide a potential for recognition[1]. The remainder of this chapter is organized as follows. The next section gives the formulation of the problem along with a presentation of the key tools that are used. The detailed algorithm is explained in Section 3.3. The substeps are illustrated as well. In Section 3.4, experimental results that aimed to demonstrate the robustness and the efficiency of the method are given. Finally, Section 3.5 summarizes the present Chapter, and proposes future projects as a continuity to the present statistical analysis.

3.2 Problem formulation

Given a set of N classes of objects $\{C_1, C_2, \ldots, C_N\}$, our goal is to decide on a class membership of an object O. To efficiently carry through such a task, we define a set of feature parameters for an object and to construct a systematic procedure for 2D surface comparison.

3.2.1 Resolution evaluation

As stated earlier, the computational complexity is a predominant factor in 3D object processing. We have experimentally also established that a uniform subdivision of

[1]Recognition here means a more refined level of classification

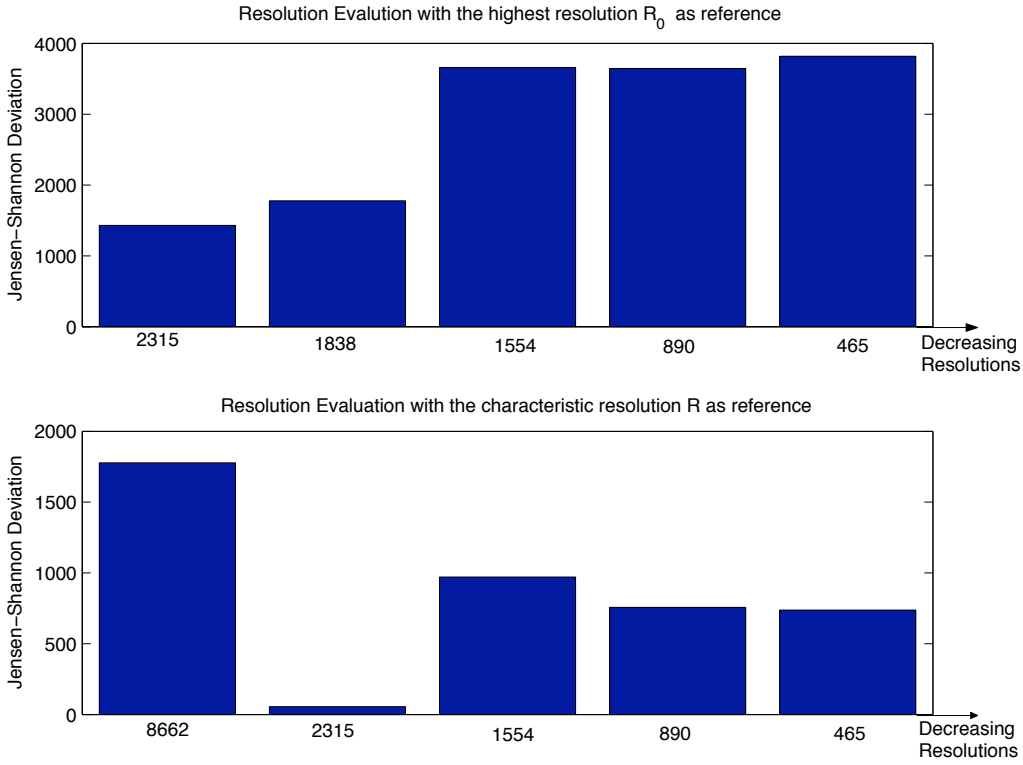


Figure 3.2: Resolution evaluation for a dog from Princeton's benchmark.

a mesh preserves a near invariance of a histogram of the GGF, for different resolutions of an object relative to that of its finest resolution. A sharp and sudden change in the overall distribution of the GGF as illustrated in Figure 3.1, takes place and the visual perception is perturbed when the resolution is too low to exhibit the desired invariance of the distribution. The reference distribution of any 3D object in a class of interest is that of its GGF computed at the finest resolution. This distribution is subsequently compared to those obtained by progressively decreasing resolutions. To this end, we use the *Jensen-Shannon Divergence* (JSD) as a distance function between two distributions [3, 25]. For K different resolutions, $R_0 > R_1 > \ldots > R_{K-1}$, of an object and their K corresponding PDFs P_{R_i}, $i = 0, 1, \ldots, K-1$, the JSD distance is defined as follows

$$JSD(P_{R_1}, P_{R_2}) = H(\sum_{l=1,2} \frac{1}{2} P_{R_l}) - \sum_{l=1,2} \frac{1}{2} H(P_{R_l}), \tag{3.1}$$

where H is the *Jensen-Shannon entropy* defined by:

$$H\big(P_{R_i}\big) = -\sum_{l=1}^{l=L} P_{R_i}(l) \log P_{R_i}(l), \tag{3.2}$$

with $L < R_i$ and $P_{R_i}(l)$, $l = 1, 2, \ldots, L$, being the elements of the discrete PDF vector P_{R_i}. Equivalently, we may write

$$2 \cdot JSD(P_{R_i}, P_{R_j}) = 2H(\frac{1}{2}P_{R_1} + \frac{1}{2}P_{R_2}) - H\big(P_{R_1}\big) - H(P_{R_2}\big). \tag{3.3}$$

With the knowledge that all 3D objects will manifest a trend similar to that illustrated in Figure 3.1, we proceed to determine the smallest (characteristic) resolution $\Re$ of any given object O. This resolution $\Re$ will also have a distribution of its geodesic function sufficiently close to that of the finest resolution. For a systematic comparison of the PDFs at various resolutions, we first define

$$X(R_i) = (JSD(P_{R_0}, P_{R_i}), JSD(P_{R_0}, P_{R_{i+1}})), \qquad i = 1, 2, \ldots, K-1. \tag{3.4}$$

We subsequently proceed to define the characteristic resolution $\Re$ as

$$\Re = \text{argmax}\Big(\text{variance}\left(X(R_i)\right)\Big), \qquad i = 1, 2, \ldots, K-1. \tag{3.5}$$

Note that this resolution evaluation is carried out during the learning step in order to alleviate the online computational burden. To reduce the computational load during testing, we try to avoid, if at all possible, calling upon the first resolution representation, and use instead the characteristic resolution as the reference. To that end, the characteristic resolution $\Re$ and its two neighbors along with the corresponding JSD distances represent the reference as detailed in Figure 3.2.

3.3 Proposed algorithm

With the tools described in Section 3.2 in hand, we propose a classification strategy that heavily relies on a training procedure during which a class parametrization is achieved in order to yield as a result, an object class membership. This result is based on an elimination principle which takes advantage of the designed parametrization. To further refine discrimination among objects, we introduce a post processing *zoom-in* procedure on PDF's comparison and to focus on more detailed dissimilarities among objects.

3.3.1 Class resolutions

As noted in Section 3.2.1, we evaluate a characteristic resolution $\Re^{(i)}$ as defined in Eq. (3.5) for each training object O_i, $i \in \Omega = \{1, 2, \ldots, M\}$, from a class C. For various objects of a class, we obtain a *class characteristic resolution* as being the maximum of all the characteristic resolutions within that class,

$$\Re_r^C = \max_{i=1,\ldots,M} \{\Re^{(i)}\}. \tag{3.6}$$

To ensure generalizability to all training samples, we define a second discrimination parameter as being the transition resolution or the minimum right neighbor, *i.e.*, the right neighbor of the minimal characteristic resolution, referred to as $\Re_t^C$. Recall that the measure of variance of JSD at $\Re_r^C$ and $\Re_t^C$, provides the third parameter; thus, we end up with three parameters.

3.3.2 Thresholding

Upon establishing $\Re_r^C$, any comparison within a class is carried out at $\Re_r^C$ by computing a pairwise JSD among all training objects in C. The highest distance obtained from the latest JSDs is the sup distance that no object within a class C would never exceed.The threshold τ is hence defined as

$$\tau = \max_{i,j \in \Omega} \{JSD(P^{(i)}, P^{(j)})\}, \tag{3.7}$$

where $P^{(i)}$, $i \in \Omega$, is the lenient notation for $P^{(i)}_{\Re_r^C}$; the PDF of the GGF associated to the object O_i from the class C and computed at the class characteristic resolution $\Re_r^C$.
When the order of $\Re_{\mathrm{r}}^C$ is important (> 1000), the computed PDF of the GGF is dependent of the sampling rate, *i.e.*, the number of bins used to compute the histogram of the GGF. Then, the threshold τ, which was a scalar in Eq. (3.7), becomes a vector whose elements are thresholds computed at different sampling rates.

3.3.3 Zoom-in operation

To detect dissimilarities between two objects O_j and O_k, with $j \neq k$, the corresponding PDFs are compared over smaller regions of their support, which we focus on by a zoom-in operation. To that end, denote by L the maximal support of the PDFs of the

GGFs for the two objects. We proceed by looking at fixed intervals L_i, with $L_i < L$, over which we determine normalized PDFs $P^{(j)}(L_i)$ and $P^{(k)}(L_i)$ for O_j and O_k respectively. The JSD between the two objects is now a function of the interval L_i and is defined as

$$f\left(L_i\right) = JSD(P^{(j)}(L_i), P^{(k)}(L_i)). \tag{3.8}$$

Establishing a critical region over which two PDFs are most similar, namely,

$$L_0 = \operatorname{argmin}\left(f(L_i)\right). \tag{3.9}$$

Our interval of interest is redefined as

$$\lambda = L - L_0. \tag{3.10}$$

The interval λ contains hence the largest diversity between O_j and O_k. A reasonable dissimilarity measure $\mathcal{D}_\lambda$ [2] is the ratio between the area on the object, either O_j or O_k, corresponding to λ and the total area of the same object $\mathcal{S}^{(j)}$ or $\mathcal{S}^{(k)}$ respectively. Recall that $\lambda \in \{[a,b] \mid 0 < a \leq b \leq\ 1\}$, and define, for O_j, for instance,

$$\mathcal{S}^{(j)}(\lambda) = \Big\{ \sum_l \delta\mathcal{S}_l \mid g(p_l) \in \lambda \Big\}. \tag{3.11}$$

Note that

$$\mathcal{S}^{(j)} = \mathcal{S}^{(j)}(]0,1]); \tag{3.12}$$

therefore, $\mathcal{D}_\lambda$ is defined as

$$\mathcal{D}_\lambda(O_j, O_k) = \frac{1}{2}\Big(\frac{\mathcal{S}^{(j)}(\lambda)}{\mathcal{S}^{(j)}} + \frac{\mathcal{S}^{(k)}(\lambda)}{\mathcal{S}^{(k)}}\Big). \tag{3.13}$$

In a more particular formulation of the problem, the zoom-in operation introduces a new distance measure $\mathcal{D}_\lambda^C$ between an object O_j and a class C by taking the second object O_k as the best representative of the class C in the sense of the nearest neighbor relatively to O_j and where JSD is the considered distance. Hence, we set $O_k = O^C$, where O^C is the object from class C of index $l \in \Omega$ that corresponds to $\min_{\substack{l\in\Omega \\ j\notin\Omega}}\{JSD(P^{(j)}, P^{(l)})\}$. It follows that

$$\mathcal{D}_\lambda^C(O_j) = \mathcal{D}_\lambda(O_j, O^C). \tag{3.14}$$

The two measures $\mathcal{D}_\lambda$ and $\mathcal{D}_\lambda^C$ are dependent of the fixed length of the interval L_i, i.e, dependent of the length of λ. Indeed, the smaller the length of λ is, the more emphasis is put into details.

[2] The subscript λ indicates that the measure is achieved for the fixed interval length of λ.

3.3.4 Algorithm

The distinct steps of the algorithm are sketched in Figure 3.4., and summarized next.

1. Define N object classes $\{C_1, C_2, \ldots, C_N\}$.
2. In the learning phase, associate to each class C_i a corresponding $\Re_r^{C_i}$, $\Re_t^{C_i}$, the class characteristic variance $\sigma_{C_i}^2$ and the threshold τ_{C_i}.
3. Sort all classes in an increasing resolution order $\Re_r^{C_i}$.
4. Construct super-classes by merging classes sharing the same parameters $\Re_r^{C_i}$, $\Re_t^{C_i}$ and $\sigma_{C_i}^2$.
5. Start from the lowest resolution $\Re_{\mathrm{r}}^{C_1}$ and set $l = 1$.
6. Compute the GGF of O at $\Re_{\mathrm{r}}^{C_l}$ and get its resolution parameters.
7. Compare the resolution parameters of O with those of C_l. If similarity is established, *i.e.*, decision is 1;
   - Termination of search if C_l is a class.

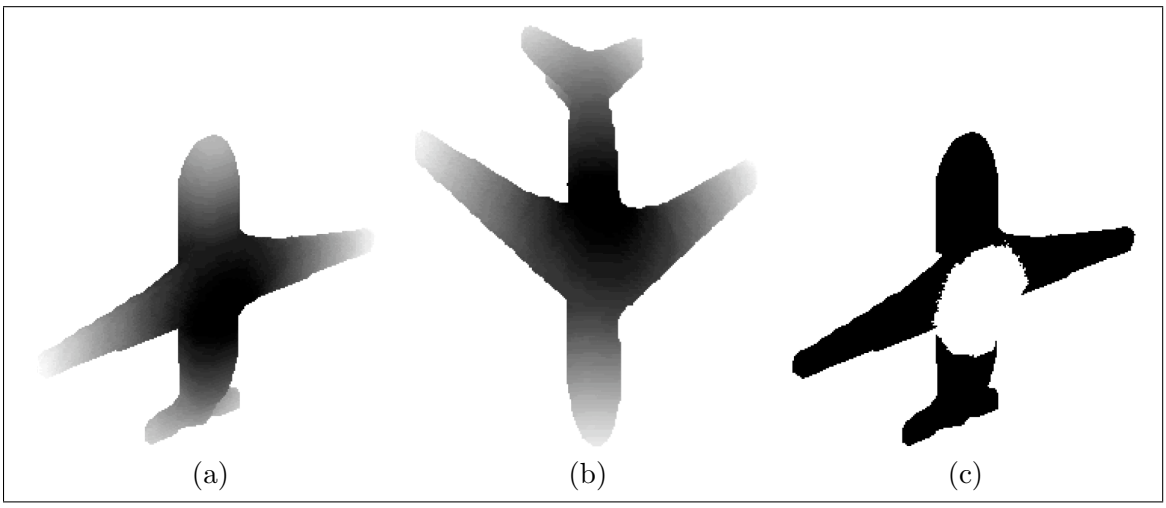


Figure 3.3: Detection of the area of dissimilarity between two airplanes using the GGF: Zoom-in operation.

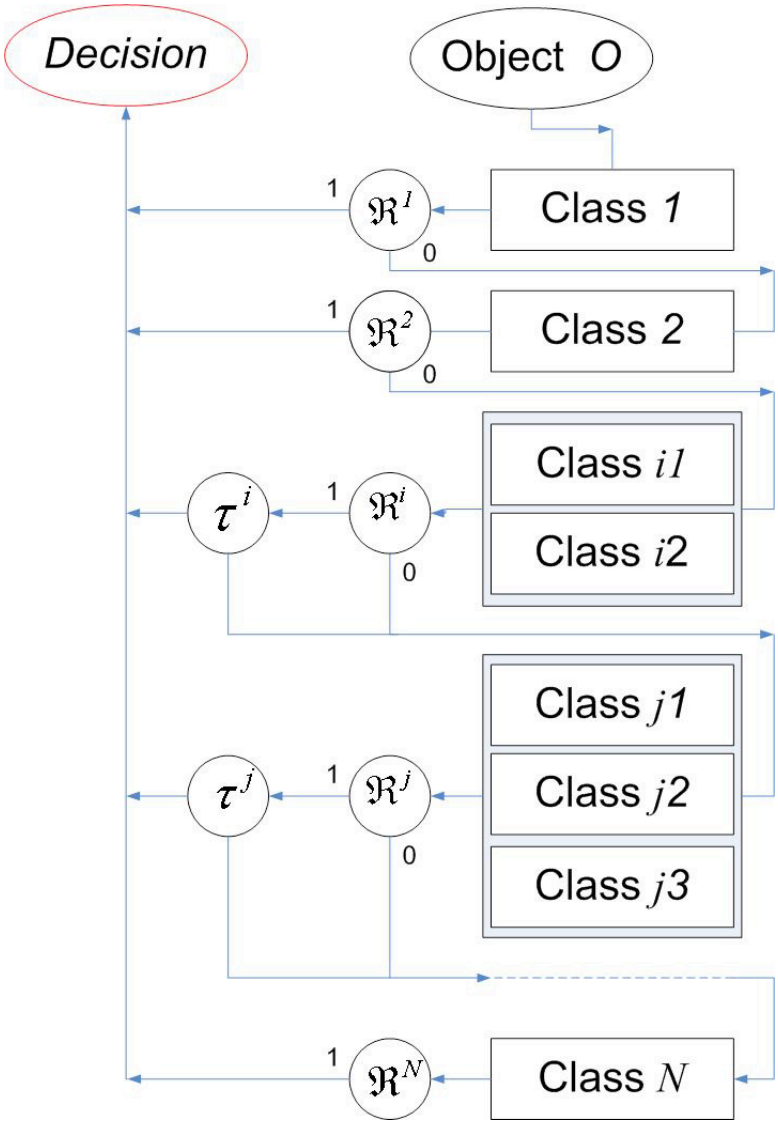


Figure 3.4: Algorithm of the discrete classification decision for 3D objects. Resolutions $\Re_{\mathrm{r}}^{C_i}$ are simply denoted $\Re^{C_i}$ in order not to overload the figure.

- Apply thresholding if C_l is a super-class.

Otherwise the decision is 0 and $l = l + 1$.

8. Go to 6.

9. Repeat operations until a decision 1 is reached.

3.4 Experimental results

3.4.1 Independence of sources of data

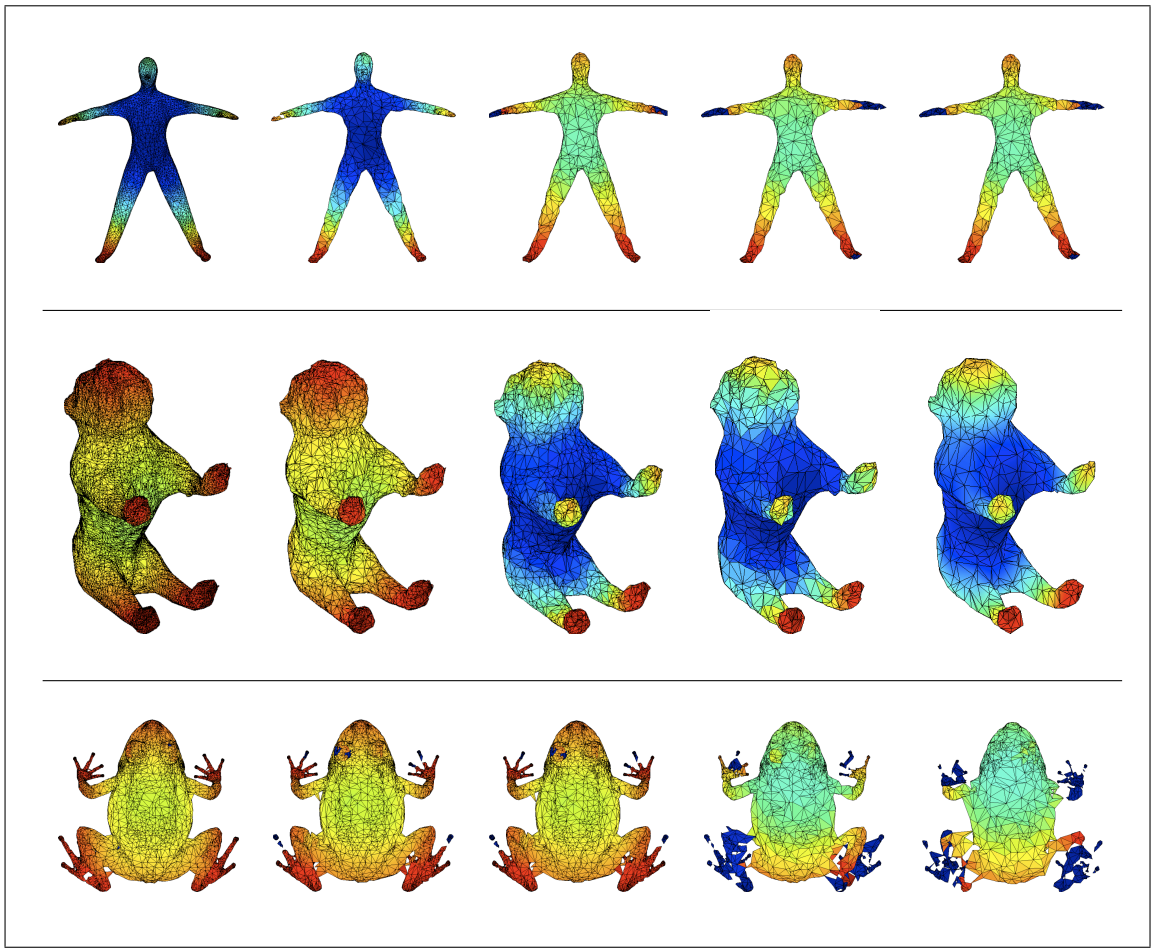


Figure 3.5: Illustration of the characteristic resolution extraction (Best visualized in color).

An important characteristic of classification algorithm is its consistent performance

Table 3.1: Characterization parameters for the class of felines.

	Parametrization from learning	Parametrization from testing	Absolute error
$\Re_r$	1942	1897	2.32 %
$\Re_t$	1501	1420	5.40%
σ^2	418000	386000	7.66%

independently of what data-base or measurement system the data originates from. So as long as the format is in readable form, we have shown that our proposed approach enjoys such a property. We made extensive use of Princeton's benchmark [16] for all our learning and testing. We have subsequently tested data from INRIA's benchmark to validate our claim. Decision tolerance is set to 10% as a maximum absolute error on each parameter. The corresponding overall recognition performance, *i.e.*, number of correct decisions overall the testing experiments, is 93.88%. To verify and illustrate the importance of the invariance properties provided by the GGF, we applied isometric transforms (rigid and non rigid) to the same dataset. The results of this new experiment show a classification performance of 97%. The parameters obtained for the class of felines, where a good representative is shown in Figure 3.1., are gathered in the following Table 3.1: We show, in Figure 3.5, additional examples of detecting the characteristic resolution for different classes of 3D models. We provide the comparison of PDFs with the JSD in the rightmost column. We, again, confirm this trend in the progression of the distribution of the GGF; indeed, for the same color mapping of the GGF on the surface of shapes, we clearly see a sudden change in the coloration of the model. This abrupt transition, corresponds each time to going below the characteristic resolution of a model/class.

3.4.2 Threshold implementation

Recall the thresholding procedure is introduced as a complementary and refined discrimination measure between two or more classes. For two closely related classes (e.g. pigs and dogs) for example, a resolution-based comparison fails while the threshold-enhanced immediately eliminates the confusion. In Figure 3.6., the "pig" object is eliminated from the class of dogs because the maximum pairwise comparison of its GGF distribution exceeds, over a region, the threshold previously computed for the class of dogs.

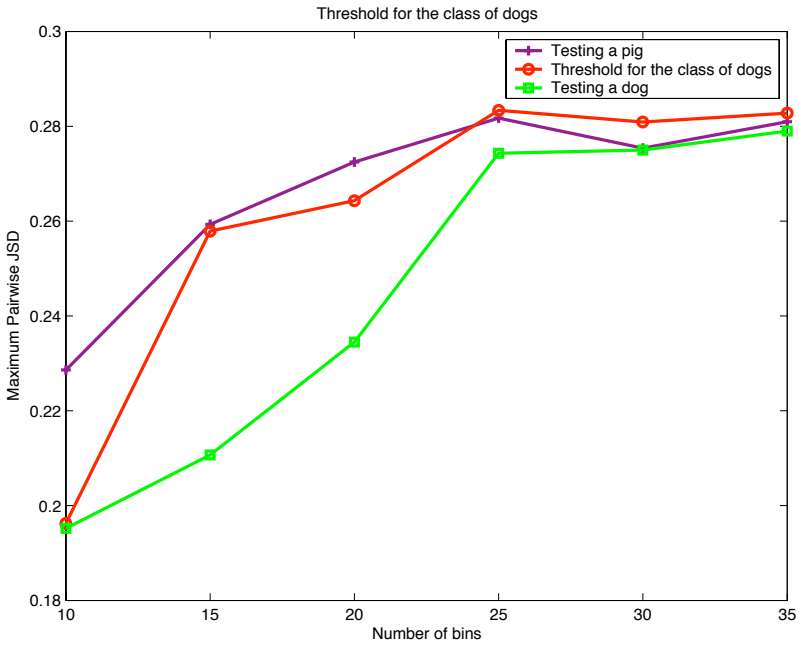


Figure 3.6: Illustration of the classification level: Thresholding.

3.4.3 Zoom-in example

The GGF is sufficiently powerful to detect and measure dissimilarities even for relatively close 3D objects. In Figure 3.3., at $\lambda = 1/10$, we look at the area of highest difference between two airplanes. In (a), a commercial airplane is represented. In (b), a more recent model of commercial airplane is shown. The zoom-in technique highlights in (c) (in white) the λ-surface of difference. This difference restricted to the fuselage, is intuitively pleasing as it confirms our visual interpretation of a fatter first airplane.

3.5 Conclusion

In this Chapter, we presented a new 3D object classification strategy based on four parameters. We have shown that it generally provides a quick discrete decision on object comparison; indeed, once the learning stage completed, the classification becomes a simple comparison of numbers with a given tolerance. The technique is based on the distribution of the GGF which, in turn, provides very interesting properties such as its invariance to

all isometric transformations; moreover, a new and precise dissimilarity measure has been introduced. More work is currently directed on the exploration of the theoretical bases of the characteristic resolution and its relation to topological changes of surfaces.

Chapter 4

3D Shape Partitioning

To simplify the matching and recognition of 3D objects, we propose to decompose a complex 3D shape into simpler primitive parts. Our proposed partitioning of objects relies on their topological Reeb graphs. Taking advantage of the properties of Morse theory, we detect the critical points of the GGF. These points define the levels at which the segmentation happens. To preserve the geometry of objects, we choose to use level curves instead of intervals. To proceed with object matching, we propose a kernel-based technique to register Reeb graphs. This optimal positioning of two Reeb graphs prepares for a pairwise comparison of the geometry of their primitives.

4.1 Introduction

It is widely believed that perception utilizes crucial topological characteristics of objects. Recent neuro-imaging studies, together with behavioral studies, provide strong evidence supporting the notion that topological properties (such as a genus, or connectedness) are primitives of visual representation in humans. To address the potentially complex topology of a 3D object (*e.g.,* holes, breaks in objects), we call upon Morse theory [19] to develop a robust and systematic technique which unfolds the topology of an object, by studying the critical points, *i.e.*, maximum, minimum or saddle points, of a Morse function defined on its surface [22]; the GGF in our case (Section 2.3). These critical points along with the regular points, fully capture the topology of a 3D object by a graph, referred to as a Reeb graph. Graph theory has enjoyed a wide popularity in problems of classification and

recognition [26]. Reeb graphs have thus proven to be instrumental in globally describing three-dimensional shapes and in representing their topology [37]. Specifically, the nodes on such graphs represent critical points of a Morse function defined on a 3D shape/object surface, while the edges capture the topologically homogeneous parts, which correspond to "primitive" shapes.

Comparing topological graphs/skeletons for the purpose of matching objects has been extensively studied. Our interest is, however, to explore these graphs for more refined comparisons. We thus propose to use the Reeb graph of an object to decompose it into simple shapes or primitives. Comparing the geometry of these shapes will therefore become an easier task as long as we ensure a good correspondence between the primitives of two objects to compare.

The remainder of this Chapter is organized as follows: We describe, in Section 4.2, our original technique for the extraction of Reeb graphs using level curves of the GGF. In Section 4.2.2, we present our partitioning technique, and propose, in Section 4.3, a framework for object comparison based on matching their graphical representation and the comparison of their primitives. Finally, in Section 4.4, we illustrate the results of our proposed technique.

4.2 Extraction of Reeb graphs

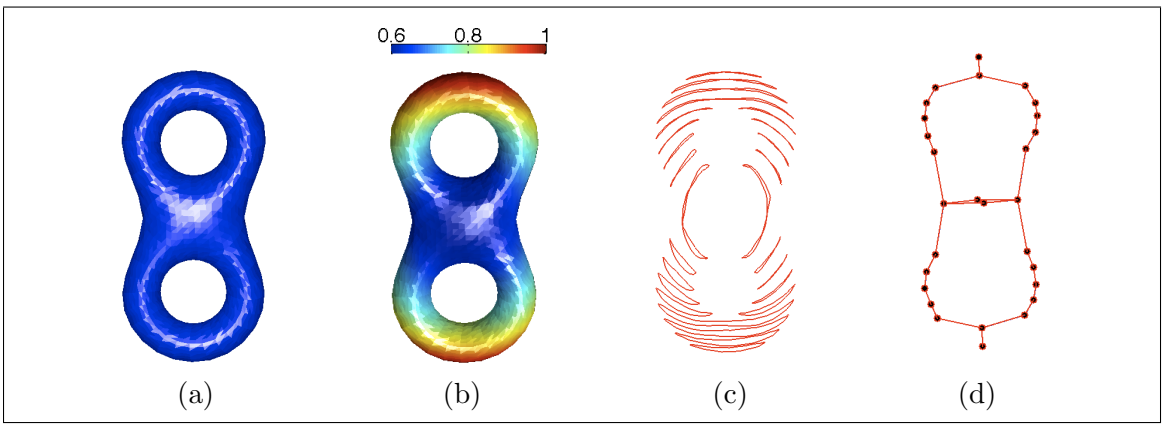


Figure 4.1: Illustration of Reeb graph extraction: (a) Initial object. (b) GGF function on the surface. (c) Iso-geodesic curves. (d) Extracted topological Reeb graph. (Best visualized in color)

With the GGF $g(\cdot)$ in hand (Section 2.3), we construct a topological Reeb graph for an object surface $\mathcal{S}$ [1, 2]. Mathematically, a Reeb graph is a quotient space $\mathcal{S}/\!\sim$, where the equivalence relation is given by $\mathbf{p} \sim \mathbf{q}$ if and only if $g(\mathbf{p}) = g(\mathbf{q})$ with $\mathbf{p}$ and $\mathbf{q}$ being two points on $\mathcal{S}$ and belonging to the same connected component of $g^{-1}(g(\mathbf{p}))$. Practically, we sample the surface $\mathcal{S}$ of a 3D object by way of level sets of a Morse function, and in our case thereof a GGF $g(\cdot)$. As illustrated in Figure 4.1 (c), the level sets of $g(\cdot)$ are closed curves, which we refer to as *iso-geodesic* curves. We present the resulting Reeb graph in Figure 4.1 (d), where each closed iso-geodesic curve is replaced by a node colored in black.

4.2.1 Level set characterization of a surface

Generically, we can reconstruct a 2-dimensional surface $\mathcal{S}$ from the set of 1-dimensional iso-curves of the GGF. A surface $\mathcal{S}$ is then a disjoint union of all iso-geodesic sets $\mathcal{A}(t)$ for $t \in [g_0, 1] \subset \mathbb{R}^+$, where $\mathcal{A}(t) = \{\mathbf{v} \in \mathcal{S} \mid g(\mathbf{v}) = t\}$, and g_0 is the minimal value of $g(\cdot)$.

$$\mathcal{S} = \bigcup_{t=g_0}^{t=1} \mathcal{A}(t), \quad \text{and} \quad \mathcal{A}(t_1) \cap \mathcal{A}(t_2) = \varnothing \quad \text{if} \quad t_1 \neq t_2. \tag{4.1}$$

For smooth and compact objects, an iso-geodesic set is the union of closed curves (*i.e.*, iso-geodesic curves). The number of these distinct curves at the same geodesic level is the cardinality $Car(t)$ of the corresponding iso-geodesic set $\mathcal{A}(t)$. In practice and on triangulated meshes, the extraction of an iso-geodesic set for a given level, say λ, entails first finding all the faces that cover this set of curves. Each iso-geodesic curve is included in a connected set of covering faces so that the resulting curve is ensured to be closed. A covering face whose vertices are $\mathbf{v}_1, \mathbf{v}_2$ and $\mathbf{v}_3$ with $g(\mathbf{v}_1) < g(\mathbf{v}_2) < g(\mathbf{v}_3)$ falls under one of the following properties:

1. $\lambda \in [g(\mathbf{v}_1), g(\mathbf{v}_2)]$
   or
2. $\lambda \in [g(\mathbf{v}_2), g(\mathbf{v}_3)]$

Two points $\mathbf{p}$ and $\mathbf{q}$ belonging to the approximated iso-geodesic set $\mathcal{A}(\lambda)$ are defined on two edges of the covering face (Figure 4.2). If Property 1 is verified, then:

$$\overrightarrow{\mathbf{v}_1\mathbf{p}} = \frac{\lambda - g(\mathbf{v}_1)}{g(\mathbf{v}_2) - g(\mathbf{v}_1)} \cdot \overrightarrow{\mathbf{v}_1\mathbf{v}_2},$$

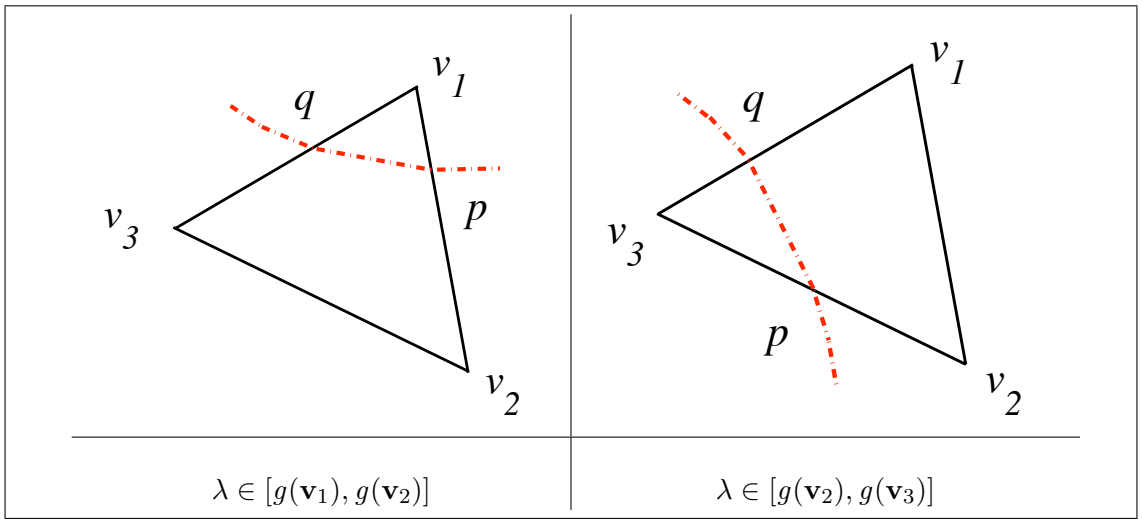


Figure 4.2: Covering face and iso-geodesic curve interpolation.

and

$$\overrightarrow{\mathbf{v}_1\mathbf{q}} = \frac{\lambda - g(\mathbf{v}_1)}{g(\mathbf{v}_3) - g(\mathbf{v}_1)} \cdot \overrightarrow{\mathbf{v}_1\mathbf{v}_3}.$$

If instead Property 2 is verified, then:

$$\overrightarrow{\mathbf{v}_3\mathbf{p}} = \frac{g(\mathbf{v}_3) - \lambda}{g(\mathbf{v}_3) - g(\mathbf{v}_2)} \cdot \overrightarrow{\mathbf{v}_3\mathbf{v}_2},$$

and

$$\overrightarrow{\mathbf{v}_3\mathbf{q}} = \frac{g(\mathbf{v}_3) - \lambda}{g(\mathbf{v}_3) - g(\mathbf{v}_1)} \cdot \overrightarrow{\mathbf{v}_3\mathbf{v}_1}.$$

Once we obtain all the possible couples $(\mathbf{p}, \mathbf{q})$, we smoothly interpolate, with a B-spline for instance, the so obtained sample of points to get the iso-geodesic curve at the level λ of the GGF; moreover, the subsurface supported by one edge has exactly one iso-geodesic curve for every value between a and b; the extremal values of the GGF along the edge. We call such subsurface $\mathcal{M}$ a *mono-cardinality* subsurface. We may view $\mathcal{M}$ as follows:

$$\begin{aligned} \mathcal{M} \;=\; \bigcup_{t\in[a,b]} C(t), \quad \text{with} \quad g_0 \le a < b \le 1, \\ \text{and} \quad C(t_1) \cap C(t_2) = \varnothing \quad \text{if} \quad t_1 \neq t_2, \end{aligned} \tag{4.2}$$

where $C(t)$ is just one iso-geodesic curve at the level t of the GGF. In the example of

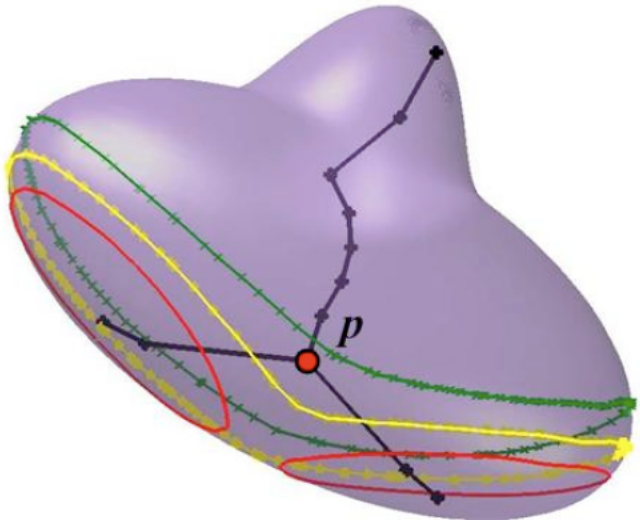


Figure 4.3: Illustration of topological change of iso-geodesic curves. The point p is the bifurcation point at which this change occurs.

