Università di Genova
Dipartimento di Matematica

Istituto di Matematica Applicata
e Tecnologie Informatiche

Analysis and Parameterization of Triangulated Surfaces

Submitted by

Giuseppe Patanè

DIMA, Università di Genova,
Via. Dodecaneso 35, 16146 Genova, Italy
http://www.dima.unige.it

CNR-IMATI, Sezione di Genova
Via. de Marini 6, 16149 Genova, Italy
http://www.ge.imati.cnr.it

Università degli studi di Genova
Dipartimento di Matematica

CNR - Istituto di Matematica Applicata e Tecnologie Informatiche – Sez. di Genova

Dottorato di Ricerca in Matematica e Applicazioni
(XVII ciclo)

Dissertation

Analysis and Parameterization of Triangulated Surfaces

Submitted by
Giuseppe Patanè

Supervisors

Dr. ***