Figure 4.3, the illustrated iso-geodesic sets correspond (from top to bottom) to the values $0.8, 0.85$ and 0.9 of the GGF, respectively. The corresponding cardinalities are $Car(0.80) = Car(0.85) = 1$, and $Car(0.90) = 2$, *i.e.*, the first two sets consist of one closed curve, whereas the third set includes two distinct closed curves.

Tracking the cardinalities of iso-geodesic sets helps us in constructing an associated Reeb graph. The continuous and smooth evolution of the iso-geodesic curves on a surface captures its topological and geometrical description. Any change in the topology of the iso-geodesic sets, *i.e.*, any smooth change in their cardinalities, hence, determines a bifurcation point as shown in Figure 4.3. This transition is translated on the object's topological graph by a node of order $\eta > 2$ that introduces η edges. We preserve all such nodes and those corresponding to critical points of the GGF along with their edges to only capture the important topological information. The resulting special skeleton, that we refer to as a *reduced 3D Reeb graph*, offers a nice structure on which the geometry of an object may easily be superposed. Thus, the geometric modeling of a complex shape is now reduced to separately modeling a partial shape along each edge of the reduced Reeb graph. We next discuss the details of modeling the geometric shape along an edge.

During the graph extraction operation, we replace each iso-geodesic curve C_i by one node N_i which is the arithmetic mean of the curve; hence, we end up with a point cloud in 3 dimensions. In order to systematically get a final topological representation of the 3D object, we define a canonical relationship $\star$ between every two nodes N_i and N_j such that:

$$N_i \star N_j = \begin{cases} 1 & \text{if } N_i \text{ and } N_j \text{ are connected,} \\ 0 & \text{, otherwise.} \end{cases}$$

To that end, we give the following postulates:

- Two nodes representing the same iso-geodesic level are disconnected. We write:

$$i = j \Longrightarrow N_i \star N_j = 0. \tag{4.2}$$

- Two nodes separated by nodes at intermediate levels are disconnected, *i.e.*,

$$j \neq i \pm l \Longrightarrow N_i \star N_j = 0, \tag{4.2}$$

  where l is the sampling step.

Based on these postulates, we constrain our search on nodes representing two different but consecutive iso-geodesic levels.

Theorem 1 (Path connectivity) *Nodes representing two consecutive iso-geodesic curves are linked (connected) if and only if there exists a continuous path joining the two curves and lying on the section limited by the same iso-geodesic curves.*

4.2.2 Graph connectivity

To partition a 3D object we need to first accurately connect all the nodes N_i. To that end, we start by defining *orthogonal curves*.

Definition 2 (Orthogonal curves) *An orthogonal curve on a surface $\mathcal{S}$ passing through a point $\boldsymbol{p}$ is the curve of minimal length linking the iso-geodesic curve containing $\boldsymbol{p}$ to another iso-geodesic curve.*

We note that the vector field generated by the gradient of the GGF gives integral curves orthogonal to the iso-geodesic curves. So, by definition, an orthogonal curve takes a point $\mathbf{p}$

on a surface $\mathcal{S}$ and geodesically projects it on another iso-geodesic curve. By considering an infinitesimal patch around a point $\mathbf{p}$ from $\mathcal{S}$, we approximate the patch by a disk on which the iso-geodesic curve C becomes a segment passing through $\mathbf{p}$. This segment represents the direction of zero variation of the GGF. We find that the projection of $\mathbf{p}$ on the next iso-geodesic segment C' follows the perpendicular to C on $\mathbf{p}$. Under the assumption that all points are uniformly distributed on the surface and since the iso-geodesic curve represents the direction of zero variation of the GGF, we conclude, by duality, that the orthogonal projection of $\mathbf{p}$ is equivalent to finding the direction e of highest variation of the GGF; hence, we construct an orthogonal curve by progressively tracking, at a point level, the direction of the highest variation of the GGF. We determine the direction e that maximizes the directional derivative of g at a point $\mathbf{p}$; thus, we extract an orthogonal curve by progressively estimating a new direction $\hat{e}$ starting at each new iso-geodesic level[1]. We may write,

$$\begin{aligned} \hat{e} &= \arg\max \big(Dg \cdot e(\mathbf{p})\big) \\ &= \arg\max(\lim_{t\to 0} \frac{g(\mathbf{p}+te) - g(\mathbf{p})}{t}). \end{aligned} \tag{4.3}$$

Starting from a point with the lowest value of the GGF, and by progressively finding $\hat{e}$, we construct an *orthogonal curve* with respect to iso-geodesic curves. By construction, iso-geodesic curves are transversal to an orthogonal curve. The orthogonal curve is the continuous path that we were after in Section 4.2. Its existence between two consecutive GGF levels implies the existence of an edge connecting the nodes representing these levels.

The subsurface supported by one edge has exactly one iso-geodesic curve for every value between a and b; the extremal values of the GGF along the edge. We call such subsurface $\mathcal{M}$ a *mono-cardinality* subsurface. We may view $\mathcal{M}$ as follows:

$$\begin{aligned} \mathcal{M} = \bigcup_{t\in[a,b]} C(t), \quad \text{with} \quad g_0 \le a < b \le 1, \\ \text{and} \quad C(t_1) \cap C(t_2) = \varnothing \quad \text{if} \quad t_1 \neq t_2, \end{aligned} \tag{4.4}$$

where $C(t)$ is just one iso-geodesic curve at the level t of the GGF. In the example of Figure 4.3, the illustrated iso-geodesic sets correspond (from top to bottom) to the values $0.8, 0.85$ and 0.9 of the GGF, respectively. The corresponding cardinalities are $Car(0.80) =$

[1]We note that we only need a starting point to extract an orthogonal curve.

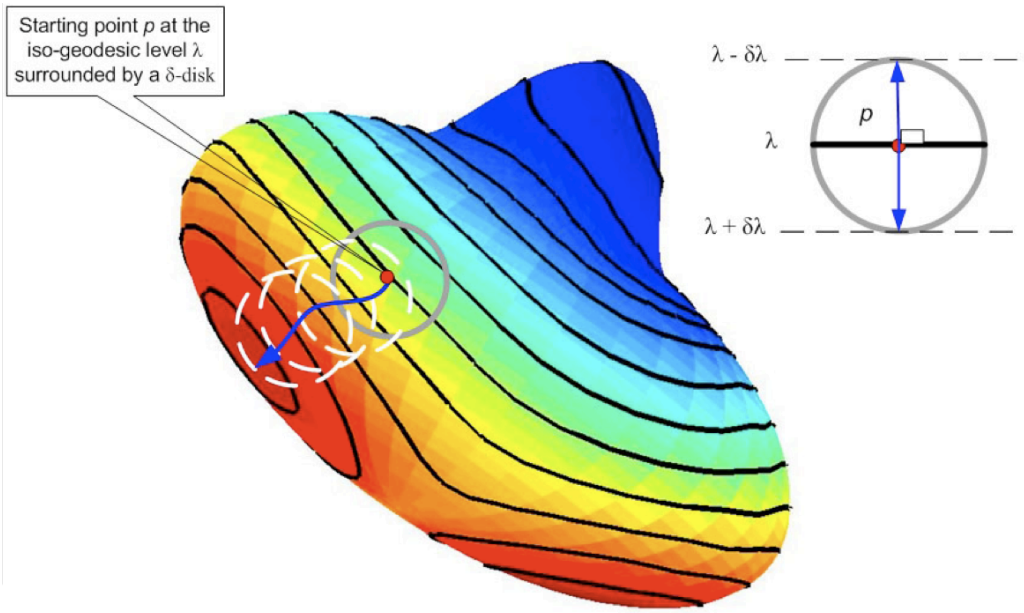


Figure 4.4: Orthogonal curve extraction using infinitesimal patches around each point at increasing values of the GGF.

$Car(0.85) = 1$, and $Car(0.90) = 2$, *i.e.*, the first two sets consist of one closed curve, whereas the third set includes two distinct closed curves.

Tracking the cardinalities of iso-geodesic sets helps us in constructing an associated Reeb graph. The continuous and smooth evolution of the iso-geodesic curves on a surface captures its topological and geometrical description. Any change in the topology of the iso-geodesic sets, *i.e.*, any smooth change in their cardinalities, hence, determines a bifurcation point as shown in Figure 4.3. This transition is translated on the object's topological graph by a node of order $\eta > 2$ that introduces η edges. We preserve all such nodes and those corresponding to critical points of the GGF along with their edges to only capture the important topological information. The resulting special skeleton, that we refer to as a *reduced 3D Reeb graph*, is the main support for a meaningful segmentation of a 3D object. Indeed, the only remaining task is to cluster all the points on the surface $\mathcal{S}$ that are delimited by two iso-geodesic curves represented by two connected nodes from the reduced graph.

4.2.3 Efficient sampling

A very redundant and important problem that is stressed when building Reeb graphs is how to solve this blind setting and define the effective sampling rate K that ensures the extraction of, and only of, the information needed to represent and reconstruct a given shape [1, 28]. The sampling rate K is usually empirically chosen, and the resulting sampling step is:

$$l = \frac{\max_{\mathbf{v}\in\mathcal{S}}\left(g(\mathbf{v})\right) - \min_{\mathbf{v}\in\mathcal{S}}\left(g(\mathbf{v})\right)}{K}. \tag{4.5}$$

The use of curvature as a criterion is certainly the first thing that comes to one's mind. Indeed, if we think of planar shapes, we know that more points (vertices) are needed to represent high curvatures in opposition to straight line segments whose curvature is zero and for which the two extremal points are sufficient to represent and reconstruct the full shape. Our intuition may further suggest to take the highest curvature on the curve/surface to be the criterion. This would be the solution if the highest curvature happened to be a dominant feature on the whole shape. In many cases, this is not true. Instead of basing our choice on a local property, we therefore rely on the global perception of shapes' curvatures.

The characteristic resolution $\Re$, defined in Eq. (3.3), is a feature that is directly related to the curvature of a shape. As we reduce $\Re$, we act on the original shape by smoothing it. This machinery is nothing but choosing a tolerance τ. "Tolerance" is the common name given for the maximal edge length between adjacent points on a surface $\mathcal{S}$. In fact, the raw data is reduced to its characteristic resolution $\Re$ and every vertex becomes now of importance to obtain an accurate representation for $\mathcal{S}$. We recall that any topological transformation that happens on $\mathcal{S}$ and that we are to detect is translated on its one dimensional Morse function $g(\cdot)$; hence, we base the sampling rule of a 3D object on the sampling of $g(\cdot)$ or of a functional of $g(\cdot)$. We view $g(\cdot)$ as a continuous random variable, or practically as a discrete random variable X ($g \equiv X$), *s.t.*, $X : \mathcal{V} \rightarrow \mathbb{R}$ is defined on the probability space $(\mathcal{V}, \mathcal{A}, P_{\Re})$, where $\mathcal{V}$ is the set of $\Re$ vertices on the triangle mesh $(\mathcal{V}, \mathcal{F})$ approximating the surface $\mathcal{S}$ (Section 2.1). $\mathcal{A}$ is a $\sigma - Algebra$ composed of all the subsets of $\mathcal{V}$. Note that we interchangeably use $\mathcal{V}$ as the set or the matrix of vertices.

Similarly to Eq. (4.5), we apply the sampling along the axis of variation of X. In order to achieve an efficient sampling step l, we need to define the largest step l that will cause no breakdowns on the mesh $(\mathcal{V}, \mathcal{F})$. Let $\gamma_1 = g^{-1}\left(g(\mathbf{p})\right)$ and $\gamma_2 = g^{-1}\left(g(\mathbf{q})\right)$

be two distinct and consecutive connected components (*i.e.*, two consecutive iso-geodesic curves). With a resolution $\Re$ and a tolerance τ associated to it, the longest acceptable orthogonal geodesic distance between γ_1 and γ_2 is equal to τ. The remaining question is to find $l = |g(\mathbf{p}) - g(\mathbf{q})| = |\lambda_1 - \lambda_2|$. If $\mathbf{p}$ and $\mathbf{q}$ are connected then $\mathbf{d}(\mathbf{p}, \mathbf{q}) = \tau$, and we have for $\mathbf{v} \in \mathcal{V}$:

$$\mathbf{d}(\mathbf{p}, \mathbf{v}) > \mathbf{d}(\mathbf{p}, \mathbf{v}) \Longrightarrow \mathbf{d}(\mathbf{p}, \mathbf{v}) = \mathbf{d}(\mathbf{q}, \mathbf{v}) + \tau, \tag{4.6}$$

Similarly,

$$\mathbf{d}(\mathbf{p}, \mathbf{v}) < \mathbf{d}(\mathbf{p}, \mathbf{v}) \Longrightarrow \mathbf{d}(\mathbf{q}, \mathbf{v}) = \mathbf{d}(\mathbf{p}, \mathbf{v}) + \tau; \tag{4.7}$$

Therefore, if there are n elements from $\mathcal{V}$ verifying (4.6), then we find:

$$\lambda_1 - \lambda_2 = \frac{(\Re - 2(n-1))}{\max_{\mathbf{v}\in\mathcal{V}} \sum_{\mathbf{w}\in\mathcal{V}} \mathbf{d}(\mathbf{v}, \mathbf{w})} \cdot \tau. \tag{4.8}$$

We may also define an upper bound for l, independent of the variable n and equal to $\frac{(\Re-2)}{\max_{\mathbf{v}\in\mathcal{V}} \sum_{\mathbf{w}\in\mathcal{V}} \mathbf{d}(\mathbf{v},\mathbf{w})} \cdot \tau$.

4.3 Reeb graph matching

Reeb graphs being topological descriptors, we require the connectivity information to define a distance between them. We actually use reduced Reeb graphs where the node order is never 2. We herein define a distance measure that is in the same spirit as the edit-distance for shape matching as defined in [4]. We compute in (4.9) the similarity score $\mathbb{S}(\cdot, \cdot)$ between two reduced Reeb graphs Γ_1 and Γ_2 with N_1 and N_2 nodes, respectively, and $N_1 \leqslant N_2$. We denote by $\{\mathbf{v}_i\}_{i=1}^{N_1}$ the nodes of Γ_1, and $\{\mathbf{w}_i\}_{i=1}^{N_2}$ the nodes of Γ_2. We define the similarity between these two graphs as follows:

$$\mathbb{S}(\Gamma_1, \Gamma_2) = \frac{2\pi\sigma}{N_1} \sum_{i=1}^{N_1} \max_{j=1,\cdots,N_2} \Big\{ \beta\Big(\mathbf{v}_i, \mathbf{w}_j\Big) \delta_i^{(\Gamma_1)}(\mathbf{w}_j) \Big\}, \tag{4.9}$$

with

$$\delta_i^{(\Gamma_1)}(\mathbf{w}_j) = \begin{cases} 1 & \text{iff} \quad |\eta_{\mathbf{v}_i} - \eta_{\mathbf{w}_j}| \leqslant 2; \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\beta(\mathbf{v}_i, \mathbf{w}_j) = \frac{1}{2\pi\sigma} \exp\left(- \frac{|g(\mathbf{v}_i) - g(\mathbf{w}_j)|}{2\sigma^2} \right), \tag{4.10}$$

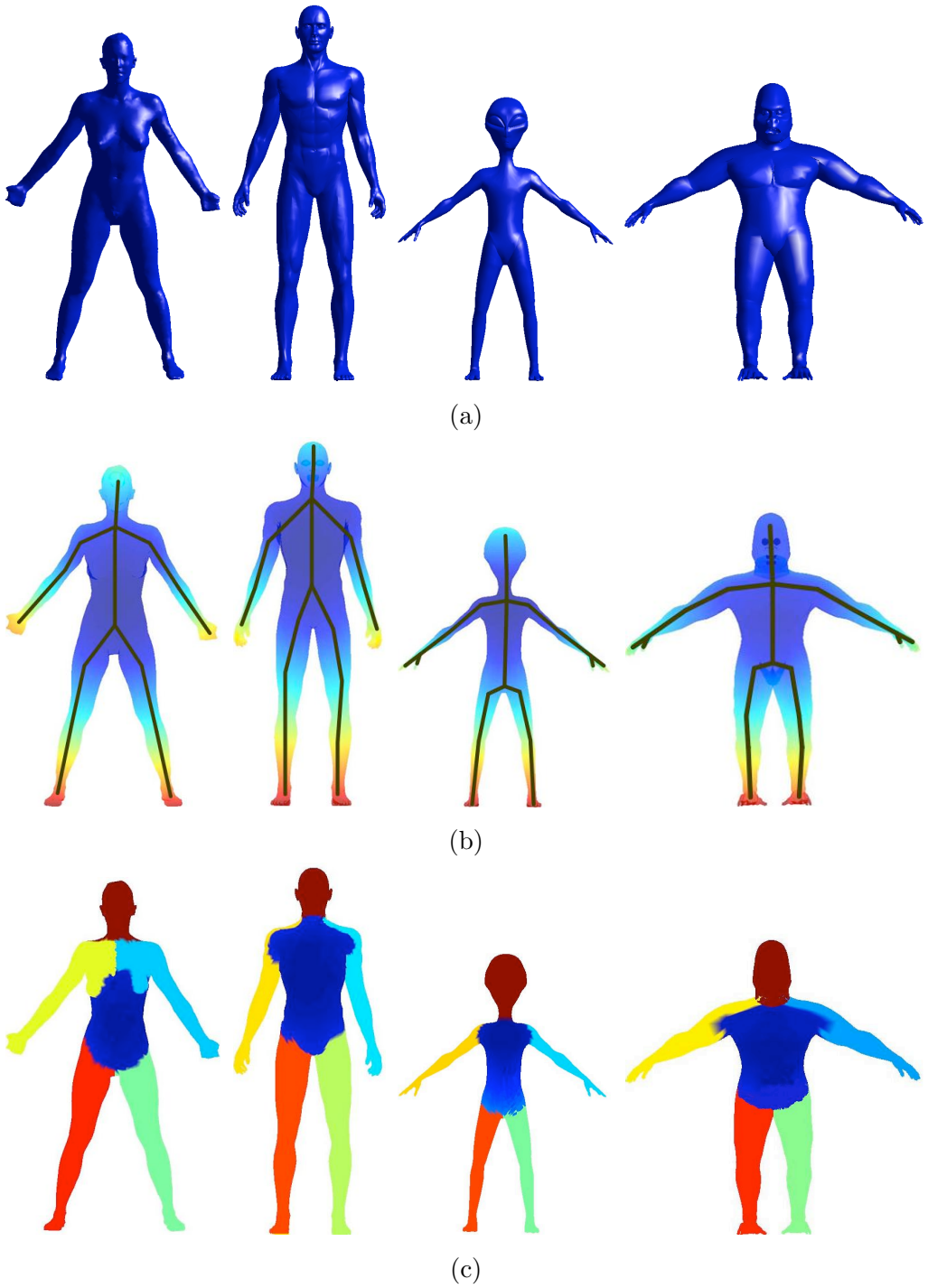
where σ is a strictly positive constant of our choosing and $\eta_{\mathbf{v}_i}$ and $\eta_{\mathbf{w}_j}$ are the orders of nodes $\mathbf{v}_i$ and $\mathbf{w}_j$, respectively. The function $\delta_i^{(\Gamma_k)}(\cdot)$ plays the role of a registration operator based on the order of graph nodes. We also choose a kernel function $\beta(\cdot,\cdot)$ that is strictly decreasing, hence the choice of the Gaussian kernel in Eq. (4.10). The intuition behind choosing the factor β is the assignment of a high similarity weight to registered nodes if they correspond to GGF levels that are close.

4.4 Illustration

In order to illustrate the proposed shape partitioning technique, we use the four objects in Figure 4.5 (a) [15]. We start by computing the GGF corresponding to each one of these shapes and extract the corresponding topological Reeb graphs (Figure 4.5 (b)). In Figure 4.5 (c), we present the segmented objects. These have the same colors for primitives that we matched using Eq.(4.9).

4.5 Conclusion

We proposed a technique for an automatic partitioning of 3D objects into primitives along with a distance to ensure a good correspondence between them. The next Chapter shows how this segmentation may be used to simplify recognition problems and for partial matching of shapes.



(a)

(b)

(c)

Figure 4.5: Overview of the partitioning technique (Best visualized in color).

Chapter 5

Squigraphs for Geometric Modeling

We propose to superpose global topological and local geometric 3D shape descriptors in order to define one compact and discriminative representation for a 3D object. While a number of available 3D shape modeling techniques yield satisfactory object classification rates, there is still a need for a refined and efficient identification/recognition of objects among the same class. In this Chapter, we add detailed geometric information to Reeb graphs, by tracking the evolution of Morse function's level curves along each primitive resulting from partitioning complex shapes. We then embed the manifold of these curves into $\mathbb{R}^3$, and obtain a single curve. We thus build new graphs rich in topological and geometric information that we refer to as squigraphs. Our experiments show that squigraphs are more general than existing techniques. They achieve similar classification rates to those achieved by classical shape descriptors. Their performance, however, becomes clearly superior when finer classification and identification operations are targeted. Indeed, while other techniques see their performances dropping, squigraphs maintain a performance rate of the order of 97%.

5.1 Introduction

Numerous 3D representation approaches have recently been proposed, as overviewed in [29] and [30]. In the present work, we propose a 3D object representation model that

offers different levels of discrimination. This model invokes an object's geometric and topological information. This rather recent technique, henceforth referred to as *topo-geometric modeling* of 3D objects, was first exploited in [31], where Meng Yu *et al.* simultaneously define topological and geometrical feature maps. They show that these feature maps are invariant to all affine transforms. This invariance includes, by definition, non uniform scaling transforms. This, in turn, implies an inaccurate dissimilarity measure between the geometry of shapes, as the geometry of shapes is only invariant to Euclidean transforms. Tung and Schmitt [2] and Baloch *et al.* [28] present a different theme by first representing the topology of a 3D shape, and subsequently enhancing it with its geometrical representation. The advantage of separating the modeling into two distinct steps is twofold: First it provides two levels of discrimination; a) a coarse level with a simple topological skeleton called a *Reeb* graph [22, 32] (Section 4.2), and b) a fine level with geometric weights assigned to each edge or node of the previously extracted topological graph. Another advantage of a topological graph representation of an object is its ability of matching objects by parts, thereby enabling a transition from a global to a localized correspondence between shapes (*e.g.,* mechanical parts, manufactured solids). In representing an object, one ideally avoids choosing a reference point extrinsic to its surface [1, 33, 34, 35, 36]. To that end, we address this limitation, evident in [28, 31], by adopting an integrated geodesic distance function as defined in [1]. Such a function ensures the invariance of the Reeb graph of an object subjected to any isometric transform. As first proposed in [1], this function was limited to only providing topological information of an object surface. Tung and Schmitt [2], in an attempt to mitigate this limitation, proposed to revert back to a Euclidean reference frame to compute differential geometric features which they use as weight attributes on the Reeb graph.

In this Chapter, we propose a novel alternative technique that is efficient, theoretically sound, practically complete, and computationally simple. Our proposed approach is rooted in modeling the geometry along edges of the object representative graph. We subsequently define a new topo-geometric skeleton that we call *squigraph*. The key idea for the geometric modeling of shapes is to embed in Euclidean space a manifold of new characteristic curves (we refer to as *iso-geodesic* curves) of an object surface. The resulting embedding space, as we show below, is $\mathbb{R}^3$, and the curve manifold is 1-dimensional, *i.e.*, a space curve in $\mathbb{R}^3$. As a result, all topological and geometric information we exploit remains intrinsic to the

modeled object, hence preserving all desired invariance properties alluded to earlier.

5.2 Representation of a surface geometry

We note that, just as in [2, 28], our 3D object modeling approach starts with a topological analysis followed by a geometrical analysis step. In our present work, however, we strive to achieve a topo-geometric representation that is both discriminative and efficient. This is why we proceed by economically exploiting, for the geometric modeling, the same entities used earlier for the topological modeling with Reeb graphs. The entities in question are the level sets of the GGF, *i.e.*, the iso-geodesic sets defined on the surface of an object. Using iso-geodesic sets allows us to further extend the invariance properties of the GGF to the geometric phase.

In addition, unlike the usual global shape descriptors, a more complete geometric representation is one which provides a local description taking into account the spatial location of points or vertices [31]. The iso-geodesic curves used in the topological representation consequently appear, once again, to be able to offer perfect local geometric descriptors. We therefore take advantage of these already extracted entities, and use them for the second phase of our modeling as described in what follows.

Broadly, our strategy is as follows: once we extract iso-levels of the GGF, we obtain a set of closed curves (namely the iso-geodesic curves) along each edge of a topological graph of an object. We use these characteristic curves to define a geometric model. We subsequently assign the models to the corresponding edges. We in turn exploit these geometric models to further compare every two edges that have been matched as part of two topologically similar graphs. In Figure 4.5 (c), we show all matched edges in the same color. We compactly represent the final topo-geometric model through a new spatial graph that we call *squigraph* (Figure 5.2 (d)). A squigraph graph differs from a classical Reeb graph by its "squiggling" curves that replace the standard straight edges and encode the geometric information about the shapes.

5.3 Whitney theorem for modeling

It is important to note that a geometric shape along an edge corresponds to what we described, in Section 4.2.1, as a mono-cardinality subsurface. So all the partial shapes along Reeb graph edges are simply equivalent to generalized cylinders. In fact, this philosophy of decomposing a complex shape into a set of cylinders goes back to the very early work on tubular representation of 3D shapes.

Equation (4.4) shows that a mono-cardinality subsurface $\mathcal{M}$ is formed by the disjoint union of an infinite number of *iso-geodesic* curves $C(t)$, monotonically and continuously parameterized by $t \in [a, b] \subset \mathbb{R}^+$, where $[a, b]$ is the range of the GGF restricted to $\mathcal{M}$ (Figure 5.1 (a)). It therefore naturally follows that $\mathcal{M}$ is a smooth path in high dimensional space whose point elements are the iso-geodesic curves $C(t)$, $t \in [a, b]$; hence, $\mathcal{M}$ constitutes a smooth 1-dimensional manifold on the topological space of closed iso-geodesic curves on which the restriction of the GGF is a homeomorphism (Figure 5.1 (c)). In what follows, we interchangeably use $\mathcal{M}$ to refer to the mono-cardinality subsurface embedded in $\mathbb{R}^3$, as well as to the corresponding curve embedded in the higher dimensional space.

Getting back to our objective of simply modeling each mono-cardinality subsurface $\mathcal{M}$, we first invoke the *Whitney embedding* theorem as stated below,

Theorem 2 (Whitney embedding theorem) *[38] Let γ be an n-dimensional compact Hausdorff C^r manifold, $2 \leq r \leq \infty$. Then, there is a C^r embedding of γ in $\mathbb{R}^{2n+1}$.*

In the present problem, the manifold γ is nothing but $\mathcal{M}$, for which the dimension n is equal to one; hence, the stated theorem asserts the existence of an embedding of $\mathcal{M}$ in $\mathbb{R}^3$. In fact, such embeddings have full measure in the set of maps or even projections into $\mathbb{R}^3$, so we can assume that a random projection works. Thus we can safely use this embedding for our geometric modeling since, by definition, an embedding preserves the geometry and does not introduce any new intersections in the new space. In practice, we approximate $\mathcal{M}$ with a finite number N of iso-geodesic curves as illustrated in Figure 5.1 (b). On each curve, we take M uniformly spaced points $\mathbf{v}_j^i$ defined by their Euclidean coordinates (x_j^i, y_j^i, z_j^i), $i = 1, 2, \ldots, N$, and $j = 1, 2, \ldots, M$. Each curve is now a point in $\mathbb{R}^{3M+1}$ represented by the column vector $\mathbf{V}_i$, $i = 1, 2, \ldots, N$ (Figure 5.1 (c)), such that,

$$\mathbf{V}_i = \left[\left[\mathbf{V}_1^i\right]^T \ldots \left[\mathbf{V}_M^i\right]^T \right]^T \text{ with } \mathbf{V}_j^i = \left[x_j^i, y_j^i, z_j^i\right]^T. \tag{5.1}$$

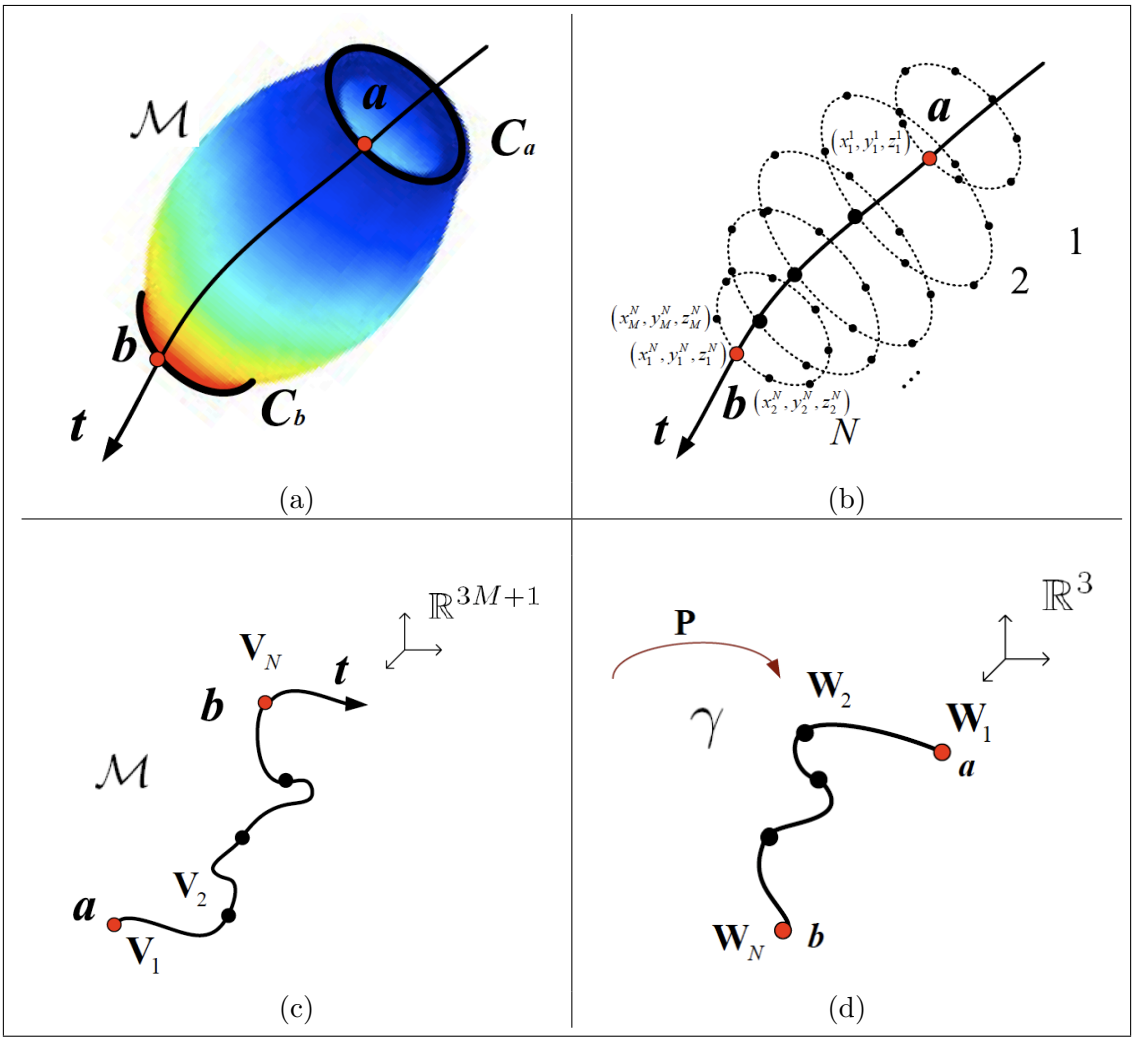


Figure 5.1: Geometric modeling of a mono-cardinality subsurface $\mathcal{M}$: (a) The GGF on $\mathcal{M}$ is strictly monotonous and takes its values in the interval $[a, b]$. As a result, the two bounding iso-geodesic curves are $C(a)$ and $C(b)$. (b) Discretized version of $\mathcal{M}$. (c) Path created by $\mathcal{M}$ in high dimensional space. (d) Final modeling curve in 3D space.

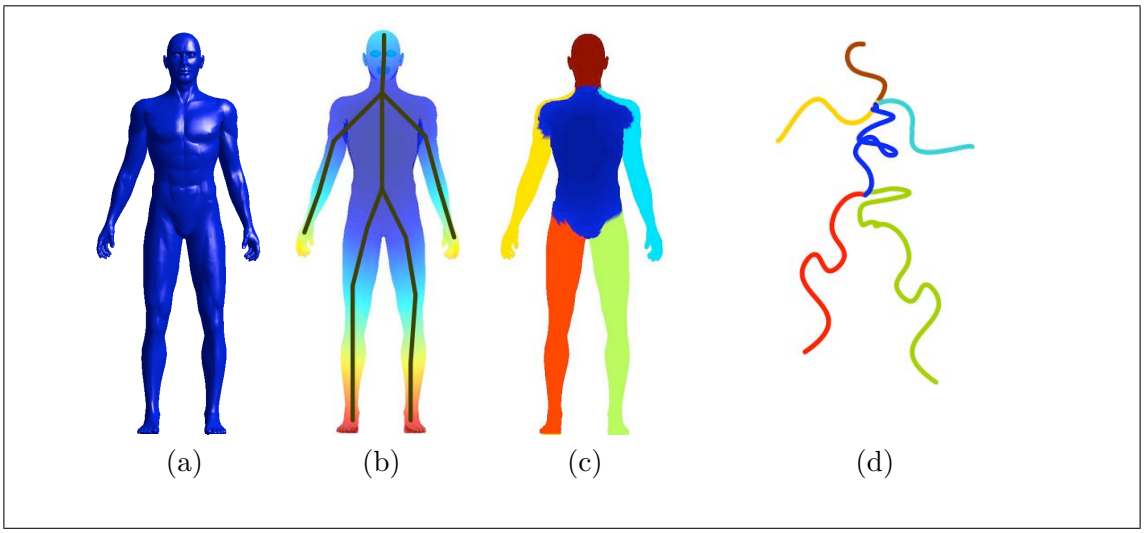


Figure 5.2: Object representation using a *squigraph*. (a) Original shape. (b) Extracted Reeb graph. (c) Partitioning into primitives. (d) Squigraph for a topo-geometric representation.

The sample set of iso-geodesic curves on $\mathcal{M}$ is a matrix $\mathbf{V}$ of dimension $3M \times N$ where $\mathbf{V} = [\mathbf{V}_1 \cdots \mathbf{V}_N]$. Applying the Whitney theorem on $\mathbf{V}$ reduces the embedding problem to a simple linear formulation;

$$\mathbf{W} = \mathbf{P}^T \mathbf{V}, \tag{5.2}$$

where $\mathbf{W}$ is a set of N points in $\mathbb{R}^3$ resulting from the projection of $\mathbf{V}$ via $\mathbf{P}$ into $\mathbb{R}^3$. In other words, as illustrated in Figure 5.1 (d), the manifold $\mathcal{M}$ is reduced to a space curve γ whose sample points are represented by the columns of the matrix $\mathbf{W}$. In Figure 5.2, we illustrate the idea of a topo-geometric modeling via the space modeling curves that we just defined; hence, we intrinsically enhance the typical 3D Reeb graph of Figure 5.2 (b). To that end, we assign one modeling curve to each edge of the Reeb graph. By so doing, we may view the final representation as a new kind of graph, that we refer to as a *squigraph*, as shown in Figure 5.2 (d). The Whitney embedding theorem guarantees that almost any projection of $\mathcal{M}$ is an embedding. In [39, 40], the notion of good *Whitney embedding* is introduced to identify a class of projections. For many applications, such as recognition, a unique representation of an object is required. As a result, we apply the notion of *optimal embedding* with the performance criterion presented in [40] and defined in (5.4) being maximized.

5.4 Optimal projection

An embedding gives us a one-to-one mapping between the path in high dimensional space $\mathcal{M} \subset \mathbb{R}^{3M}$, and the path $\gamma \subset \mathbb{R}^3$, so that no two points in $\mathcal{M}$ will collapse as a result of the mapping in (5.2). An implementation of this principle is presented in [40] and referred to as a *secant based method*. As just explained, no two points in $\mathbb{R}^{3M}$ are mapped to the

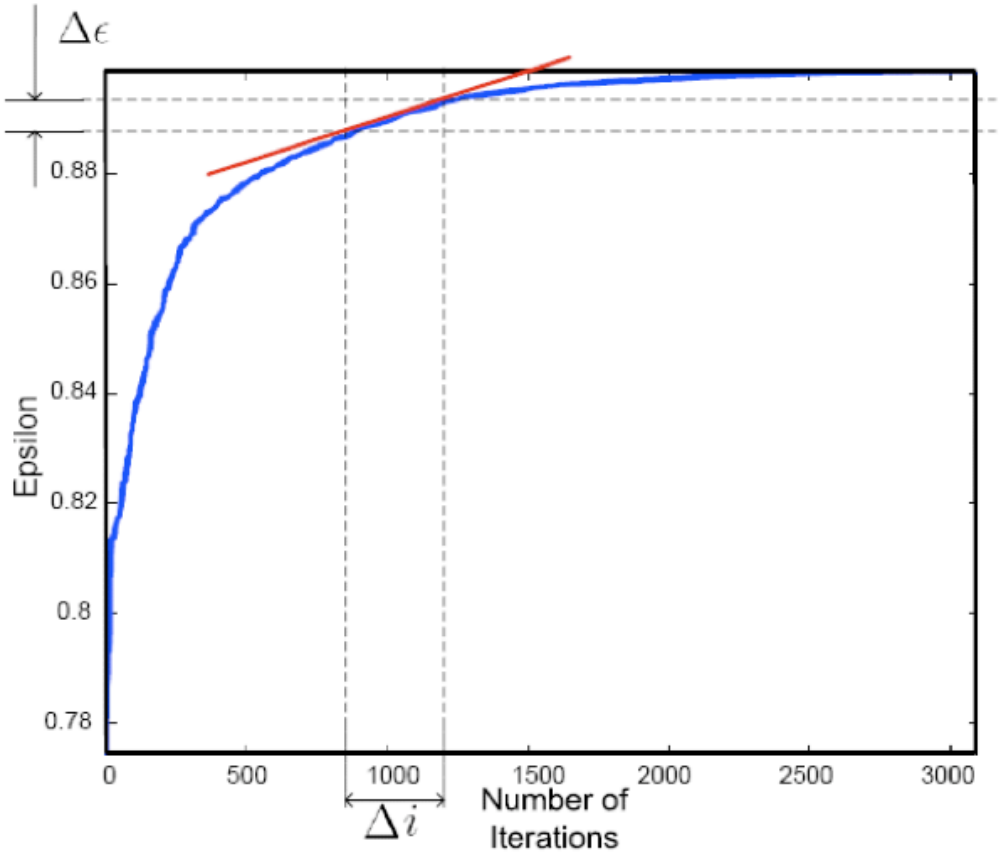


Figure 5.3: Asymptotic convergence of LTMADS applied to 3D shape dataset.

same point in $\mathbb{R}^3$. This means that the projecting vector is linearly independent from any possible secant[1] in $\mathbb{R}^{3M}$. We define the set of all possible unit secant vectors from the initial

[1] *Secant* or *secant line* is a line that intersects two points from a curve.

data $\mathbf{V}$ as follows:

$$\begin{aligned} \mathbf{\Psi} &= \left\{ \frac{\mathbf{V}_i - \mathbf{V}_j}{\|\mathbf{V}_i - \mathbf{V}_j\|},\ (i,j) \in \{1,\ldots,N\}^2 \text{ and } i \neq j \right\} \\ &= \{\mathbf{\Psi}_i, \quad i = 1,\ldots,L\}, \end{aligned} \tag{5.3}$$

with $L = \frac{N!}{(N-2)!2!}$.
The performance of the projection operator $\mathbf{P}$ is then reflected by:

$$\epsilon = \min_{i=1,\ldots,L} \left[\| \mathbf{P}^T \mathbf{\Psi}_i \|_2 \right]. \tag{5.4}$$

The closer ϵ is to 1, the better spread out the projected points in $\mathbb{R}^3$ are, and the better the choice of $\mathbf{P}$ is; hence, finding a good estimate of the optimal projection is reduced to the following minimization problem:

$$\begin{aligned} \hat{\mathbf{P}} &= \arg\min \left[- \min_{i=1,\ldots,L} \| \mathbf{P} \mathbf{\Psi}_i \|_2 \right] \\ &= \arg\min F(\mathbf{P}). \end{aligned} \tag{5.5}$$

Since the function $F(\mathbf{P})$ is non differentiable, we need a direct search algorithm that does not require any derivative of the function to minimize it. As presented in [41], the Lower *Triangular Mesh Adapted Direct Search algorithm* (LTMADS) has been adapted to Riemannian manifolds and tested on an example of a Whitney embedding. The solution given by LTMADS is highly dependent on the initial data. On our 3D objects, the performance of LTMADS converges asymptotically to 1 (Figure 5.3). We define a stopping criterion for the LTMADS algorithm adapted to each data by considering the slope $\Delta\epsilon = |\epsilon_i - \epsilon_{i-1}|$, where ϵ_i and ϵ_{i-1} respectively represent the performances of the optimization algorithm at iterations i and $i-1$.

5.5 Space marking

It is important to ensure that all iso-geodesic curves' discrete representations are uniform and consistent within a coordinate space. Visually, we may equate this task to marking the surface $\mathcal{M}$. By marking, we mean drawing a well defined reference line on the surface $\mathcal{M}$. In Figure 5.1 (a), this reference line is the one curve going through the two red points corresponding to levels a and b. This special line enables us to construct the vectors

$\mathbf{V}_i$ in an accurate and consistent manner. To that end, we use the *orthogonal curve* defined in Eq. (4.3).

We note that we only need a starting point to extract an orthogonal curve. This point, however, is not unique as it is only required to be a point from a bounding iso-geodesic curve ($C(a)$ or $C(b)$ in Figure 5.1). In fact, thanks to the construction proposed in Eq. (5.1), choosing a different starting point will only result in a mere rotation of the reference axes in $\mathbb{R}^{3M}$. Besides, because in the present geometric modeling we require a full invariance to Euclidean transforms, this rotation will have no effect on the final modeling curve (γ in Figure 5.1 (d)).

Chapter 6

Correspondence Preserving Similarity Invariant for Space Curves

We present a new similarity invariant signature for space curves. This signature is based on the information contained in the turning angles of both the tangent and the binormal vectors at each point on the curve. For an accurate comparison of these signatures, we define a Riemannian metric on the space of the invariant. We show through relevant examples that, unlike classical invariants, the one we define in this paper enjoys multiple important properties at the same time, namely, a high discrimination level, independence of any reference point, uniqueness, as well as a good preservation of the correspondence between curves; moreover, we show how to use the proposed signatures for both full and partial matchings of 3D objects.

6.1 Introduction

Multiple 3D modeling methods use spatial curves for recognition and matching of objects. Spatial curves are exploited in different configurations. They may, for instance, be extracted as contours of landmark surfaces [7], as level curves of a Morse function [8], or also as elements of curved skeletons [9, 10]. All these techniques, despite their differences, agree in relying on curves' properties in solving computer vision problems. This common

approach is motivated by the fact that curves in 3D are fairly well known geometric entities; moreover, under some conditions, they can accurately describe the overall geometry of an object in 3D space [7]. Translating the constraints of 3D shape representation techniques to curves reduces the level of difficulty associated with the representation problem and makes it more tractable.

Pose invariance of surfaces is a common requirement in object modeling. It is also a good illustration of the simplification from surfaces to curves. Indeed, an effective and economical solution for curves pose invariance may be provided through Euclidean/similarity invariants or invariant signatures [42]. Besides this pose invariance property, additional constraints are imposed on 3D curves as a direct result of the nature of 3D shape recognition applications. These constraints may be summarized as follows: invariance to a group of transforms, uniqueness, local characterization (local support), ability to determine shape properties such as symmetries and part correspondences. To the best of our knowledge, none of the available references seem to gather all these properties at once. The most complete work is the one of Mokhtarian and Bober [7], as they succeed in citing and addressing all the necessary properties; however, to provide an invariance to scaling transforms, the authors use a multi-resolutional procedure. In the present work, we provide the advantage of having an invariant that is, by definition, *i.e.*, without additional steps, fully invariant to all similarity transforms. The key contribution of our work resides in using turning angles based on curvature and torsion instead of using curvature and torsion directly. We further ensure a natural registration of all the invariants on one curved space which leads to defining an accurate and computationally easy metric for curves comparison. Indeed, we show that while torsion and curvature are clearly variant with scaling, turning angles are not. The first inspiration of our work comes directly from [43], where an invariant for planar curves is defined. This invariant has the particularity to be an information theoretical measure of local geometric properties of curves. Moreover, this invariant comes as a proof for long-time psychological assumptions on mental shape perception. The operation of comparing invariants, although often overlooked, is crucial in assessing the properties of the invariant, and accurately achieving recognition operations. This is why, in the present work, we complete the proposed invariant signature by defining its Riemannian space equipped with an intrinsic measure.

The present Chapter is organized as follows: In Section 6.2, we explain the mo-

tivation behind this work. In Section 6.3, we review important geometric and information theoretical notions used throughout this paper. We briefly cover the work of Feldman and Singh [43] in Section 6.4, as it was the precursor and the inspiration for this effort. In Section 6.5, we introduce our new similarity invariant signatures for space curves and define a Riemannian metric for their accurate comparison in Section 6.6. We prove and illustrate the property of correspondence preservation in Section 6.7. Finally, in Section 6.8 we show direct applications of the proposed signatures in 3D object matching and comparison.

6.2 Motivation and related work

We say that two space curves have the same shape if there exists a rigid transformation (rotation, translation) or a uniform scaling or a combination thereof that will cause one of these curves to completely overlap the other. In addition, we talk about curves that are partial matches. This practically translates into shapes subjected to occlusions, or problems of reconstruction from multiple parts. In what follows we provide a list of properties that models of space curves are to verify;

1. Invariance: In this work, we view a shape as a geometrical entity that is independent of the coordinate space. It is therefore necessary to have a representation that eliminates the effects of rigid motions and uniform scalings. In other words, we request our modeling to be invariant to the group of similarity transforms.

2. Uniqueness: Curves with the same representation are considered to be curves with the same shape. If curves' models are different, then these curves are geometrically different. This requirement implies using a *complete system* of invariant functions whose combination will define the final model.

3. Local support: This property is related to the partial matching problem. In this work our objective is to respect the correspondences between parts of curves. This is why we highlight the importance of having a locally supported representation.

4. Independence of parametrization and of the starting point: This property will show to be important in defining the appropriate metric to compare the final models for space curves.

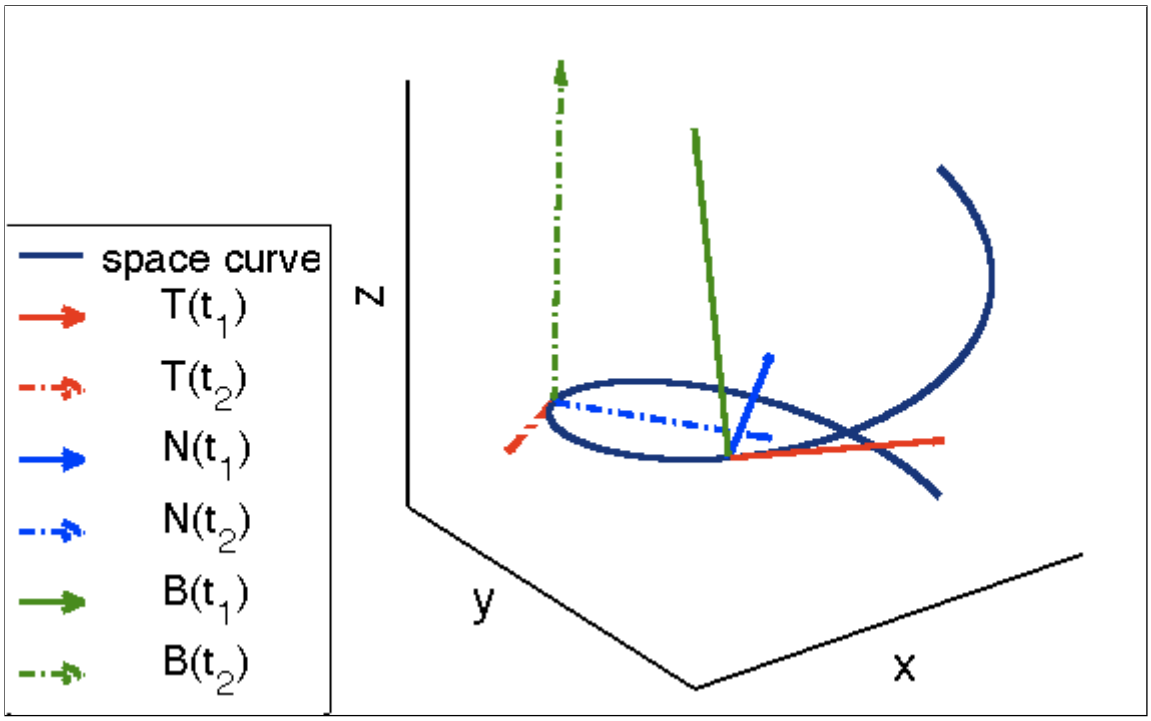


Figure 6.1: Frenet Frame.

5. Determination of shape properties: It is sometimes of interest to be able to determine some shape properties such as symmetries or periodicities.

As we progress in this paper, we will relate our results to the motivations listed above. We start this work by identifying appropriate geometric features to use in defining our model for space curves. We propose to rely on the curvature and the torsion functions of a space curve since these two uniquely specify a space curve up to a rotation and a translation; however, they are not invariant to scaling which leads us to using the corresponding turning angles, instead.

6.3 Background and formulation

6.3.1 Turning angles

A space curve is uniquely determined, up to a Euclidean transform, by its curvature function $\kappa(t)$, and torsion function $\tau(t)$, both continuous functions of the parameter t;

hence, we naturally use these measurements to define an adequate invariant signature curve; however, since we target the group of similarity transforms, and knowing that curvature and torsion are not scale invariant, we use turning angles as the geometric features describing space curves [43]. In what follows, we show how it is possible to relate curvature and torsion as linear functions of turning angles. Using the Frenet-Serret formulae (Figure 6.1),

$$\frac{d\mathbf{T}}{dt} = \kappa\mathbf{N}, \tag{6.1}$$

$$\frac{d\mathbf{N}}{dt} = -\kappa\mathbf{T} + \tau\mathbf{B}, \tag{6.2}$$

$$\frac{d\mathbf{B}}{dt} = -\tau\mathbf{N}, \tag{6.3}$$

where $\mathbf{T}$, $\mathbf{N}$ and $\mathbf{B}$ are the tangent, the normal and the binormal vectors, respectively, we define two turning angles α_T and α_B. α_T is the change in the direction of $\mathbf{T}$, and α_B is the change in the direction of $\mathbf{B}$ such that: $\alpha_T(t) \approx \kappa(t) \cdot dt$ and $\alpha_B(t) \approx \tau(t) \cdot dt$. By choosing to use curvatures as main descriptors, we systematically verify properties 1) and 2) in the list of motivations, as well as the independence of parametrization. We also start touching upon properties 3) and 4).

6.3.2 Shannon surprisal

We extract the information contained in the proposed turning angles, *i.e.*, our chosen geometric measures, by using a notion in information theory known as "Shannon surprisal" [44, 43]. We assume that $\alpha(t)$ follows a *von Mises* distribution with a zero mean and a spread parameter b equal to 1. We then define the probability density function of $\alpha(t)$ as follows:

$$f_\alpha\left(\alpha(t)\right) = A \exp\left(\cos(\alpha(t))\right), \qquad \forall\, t, \tag{6.4}$$

with $\alpha \in [-\pi, +\pi]$ and $A = \frac{1}{2\pi \cdot Bessel(0,b)}$. $Bessel(0, b)$ being the Bessel distribution of mean 0 and variance b. The surprisal of $\alpha(t)$ is by definition:

$$\begin{aligned} \theta(t) &= u\left(\alpha(t)\right) \\ &= -\ln\left(f_\alpha\left(\alpha(t)\right)\right) \\ &= -\ln(A) - \cos(\alpha(t)), \qquad \forall\, t. \end{aligned} \tag{6.5}$$

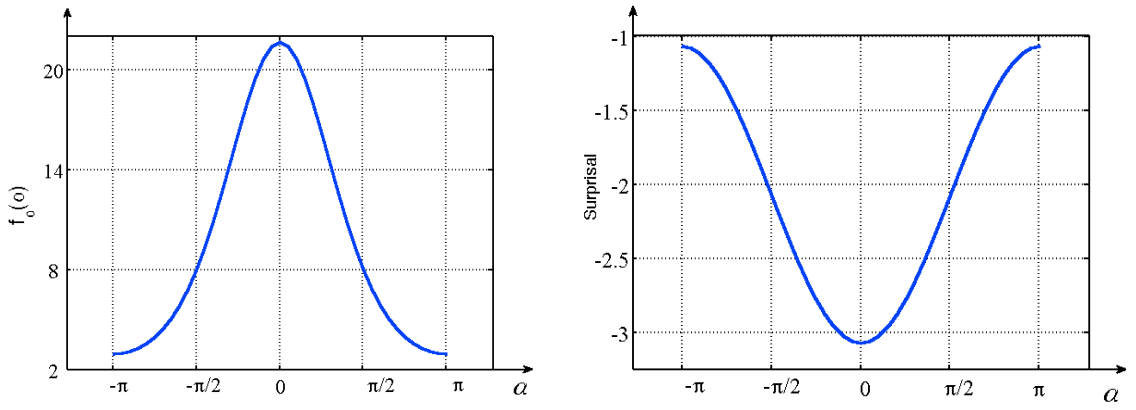


Figure 6.2: (a) The probability density function of a von Mises distribution. (b) The corresponding surprisal, which is also the defined manifold $\mathcal{N}$.

6.4 Invariant signature for planar curves

6.4.1 Definition

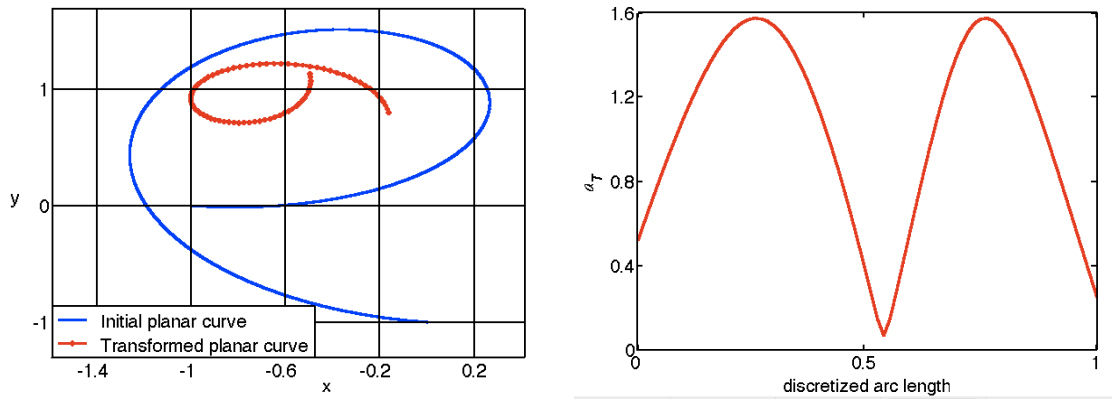
In [43], Feldman and Singh present an invariant signature for planar curves. This invariant is based on the turning angle $\alpha(t)$, the change in the direction of the tangent vector at the instant t. The actually considered invariant signature is the information gained when measuring $\alpha(t)$ at all instants t. This exactly corresponds to $\theta(t)$, the surprisal of $\alpha(t)$ as defined in (6.5). For a simple planar curve of length L sampled at N equal intervals of arc length Δt, $\alpha(t)$ is related to the curvature $\kappa(t)$ at a given point by the following approximation,

$$\alpha(t) \approx \kappa(t) \cdot \Delta t. \tag{6.6}$$

We note that scaling the curve implies scaling both $\kappa(t)$ and L while keeping the value of $\kappa(t)\frac{L}{N}$ invariant. It thus follows that $\alpha(t)$ is a measure equivalent to $\kappa(t)$ except that it is scale-invariant. The actually considered invariant signature is the information gained when measuring $\alpha(t)$ at all instants t, which is the surprisal function $\theta(t)$ as defined in (6.5).

6.4.2 Illustration

In Figure 6.3 (a), we illustrate an example of a planar curve (solid blue) and its transformed version (dotted red). The applied transformation is a combination of a trans-



(a) (b)

Figure 6.3: Two similar planar curves are illustrated in (a). In (b), the same turning angle α_T is obtained for the two planar curves in (a).

lation, rotation, and scaling, *i.e.*, a similarity transform. The resulting invariant signatures $\theta(t)$ for these two curves are exactly the same. We show the corresponding turning angle function in (Figure 6.3 (b)).

6.5 Invariant signature for space curves

6.5.1 Definition

The following section contains the major contributions of this paper. We start with a generalization of the invariant signature we presented in Section 6.4 from 2D to 3D. In our case, we deal with space curves instead of planar curves. For this reason, we require two turning angles, α_T and α_B, versus one in the planar case; hence, we naturally use these two measurements to define an adequate invariant signature curve. α_T is the change in the direction of $\mathbf{T}$. Informally, we view it as a measure of how much a curve is curved and diverges from a line. α_B is the change in the direction of $\mathbf{B}$. Similarly, we view it as a measure of how much a curve is twisted and diverges from a plane.

$$\alpha_T(t) \approx \kappa(t) \cdot \Delta t \quad \text{and} \quad \alpha_B(t) \approx \tau(t) \cdot \Delta t. \tag{6.7}$$

Similarly to $\alpha(t)$, $\alpha_T(t)$ and $\alpha_B(t)$ follow *von Mises* distributions of mean zero and with a spread parameter b equal to 1. Thus, exactly as in (6.5), these two measures introduce the

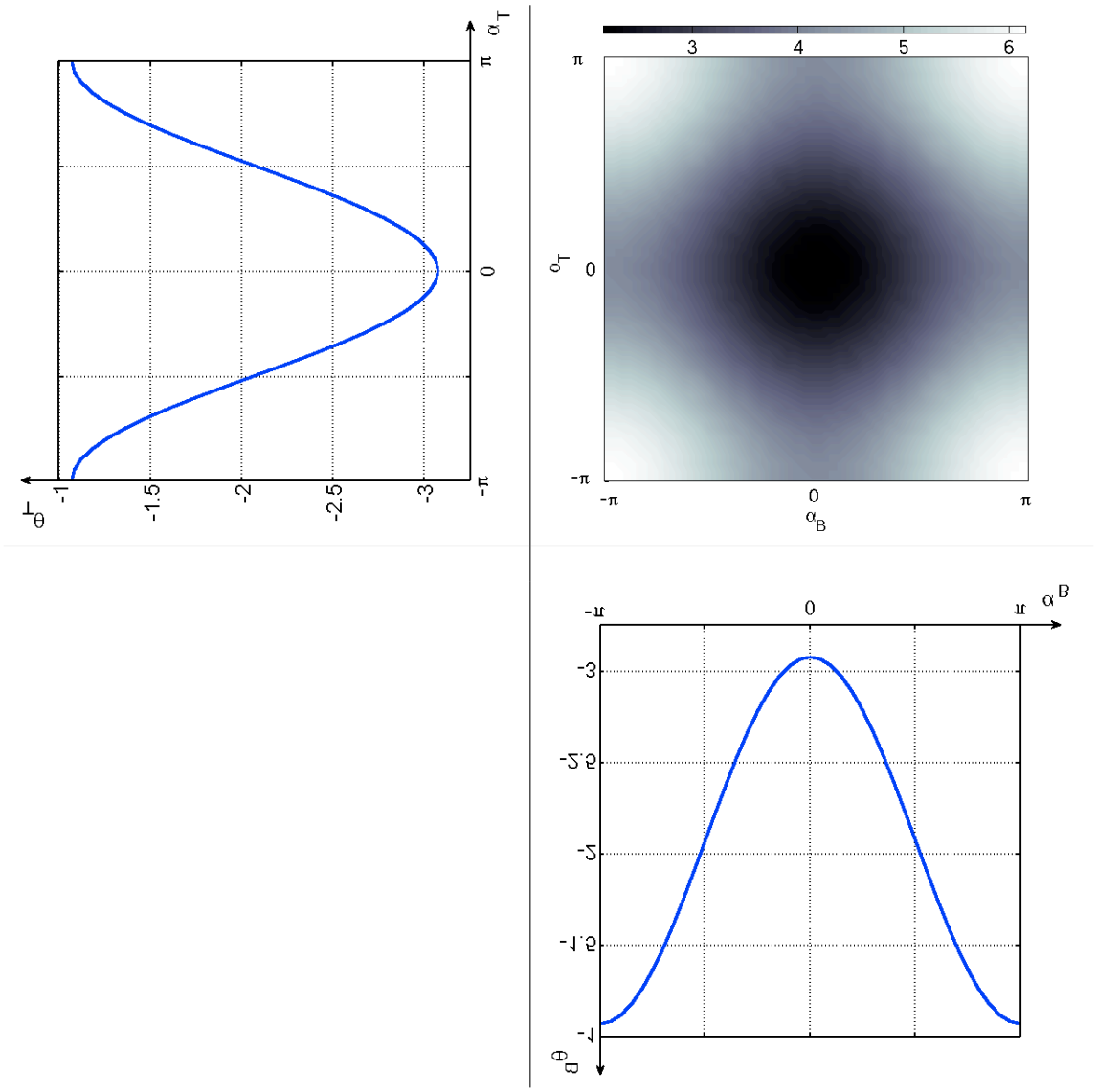


Figure 6.4: From the surprisal of two marginals to the surprisal of one binary distribution.

following independent invariant signatures for space curves:

$$\theta_T(t) = -\log(A) - \cos(\alpha_T(t)) \tag{6.8}$$

and

$$\theta_B(t) = -\log(A) - \cos(\alpha_B(t)). \tag{6.9}$$

Instead of separately considering the marginals of the two random variables $\alpha_T(t)$ and

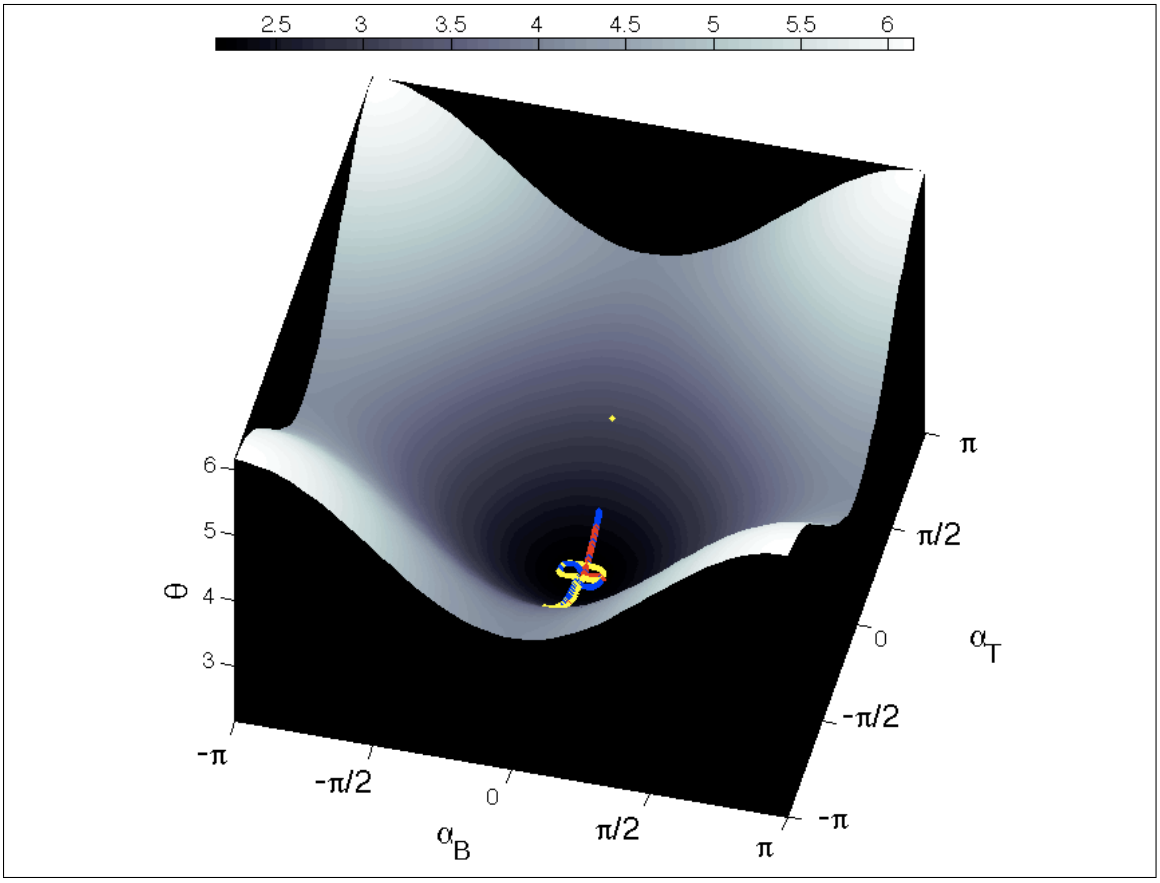


Figure 6.5: Manifold defined by the invariant signature curve as defined in Eq.(6.11).

$\alpha_B(t)$, we define a third invariant term that considers the random vector $[\alpha_T(t), \alpha_B(t)]^T$. Thus, the distribution f_α of interest becomes the binary *von Mises* distribution of the independent variables $\alpha_T(t)$ and $\alpha_B(t)$, such that:

$$f_\alpha(\alpha_T(t), \alpha_B(t)) = A^2 \exp(\cos(\alpha_T(t)) + \cos(\alpha_B(t))). \tag{6.10}$$

The corresponding surprisal function $\theta(\alpha_T(t), \alpha_B(t)) \equiv \theta(t)$ becomes,

$$\begin{aligned} \theta(t) &= -\ln\left(f_\alpha(\alpha_T, \alpha_B)\right) \\ &= -2\ln(A) - \cos(\alpha_T(t)) - \cos(\alpha_B(t)), \end{aligned} \tag{6.11}$$

with $(\alpha_T(t), \alpha_B(t)) \in ([-\pi, \pi])^2$.

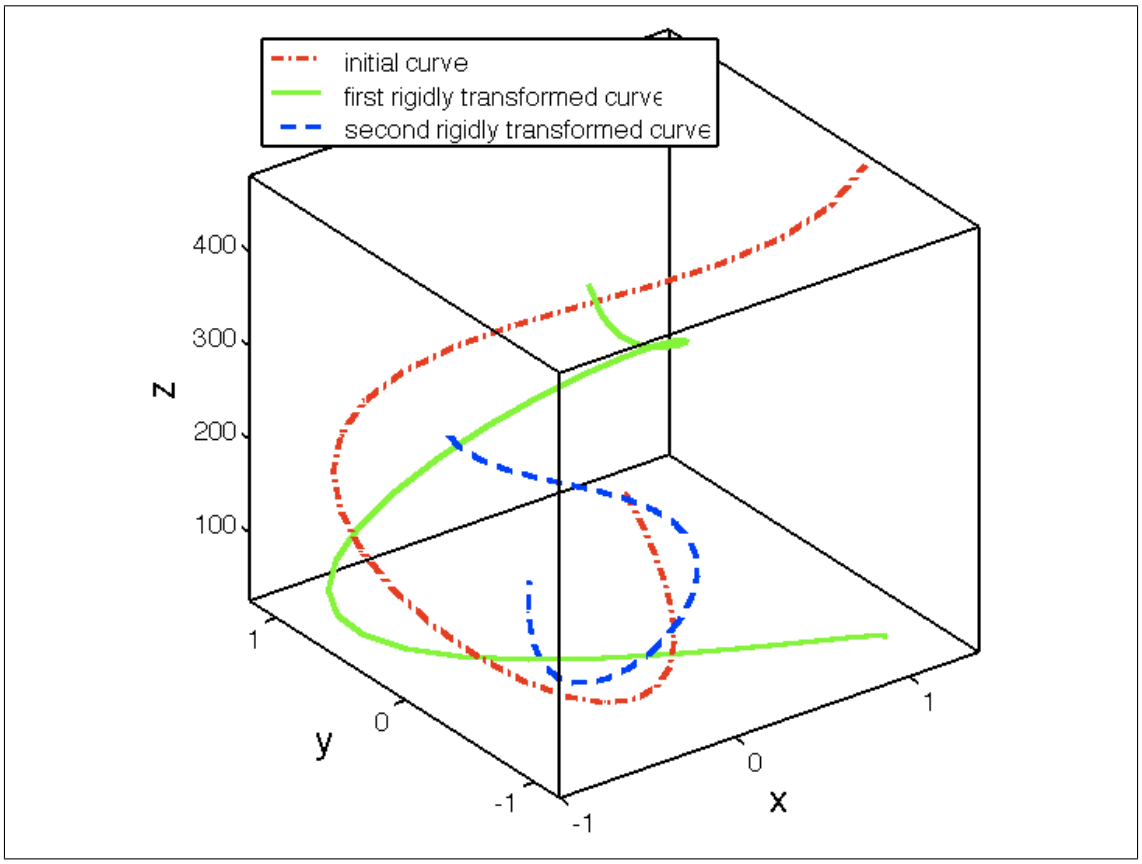


Figure 6.6: Two different sets of synthetic space curves. In (a), T1 and T2 are two similarity transforms. In (b), $C1$, $C2$ and $C3$ are three similar curves except at one inflection point.

6.5.2 Illustration on synthetic space curves

We test the invariant signature defined in (6.11) on some synthetic space curves. In Figure 6.6, we show the initial curve $C1$ defined by,

$$C1 : t \in [0, \pi] \rightarrow (\cos(2t), \cos(3t), \cos(5t)). \tag{6.12}$$

We define two similarity transforms through the matrices $T1$ and $T2$; therefore, all the curves, except $C2$, are similar to $C1$. Indeed, we find two sets of invariants as shown in Figure 6.7. Those corresponding to the family of $C1$ are represented in red in (a) and (c) (left column). (b) and (d) correspond to $C2$ (right column).

This example illustrates the invariance of the turning angles to similarity transforms. It is actually a property that is always verified thanks to the intrinsic nature of

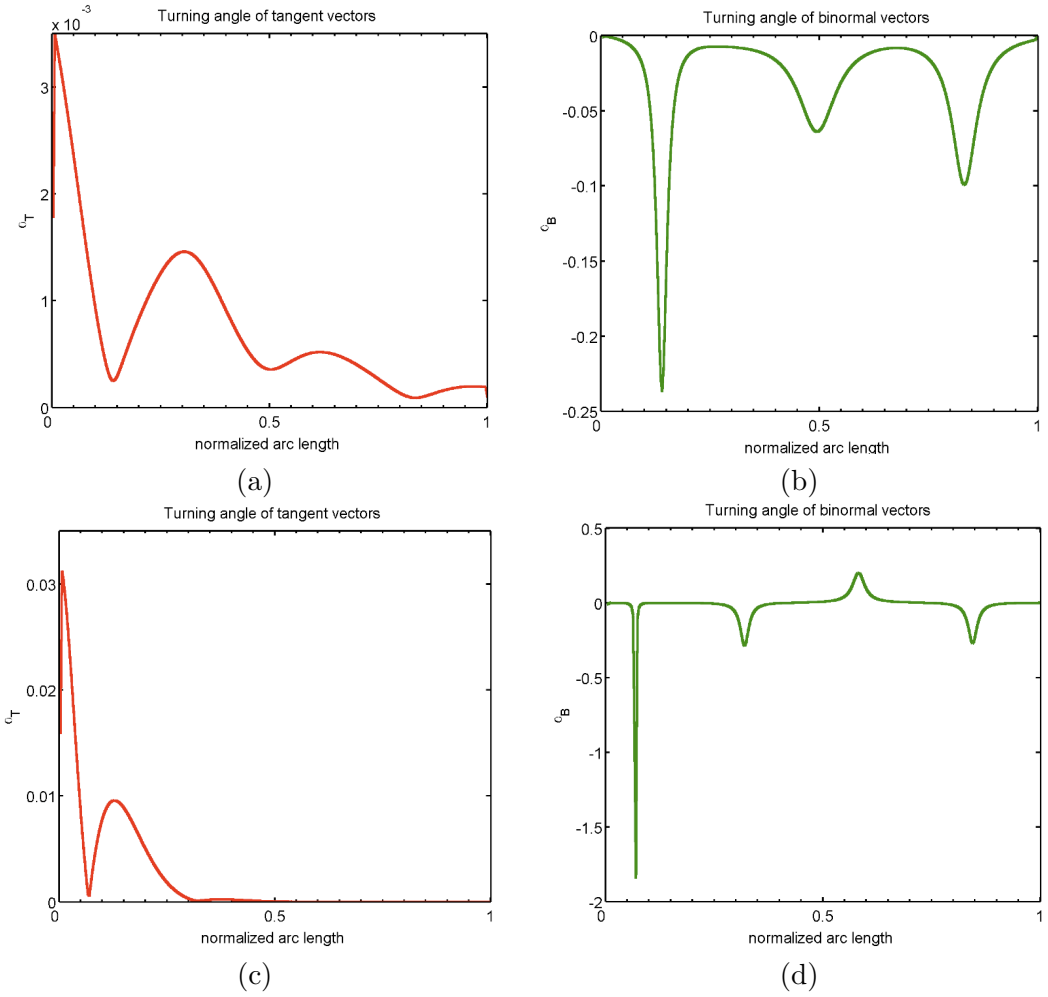


Figure 6.7: Invariants for the space curves of Figure 6.6. (a) and (b) correspond to turning angles of the binormal vectors. (c) and (d) correspond to turning angles of the tangent vectors. In red are the signature curves for $C1$'s family. In green are the ones for $C2$.

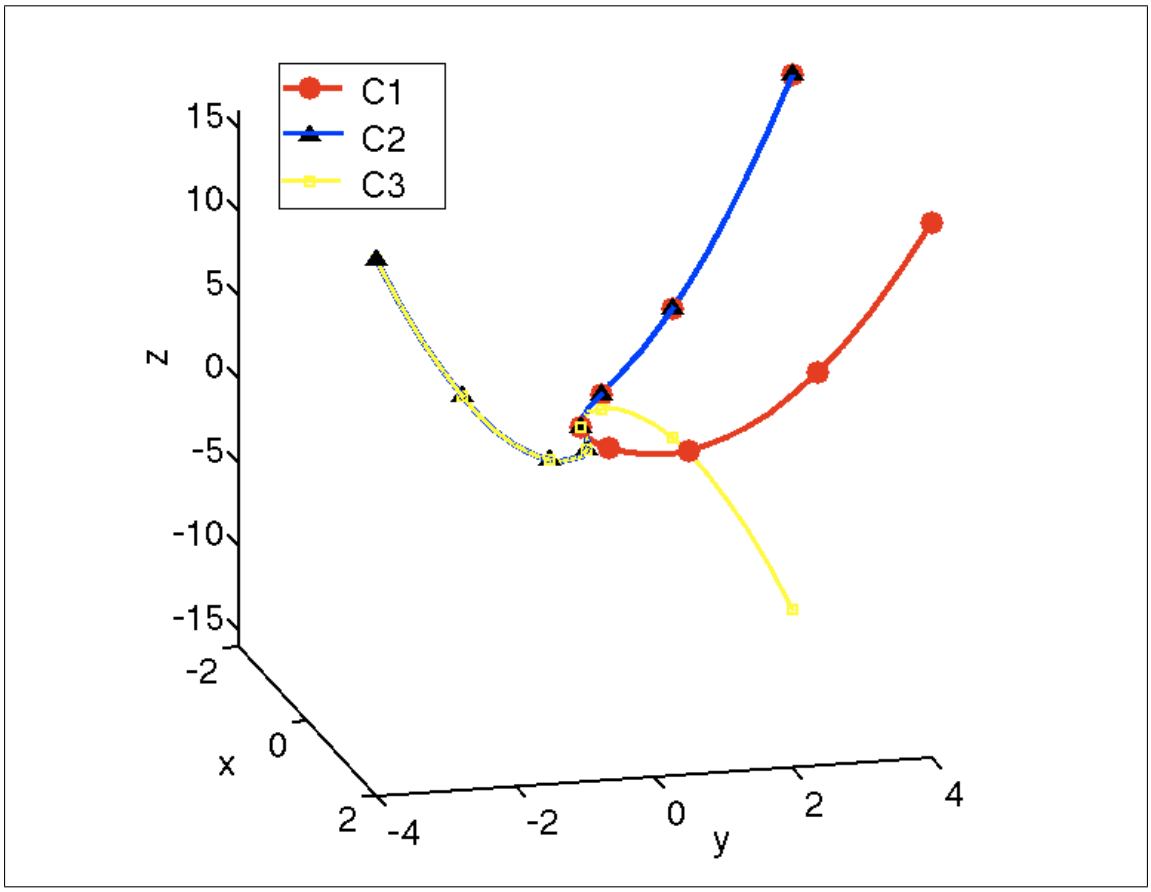


Figure 6.8: Two different sets of synthetic space curves. In (a), T1 and T2 are two similarity transforms. In (b), $C1$, $C2$ and $C3$ are three similar curves except at one inflection point.

curvatures (κ and τ).

6.5.3 Signed angles and their implementation

We test a tricky case where local versus global representations are confronted. We use the new curves $C1$, $C2$ and $C3$ illustrated in Figure 6.8, and defined as follows:

$$C1 \quad : \quad t \in [-\pi, \pi] \rightarrow (t, t^2, t^4); \tag{6.13}$$

$$C2 \quad : \quad t \in [-\pi, \pi] \rightarrow (t, \text{sign}(t) \times t^2, t^4); \tag{6.14}$$

$$C3 \quad : \quad t \in [-\pi, \pi] \rightarrow (t, \text{sign}(t) \times t^2, \text{sign}(t) \times t^4); \tag{6.15}$$

with

$$\text{sign}(t) = \begin{cases} 1 & \text{iff} \quad t \geqslant 0; \\ -1 & \text{otherwise.} \end{cases}$$

Using the turning angles as defined in Section 6.3, we find the turning angles represented in Figure 6.9.

Since we have the analytical expression of the curves $C1, C2$ and $C3$, we may compute their curvature functions $\kappa 1(t)$, $\kappa 2(t)$, and $\kappa 3(t)$, respectively, as well as their torsion functions $\tau 1(t), \tau 2(t)$, and $\tau 3(t)$, respectively. We, thus, find

$$\kappa 1(t) = \frac{\|(16t^3, 12t^2, 2)\|}{\|(1, 2t, 4t^3)\|^3}; \tag{6.16}$$

$$\tau 1(t) = \frac{48t}{\|(16t^3, -12t^2, 2)\|^2}; \tag{6.17}$$

and

$$\kappa 1(t) = \kappa 2(t) = \kappa 3(t);; \tag{6.18}$$

$$\tau 1(t) = -\text{sign}(t) \times \tau 2(t) = \text{sign}(t) \times \tau 3(t). \tag{6.19}$$

We notice that these curves differ in their torsion functions. Nevertheless, the turning angles signatures of these curves fail to describe their difference. This phenomenon of local similarity versus global dissimilarity shows the necessity of taking into account the whole shape representation. This is equivalent to talking about a memoryless description versus a description with memory. Therefore, we modify the initial definition of the turning angles α_T and α_B. The idea is to include the sign negative or positive for our turning angles. To that end, we come up with a sign convention for our angles in space. So a Frenet frame at a time t is now considered with respect to the Frenet frame at time $(t - \delta t)$, or in our discretized case at $(t-1)$. Going through the computations and simplifications of Appendix B, we present the following definition of the signed turning angles:

$$\alpha_T(t) = \text{sign}(\mathbf{T}(t) \cdot \mathbf{N}(t-1)) \arccos(\mathbf{T}(t) \cdot \mathbf{T}(t-1)); \tag{6.20}$$

$$\alpha_B(t) = \text{sign}(\mathbf{B}(t) \cdot \mathbf{T}(t-1)) \arccos(\mathbf{B}(t) \cdot \mathbf{B}(t-1)). \tag{6.21}$$

We present in Figure 6.10 the turning angles computed using (6.20) and (6.21). In Figure 6.10 (a), we confirm the results of (6.18) and observe a perfect correspondence between the turning angles corresponding to $\kappa 1$, $\kappa 2$, and $\kappa 3$. We recall that the difference between

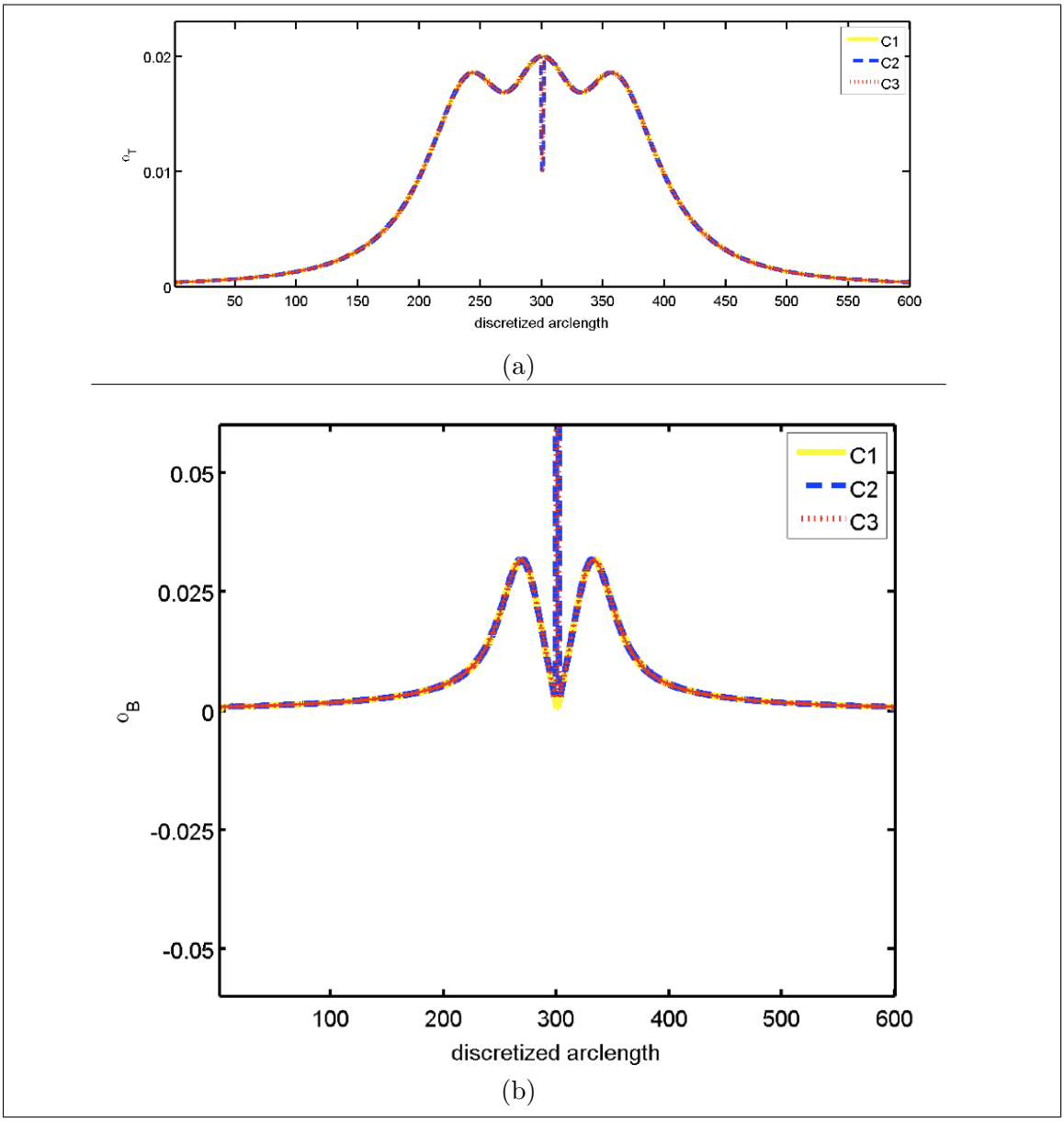


Figure 6.9: Non signed turning angles for the space curves of Figure 6.8.

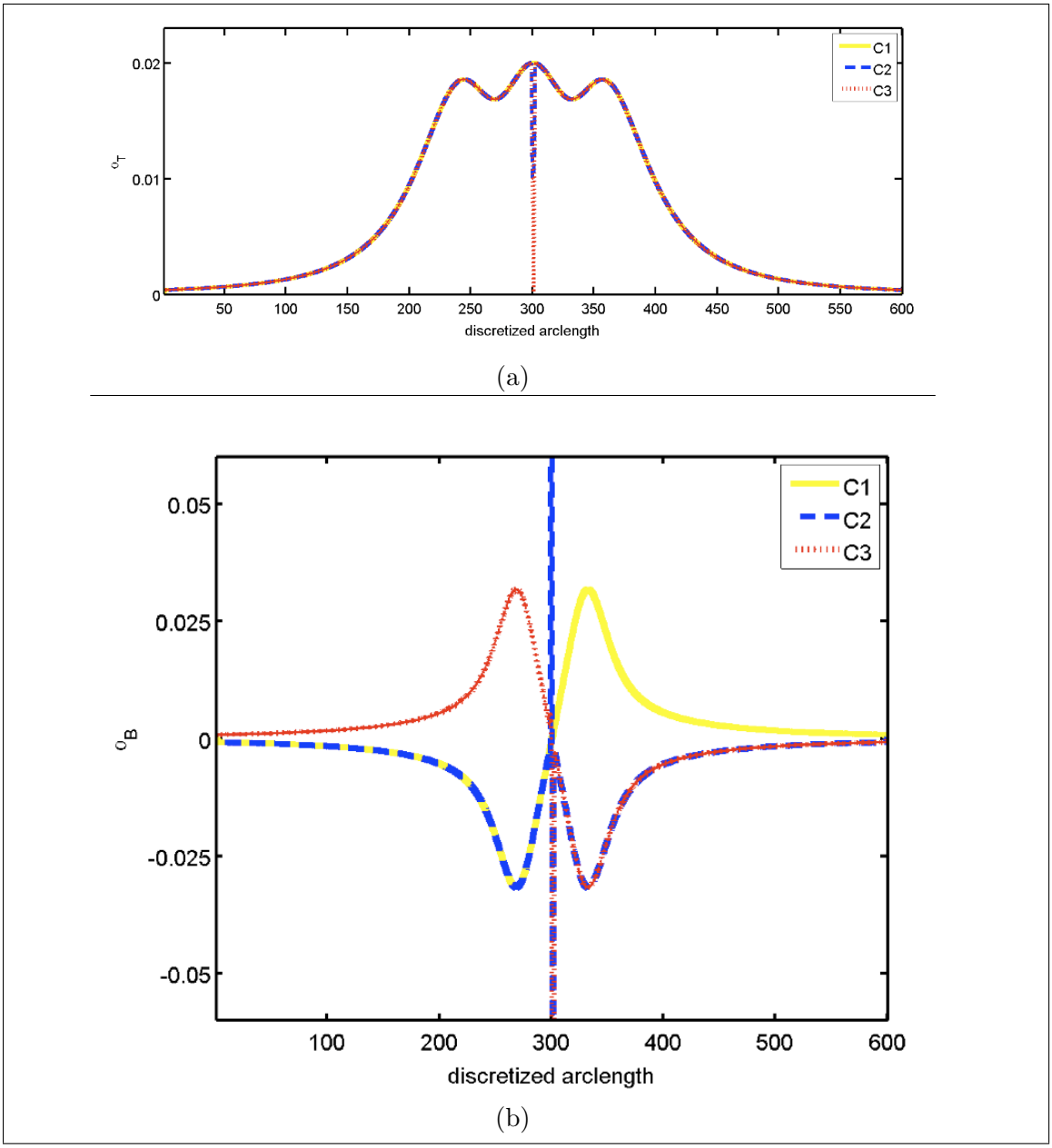


Figure 6.10: Turning angles for the space curves of Figure 4 (b).

the three curves is in the sign of their α_T functions at different semi-open intervals. Using the formulation in (6.21), we are able to see, experimentally, in Figure 6.10 (b), overlaps

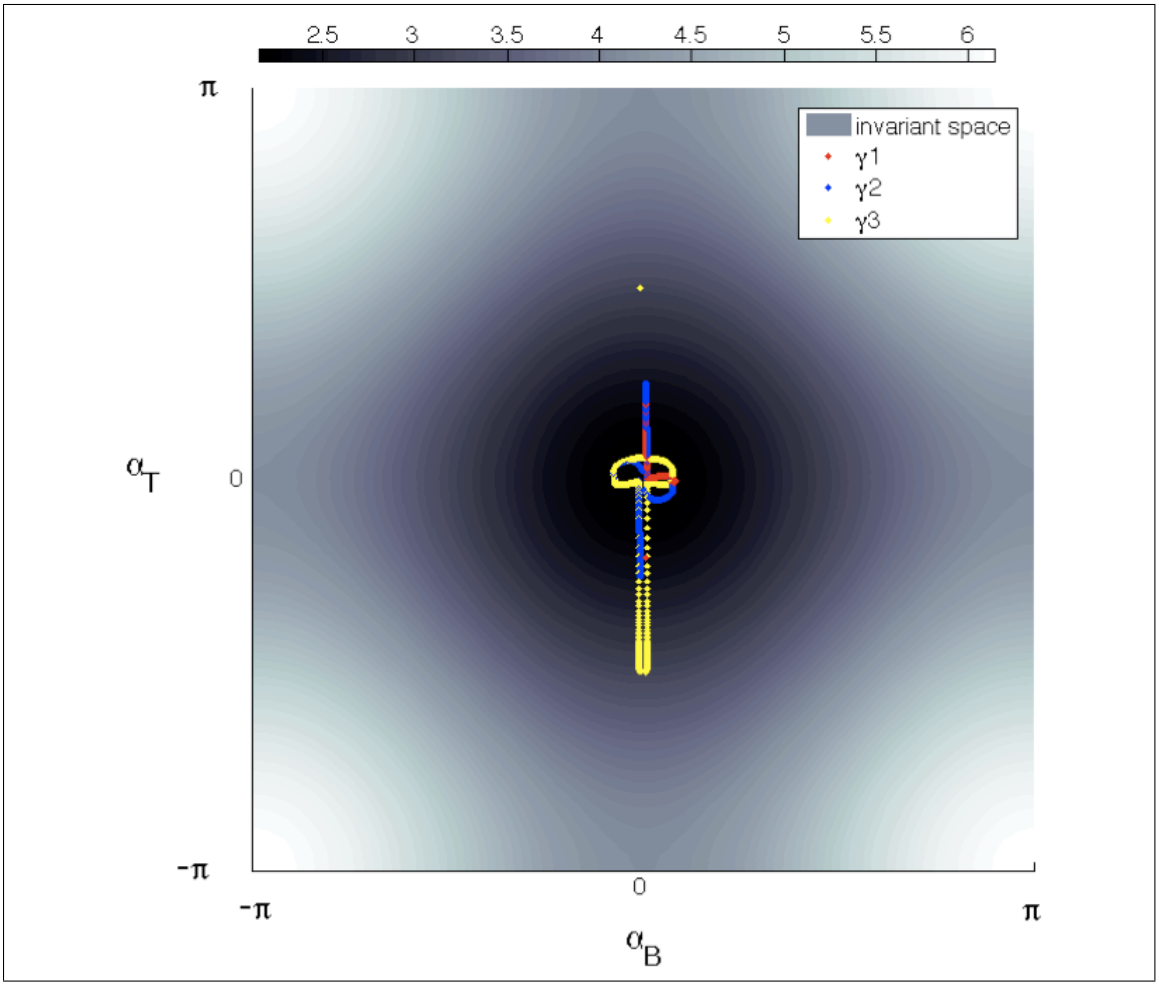


Figure 6.11: Invariant signature curves for 3D modeling curves $\gamma 1$, $\gamma 2$ and $\gamma 3$; the same curves appearing in Figure 6.5.

and symmetries between positive and negative parts of the turning angles. This observation perfectly complies with the result derived in (6.21). This change in the sign of torsion functions translates the effect of inflection points on the curves $C1$, $C2$, and $C3$. Actually, what we represented in Figure 6.10, are the marginals of the binary distribution presented in (6.10). The actual signature curves are the surprisal of this distribution. Their accurate representation should be on the curved space defined in (6.11). We show, in Figure 6.11, the signature curves, for $C1$, $C2$, and $C3$, sitting on this space. In Figure 6.10 (a) and (b), we observe overlaps and symmetries between some parts of the turning angles. This

observation exactly translates what is happening at the curves level because of the effect of the inflection points. The actual signature curves sitting on a subset of the space defined in (6.11).

6.6 Comparison of invariants

The choice of the metric to use on any model is very important in assessing the accuracy of this one. Indeed, an ineffective metric may lead to a failing model despite the effectiveness of the model itself. For this very reason, we propose to walk through the ideas that led to the final metric that we use in combination with the proposed representation for space curves.

To compare two given curves γ_i and γ_j, we compare their invariants. In this section, we explain the reason behind the transition from (6.8) to (6.11). Why is it that we used the binary distribution instead of the marginals even though the two variables α_T and α_B are assumed to be independent ?

Let us assume that we used the marginals $f_\alpha(\alpha_T)$ and $f_\alpha(\alpha_B)$. In such case, we approach the two turning angles α_T and α_B similarly but separately. As a consequence, we go back to using their corresponding surprisal functions $\theta_T(t)$ and $\theta_B(t)$, respectively, instead of $\theta(t)$. We clearly see that what applies to θ_T applies to θ_B; hence, we proceed by defining a metric for θ_T and expand it to θ_B.

If the curve γ_i is represented by N sampling points, its model becomes a vector Θ_T^i of length N. Each element of this vector $\Theta_T^i(k)$, $k = 1, \cdots, N$, is a realization of θ_T that takes its values from $u([-\pi, \pi])$; a smooth planar curve $\mathcal{N}$ (or a 1-dimensional manifold) as shown in Figure 6.14, and defined as: $u : [-\pi, \pi] \to \mathcal{N} \subset \mathbb{R}^2$.

To compare γ_i and γ_j, γ_j needs to be sampled with exactly the same number of points N. That is γ_j would be assigned a similar vector Θ_T^j. Comparing γ_i and γ_j in a strictly planar case[1] becomes equivalent to comparing the two vectors Θ_T^i and Θ_T^j.

The simplest metric to use would be a mere Euclidean distance between the two vectors Θ_T^i and Θ_T^j; however, it is easy to show that such a distance would be inaccurate in the proposed configuration. Indeed, if we look at the example of Figure 6.12, we find

[1] In 3D, in addition to the information for curvature, one needs to account for the equivalent information for torsion, *i.e.*, some vectors Θ_B^i and Θ_B^j. We only consider the planar case here in order to simplify our explanations.

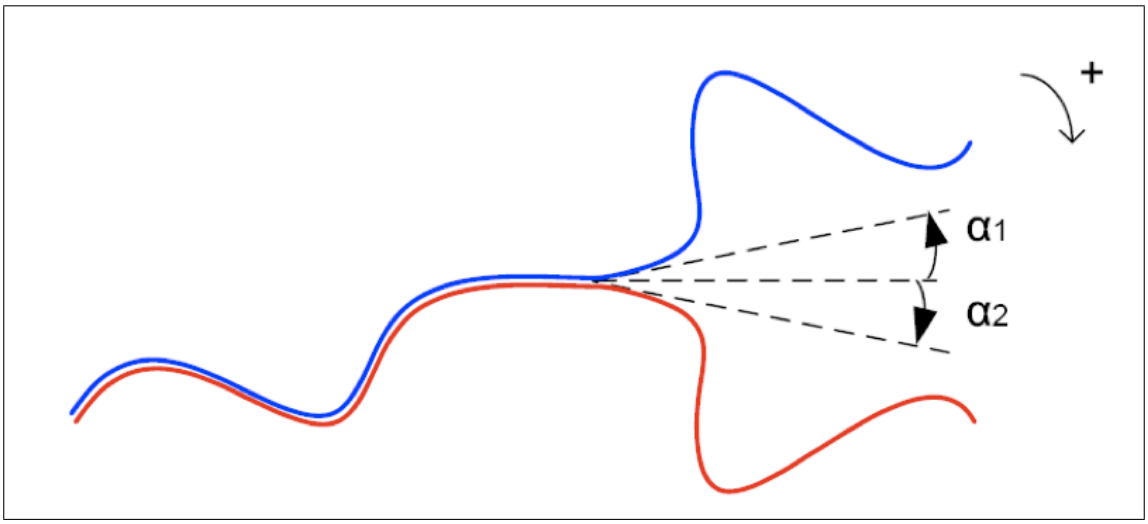


Figure 6.12: Euclidean versus Riemannian metric. We notice that for two turning angles α_1 and α_2 that are the same, but in opposite directions, a Euclidean metric cannot distinguish between their corresponding surprisals.

that for two turning angles α_1 and α_2 such that $\alpha_1 = -\alpha_2$, a Euclidean distance clearly fails to distinguish between these two curves as it would give a null distance. It is therefore necessary to apply a Riemannian metric $\mathfrak{L}(.,.)$ on the manifold $\mathcal{N}$. We may think of using a normalized geodesic distance between two realizations θ_i and θ_j. Specifically:

$$\begin{aligned} \mathfrak{L} &: \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{R}; \\ \mathfrak{L}(\theta_i, \theta j) &= \frac{1}{2\pi} \cdot \int_{t_0}^{t_1} \sqrt{1 + \left(\frac{du(t)}{dt}\right)^2}\, dt, \end{aligned} \tag{6.22}$$

with $u(t_0) = \theta_i$ and $u(t_1) = \theta_j$.

While this metric is more accurate, we still have to ensure that the two curves γ_i and γ_j are sampled with exactly the same number of points. Enforcing this condition may cause correspondence problems as shown in Figure 6.14. One may think to deal with this problem by using multiple paths/comparisons without considering or trying to assign a perfect one-to-one matching between sampling points.

Assuming that we define the metric that does the work for the comparison of Θ_T^i and Θ_T^j, we would use the same metric to compare Θ_B^i and Θ_B^j. A new question would still arise; How do we combine the two metrics $\mathfrak{L}(\Theta_T^i, \Theta_T^i)$ and $\mathfrak{L}(\Theta_B^i, \Theta_B^i)$. What would justify a uniformly weighted average, for instance.

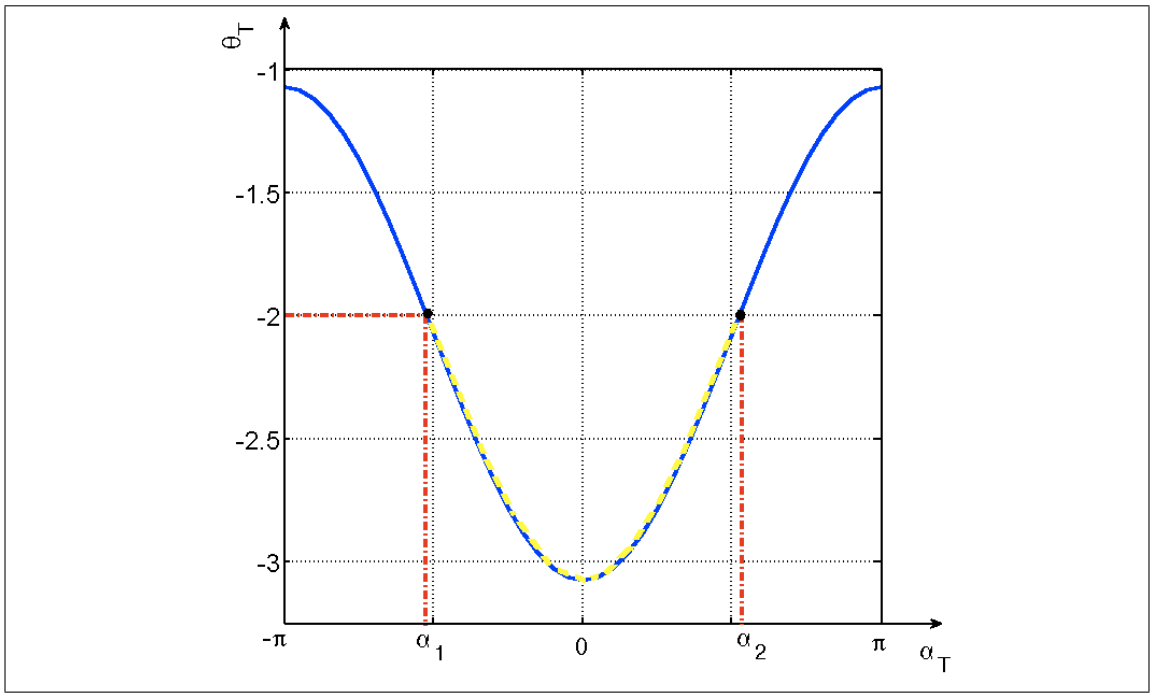


Figure 6.13: Euclidean versus Riemannian metric. We notice that for two turning angles α_1 and α_2 that are the same, but in opposite directions, a Euclidean metric cannot distinguish between their corresponding surprisals.

In order to avoid any arbitrary considerations, and mainly to solve all correspondence problems, we choose to use the binary surprisal function $\theta(t)$ instead of the marginals $\theta_T(t)$ and $\theta_B(t)$. For an accurate comparison of the signature curves, we define a Riemannian metric on the space $\mathcal{T}$ of the invariant. The invariant we defined in Eq. (6.11) is a signature curve embedded in the curved space $\mathcal{T}$ created by the two variables α_T and α_B (Figure 6.5). All the invariant signature curves that we are to compare are thus constrained to live on the defined invariant space that we call $\mathcal{T}$. Defining a space $\mathcal{T}$ that holds all the possible invariants is a natural way to register them, and therefore deal with problems of correspondence between signature curves. As a consequence, we may directly apply a distance measure to compare these invariant curves without worrying about ensuring a prior registration. We thus choose to compare two signature curves $\lambda 1$ and $\lambda 2$, corresponding to two space curves $\gamma 1$ and $\gamma 2$, by considering the oriented curve $\lambda_{\triangle} = \lambda 1 - \lambda 2$. We use tools from measure theory and choose to refer to their physical intuition in relating them to our

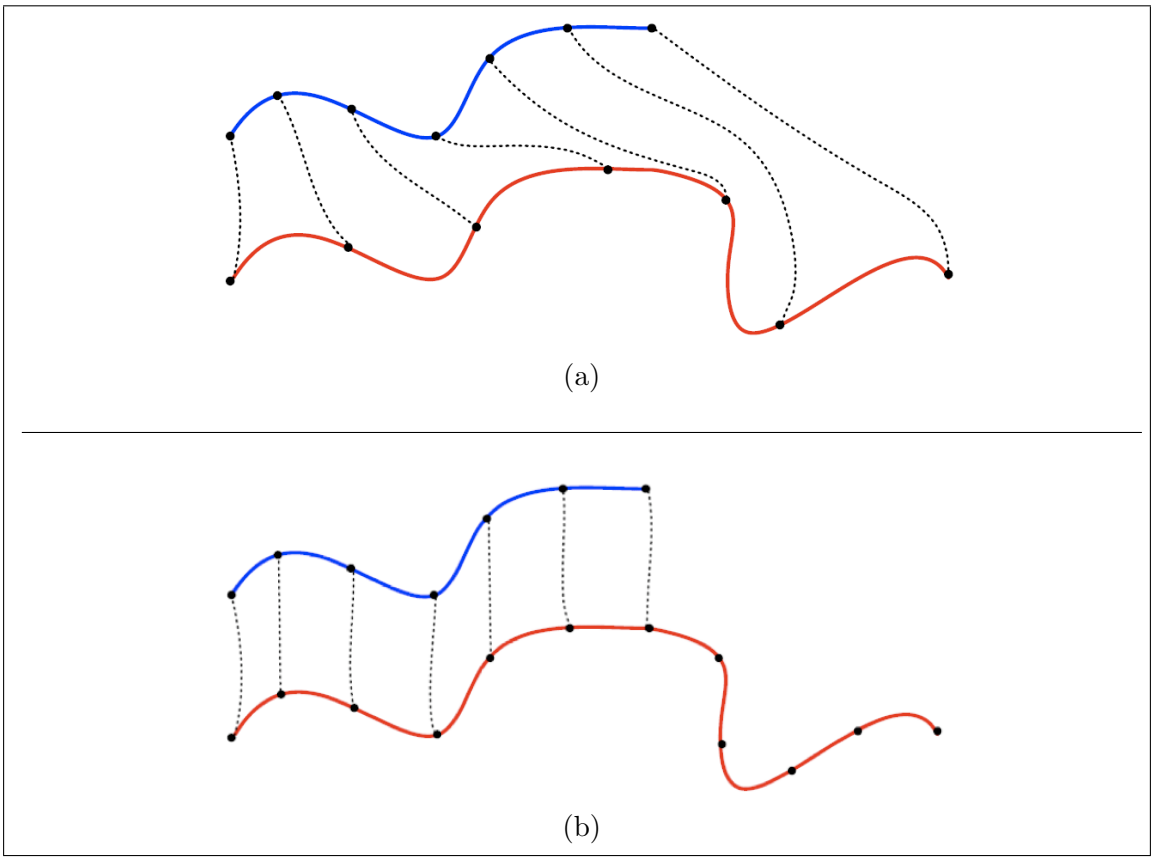


Figure 6.14: Illustration of the correspondence problem

problem [45]. We start by viewing the oriented version of the space $\mathcal{T}$ as a vector field $\overrightarrow{\mathbf{F}}$ on the $(2\pi \times 2\pi)$ plane defined by the variables α_T and α_B. We directly relate $\overrightarrow{\mathbf{F}}$ to $\nabla\theta(t)$, the gradient vector field of $\theta(t)$, and define it as follows:

$$\begin{aligned} \overrightarrow{\mathbf{F}} : ([-\pi, \pi])^2 &\rightarrow \mathbb{R}^2 \\ (\alpha_T, \alpha_B) &\mapsto \sin(\alpha_T)\overrightarrow{i} + \sin(\alpha_B)\overrightarrow{j}. \end{aligned}$$

We also define λ^*_{Δ}; the projection of λ_{Δ} on the $(2\pi)^2$ plane. λ^*_{Δ} is a 1-current in the space dual to the space of 1-forms $\mathcal{D}^1([-\pi, \pi])^2$. This means that if we consider $\overrightarrow{\mathbf{F}}(t) \equiv$

$\overrightarrow{\mathbf{F}}\left(\alpha_T(t), \alpha_B(t)\right)$ and for all ϕ from $\mathcal{D}^1([-\pi, \pi])^2$,

$$\lambda^*_{\Delta}(\phi) := \int_{\lambda^*_{\Delta}} \phi, \tag{6.23}$$

$$= \int_{\lambda^*_{\Delta}} \phi(\overrightarrow{\mathbf{F}}(t))\, dt. \tag{6.24}$$

With these notions of measure theory, we naturally use the flat norm $\mathbb{F}(\gamma^*_{\Delta})$ as the intrinsic distance between two curves $\gamma 1$ and $\gamma 2$ whose invariants are $\lambda 1$ and $\lambda 2$, respectively. We thus may write,

$$\mathbb{D}\Big(\gamma 1, \gamma 2\Big) = \mathbb{F}\Big(\lambda^* 1 - \lambda^* 2\Big), \tag{6.25}$$

$$:= \sup\{\lambda^*_{\Delta}(\phi) : \ \|d\phi\| \leqslant 1 \text{ for all } \|\phi\| \leqslant 1\}, \tag{6.26}$$

where $\lambda^*_{\Delta} = \Big(\lambda^* 1 - \lambda^* 2\Big)$.

6.7 Correspondence preservation

In what follows we prove and simulate different properties that the proposed invariants along with their metric are to verify. We use synthetic space curves and simulate different scenarios.

The invariance of Euclidean transforms and the invariance of parametrization are naturally verified from the properties of curvature and torsion. We also showed the invariance to scaling in Section 6.4. We therefore start by looking at the property of correspondence preservation. Our objective is to show the advantages of registering all the invariant signature curves on the space $\mathcal{T}$. Some of these advantages are: the model's independence of any starting point, correspondence preservation in occluded curves or ensuring a partial matching of curves.

Let us consider two space curves $\gamma 1$ and $\gamma 2$ represented by two invariant curves $\lambda 1$ and $\lambda 2$, and defined as follows:

$$\gamma 1 : [a, b] \to \mathbb{R}^3; \tag{6.27}$$

$$\gamma 2 : [c, d] \to \mathbb{R}^3, \tag{6.28}$$

such that $a \leqslant c \leqslant d \leqslant b$ and $\gamma 1([c, d]) = h(\gamma 2([c, d]))$, where $h(\cdot)$ is a similarity transform. In other words, the curve $\gamma 2$ perfectly matches a portion of $\gamma 1$ after a registration operation,

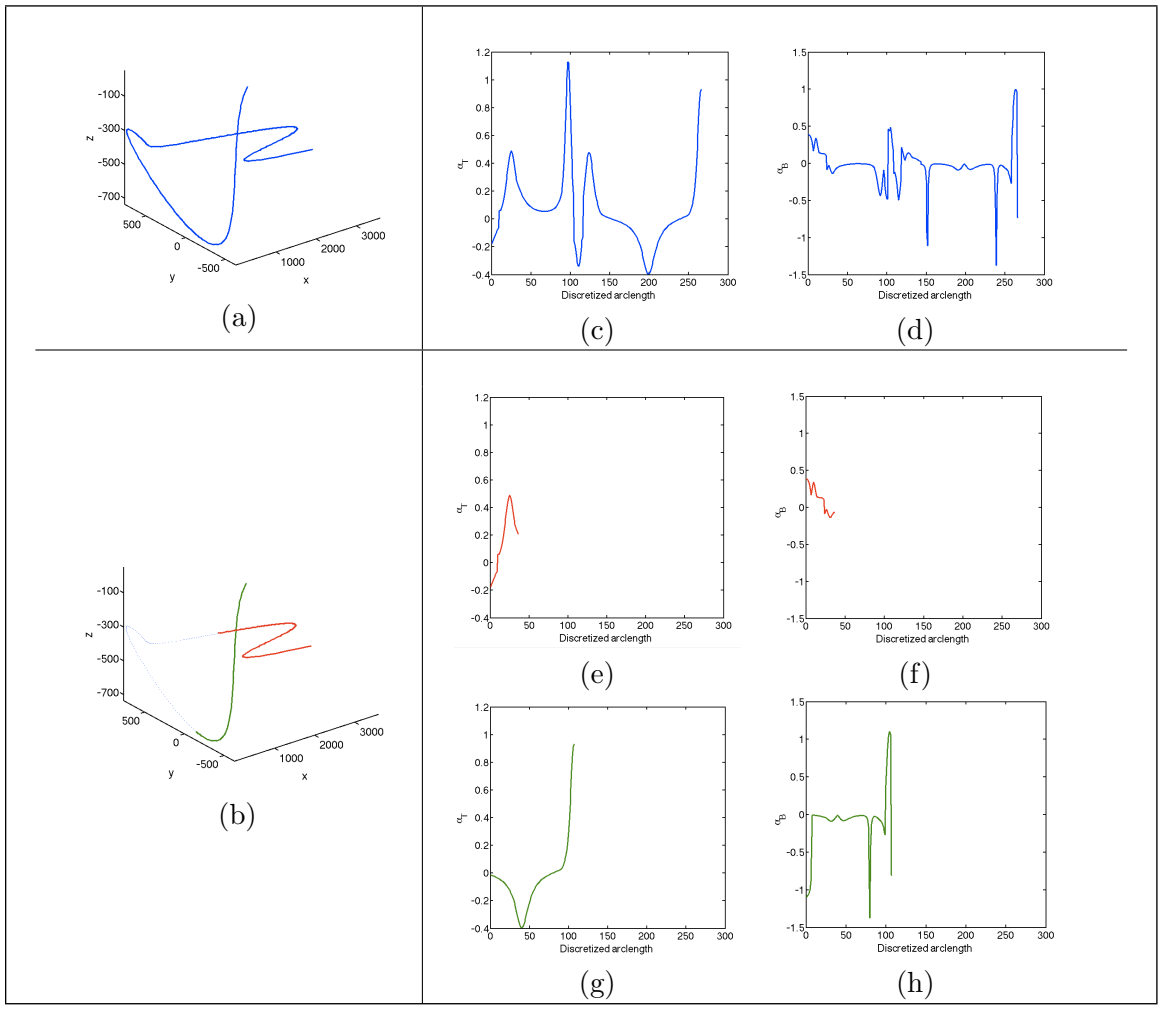


Figure 6.15: Illustration of the problem of occlusion: (a) Initial curve $\gamma 1$. (b) Two curves $\gamma 2$ and $\gamma 3$ resulting from occluding $\gamma 1$. (c), (d) represent the turning angles for $\gamma 1$. (e), (f) are the turning angles for $\gamma 2$, and (g), (h) for $\gamma 3$.

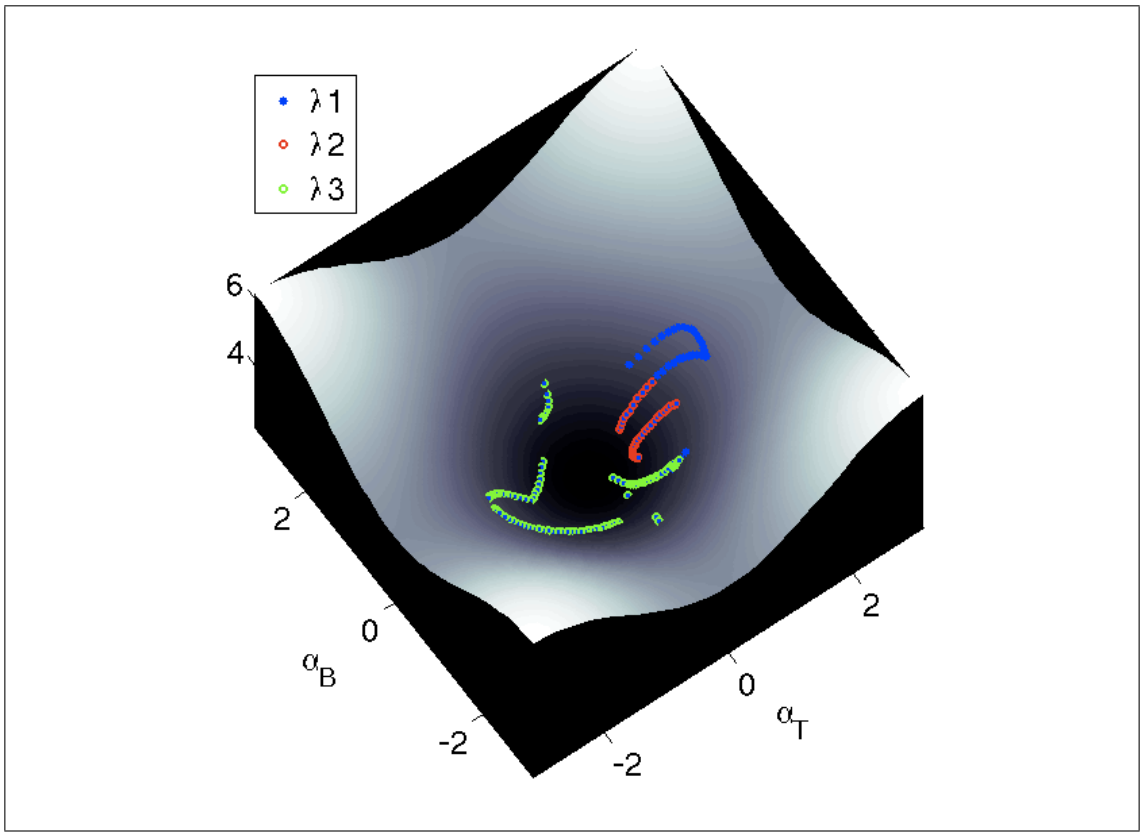


Figure 6.16: Comparison of the 3D objects in (a) and (c) using their squigraph representations in (b) and (d), respectively.

i.e., $h^{-1}(\cdot)$, in $\mathbb{R}^3$.

Since the invariant curves are representations that are locally supported, λ_{Δ} as defined in Eq. (6.23), becomes the invariant curve representing $\gamma 1([a,c])$ and $\gamma 1([b,d])$ that we call $\lambda 1_{[a,c]}$ and $\lambda 1_{[b,d]}$. The distance $\mathbb{D}\Big(\gamma 1, \gamma 2\Big)$ is therefore equivalent to $\mathbb{F}(\lambda 1_{[a,c]}) + \mathbb{F}(\lambda 1_{[b,d]})$. This correspondence problem is very important for shapes or curves that are subjected to occlusion or provided as partial views. Let us for instance consider the curve $\gamma 1$ in Figure 6.15 (a). We show the corresponding turning angles α_T and α_B in Figure 6.15 (c) and (d). In Figure 6.15 (b) we illustrate the same curve $\gamma 1$ but this time subjected to an occlusion, which created two separate curves $\gamma 2$ (in red) and $\gamma 3$ (in green). Computing the turning angles for each one of these two curves requires using two distinct starting points; hence, we find the two sets of signatures (e) and (f) for $\gamma 2$, and (g) and (h) for $\gamma 3$. It is

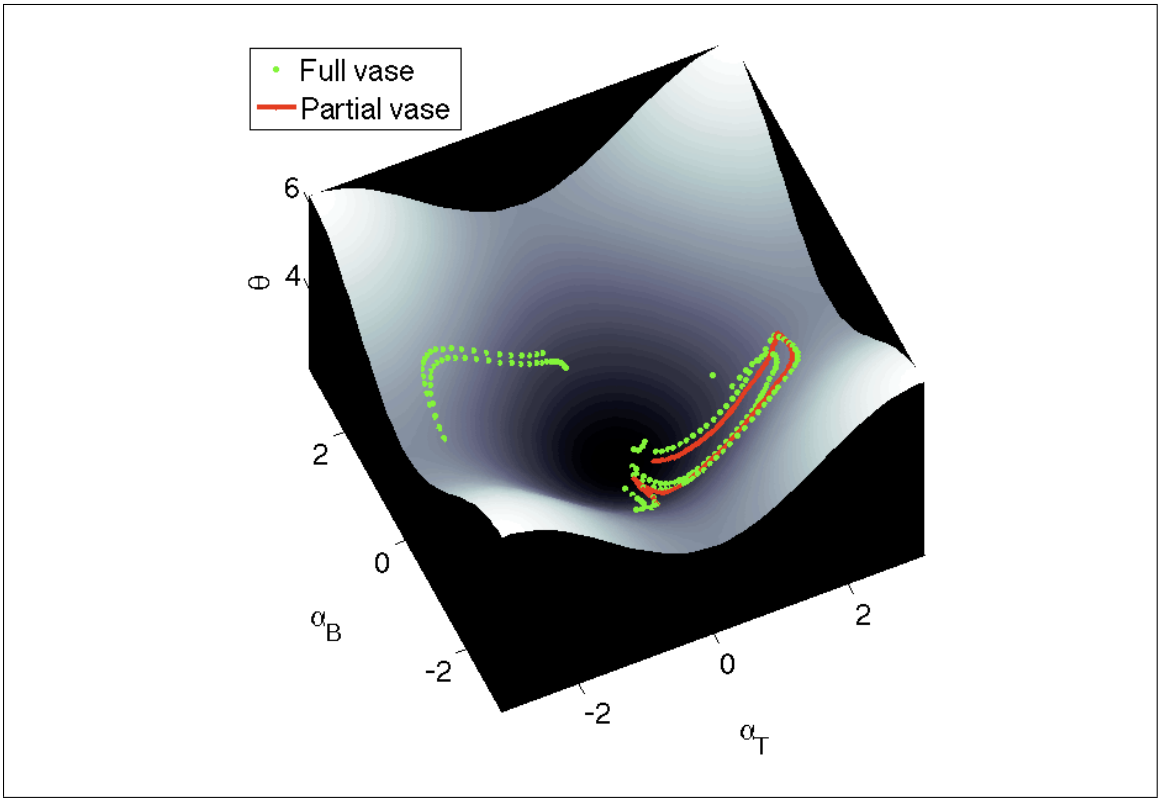


Figure 6.17: Invariant for partial matching.

clearly difficult to directly match the signatures of $\gamma 1$ with those corresponding to $\gamma 3$; we face a correspondence problem caused by occluded parts of a curve.
The problem becomes less challenging when using the invariant space $\mathcal{T}$ as a way to register the signatures. Indeed, Figure 6.16 presents the three invariant signature curves $\lambda 1$, $\lambda 2$, and $\lambda 3$ respectively representing $\gamma 1$, $\gamma 2$ and $\gamma 3$.

6.8 Partial matching of primitive shapes

In what follows we give an example of an partial matching of 3D objects (Figure 6.18). Using the modeling technique proposed in [10], we may reduce the geometry of a primitive 3D object to one curve in $\mathbb{R}^3$ that we refer to as a modeling curve; thus the full vase in Figure 6.18 (a) may be mapped from its sampling (Figure 6.18 (b)) to just one

curve (Figure 6.18 (c)). We may do the same for the top part of the vase. We plot the two resulting modeling curves, for the partial and the full vase, in Figure 6.18 (c). We notice that it is nearly impossible to see any partial matching by simply looking at these curves. Instead, we extract our proposed invariant signature curves. We find, in Figure 6.17, a nice partial matching of these signatures.

6.9 Squigraph comparison

We further apply these signature invariant curves to compare 3D shapes through their squigraph representations. The particularity of these skeletons is that they have spatial curves replacing their edges. These characteristic curves are new means to compactly carry the geometric information of 3D shapes. In Figure 5.2 we illustrate a typical example for which the practical importance of the proposed invariant signature becomes obvious. The 3D shape comparison technique simplifies to comparing the signatures of the edge curves with the same colors in Figure 4.5. For these skeletons, we define a new global metric based on Eq. (6.25). We thus consider that $C1$ and $C2$ are now two sets of curves in 3D such that each set contains N curves C_1^i and C_2^i, $i = 1, \cdots, N$, respectively representing the geometrical shapes of the 3-dimensional parts S_1^i and S_2^i, $i = 1, \cdots, N$, that constitute each 3D object. We define in Eq. (6.29) the new distance between the two sets $C1$ and $C2$, which is also the distance between the corresponding objects $S1$ and $S2$. We show in Chapter 7, the results of using this distance in comparing 3D objects.

$$\mathbb{D}\Big(C1, C2\Big) = \frac{1}{2}\sum_{i=1}^{N}\left(\frac{\text{area}\Big(S_1^i\Big)}{\text{area}\Big(S1\Big)} + \frac{\text{area}\Big(S_2^i\Big)}{\text{area}\Big(S2\Big)}\right)\cdot \mathbb{D}\Big(C_1^i, C_2^i\Big). \tag{6.29}$$

6.10 Conclusion

In this Chapter, we presented a new similarity invariant signature for space curves. This invariant, since based on the tangent and the binormal turning angles, has the advantage of being local, unique and fully invariant to similarity transforms. The proposed invariant proves to be very practical to use in 3D shape modeling/comparison problems.

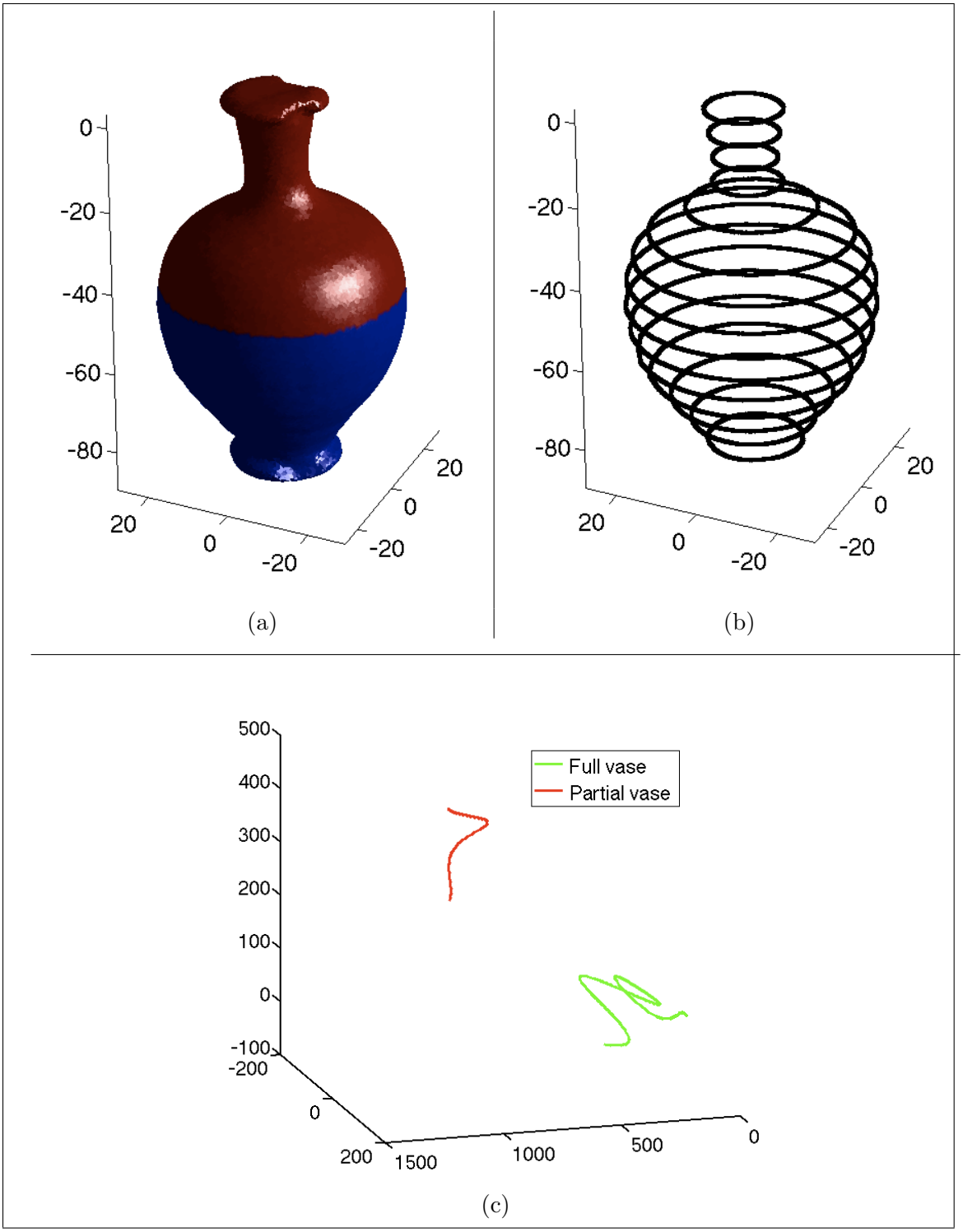


Figure 6.18: Comparison of the 3D objects in (a) and (c) using their squigraph representations in (b) and (d), respectively.

Chapter 7

Experimental Results

We experimentally investigate the performance of the proposed shape modeling technique. Throughout our definition and presentation of squigraphs, we have shown how rich they are in information. In what follows, we illustrate how to exploit this information for different levels of discrimination and for different application objectives; furthermore, we investigate the robustness properties of squigraphs through some pertinent examples.

7.1 Discrimination power

To assess the overall discrimination power of squigraphs, we compare their performance to those of well established approaches. Our comparison involves the following techniques:

- Probability density function (PDF) descriptors: Using the distributions of surface features is a technique that was first clearly defined and analyzed by Funkhouser *et al.* [50]. In our experiments, we use the GGF as the surface feature, and its distribution as the shape descriptor. We choose the *Jensen-Shannon* Divergence (JSD) as the dissimilarity measure between the GGF distributions [33, 48, 49, 50].

- Classification by Characteristic Resolution (CCR) [23, 51]: We have shown in [23] that each class of shapes determines a characteristic resolution[1]. This parameter is extracted through a global comparison of the distribution of the GGF for each class of

[1]Resolution: is the number of vertices used to describe a given shape.

3D objects. In short, we may define the characteristic resolution as being the lowest resolution at which all class members will be accurately represented.

- Augmented Multi-resolution Reeb Graph (aMRG) [2]: This technique is the closest to squigraphs because of its two-level philosophy, and because of the similar Morse function it primarily uses in defining a Reeb graph. aMRG is a technique that has been experimentally established to outperform many 3D classification techniques including the method of spherical harmonics [5].

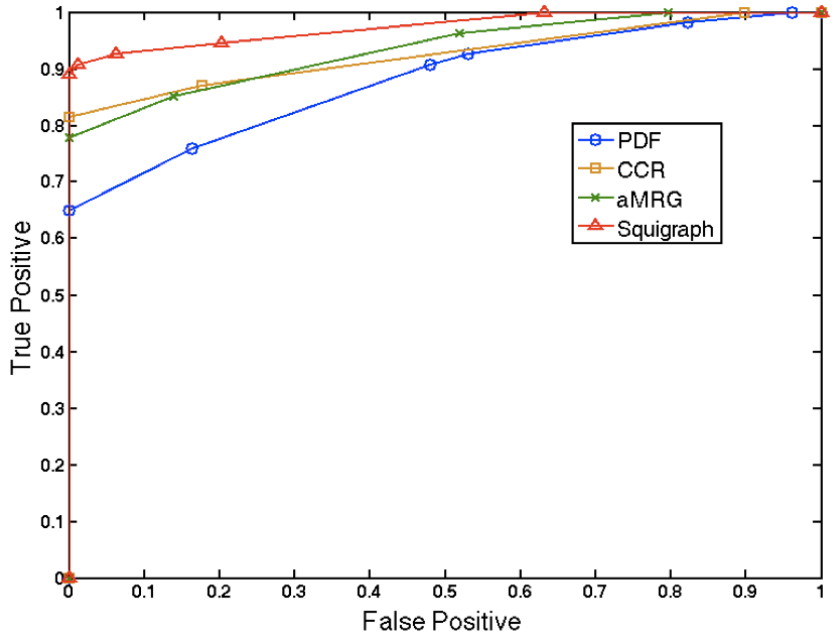


Figure 7.1: ROC curves comparing different classification techniques.

To be able to compare all these different techniques on the same basis we use Receiver Operating Characteristic curves (ROC) as illustrated in Figure 7.1. We then use the Area Under the Curve (AUC) as our measure for classification performance. We use the Princeton [15] and the Technion datasets [52, 53, 54] for a total of 17 classes and 239 objects. The overall performance AUC for each technique is summarized in Table 7.1. With a little more than 97% of overall performance, we note that *squigraphs* outperform other techniques. Doing better than PDF and CCR is quite expected as these two techniques only provide a global

Table 7.1: Overall performance summary.

Method	PDF	CCR	aMRG	Squigraph
AUC	87.95%	92.53%	93.389%	97.35%

description of a complex shape. It becomes more challenging to clearly distinguish between two objects whose global shapes are very similar. As presented earlier in Section 5.1, aMRG and *squigraphs* are multi-level descriptors which enable them to capture global, as well as local features. We explain the better performance of *squigrpahs* over aMRG by the nature of the features extracted locally. Indeed, while both methods extract coarse topological graphs from fully intrinsic representations, *i.e.*, a geodesic distance independent of any reference point, aMRG relies on local features that relate to a Euclidean-based distance. So, the key advantage of the *squigraph* representation technique, that played in the results of Figure 7.1, is to consistently use intrinsic features and geodesic distances.

7.2 Primitive shape analysis

In order to thoroughly understand *squigraphs* and their modeling abilities, we propose, in the following set of experiments, to investigate the different properties a 3D descriptor is required to enjoy. Since the proposed technique proceeds to separately consider homogeneous parts constituting an object, *i.e.*, mono-cardinality subsurfaces, it is intuitively easier to start visualizing the different properties of objects with *primitive geometries*. By primitive geometry, we mean that the corresponding topological graph is just one edge.

7.2.1 Invariance to pose

As noted earlier, given that the shape of an object remains unaltered when subjected to a similarity transform, we require a geometric descriptor to be strictly and completely invariant to similarity transforms. To demonstrate the invariance of a GGF to isometries by way of geometry modeling, we carry out the following experiment. Starting with the shape in Figure 7.2 (a), we apply each of the following transformations to it: (b) rotation, (c) scaling, (d) shearing, and (e) translation. In Figure 7.3 (a), we present the resulting five modeling curves that correspond to each shape of Figure 7.2. To better visualize the differences of the curves in $\mathbb{R}^3$, we propose to project them on the plane as shown

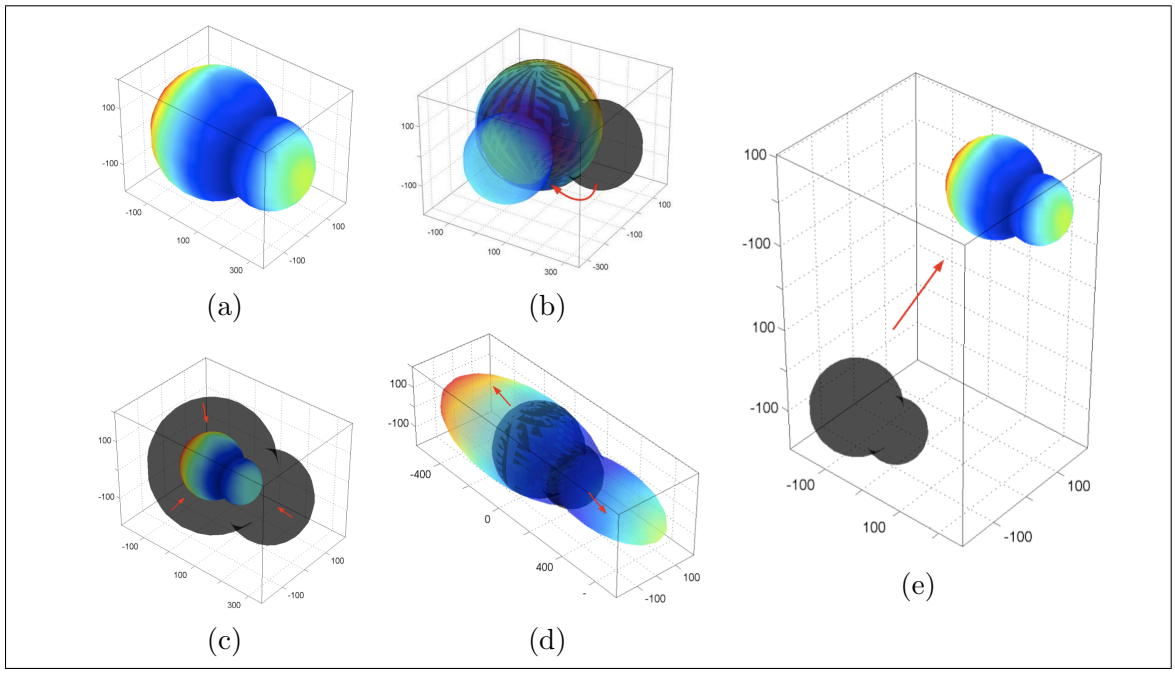


Figure 7.2: Invariance to pose. The original object represented in (a) is subjected to: (b) a rotation transform, (c) a scaling transform, (d) a shearing transform and in (e) to a translation. (Best visualized in color)

in Figure 7.3 (b). We note that all the modeling curves show, up to a similarity transform, a similar variation to that of the curve in black corresponding to the initial shape. This experimental result is consistent with the results of Appendix A.

We say that two shapes are geometrically identical if their modeling curves are identical up to a similarity transform. For this reason, we use the representation of Chapter 6, where the effects of similarity transforms are eliminated and all traces corresponding to the modeling curves are registered. Identical geometries will hence see their traces fully overlap. We present the traces corresponding to this experiment in Figure 7.4. We show the top view of these traces in Figure 7.5. These traces are, by definition, invariant to all similarity transforms. We thus expect to see the result of Figure 7.5, where all the traces overlap except for the one corresponding to the shearing transform; moreover, we find that the distance $\mathbb{D}$ (as defined in Eq. (6.29)) between the original shape and the sheared one is equal to 4.65 while the distance of other shapes away from the original one is between 0.08 and 0.29. Practically, we do not find a null dissimilarity between geometrically similar

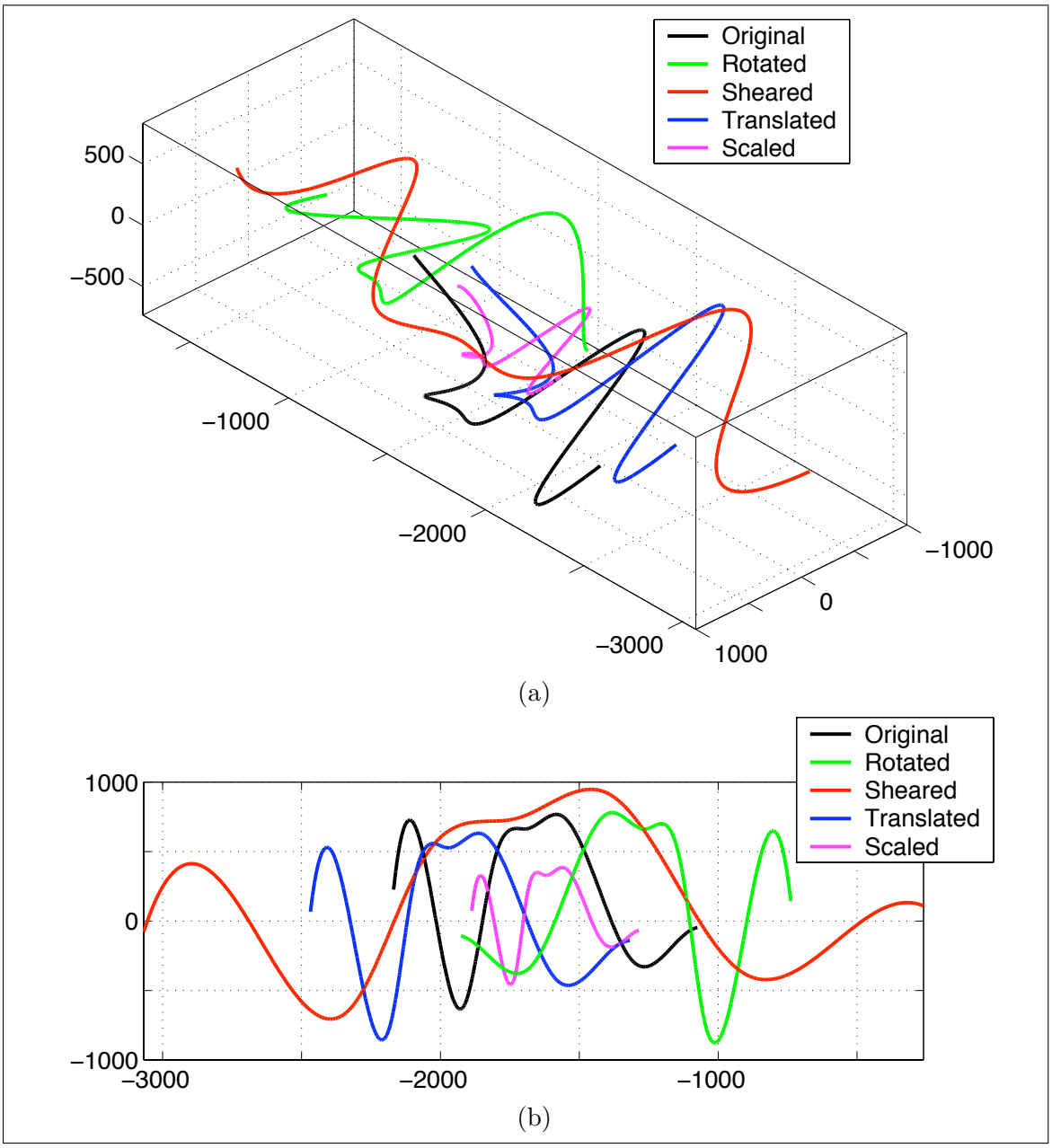


Figure 7.3: Modeling curves corresponding to the objects of Figure 7.2. (a) Modeling curves in space. (b) Projection of the modeling curves in (a) on the horizontal plane.

shapes because of slight differences due to approximations by sampling and the resulting Whitney embeddings.

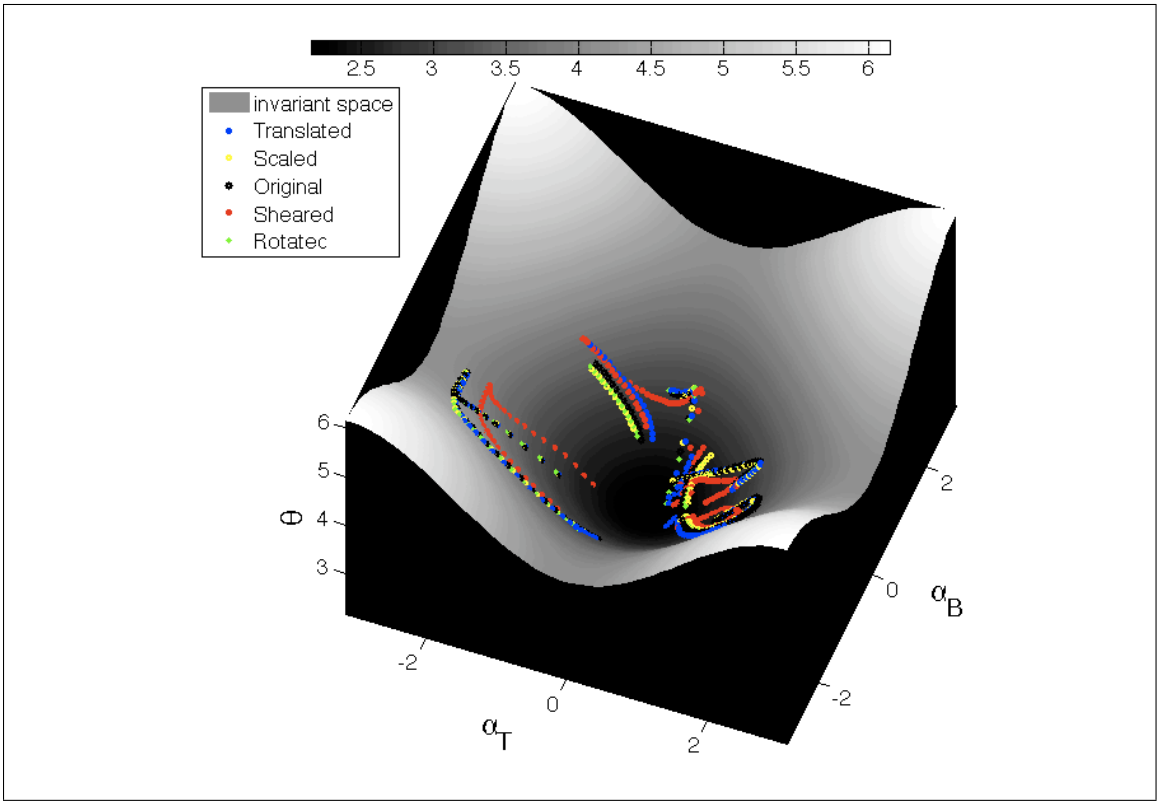


Figure 7.4: Invariant traces for curves in Figure 7.3.

7.2.2 Progressive deformation

In order to analyze the progression of the distance $\mathbb{D}$, as a result of progressively deforming a primitive shape, we use the first ellipsoid starting from left in Figure 7.6 (a). This ellipsoid is defined by the following equation:

$$\frac{(x-x_0)^2}{(r_x)^2} + \frac{(y-y_0)^2}{(r_y)^2} + \frac{(z-z_0)^2}{(s \times r_z)^2} = 1, \tag{7.1}$$

with $(x_0, y_0, z_0) = (0, 0, 0)$, $(r_x, r_y, r_z) = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, and $s = \frac{1}{4}$.

Each time, we apply a vertical shearing transform defined by the matrix $T(s) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & s \end{pmatrix}$, s being the shearing factor. We vary s from $\frac{1}{4}$ to 3. We denote the resulting ellipsoid for a given s by $\mathcal{E}(s)$. We note that $\mathcal{E}(1)$ is the 2-dimensional sphere centered at the origin (0,

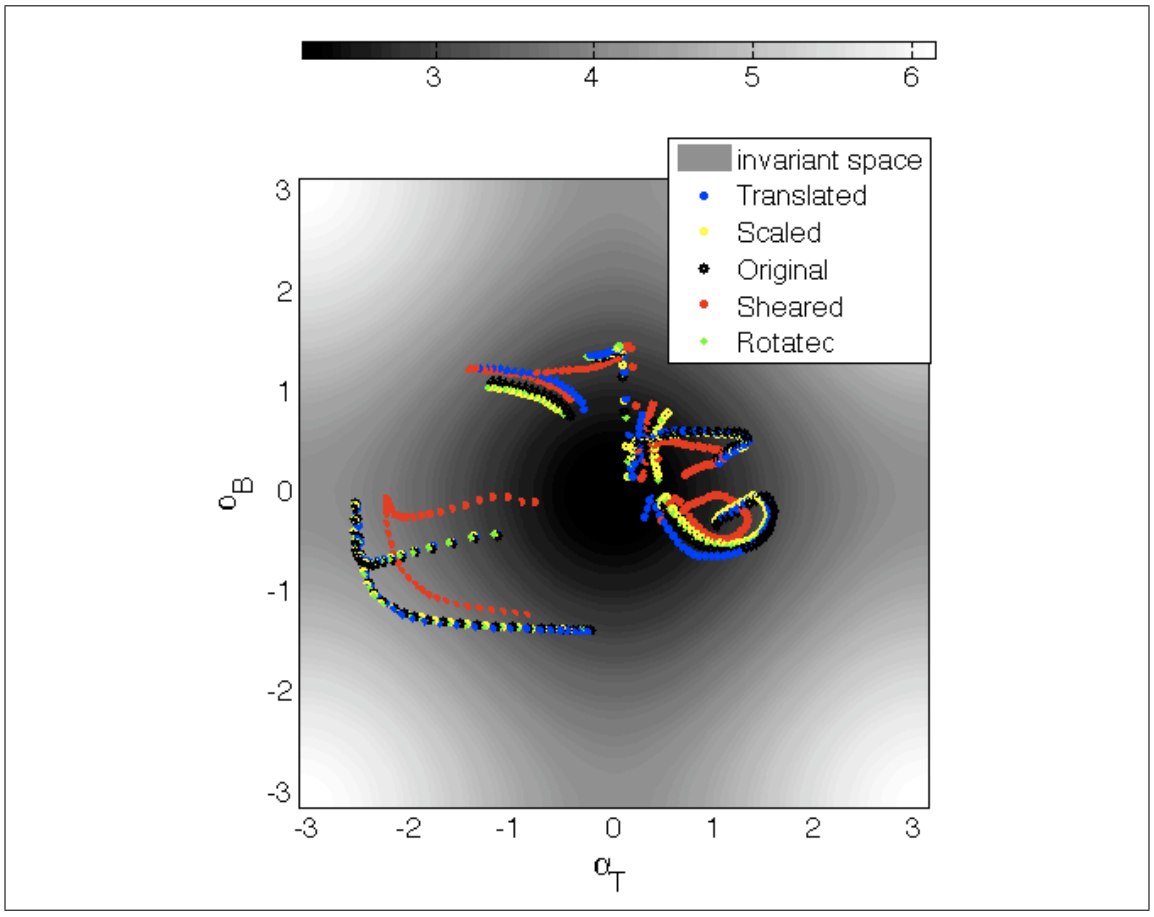


Figure 7.5: Invariant traces for curves in Figure 7.3.

0, 0) with a radius equal to $\frac{1}{2}$. Before proceeding with the analysis of the present experiment, we have to answer a very important question that arises when considering singular shapes such as spheres. We first recall that we only talk about a Morse function when we ensure that all critical points are non-degenerate; thus, the GGF on a spherical shape is not a Morse function. Our proposed solution is to slightly disturb such a singular shape. For a spherical shape for instance, we may disconnect a polar point from other points. As illustrated in Figure 7.7, the distribution of the GGF on a spherical surface quickly changes from a simple dirac function to the distribution shown in Figure 7.7 (d).

We apply this same perturbation technique to all the shapes in Figure 7.6 (a), and extract a modeling curve for each set of iso-geodesic curves in Figure 7.6 (b). Figure 7.8 illustrates

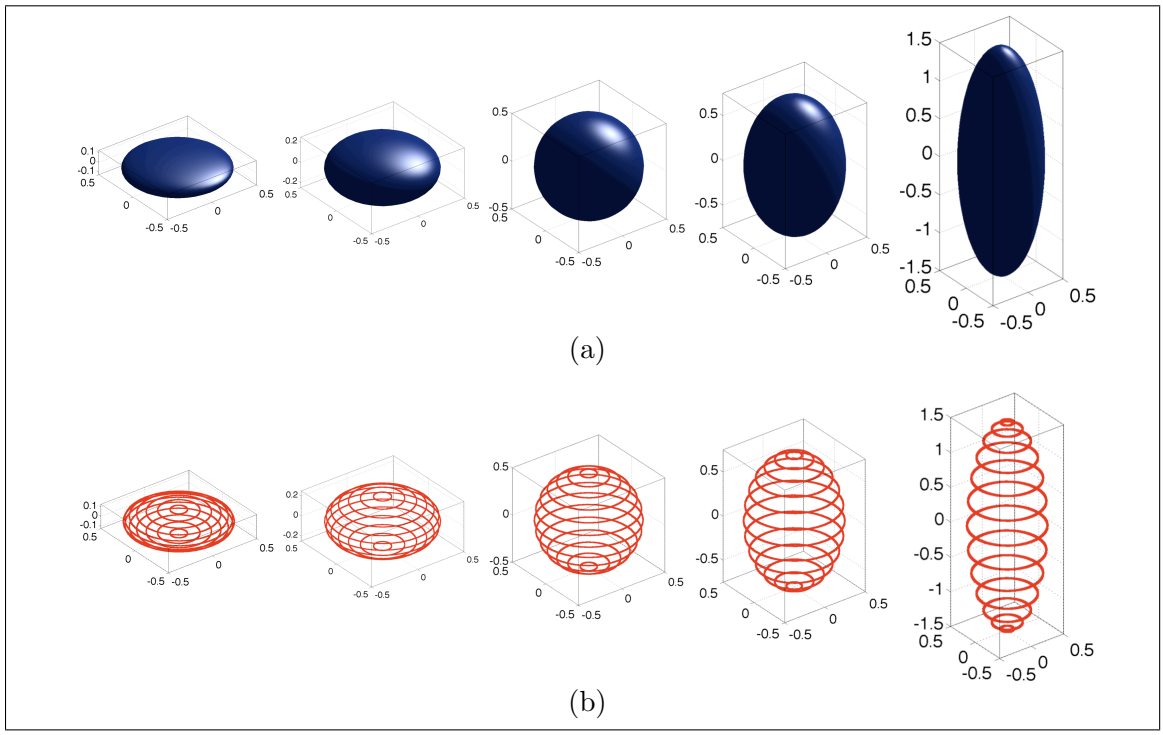


Figure 7.6: Comparison of the geometry of five deformed spheres. (a) shows the 5 considered shapes, and (b) shows the correspondingh spatial modeling curves.

the resulting modeling curves. Our first observation is that these curves are planar, *i.e.*, in $\mathbb{R}^2$, while the proposed Whitney modeling technique is in $\mathbb{R}^3$. This is due to the simplicity of the evolution of the iso-geodesic curves. Very briefly, we state that more variations imply more dimensions.

Focussing on this important aspect of predicting and understanding the modeling curves constitutes a future research direction. Indeed, the present experiment and its results lead to further investigations toward defining the applicability and limitations of the strong Whitney embedding versus the easy Whitney embedding theorem[2].

Our choice of a simple progressive vertical shearing transform is motivated by two points: first, applying a simple directional geometric deformation, and second, being able to quantitatively describe this deformation. In the present example we use the shearing factor s for this quantification; hence, we are able to visualize the variation of $\mathbb{D}$ versus s; thus, the red

[2]Indeed, while the easy Whitney embedding theorem allows embedding an n-dimensional Hausdorff manifold into $\mathbb{R}^{2n+1}$, the strong Whitney embedding allows going lower and defining an embedding in $\mathbb{R}^{2n}$ [38].

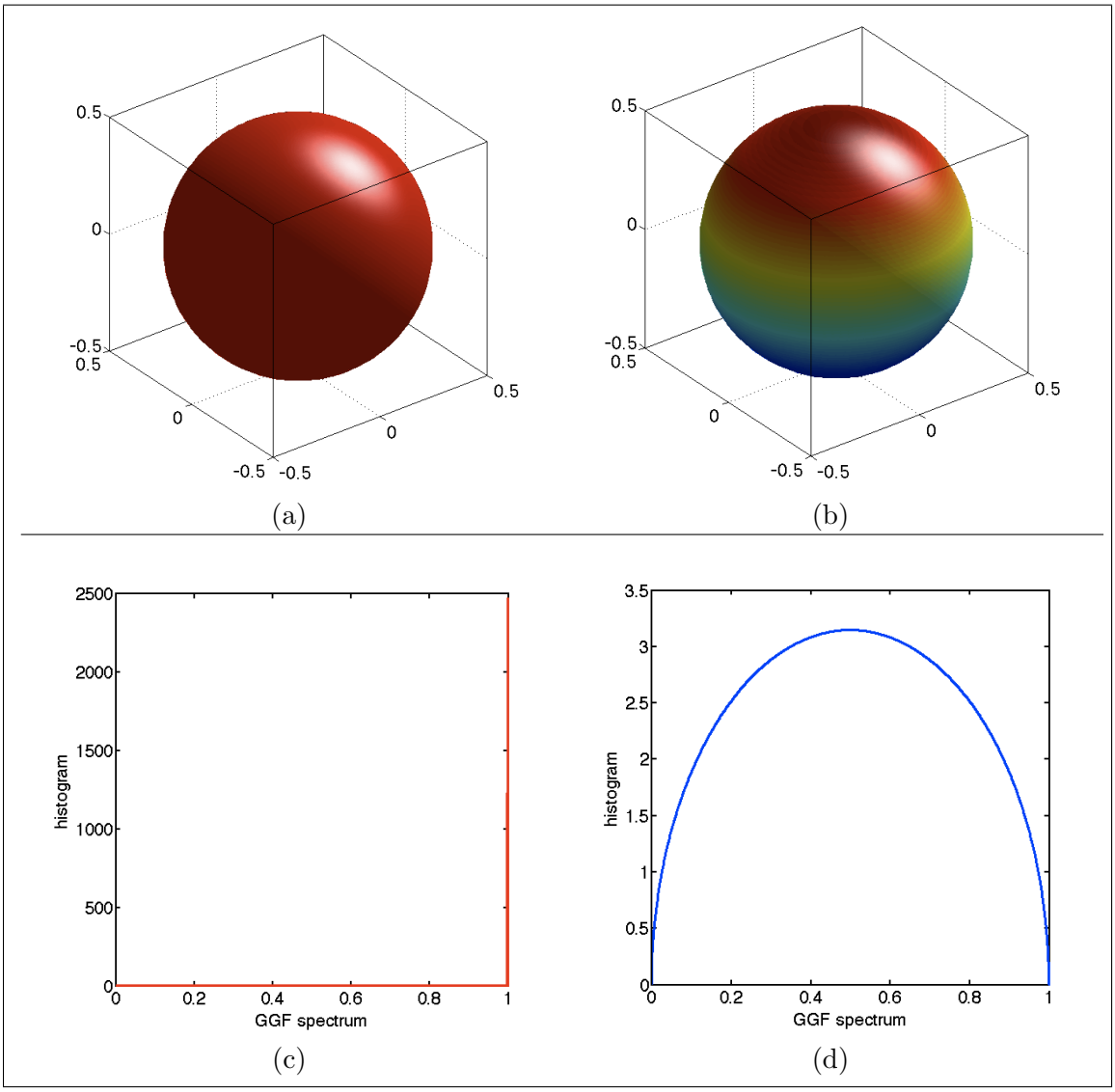


Figure 7.7: The original sphere represented in (a) is disturbed to create a non-degenerate GGF function, whose color mapping is shown in (b). The corresponding distributions of the GGF: (c) Before, and (d) after disturbing the sphere.

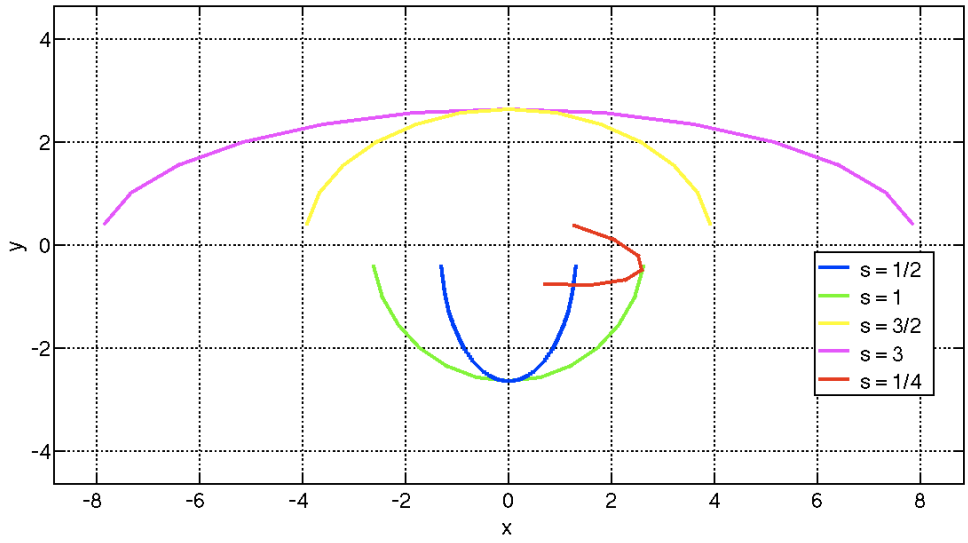


Figure 7.8: Analysis of the effect of shearing transforms on the modeling curves.

full curve in Figure 7.9 shows the smooth evolution of the distance between $\mathcal{E}(\frac{1}{4})$ and other ellipsoids with $\frac{1}{4} < s \leqslant 3$. Taking a sphere as the reference, *i.e.*, $\mathcal{E}(1)$, we may see the effect of the vertical deformation in the two directions $s < 1$ and $s > 1$. The resulting distance curve is illustrated in Figure 7.9 with a dotted blue line. We note that the distance $\mathbb{D}$ follows an exponential behavior as it starts slowing down for large values of Δs. One may explain this tendency by referring back to the human perception. In fact, looking at Figure 7.6 (a), we may classify these ellipsoids into two categories: category 1, corresponding to $s < 1$; and category 2, corresponding to $s > 1$, the sphere ($s = 1$) representing the transition point. We then may say that when an ellipsoid is deformed to be away from its category, the distance $\mathbb{D}$ is relatively high, but once it reaches a new category, additional deformations in the same direction will only add slight distances. We find, for instance, that $\mathbb{D}(\mathcal{E}(2), \mathcal{E}(3)) = 0.045$ while $\mathbb{D}(\mathcal{E}(0.5), \mathcal{E}(1.5)) = 0.22$, that is about 5 times the first distance and for the same Δs. The present result constitutes an important step in understanding the geometry of shapes, and translating the human perception of geometry.

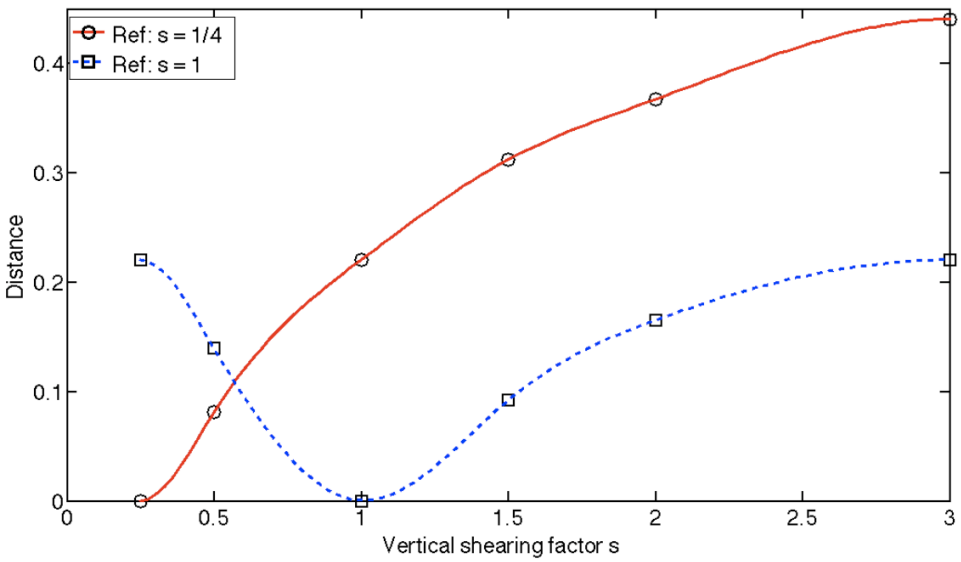


Figure 7.9: Analysis of the effect of shearing transforms on the modeling curves.

7.2.3 Mono-cardinality subsurfaces

In this experiment we continue our observation and analysis of mono-cardinality subsurfaces, but this time we consider more complex shapes found in real-world objects. We thus compare selected parts that constitute the edges of the *squigraphs* extracted for the datasets used in Section 7.1.

We summarize our comparison results with the confusion matrix in Figure 7.10. We note that when legs are bent, the modeling curves are still able to detect a difference between stretched and bent legs. This phenomenon is equivalent to the example of Subsection 6.5.3 where we illustrated how the modeling curves, and *squigraphs* in general, are able to detect an inflection point. Depending on the application, one might think of bent and stretched legs (or arms) as being the same. In this case, we note that, while modeling curves are still able to be lenient on the effect of one inflection point, we recommend for simplicity and efficiency to directly use the distribution of the GGF along each part.

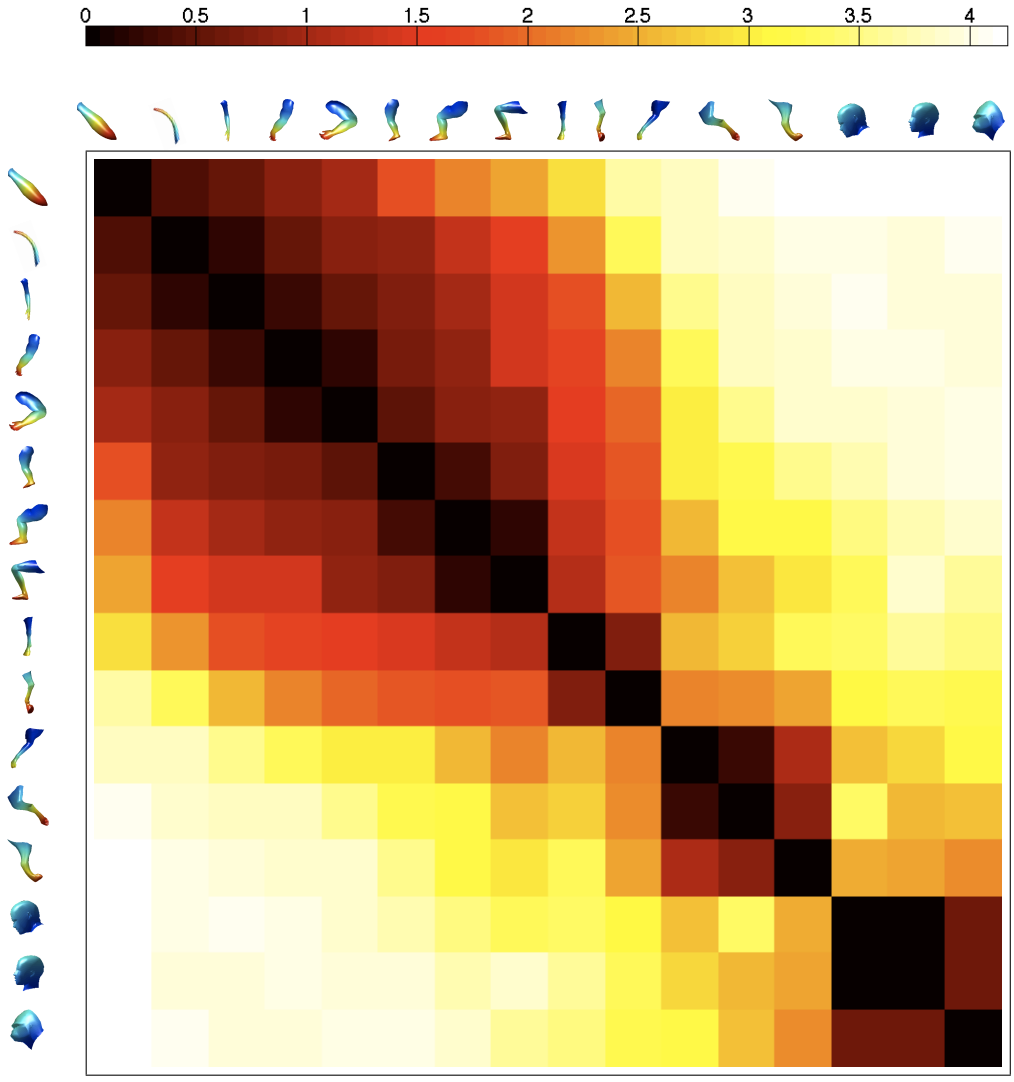


Figure 7.10: Matching and comparison of the geometry of mono-cardinality subsurfaces.

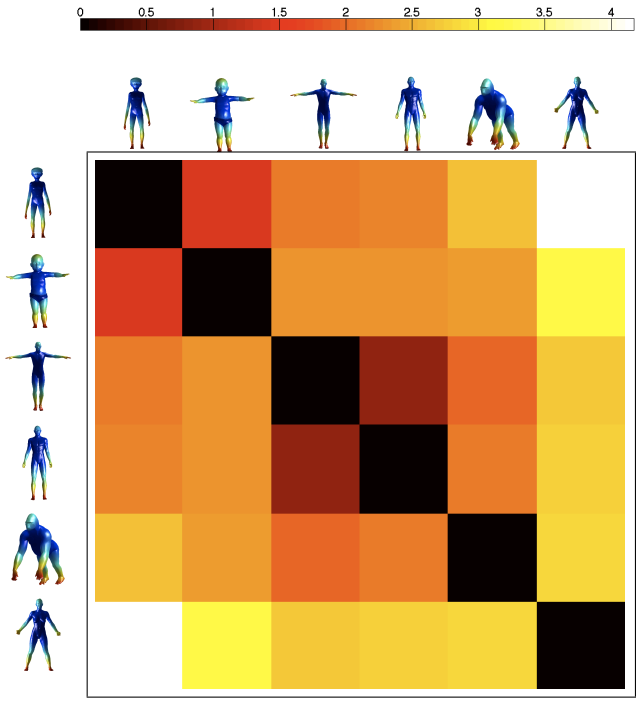


Figure 7.11: Comparison and matching of 6 different bipedal subjects.

7.2.4 Shape matching

The synthesis of the distances between different mono-cardinality subsurfaces provides a global distance for complex shapes as given by Eq. 6.29. We present a couple of relevant examples of confusion matrices for different classes and subclasses of 3D shapes. In Figure 7.11, we note the power of the *squigraph* technique in differentiating between objects that are topologically similar, but who belong to distinct object classes. In Figure 7.12, we present the result of comparing the same object/subject, but in different poses. We note again the ability of *squigraphs* in finding differences due to both rigid and non-rigid transformations. The best way to understand these results is to go back to the analysis of changes that occur on primitive shapes. In this case, it means observing what happens to each part of a human body separately.

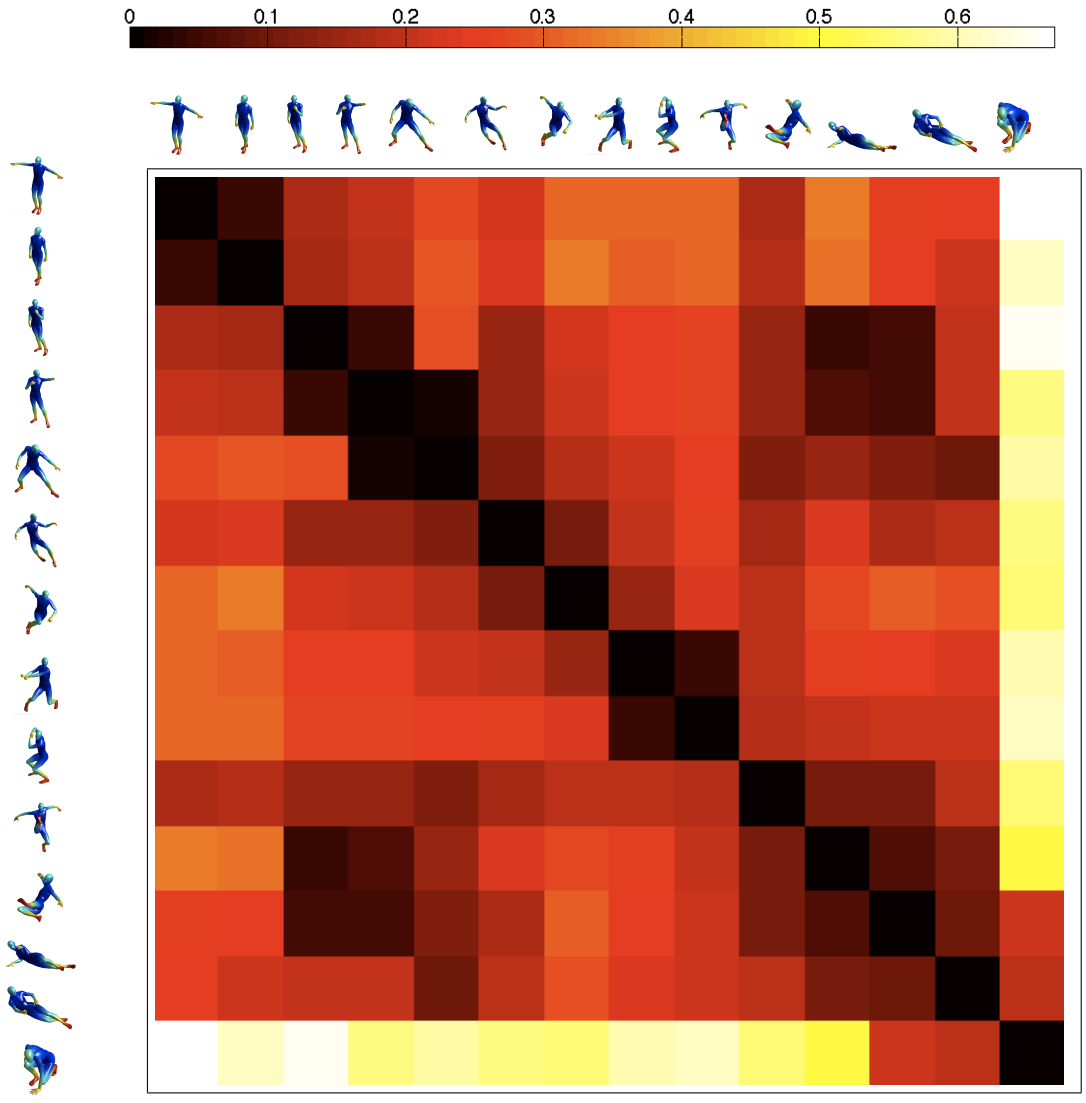


Figure 7.12: Comparison and matching of the 14 different poses for the same subject

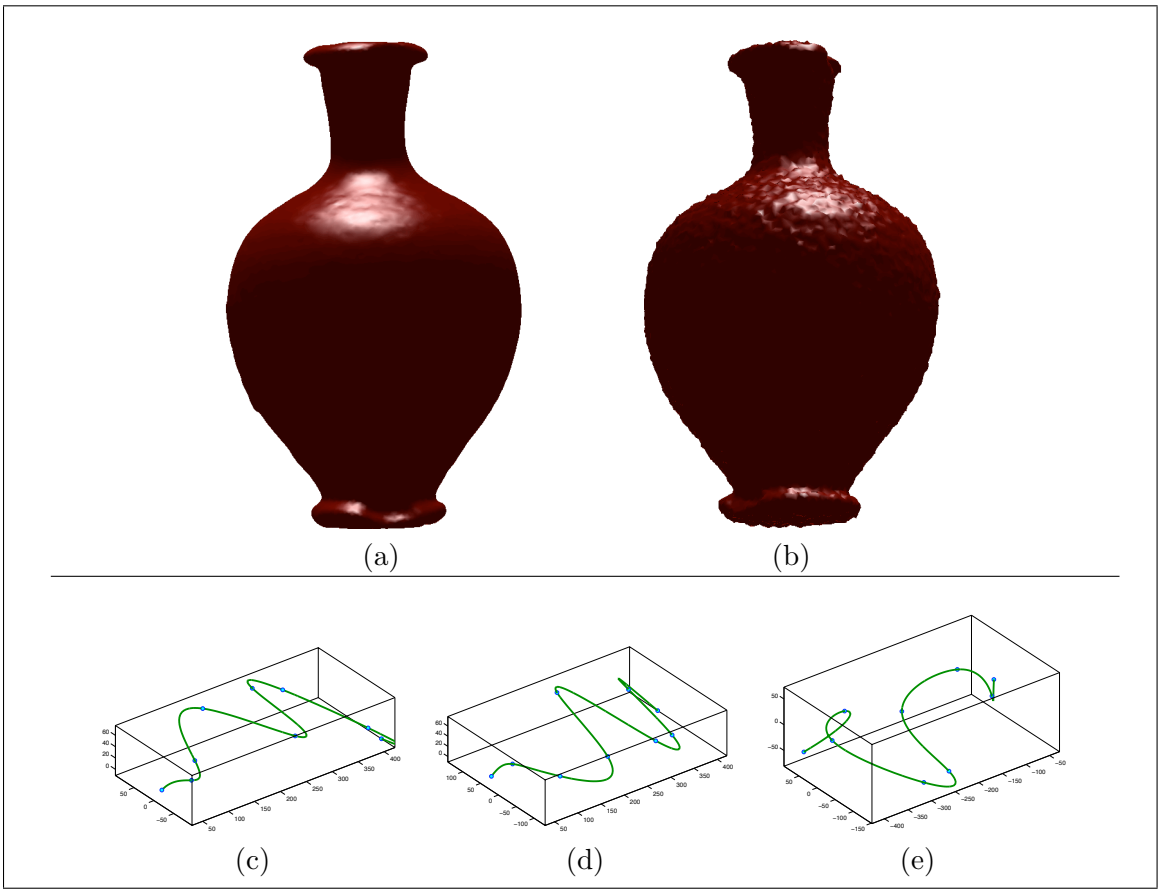


Figure 7.13: Robustness to noise by short secants filtering.

7.3 Robustness evaluation

For the practical applicability of the modeling space curves and the final squigraphs, it is important to evaluate their robustness to both noise and object discrete decimation.

7.3.1 Robustness to noise

As for any practical problem, the measurements, when scanning 3D models, contain additive noise. The secant-based method (described earlier), being directly dependent on the data, raises natural concern about the level of accuracy of the modeling curves in presence of noisy measurements. For testing purposes, we add Gaussian noise with an am-

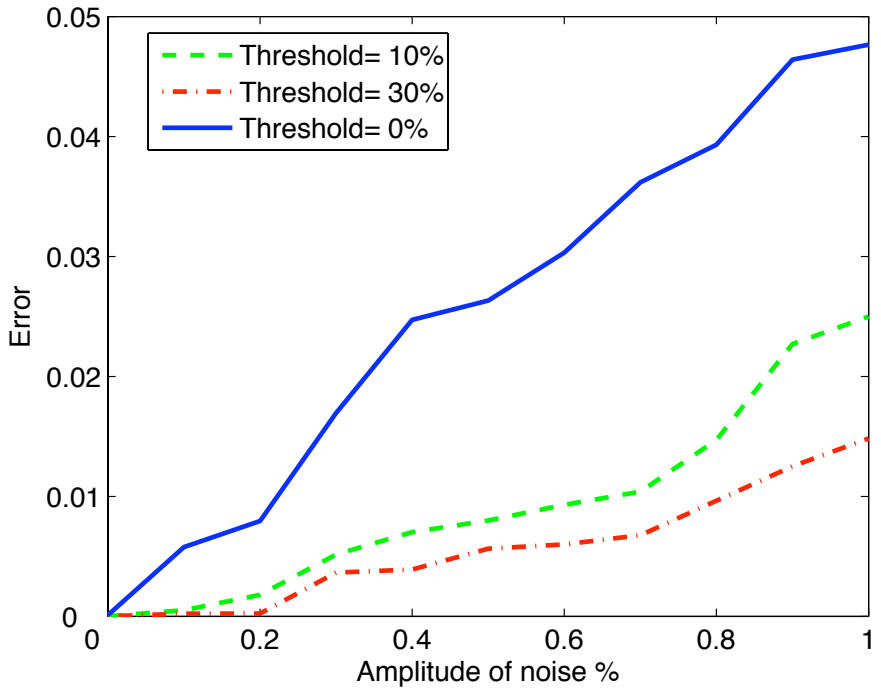


Figure 7.14: Secants filtering with different thresholds

plitude ranging between 0 to 1% of the shape bounding radius for each considered object. Figures 7.13 (a) and (b) illustrate the object "Vase" with and without additive noise. In Figure 7.13 (e), we observe the impact of noise on the modeling curve. The dissimilarity score between the initial shape "Vase" and its noisy version is of 3.08. This basically indicates that the two modeling curves are very different and may induce an erroneous recognition result. Broomhead and Kirby [40] propose a solution to alleviate the effect of noise on embeddings. They show that the shortest secants induce the most severe distortion. We propose, as a result, to filter the shortest secants prior to applying the LTMADS algorithm. We test the effect of filtering out the short secants where the threshold is a fraction of the longest secant. Figure 7.14 illustrates the result of 250 Monte Carlo simulations. It is clear that the errors due to noise are reduced after filtering, with a more dramatic impact for larger thresholds.

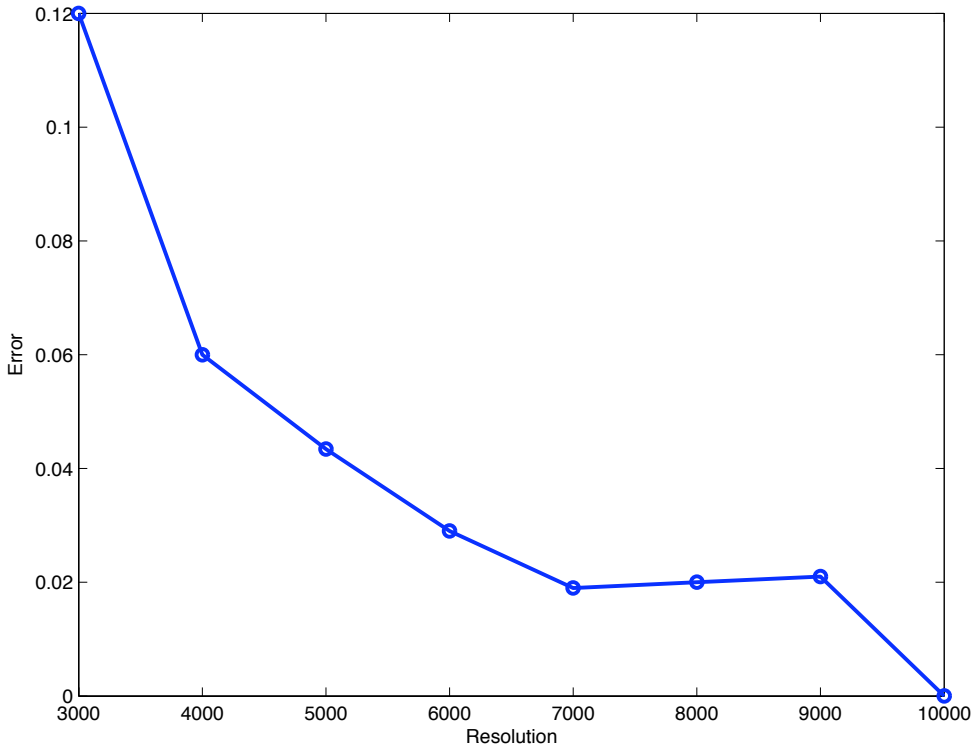


Figure 7.15: Robustness to decimation.

7.3.2 Robustness to decimation

The resolution of a 3D object measurement/scanning may vary with applications, and it is hence important to evaluate our proposed geometric modeling in presence of such variations. To that end, we consider an objet "Frog" to assess the robustness of the modeling to decimation. Taking the highest resolution, of 10000 vertices, as a reference, we measure the dissimilarity induced by a progressive decrease in resolution for object "Frog". We illustrate this experiment in Figure 7.15. As explained in [23], using the GGF, we may assign a particular resolution, to each shape, referred to as a *characteristic resolution*. By definition, the characteristic resolution corresponds to the lowest number of vertices able to accurately represent a considered shape. For the shape "Frog", we find that the characteristic resolution is about 5000 points. Analyzing the results of Figure 7.15, we may conclude that for "Frog", our modeling is accurate for resolutions greater or equal to the

characteristic resolution of 5000 points. Thus, our geometric modeling is consistent and nearly invariant to decimation.

7.4 Conclusion

In this Chapter, we presented a novel and effective geometric modeling technique which significantly reduces a 3D shape representation to a *squigraph*. The first positive results are encouraging to warrant further study and more expanded applications. The overall objective of our work is to develop a comprehensive and efficient classification algorithm which can further translate the human understanding of the geometry of shapes from a local to a global viewpoint.

Chapter 8

Adaptive Embedding and its Application to Network Failure Detection

We define a new adaptive embedding approach for data dimensionality reduction applications. Our technique entails a local learning of the manifold of the initial data, with the objective of defining local distance metrics that take into account the different correlations between the data points. We choose to illustrate the properties of our work on the isomap algorithm. We show through multiple simulations that the new adaptive version of isomap is more robust to noise than the original non-adaptive version; moreover, we use our proposed technique to detect intrusions in sensor networks.

8.1 Introduction

In recent years, data sizes have drastically increased. As a result, there has been a great research focus on improving and defining effective dimension reduction techniques. These efforts are extremely relevant if not crucial to data storage, visualization, and analysis application. The growing demand of storage and archiving resources, together with their inefficient current exploitation, and their increasing cost, have also made this line of research very relevant. The objective behind learning and reducing the dimension of data is to eliminate any redundant information while still preserving the intrinsic and underlying

structure. One may think of this problem as an attempt to find all the variables that may be combined into fewer variables without destroying the interactions between the data points. To formulate this, we proceed as follows.

8.1.1 Problem statement

Given a data point sample of N points X_i, $i = 1, \cdots, N$, from a d-dimensional smooth manifold $\mathcal{M}$, where $\mathcal{M} \subset \mathbb{R}^n$ and $d < n$, in a dimension reduction problem, we aim at finding a homeomorphism $f(\,\cdot\,)$, or the image of $\{X_i\}$ by a homeomorphism $f(\,\cdot\,)$ such that:

$$\begin{aligned} f : \mathcal{M} \subset \mathbb{R}^n &\rightarrow \mathbb{R}^d \\ X_i &\mapsto f(X_i) = Y_i, \ \text{ for } \ i = 1, \cdots, N. \end{aligned} \tag{8.1}$$

The goal is to effectively determine the data $\{Y_i\}$ of reduced dimension d that reflects all the information present in the initial data $\{X_i\}$. Explicitly finding the function $f(\cdot)$ is however, not required in dimension reduction applications, as the approach is in spirit non-parametric and data driven. We may distinguish two classes of dimension reduction approaches; The first class includes all the classical methods, or linear methods, such as Principal Component Analysis (PCA) [55], and Multidimensional Scaling (MDS) [56]. In contrast, the second class corresponds to non-linear techniques [58], also referred to as manifold learning methods. The linear algorithms are well known and fully understood. They are often used to preprocess data before proceeding to further analysis. The fundamental idea of PCA, a widely used linear algorithm, is to only keep predominant dimensions of the data. Geometrically speaking, this corresponds to linearly projecting the data on the vectors of highest variances; however, this only works if the data has a linear, or close to linear, structure. Any nonlinearity in the data can only be accounted for by more general methods extending beyond linear structure. The second class of manifold-based algorithms tends to find the same predominant feature axes using more relaxed assumptions. The common assumption of these more powerful techniques, is that all the data samples live on a Riemannian manifold embedded in a high dimensional Euclidean space. These techniques have proven so far, to be the most successful approaches [57]. When the source of data has a relatively few degrees of freedom, it becomes possible to reduce the dimensionality of this data while keeping its inherent structure. In fact, a lot of real world applications fall

within this class of underlying lower dimensional systems which may easily be formulated in a manifold learning setting. Such applications include face recognition [68], face pose detection [65], gait analysis [66], and human motion data interpolation [67]. There are about four widely known manifold learning techniques; Locally Linear Embedding (LLE) [59], Laplacian eigenmap [60, 61], Hessian eigenmap [62], and isomap [63]. Manifold learning algorithms always assume the observed cloud of data points as part of a smooth manifold. To proceed with the analysis of this data, we start by constructing a graph connecting all the data points and preserving the structure of the manifold. One usually defines an $\epsilon-$ball neighborhood of fixed radius around each data-point to carry out the analysis. All these techniques have shown very successful results in ideal conditions; nevertheless, there is very limited work in addressing the effect of noise and the choice of the neighborhood size. Very simple experiments may show how crucial it is to take these considerations into account.

Our goal in this work is to address the noise problem, and propose a way to develop a new manifold-learning technique, with a built in robustness to noise. The key idea is to replace the arbitrary choice of an approximate Euclidean distance, and to instead use a locally adaptive distance. To achieve that, we propose to account for sample data points' correlations in defining their neighborhood. We follow the approach of Carlsson *et. al* in addressing the problem of an "adequate" neighborhood size [72]. So, to go beyond the one neighborhood size which achieves some particular embedding, and to gain further insight, we argue that, geometrically and topologically, there is more interest to discover by evolving a neighborhood size. We demonstrate in addition the flexibility of this approach in a sensor network intrusion detection application.

The remainder of the paper is organized as follows: Section 8.1.2 is an overview of the different existing approaches to robust manifold embedding techniques, with more emphasis on the *isomap* algorithm. In Section 8.2, we discuss the classical isomap algorithm which, in contrast to our proposed technique, is non-adaptive. In Section 8.3, we describe in detail our proposed adaptive method. We evaluate the benefit of an adaptive/non-adaptive isomap in Section 8.4, using a residual covariance as a performance measure. We illustrate our proposed technique by analyzing synthetic data in Section 8.5, and demonstrate its wider applicability by solving a sensor network failure detection.

8.1.2 Related work

Many of the existing manifold learning techniques show successful results on some well chosen data; and they all share a limiting failure when in presence of more challenging datasets. The difficulties are often due to the *intrinsic topological* and *geometric structure* of the manifold with quick variations in their curvature and non-convex boundaries [69]. Additional difficulties result from the properties of the real data such as the *sampling distribution* and the *nature and level of the prevailing noise*.

To address existing limitations and to further improve the embedding results, and extend the applicability of the current manifold learning techniques, we propose to account for these overlooked characteristics. The idea is to progressively adapt to the data at hand in tracing the local connectivity between the point samples. Some recent efforts have explored adaptive manifold learning, by specifically focusing on two parameters: the *intrinsic dimensionality* of the data, and the *size of the neighborhood*. Wang *et al.* propose in [69] a method to adaptively select the neighborhood size. They base their technique on determining the alignment space of local tangents. Cost *et al.* on the other hand define an intrinsic dimensionality using K-nearest neighbors graphs [70]. In [71], Levina *et al.* adopt a local estimate for the intrinsic dimension at each point.

Our present work also focusses on an adaptive embedding of the manifold; we, in contrast, point out an additional characteristic that appears to be as critical as the choice of the embedding dimension or the neighborhood size. We indeed show in what follows, that the choice of a Euclidean distance is suboptimal in determining the local connectivity between data points, and therefore, introduce a new adaptive distance locally defined for each point.

8.2 Non-linear manifold searching techniques

As noted earlier, our proposed effort builds on existing manifold embedding techniques; therefore, we start by recalling the preliminary steps of a non-linear manifold learning technique.

8.2.1 Isometric feature mapping (Isomap)

Among the multiple manifold learning algorithms, we choose to use the *isomap* algorithm to illustrate our ideas. This choice is due to the *isomap* success in numerous

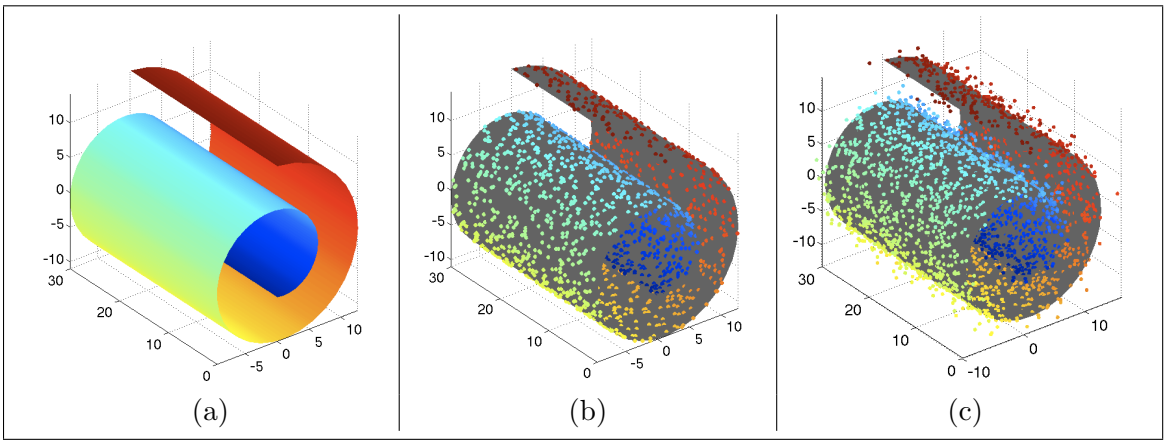


Figure 8.1: Different phases of isomap.

embedding problems and its well established properties [63]. The principle and motivations of this work are, nevertheless, extendable to other mapping (embedding) algorithms.

Isomap is a nonlinear mapping algorithm which starts with the assumption that the high dimensional data lie on a Riemannian manifold [64]. To achieve the dimension reduction, isomap defines a mapping that aims to preserve the geodesic distances on the initial manifold. We may describe isomap as merely an improved version of Multidimensional Scaling (MDS) embedding where the interpoint distance is a geodesic, i.e., restricted to lie on the initial manifold of the data.

Concretely, for a data point sample of N points X_i, $i = 1, \cdots, N$, from a d-dimensional manifold $\mathcal{M}$, where $\mathcal{M} \subset \mathbb{R}^n$ and $d < n$, we detail the different steps of the *isomap* embedding algorithm in Table 8.1.

8.2.2 Local connectivity graph

The very key idea of *isomap* is to consider geodesic distances along the manifold of data. To practically approximate the intrinsic geodesic distance on a manifold, we need to locally connect each point to its k nearest neighbors, or equivalently to the neighbors within the ϵ-neighborhood. By so doing, we result in a graph that approximates the real manifold. In Figure 8.1 we illustrate the different phases of isomap; from a continuous manifold Figure 8.1 (a) to its approximating graph Figure 8.2 (a), and finally the embedding Figure 8.2 (b). We show in Figure 8.2 (b) how severely a connectivity graph may be affected

Table 8.1: Non-adaptive isomap algorithm.

Step 1: Construct a weighted graph G;

Let $G = \{\mathcal{A}, \mathbf{X}\}$, where:
- $\mathcal{A}$ is an $(N \times N)$ adjacency matrix;
- $\mathbf{X} = [X_1, \cdots, X_N]^T$;

Compute $\mathbf{D}_E$, the matrix of Euclidean distances between each two points in $\{X_i\}_{i=1}^N$.
Choose ϵ, the neighborhood radius.

```
for i, j ∈ {1, ··· , N} do
    if D_E(i, j) < ε do
        A(i, j) = D_E(i, j);
    else
        A(i, j) = ∞;
    end if
end for
```

Step 2: Compute geodesic distances on G.

Let $\mathbf{D}_G$ be the matrix of geodesic distances between each two points in $\{X_i\}_{i=1}^N$.

```
do D_G = A; (initialization)
for i, j ∈ {1, ··· , N}; k = 1; do
    while D_G(i, j) ≠ D_G(i, k) + D_G(k, j) do
        for k ∈ {1, ··· , N} do
            D_G(i, j) = min(D_G(i, j), D_G(i, k) + D_G(k, j));
        end for
    end while
end for
```

Step 3: Apply MDS on $\mathbf{D}_G$.

in the presence of noise. This consequently yields an inaccurate embedding of a given manifold as shown in Figure 8.2 (d).

8.3 New adaptive distance

The graphs illustrated in Figure 8.2 (b) and (d) are the result of considering a Euclidean neighborhood. We herein maintain that the choice of the distance is crucial in constructing good connectivity graphs. Our objective is to define a more appropriate distance that alleviates the effect of noise and obtain accurate graphical approximations. In what follows we provide the intuitive rationale for the choice of a new adaptive distance.

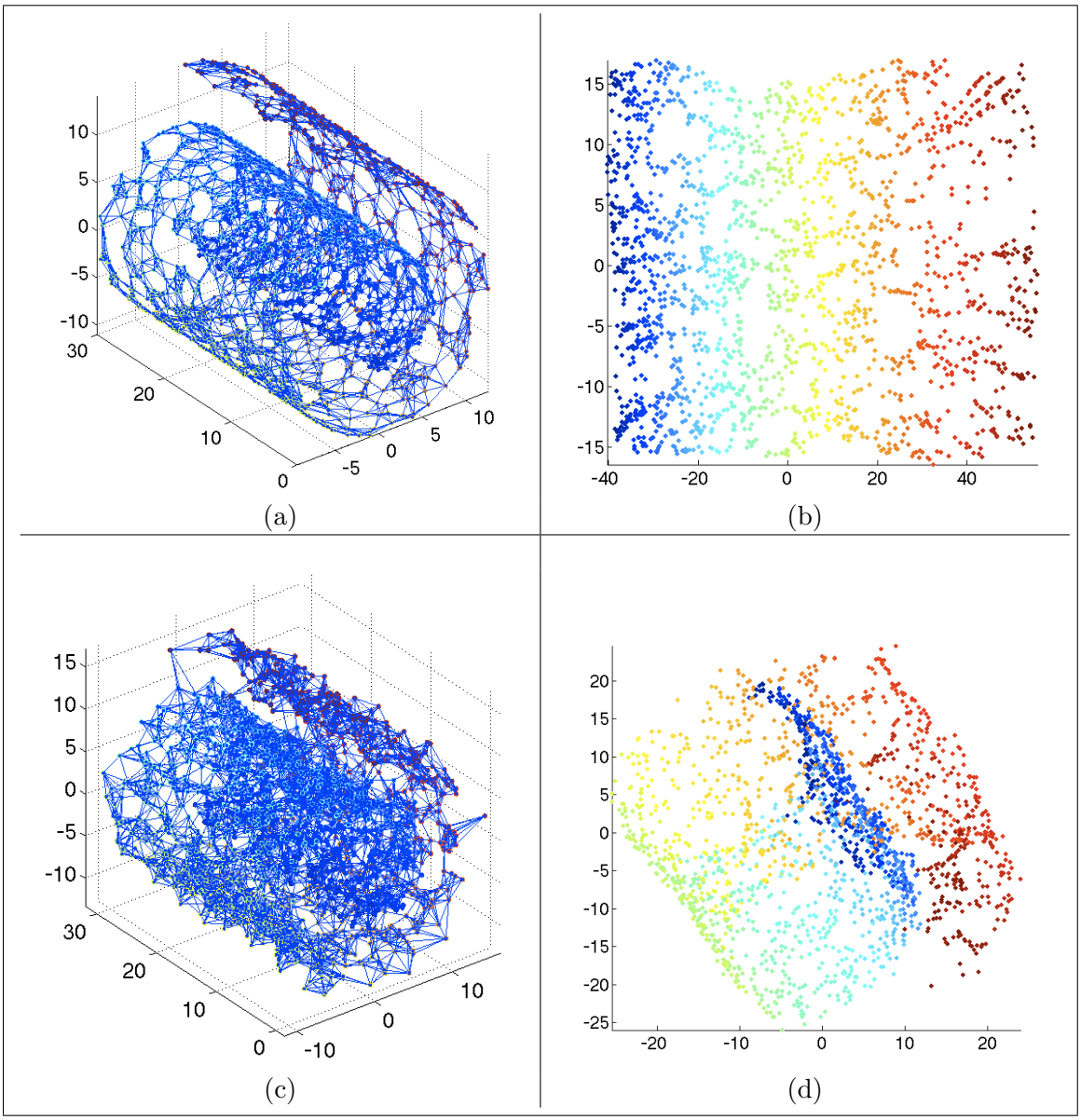


Figure 8.2: Failure of isomap in a noisy setting.

We subsequently present a mathematical formulation of new solutions to the embedding problem to result in an improved technique described in Table 8.1.

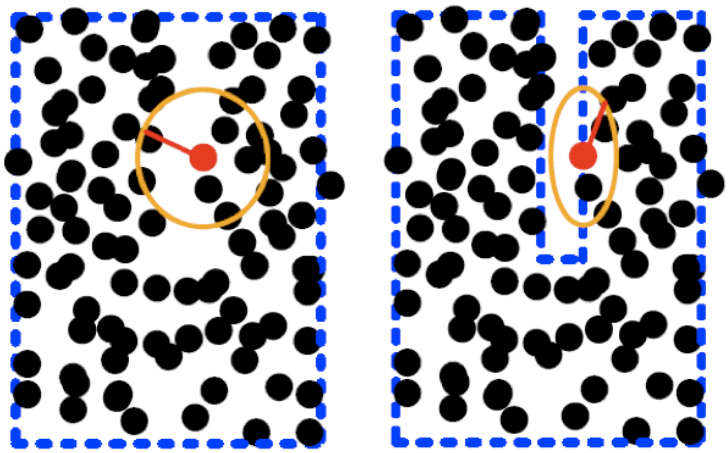


Figure 8.3: Mahalanobis vs Euclidean distance: On the left, the result of using a Euclidean distance. On the right, the result of using a Mahalanobis distance.

8.3.1 Motivation

In spite of the isomap good embedding results, it remains very unstable and sensitive to noise, as well as to the choice of the parameter ϵ and the distance function used prior to applying MDS. Changing the distance from Euclidean to geodesic thus appears to be insufficient to completely preserve the intrinsic geometric structure of the initial manifold $\mathcal{M}$. In what follows, we propose to account for the statistical properties of the observed data. Specifically, our technique consists in considering the correlation between each point and the rest of the observed data points, and subsequently exploit this information to connect it to its neighbors. This idea is exactly equivalent to using a *Mahalanobis* distance [73]. To better understand the intuition behind our choice, we illustrate the result of constructing a graph connectivity for the sample points in Figure 8.3. We note that when using a Euclidean distance to determine the neighbors of each data point causes a miss of some details in the structure of the data set.

8.3.2 Adaptive embedding algorithm

In what follows we propose a new distance matrix $\mathbf{D}_M$ to replace $\mathbf{D}_E$ in the algorithm described in Table 8.1. Our objective is to define, each time, a distance that is fully dependent on the sample point $\{X_i\}_{i=1}^N$; hence, we rescale the data coordinates based

Table 8.2: Description of the learning step (new adaptive distance)

Step 0: Compute $\mathbf{D}_M$, the new distance on $\mathcal{M}$.

Choose ϵ_1, the neighborhood radius for manifold learning;
and ϵ_2, the neighborhood radius for the construction of G.

for $i \in \{1, \cdots, N\}$ **do**
  $\mathbf{Y}_i = X_i^T$; (initialization)
  for $j = 1, cdots, N$ **do**
    while $\mathbf{D}_E(i,j) < \epsilon_1$ **do**
      $\mathbf{Y}_i = [\mathbf{Y}_i; X_j^T]$;
    end while
  end for
  $\sum_i = \text{cov}(\mathbf{Y}_i^T)$, cov$(\cdot)$ being the covariance matrix;
end for
for $i \in \{1, \cdots, N\}$ **do**
  for $j \in \{1, \cdots, N\}$ **do**
    $\mathbf{D}_M(i,j) = (X_j - X_i) \sum_i^{-1} (X_j - X_i)^T$;
  end for
end for
do $\epsilon = \epsilon_2$; $\mathbf{D}_E = \mathbf{D}_M$;
go to Step 1. (See Table 8.1)

on their distributions on $\mathcal{M}$ as well as their correlations. Since this technique relies on a learning procedure and directly uses isomap to build on, we refer to it as an adaptive isomap algorithm. We hence use the algorithm of Table 8.1 with a learning step, *i.e*, Step 0, as described in Table 8.2.

8.4 Performance comparison

In this section, we qualitatively and quantitatively compare the performances of the two versions (adaptive and non-adaptive) of isomap embeddings. To that end, we start by defining a performance measure. We subsequently simulate different classical examples of manifolds to embed in a lower dimensional space. The choice of our examples is such that one may visually inspect and verify the properties as well as the intuition behind each technique.

8.4.1 Residual variance

We choose a residual variance ρ to be our performance indicator. In further applications, we may use it to investigate the topological structure of a manifold. Residual variance ρ is defined in Eq. (8.2).

$$\rho = 1 - \big(\text{corrcoef}(\mathbf{D}^X, \mathbf{D}^Z)\big)^2, \tag{8.2}$$

where:

- $\mathbf{D}^X$: is the distance matrix for the initial data in $\mathcal{M}$. For an isomap embedding technique, this matrix is the geodesic distance matrix $\mathbf{D}_G$ which is exploited by MDS. We note that $\mathbf{D}_G$ may vary as it depends on the first distance matrix used to identify a neighborhood. For the classical non-adaptive isomap, this distance was simply Euclidean, *i.e.*, $\mathbf{D}_E$, while for the adaptive case we define it as $\mathbf{D}_M$. This difference is crucial to comparing the performance improvement of the adaptive isomap algorithm over its classical counterpart.

- $\mathbf{D}^Z$: is the distance matrix for the final (embedded) data of reduced dimension p ($p < n$). We take advantage of the simplicity of the geometries of our examples (swiss rolls, hemispheres, parallel sheets), and take p equal to 2 and consider $\mathbf{D}^Z$ to be Euclidean. It becomes trivial to visually verify the accuracy of our assumption. For more complex geometrical and topological structures, one would, however, need to compute geodesic distances on the new manifold embedded in $\mathbb{R}^p$.

- corrcoeff$(\cdot,\cdot)$: is the linear correlation coefficient. If we note $\{d_X\}$ and $\{d_Z\}$ as two ordered sets of distances (matrix elements) of $\mathbf{D}^X$ and $\mathbf{D}^Z$, respectively, then:

$$\text{corrcoef}(\mathbf{D}^X, \mathbf{D}^Z) = \frac{\sigma_{XZ}}{\sigma_X \sigma_Z}, \tag{8.3}$$

  σ_{XZ} being the correlation between the two sets $\{d_X\}$ and $\{d_Z\}$, and σ_X and σ_Z being the standard deviations of $\{d_X\}$ and $\{d_Z\}$, respectively.

8.4.2 Simulation examples

We saw that in the presence of noise, the performance of non-adaptive isomap drastically deteriorates and it only makes sense to evaluate performance change when we

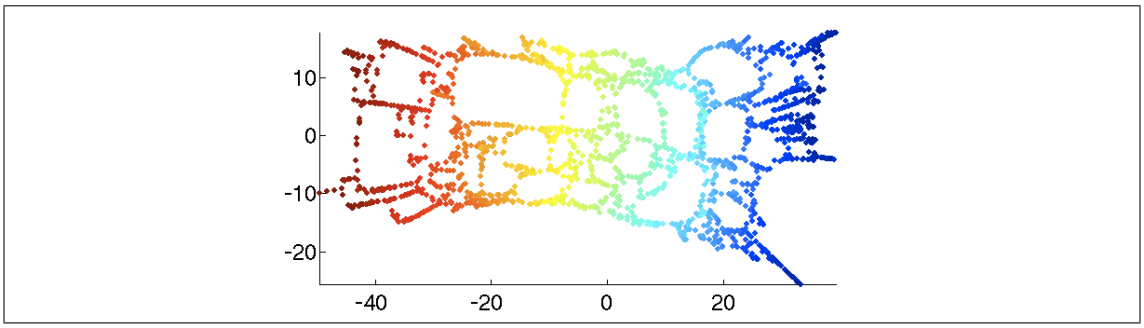


Figure 8.4: Adaptive embedding of the noisy swiss roll in Figure 8.1.

analyze the same noisy data sets using the adaptive *isomap*. We consider the noisy swiss roll example for which we determine an embedding as shown in Figure 8.4. We next treat a slightly more challenging case constituted of two adjacent hemisphere and two parallel sheets. Figure 8.5 shows the results obtained for the hemispheres. We note that the result of the non-adaptive isomap is not an embedding. This is due to the connectivity resulting from using a Euclidean neighborhood. Indeed, the two hemispheres end up connected through at least 2 points. We avoid this connection by using a Mahalanobis distance. Figure 8.5 (c) shows the final embedding resulting from using the proposed adaptive isomap.

In Figure 8.6, we show the embedding results for parallel sheets shown in Figure 8.6 (a).

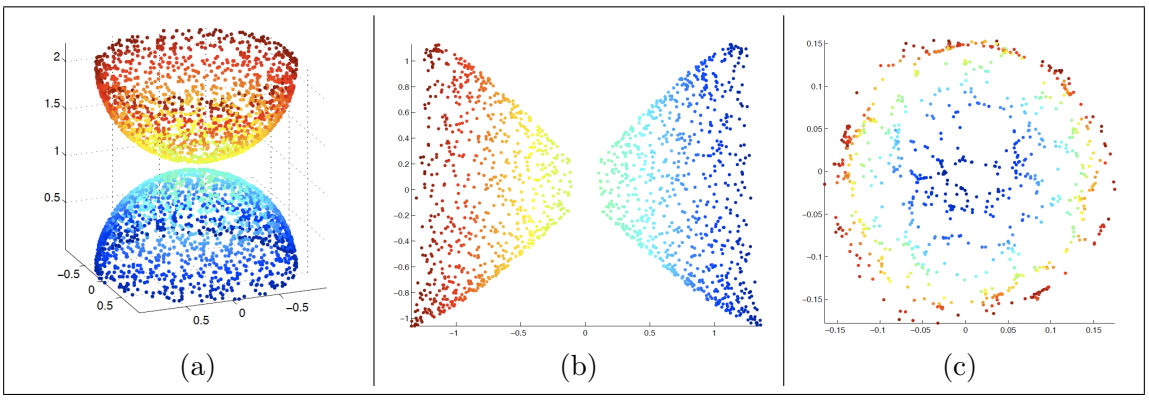


Figure 8.5: Embedding two adjacent hemispheres: (a) Hemispheres, (b) the result of a non-adaptive isomap embedding, (c) the result of the proposed adaptive isomap.

For $k = 10$, we see in Figure 8.6 that non-adaptive isomap fails again to define an embedding of the two sheets. The reason is again the connection that occurs when using a Euclidean

distance for a graph construction. The result of the adaptive isomap is the disconnection of the two sheets. As they should, they remain two distinct structures and are hence separately embedded. The result of embedding one sheet is shown in Figure 8.6 (b). Note that it is exactly the sheet itself, but now in 2-dimensions instead of 3-dimensions. For a different value of the neighborhood size $k > 10$, we find the results in (d) and (e) for the adaptive and adaptive isomaps, respectively. We notice the sensitivity of the mapping results to the neighborhood size; we also see that the separation between the two sheets did not happen in both cases; however, in the adaptive case, the two sheets are clearly spread on a plane, while we lose one sheet in the non-adptive case, as all the points collapse into one line.

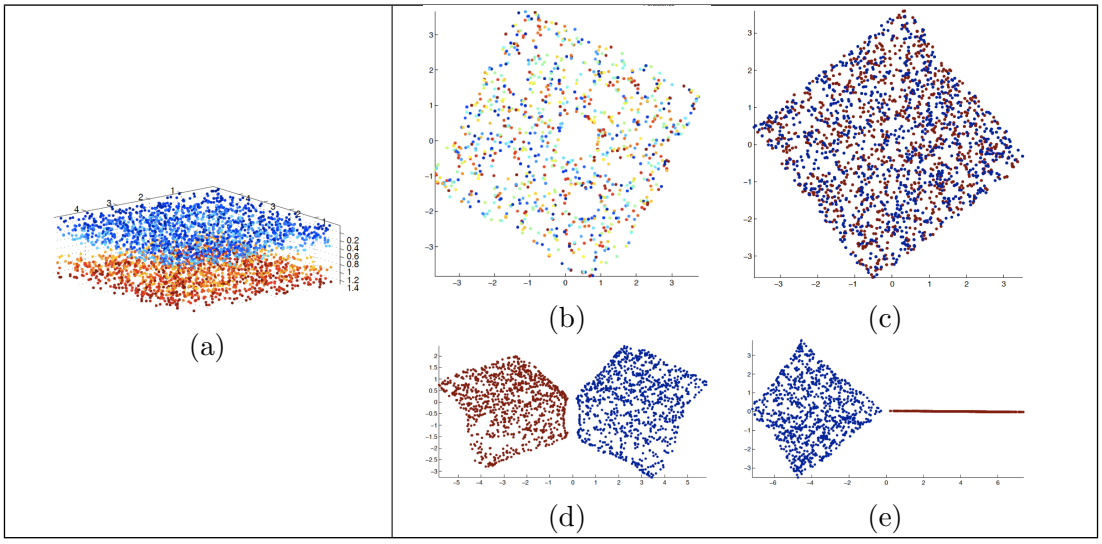
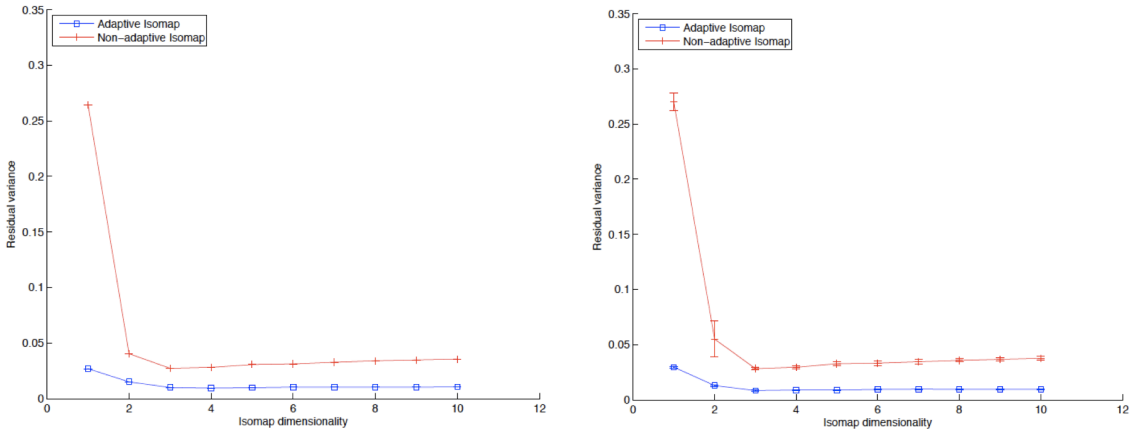


Figure 8.6: Embedding of two parallel sheets: (a) the initial data. (b) Adaptive isomap with $k = 10$ and $k1 = 55$. Only 50% of the initial points are represented here. (c) Non-adaptive isomap with $k = 10$. 100% of the initial points are represented and the two sheets are overlapped. Adaptive isomap (d) and non-adaptive isomap (e) with $k = 15$. 100% of the initial points are represented.

To further evaluate the effect of noise on our proposed embedding technique, we increase the amount of Gaussian noise to vary between 0% and 8% of the orthogonal distance between the two parallel sheets, the normal distance between two consecutive levels of the swiss roll, and the orthogonal distance between the poles of the two adjacent hemispheres. By way of Monte Carlo simulations on the data in hand, we obtain the results shown in Figure 8.7 (a) and (b). We establish that the adaptive isomap technique consistently

outperforms the non-adaptive isomap technique.



(a) (b)

Figure 8.7: Monte Carlo simulations.

8.5 Network failure detection

A natural source of data which share similar characteristics with much of the data explored above is that of sensors in a given network topology. We propose to fully exploit the adaptive isomap algorithm for a statistical analysis of networks in the presence of attacks or intrusions with the ultimate objective of detecting them and potentially localizing them. Using the correlation between the different nodes of a network is a way to unfold the interdependencies between them.

Our approach starts by abstracting networks from their physical layered description; we view a network as a set of features/measurments. The same feature space may house multiple networks that are not necessarily the same. Our hypothesis is that each network is a smooth topologically homogeneous manifold embedded in the network feature space. We consider each data point on a manifold as a node in a network. In addition, an attack or any intrusion is assumed to have the effect of destroying the initial topological structure of a manifold. Our intrusion detection task is thus equivalent to detecting and determining any topological changes on the nodes manifold.

Before considering the dimensionality reduction of a network data, it is important to adap-

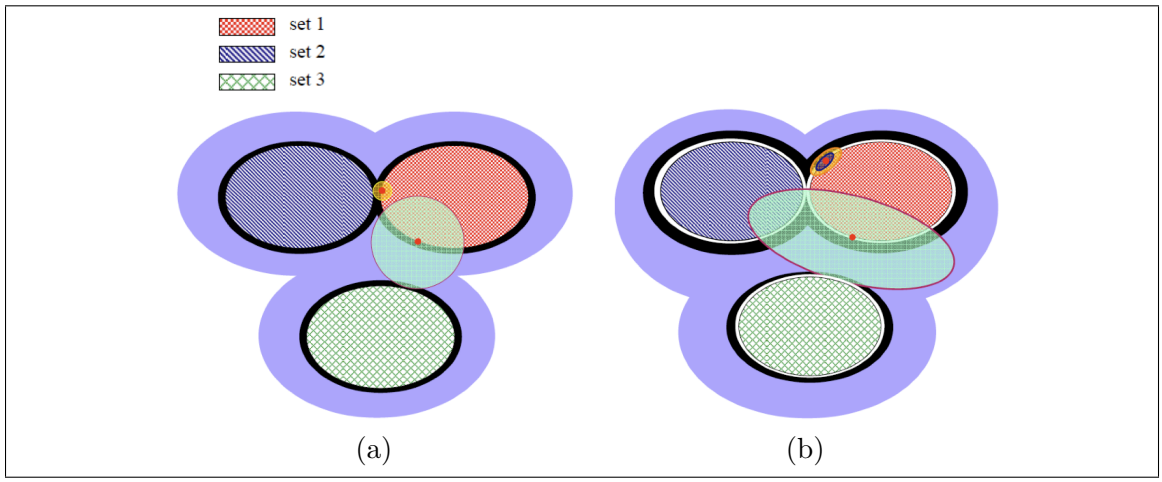


Figure 8.8: Sets connectivity versus neighborhood size. By progressively varying the size of the neighborhood, we identify and count the parts that get merged. We talk about the persistence of these parts (Figure 8.9).

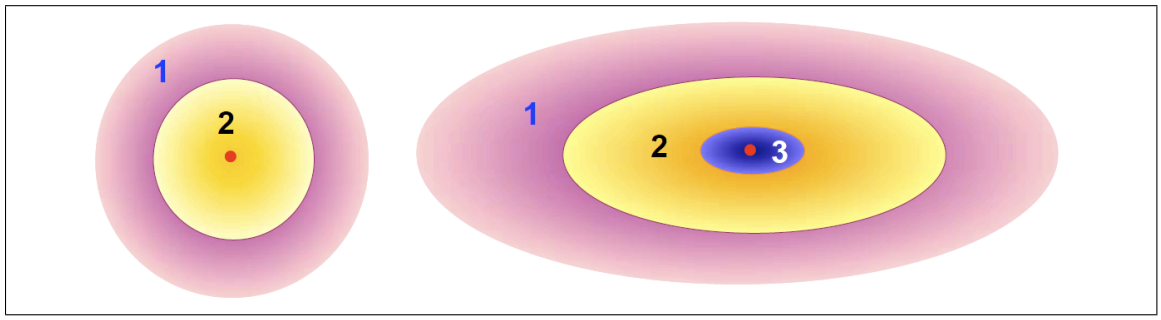


Figure 8.9: Detection of different events. (a) Two events detected when using a Euclidean distance. (b) Three events detected when using a Mahalanobis distance.

tively define the connectivity between the different nodes of a network. In the present experiment, the connectivity is dependent on the correlation between the measurements collected at each node. Using a mahalanobis distance becomes, thus, crucial as it is independent of any scaling we may apply to a specific set of measures. We illustrate this idea in Figure 8.8, where we observe all the connectivity events associated with the three provided sets. By evolving the size of the neighborhood, we connect the different data points and find the number of the final resulting structures (connected sets). We determine, by using a Euclidean distance that there are two detected events, while we discover three events when using a Mahalanobis distance (Figure 8.9).

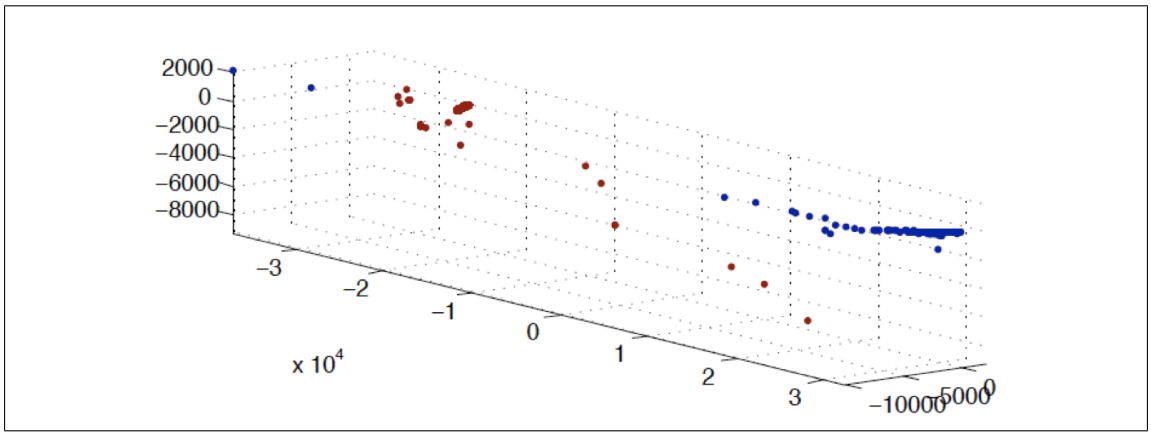


Figure 8.10: Distinction between attacked nodes in red and normal ones in blue.

Given the correlated and noisy nature of network data, we adopt the adaptive isomap algorithm. We use the DARPA network data available online from the Knowledge Discovery and Data (KDD) Mining Cup 1999 website. We use all 41 parameters to define each node, *i.e.*, $N = 41$. We initially only use data that is subjected to a single attack. We find the embedding result of the adaptive isomap in Figure 8.10.

The first comment is that there is a clear separation between the safe and attacked nodes; nevertheless, we need to verify that this visualization makes sense and that we are in presence of an accurate embedding. To that end, we look at the embedding dimensionality to verify that it is indeed equal to three. We find in Figure 8.11 (a) the residual variance corresponding to this experiment. We may read that "3" is the the accurate embedding dimensionality when the network is subjected to one attack; however, in the presence of multiple attacks, the embedding dimensionality increases and becomes difficult to visualize.

8.6 Conclusion

We proposed a new adaptive embedding algorithm that first learns from the correlations between the neighboring points. We showed that combining an adaptive distance with an existing embedding algorithm leads to better embedding results with a higher robustness to noise. We illustrated this technique on the *isomap* algorithm, while conceptually

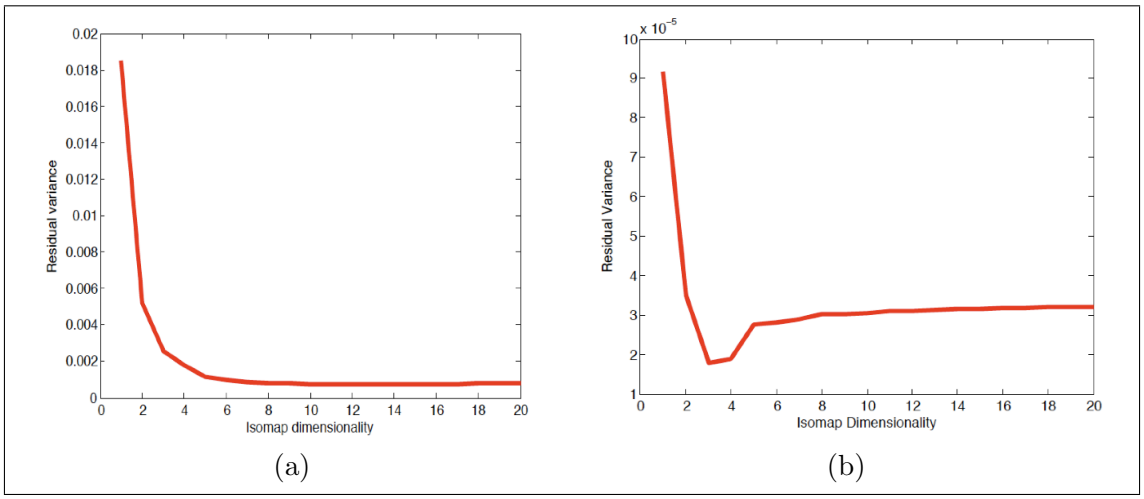


Figure 8.11: Embedding dimensionality of networks under different attacks. (a) Under multiple attacks. (b) Under a single attack.

compatible with any manifold learning technique that relies on a connectivity graph for the initial data. Our initial experiments on network data show promising results that we plan to further investigate.

Chapter 9

Future research directions

In this thesis, we have presented different techniques to compactly represent manifold-like datasets, with an emphasis on 3-dimensional objects. All our research work was primarily motivated by applications in object classification, recognition and identification. Many fundamental questions arose in the course of this work, such as specifying the sampling rate of a height function defined on a complex 3D shape, remain open research questions and constitute a very rich avenue for future research.
Another very promising and new theme of research directly impacts networking applications, as quickly illustrated in Chapter 8, and attempts to exploit novel structures of data which may approximately lie on nonlinear curved spaces or manifolds. For concreteness and conciseness, we concentrate on and elaborate on these two directions of research, the first one with more of a fundamental interest than the second.

9.1 Fundamentals of shape modeling

In order to achieve an efficient modeling of 2 and 3D shapes, a good understanding of their geometry is required. To that end, we proceed to detail the crucial elements for an accurate shape representation. Specifically,

1. Optimal reduction of triangulated meshes.

2. Efficient sampling of 2-dimensional surfaces via level curves.

3. Surface reconstruction.

9.1.1 Optimal reduction of triangulated meshes

In practice, a surface $\mathcal{S}$ is arbitrarily closely approximated by a triangular mesh consisting of a certain number of nodes (points). As discussed in Chapter 3, each surface $\mathcal{S}$ is characterized by a parameter $\Re$, referred to as the characteristic resolution of $\mathcal{S}$. This parameter effectively defines the minimal number of points required to "correctly" represent a shape $\mathcal{S}$.

The role that $\Re$ plays in reducing the computational cost of representing a shape with minimal quality loss is pivotal. While this approach has demonstrated a representation efficiency improvement of up to an order of magnitude, the illustration remains statistical and empirical and hence theoretically unsubstantiated, hence raising the fundamental question of its source and overall viability.

The reduction of $\Re$ of a shape as shown, was equivalent to smoothing the shape, hence affecting it. Analogously, a resolution of a flat norm corresponds to a certain scale of curvature, which in turn is directly related to the characteristic resolution $\Re$ and thus to the flat norm in $\mathbb{R}^3$. A more robust and systematic modeling of 3D shapes (for instance), would likey incude the multi-scale nature of the flat norm.

9.1.2 Efficient sampling

We have seen in Eq. (4.1) how a surface is represented by a set of disjoint closed curves. Sampling a surface is then reduced to keeping a subset of these curves. The obvious, and crucial question is what to keep and what to leave? To the best of our knowledge, such a sampling rate K is only empirically chosen [1, 28]. Finding the effective sampling rate K that ensures the extraction of, and only of, the information needed to represent and reconstruct a given shape, would thus be revolutionary in the automatization of level set based algorithms.

We may start approaching the problem by first considering surfaces (or subsurfaces) of cardinality one[1]. Let us revert to the probability space $(\mathcal{V}, \mathcal{A}, P_{\Re})$ as defined in Section 4.2.3. For a uniformly sampled space, it is easy to note that the instantaneous value $P_{\Re}(i)$ is proportional to the length of the level curve at the value i of the GGF. We thus propose to measure the variability $\Delta P = |P_{\Re}(i) - P_{\Re}(j)|$ induced by the monotonous and

[1]Cardinality of a surface is the number of closed level curves for a given value of the considered Morse function on that surface.

continuous evolution of the level curves. In fact, it is this variability that will guide an efficient sampling of a surface. ΔP is nothing but the change in the length of C_i and C_j up to a constant. In summary, the idea is to determine ΔP as the minimal variation to be detected. Then, we need to sample $P_{\Re}$ at every ΔP variation.

Many questions may follow from the proposed preliminary direction in investigating an efficient sampling rule for surfaces. One may consider achieving an adaptive sampling where the criterion ΔP changes as needed. An even more challenging problem is to consider complex surfaces where the cardinality is no longer equal to one.

9.1.3 Surface reconstruction

Once we determine how detailed the representation of a 3D object needs to be, it becomes possible to ensure a good surface reconstruction. Thus, the inverse problem to the one of Section 9.1.2 may be posed, *i.e.*, we are given few level curves and we want to get the initial shape back.

Considering the observed closed sampling curves as points in a hyperspace, we may write them in the form of a matrix $\mathbf{X}$, where each column corresponds to one curve. At this level, the problem is reduced to interpolating this space curve. In order to further simplify the problem, we propose to use linear algebra and compute the SVD[2] decomposition of $\mathbf{X}$.

$$\mathbf{X} = \mathbf{U}\left(\mathbf{\Sigma}\mathbf{V}^{\dagger}\right) = \mathbf{U}\mathbf{P}. \tag{9.1}$$

Our problem is thus reduced to finding the interpolated version of $\mathbf{P}$, say $\mathbf{Q}$. The interpolated version of $\mathbf{X}$, say $\mathbf{Y}$, will follow as:

$$\mathbf{Y} = \mathbf{U}\mathbf{Z}\mathbf{Q}, \tag{9.2}$$

where $\mathbf{Z}$ is a diagonal filtering matrix. This simple technique is proposed for surfaces of cardinality one. Again, there is a need to generalize this approach to generic shapes. It may also be of interest to combine recognition and reconstruction in order to efficiently restore occluded objects.

[2]SVD: Singular Value Decomposition

9.2 Behavioral Modeling

We have shown throughout this thesis, how to exploit the same topological-geometric techniques in modeling various data sets (*e.g.,* 3D objects, sensor network data). Modeling the human behavior is one more possible setting, that may benefit from our proposed set of tools. The ultimate goal of this research is to implement an automatic system capable of understanding complex dynamic scenes. One may start by studying elementary activities (*i.e.,* behaviors) such as walking, running, and jumping. For such a modeling problem, it is of interest to exploit both spatial and temporal information captured in video frames. We thus may view each frame as a visual/spatial description of the studied action at an instant t. The description in question is the precise placement of the subject with respect to his/her environment, which may first be limited to the floor. We propose to extract this description

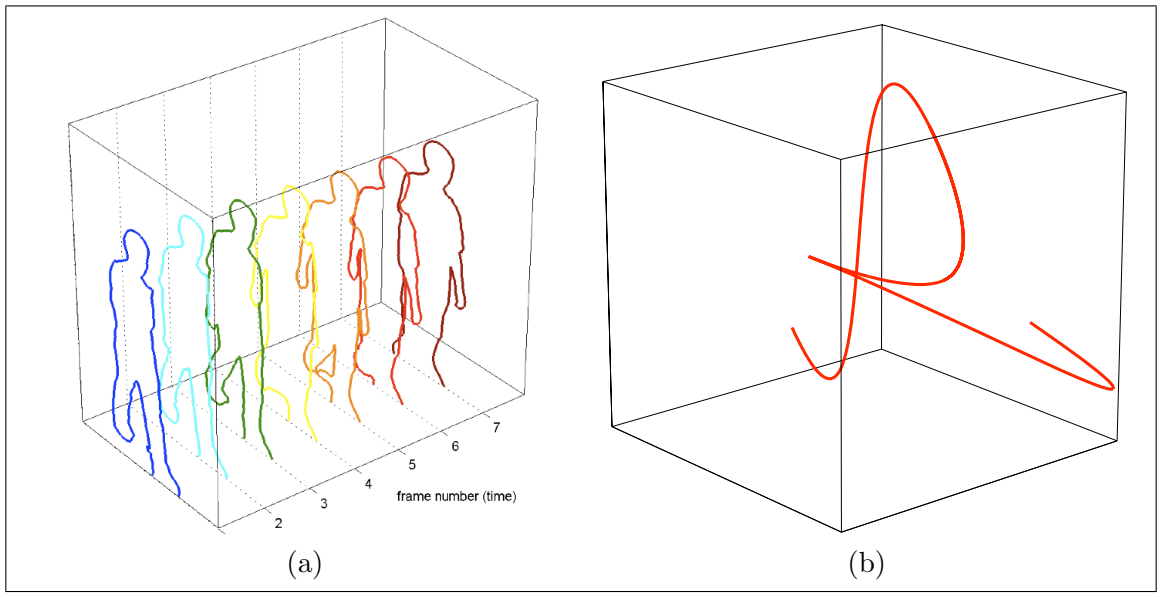


Figure 9.1: Modeling of the action of walking.

through the contour of the subject and its relative location. We, then, may recreate the dynamic scene by interpolating the surface of all the contours parameterized by time. By so doing, we construct the one dimensional manifold $\mathcal{M}$ of the scene. Our modeling consists therefore in embedding $\mathcal{M}$ in $\mathbb{R}^3$ thanks to an implementation of the Whitney embedding

theorem (Section 5.3).

We investigate the discriminative capabilities of the proposed method by looking at three scenes; two of which illustrate the action of *jumping* (Figure 9.2) and the third one being the action of *walking* (Figure 9.1). The preliminary results comfort the idea of

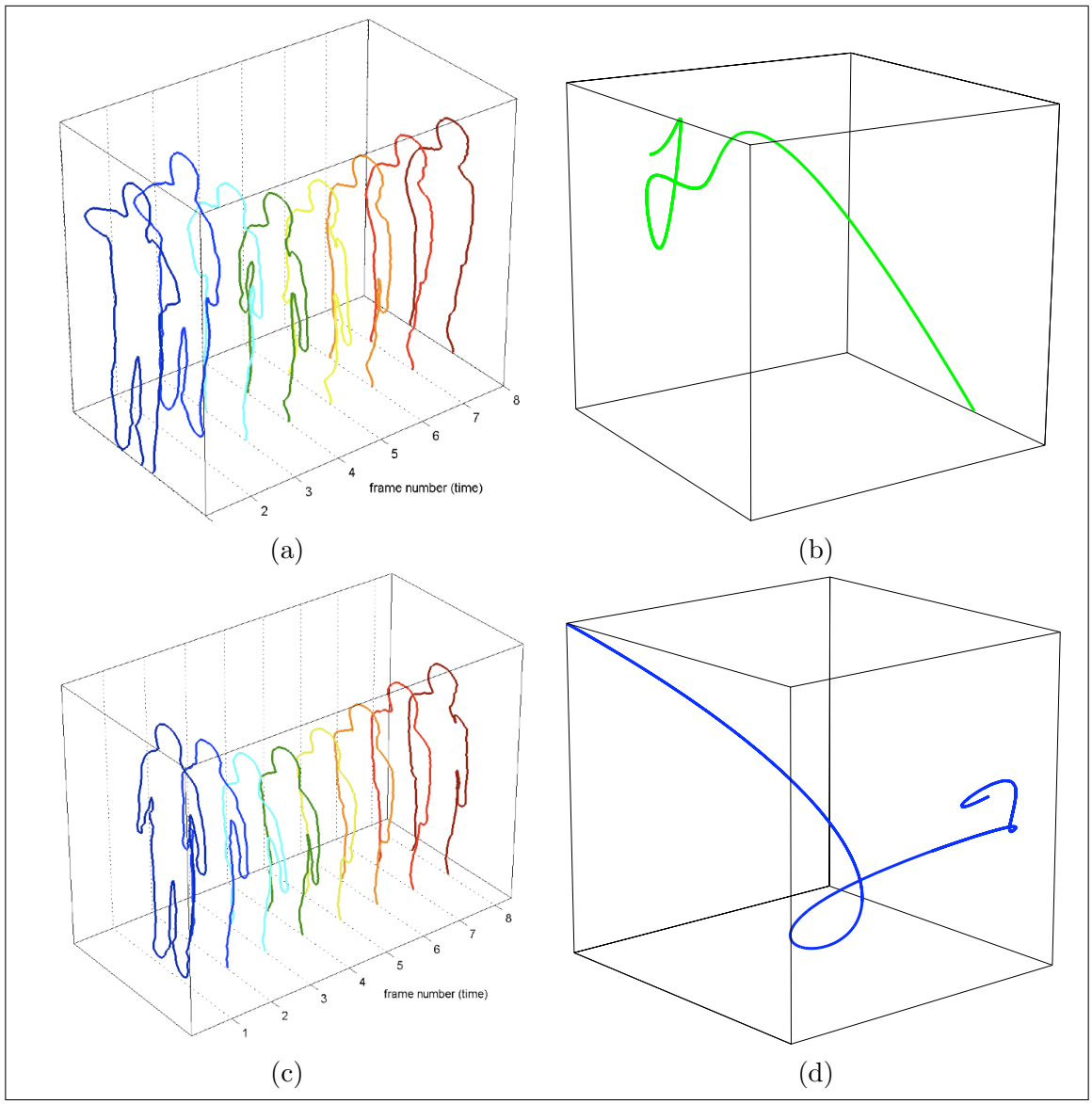


Figure 9.2: Modeling of the action of jumping for different sequences.

a signature assigned to each elementary behavior. There are nevertheless many questions that one may needs to answer in order to fully investigate the effectiveness of the proposed technique. Below is a non exhaustive list of these important questions:

1. How accurate is it to only consider the exterior contours?
2. How much does the change in topology affect the final model?
3. How many frames are needed to represent each movement?
4. How to deal with complicated configurations where severe occlusion and noise are involved?
5. How to combine many actions in one scene?

Bibliography

[1] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii: "Topology Matching for Fully Automatic Similarity Esimation of 3D Shapes", Proc. SIGGRAPH, pp. 203-212, August 2001.

[2] Tony Tung, Francis Schmitt, "The augmented multiresolution Reeb graph approach for content-based retrieval of 3d shapes". International Journal of Shape Modeling. vol. 11, no.1, pp. 91-120, 2005.

[3] Y. He, A. B. Hamza, H. Krim: "A Generalized Divergence Measure Registration for Robust Image Registration", IEEE Transactions on Signal Processings, vol. 51, pp.1211-1220,2003.

[4] Philip N. Klein, Thomas B. Sebastian, Benjamin B. Kimia,"Shape matching using edit-distance: an implementation", pp. 781-790, SODA, 2001.

[5] M. Kazhdan, and T. Funkhouser: "Harmonic 3D Shape Matching", SIGGRAPH 2002 Technical Sketches, pp. 191, July, 2002.

[6] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin: "Matching 3D Models with Shape distributions".

[7] Farzin Mokhtarian and Miroslav Bober, "Curvature Scale Space Representation: Theory, Applications, and MPEG-7 Standardization", Kluwer Academic Publishers, Norwell, MA, USA, 2003.

[8] Djamila Aouada and Hamid Krim, "3d object recognition using fully intrinsic skeletal graphs", 2008, vol. 6814, p. 681409, doi:10.1117/12.774868, SPIE.

[9] Kaleem Siddiqi, Sylvain Bouix, Allen Tannenbaum, and Steven W. Zucker, "The hamilton-jacobi skeleton", Image Processing, 2005. ICCV 2005. IEEE International Conference on Computer Vision, pp. 828-834, Sept 1999.

[10] Djamila Aouada, David W. Dreisigmeyer, and Hamid Krim, "Geometric modeling of rigid and non-rigid 3d shapes using the global geodesic function", IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008. CVPR Workshops 2008., pp. 1-8, Jun. 2008.

[11] Djamila Aouada and Hamid Krim, "Squigraphs for fine and compact modeling of 3D shapes", IEEE Transactions on Image Processing - In peer review.

[12] Djamila Aouada and Hamid Krim, "Novel similarity invariant for space curves using turning angles and its application to object recognition", IEEE Int. Conf. on Acoust., Speech and Signal Proces. (ICASSP'09), Taipei, Taiwan, Apr. 2009.

[13] Djamila Aouada and Hamid Krim, "Correspondence preserving similarity invariant and its application to object recognition", IEEE Transactions on Image Processing - In peer review.

[14] http://www.cim.mcgill.ca/ shape/benchMark/

[15] Philip Shilane, Patrick Min, Michael Kazhdan, and Thomas Funkhouser, "The Princeton Shape Benchmark", Shape Modeling International, Genova, Italy, June 2004.

[16] http://shape.cs.princeton.edu/benchmark/

[17] http://www-c.inria.fr/Eric.Saltel/download/download.php

[18] http://www.3dcafe.com/

[19] J. Milnor, "Morse theory", Princeton University Press, New Jersey, 1963.

[20] A. T. Fomenko, and T. L. Kunii, "Topological Modeling for Visualization". Springer-Verlag, 1997.

[21] X. Ni, M. Garland, and J. C. Hart, "Fair morse functions for extracting the topological structure of a surface mesh", ACM Transactions on Graphics, pp. 613-622, 2004.

[22] Yoshihisa Shinagawa, Tosiyasu L. Kunii, Alexander G. Belyaev, and Taketo Tsukioka, "Shape Modeling and Shape Analysis Based on Singularities". International Journal of Shape Modeling, 2(1):85-102, March 1996.

[23] Djamila Aouada, Shuo Feng, and Hamid Krim, "Statistical analysis of the global geodesic function for 3D object classification", IEEE Int. Conf. on Acoust., Speech and Signal Proces. (ICASSP'07), Hawaii, USA, Apr. 2007.

[24] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: "Introduction to Algorithms", Second Edition, September 2001. MIT Press, Cambridge, Massachusetts.

[25] J. Lin: "Divergence Measures Based on Shannon entropy", IEEE Transactions. Inform. Theory, vol. 37, pp.145-151, 1991.

[26] M. I. Jordan, "Learning in Graphical Models", MIT press, 1998.

[27] Yoshihisa Shinagawa and Tosiyasu L. Kunii, "Constructing a Reeb Graph From cross sections", IEEE Computer Graphics and Applications, 11(6), pp. 44-51, November 1991.

[28] S. Baloch, H. Krim, I. Kogan, and D. Zenkov: "Topological-Geometric Shape Model For 3D Object Representation". Proc. of European Signal Processing Conference (EUSIPCO'05), 4 pp., 2005.

[29] Johan W. H. Tangelder and Remco C. Veltkamp, "A survey of content based 3D shape retrieval methods", Multimedia Tools and Applications, vol. 39, no. 3, Sep. 2008, pp. 441-471.

[30] V. Jain, H. Zhang, "Robust 3D shape correspondence in the spectral domain", in: Proc. of Shape Modeling International (SMI), 2006, pp. 118-129.

[31] Meng Yu; Atmosukarto, I.; Wee Kheng Leow; Zhiyong Huang; Rong Xu, "3D model retrieval with morphing-based geometric and topological feature maps" Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on , vol.2, no., pp. II-656-61 vol. 2, 18-20 June 2003.

[32] G. Reeb,"Sur les points singuliers d'une forme de pfaff complèment intégrable ou d'une fonction numérique", *Comptes Rendus de L'Académie ses Séances*, Paris, 222, pp. 847-849, 1946.

[33] A. B.Hamza, and H. Krim: "Geodesic Matching of Triangulated Surfaces", IEEE transactions on image processing, vol. 15, no. 8, pp 2249-2258, August 2006.

[34] A. B. Hamza, and H. Krim: "Geodesic Object Representation and Recognition", Proc. DGCI, pp. 378-387, November 2003.

[35] Juan Zhang, Kaleem Siddiqi, Diego Macrini, Ali Shokoufandeh, and Sven Dickinson, "Retrieving Articulated 3-D Models Using Medial Surfaces and Their Graph Spectra". EMMCVPR 2005, LNCS 3757, pp. 285-300, 2005.

[36] Anne Verroust and Francis Lazarus,"Extracting skeletal curves from 3D scattered data". The Visual Computer, vol. 16, no. 1, pp. 15-25, 2000.

[37] Fomenko, A. T., and Kunii, T. L. "Topological Modeling for Visualization". Springer-Verlag, 1997.

[38] Morris W. Hirsh. Differential Topology. Graduate Texts in Mathematics. Vol.33. Springer-Verlag, 1976.

[39] D.S.Broomhead and M.Kirby, "New approach for dimensionality reduction: Theory and algorithms" SIAM Journal. of Applied Mathematics, vol. 60, no. 6, pp. 2114-214, 2000.

[40] D.S Broomhead and M. J. Kirby, "The Whitney Reduction Network: a method for computing autoassociative graphs", Neural Computation, vol.13, pp. 2595-2616, 2001.

[41] David W. Dreisigmeyer. Direct search algorithms over Riemannian manifolds. 2007. http://ddma.lanl.gov/Documents/publications/dreisigm-2007-direct.pdf

[42] P. Olver, G. Sapiro, and A. Tannenbaum, Differential invariant signatures and ows in computer vision: A symmetry group approach, in Geometry Driven Diffusion in Computer Vision, B. M. ter Haar Romeny, Ed. Kluwer Academic Publishers, Dordrecht, 1994.

[43] Feldman, J., Singh, M. (2005). Information along contours and object boundaries. Psychological Review, 112(1), 243252.

[44] Shannon, C. (1948), "A mathematical theory of communications", The Bell System Technical Journal; 27, 379-423.

[45] Frank Morgan. Geometric Measure Theory: A Beginner's Guide . Academic Press, 3rd edition, 2000.

[46] Simon P. Morgan and Kevin R. Vixie, "L1TV Computes the Flat Norm for Boundaries", abstract and Applied Analysis, vol. 2007, Article ID 45153, 14 pages, 2007. doi:10.1155/2007/45153

[47] Marc Vaillant and Joan Glauns, "Surface matching via currents". In Proceedings of Information Processing in Medical Imaging (IPMI 2005), volume 3565 of Lecture Notes in Computer Science. Springer, 2005.

[48] J. Lin: Divergence Measures Based on Shannon entropy, IEEE Transactions. Inform. Theory, vol. 37, pp.145-151,1991.

[49] Y. He, A. B. Hamza, H. Krim: "A Generalized Divergence Measure Registration for Robust Image Registration", IEEE Transactions on Signal Processing, vol. 51, pp.1211-1220, 2003.

[50] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions", ACM Transactions on Graphic, vol. 21, pp. 807-832, 2002.

[51] D. Aouada and H. Krim, "Global Geodesic Function in 3D Object Classification", International Journal of Computer Vision, 3D Object Retreival (3DOR), Special Issue. In Peer Review.

[52] A. M. Bronstein, M. M. Bronstein, R. Kimmel, "Numerical geometry of non-rigid shapes", Springer Verlag, 2007.

[53] A. M. Bronstein, M. M. Bronstein, R. Kimmel, "Calculus of non-rigid surfaces for geometry and texture manipulation", IEEE Trans. Visualization and Computer Graphics, Vol. 13/5, pp. 902-913, 2007.

[54] A. M. Bronstein, M. M. Bronstein, R. Kimmel, "Efficient computation of isometry-invariant distances between surfaces", SIAM Journal of Scientific Computing, Vol. 28/5, pp. 1812-1836, 2006.

[55] I. T. Joliffe. Principal Component Analysis. Springer-Verlag, New York, 1986.

[56] W. S. Torgerson. Multidimensional Scaling, I: Theory and methods. Psychometrica, 17:401-419, 1952.

[57] X. Huo, X. Ni, and A. K. Smith, A survey of manifold-based learning methods.

[58] J. B. Tanenbaum, V. de Silva, and J. C. Langford, A global geometric framework for nonlinear dimensionality reduction. Science 290(5500) (2000) 2319-2323

[59] S. T. Roweis and L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500) (2000) 2323-2326

[60] M. Belkin and P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering. In Dietterich, T. G., Becker, S., and Ghahramani, Z. (Eds.), Advances in Neural Information Processing Systems, 14: 585-591.

[61] M. Belkin and P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation, 15 (6): 1373-1396.

[62] D. L. Donoho, C. E. Grimes, Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. Proceedings of the National Academy of Arts and Sciences 100 (2003) 5591-5596

[63] D.L. Donoho and C. Grimes. When does isomap recover natural parameterization of families of articulated images? Technical Report 2002-27, Department of Statistics, Stanford University, August 2002.

[64] J.B. Tanenbaum, S. Mika, B. Schlkopf, and R.C. Williamson. Regularized principal manifolds. Journal of Machine Learning Research, 1:179-209, June 2001.

[65] A. Hadid, O Kouropteva and M. Pietikainen, Unsupervised learning using locally linear embedding: experiments in face pose analysis. In Proc. the 16th International Conference on Pattern Recognition, pages I:111-114, 2002.

[66] A. Elgammal and C. S. Lee, Inferring 3D body pose from silhouettes using activity manifold learning. In Proc. the IEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pages 681-688, 2004.

[67] O. Jenkins and M. Mataric, A sptio-temporal extension using packing numbers and design of principal curves. IEEE transactions on Pattern Analysis and Machine Intelligence, 22(3):281-297, 2000.

[68] M. H. Yang, Face recognition using extended isomap. In IEEE International Conference on Image Processing, pages II: 117-120, 2002.

[69] J. Wang, Z. Zhang and H. Zha, Adaptive manifold learning. NIPS, 2004.

[70] J. Costa, A. Girotra and A.O. Hero, Estimating local intrinsic dimension with k-nearest neighbor graphs. In IEEE Workshop on Statistical Signal Processing (SSP), Bordeaux (2005).

[71] E. Levina and P.J. Bickel, Maximum likelihood estimation of intrinsic dimension. In K.L. Saul, Y. Weiss and I. Bottou, Adances in Neural Information Processing Systems 17. MIT Press, Cambridge, MA (2005) 777-784.

[72] http://comptop.stanford.edu/preprints/multidimensional.pdf.

[73] De Maesschalck, R.; D. Jouan-Rimbaud, D.L. Massart (2000) The Mahalanobis distance. Chemometrics and Intelligent Laboratory Systems 50:118.

Appendices

APPENDIX A

If a 3D shape is subjected to a similarity transform, then each point $\mathbf{V}_j^i$ in $\mathbb{R}^3$ defined by the coordinates $\left(x_j^i, y_j^i, z_j^i\right)$ is transformed as follows

$$\mathbf{V'}_j^i = \alpha \mathbf{Q} \mathbf{V}_j^i + \mathbf{K}, \tag{.3}$$

where α is a scaling factor, $\mathbf{Q}$ is a (3×3) unitary matrix , *i.e.,* $\mathbf{Q}\mathbf{Q}^T = \mathbf{I}_3$, and $\mathbf{K}$ is a (3×1) translation vector. Considering the construction in (**??**) and the transform in (.3), each point $\mathbf{V}_i$ in $\mathbb{R}^{3M}$ is transformed by:

$$\mathbf{V}'_i = \alpha \underline{\mathbf{Q}} \cdot \mathbf{V}_i + \underline{\mathbf{K}}, \tag{.4}$$

with

$$\underline{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q} & & & \\ & \mathbf{Q} & & \\ & & \ddots & \\ & & & \mathbf{Q} \end{bmatrix}, \quad \text{and} \quad \underline{\mathbf{K}} = \begin{bmatrix} \mathbf{K} \\ \mathbf{K} \\ \vdots \\ \mathbf{K} \end{bmatrix}.$$

The question of concern is to find the transform applied to a modeling curve when the 3D shape it originates from is subjected to a similarity transform. If $\mathbf{P}$ is defined by (**??**), we may define $\mathbf{P'}$ as follows

$$\hat{\mathbf{P}}' \;=\; \arg\min \left[- \min_{i=1,\ldots,L} \parallel \mathbf{P}^T \mathbf{\Psi}'_i \parallel_2 \right].$$

Since the new secants $\mathbf{\Psi}'_i$, $i = 1, \cdots, L$, result from the same equation (.4) as $\mathbf{V}'_i$, we conclude that $\mathbf{P'}$ is directly related to $\mathbf{P}$ through a permutation (or a rotation) matrix. In other words: $\mathbf{P'} = \underline{\mathbf{Q}}\mathbf{P}\mathbf{U}$, where $\mathbf{U}\mathbf{U}^T = \mathbf{I}_3$; thus, we may write:

$$\begin{aligned} \mathbf{W}' &= \mathbf{P'}^T \cdot \mathbf{V}', \\ &= \left(\underline{\mathbf{Q}}\mathbf{P}\mathbf{U}\right)^T \cdot \left(\alpha \underline{\mathbf{Q}}\mathbf{V} + \underline{\mathbf{K}}\right), \\ &= \alpha \left(\mathbf{P}\mathbf{U}\right)^T \cdot \underline{\mathbf{Q}}^T \cdot \underline{\mathbf{Q}}\mathbf{V} + \left(\underline{\mathbf{Q}}\mathbf{P}\mathbf{U}\right)^T \cdot \underline{\mathbf{K}}, \\ &= \alpha \mathbf{U}^T \cdot \mathbf{P}^T \mathbf{V} + \mathbf{L}, \\ &= \alpha \mathbf{U}^T \cdot \mathbf{W} + \mathbf{L}. \end{aligned} \tag{.5}$$

From (.5), we conclude that if a shape in 3D is subjected to a similarity transform (translation, rotation, scaling, or a combination thereof), then its modeling curve in 3D is subjected to the same group of transforms.

APPENDIX B

Let $\mathbf{T}(t)$, $\mathbf{N}(t)$ and $\mathbf{B}(t)$ be the tangent, normal and binormal vectors, respectively, of the Frenet frame at the time index t. Applying Graham-Schmidt normalization to this frame, we define the orthonormal basis (e_1, e_2, e_3) such that:

$$\|e_2\| = e_2 \cdot e_2 = 1; \tag{.6}$$

$$e_2 \cdot e_1 = 0; \tag{.7}$$

$$e_3 = e_1 \times e_2. \tag{.8}$$

Our task is to define the two turning angles $\alpha_T(t)$ and $\alpha_B(t)$. We recall that $\alpha_T(t)$ is the signed angle formed by the two tangent vectors $\mathbf{T}(t)$ and $\mathbf{T}(t+1)$, and $\alpha_B(t)$ is the signed angle formed by the two binormal vectors $\mathbf{B}(t)$ and $\mathbf{B}(t+1)$. We start by assuming that $\mathbf{T}(t)$ and $\mathbf{T}(t+1)$ construct a plane, *i.e.*, we assume that $\alpha_T(t)$ is not null. We furthermore restrict $\alpha_T(t)$ to be in $]0, \frac{\pi}{2}]$. It follows that $1 > \sin(\alpha_T(t)) > 0$ and $1 > \cos(\alpha_T(t)) > 0$. We may write:

$$\begin{aligned} e_1 &= \mathbf{T}(t); \\ e_2 &= \frac{1}{b}\big(\mathbf{T}(t+1) - a\mathbf{T}(t)\big), \quad \text{with} \quad a, b \neq 0, \end{aligned} \tag{.9}$$

where $a = \cos(\alpha_T(t))$, and $b = \sin(\alpha_T(t))$. The frame at $(t+1)$ needs to verify the following conditions:

$$\begin{aligned} e_1 &= \mathbf{T}(t); \\ e_3 &= \text{sign}(\mathbf{N}(t) \cdot e_3'); \\ e_3' &= \frac{\mathbf{T}(t) \times \mathbf{T}(t+1)}{\|\mathbf{T}(t) \times \mathbf{T}(t+1)\|}. \end{aligned} \tag{.10}$$

$$e_2 = 0 \cdot e_1 + \cos(\alpha_T) \cdot e_3 + \sin(\alpha) \cdot e_2; \tag{.11}$$

$$\text{or} \quad e_2 = \frac{1}{\sin(\alpha)} \cdot \big(\mathbf{N}(t) - \cos(\alpha) e_3\big). \tag{.12}$$

Solving these equations leads to the following result:

$$\Theta = \text{sign}\big(\mathbf{T}(i+1) \cdot \mathbf{N}(t) - \cos(\alpha)\mathbf{T}(t+1) \cdot e_3\big). \tag{.13}$$

Since $\text{sign}(\mathbf{T}(t+1)\cdot e_2) = \text{sign}(\mathbf{T}(t+1)\cdot\mathbf{N}(t) - \cos(\alpha)\mathbf{T}_2\cdot e_3) = \text{sign}(\mathbf{T}(t+1)\cdot\mathbf{N}(t))$, then:

$$\Theta = \text{sign}(\mathbf{T}(t+1)\cdot\mathbf{N}(t))\cdot\arccos(\mathbf{T}(t)\cdot\mathbf{T}(t+1)). \tag{.14}$$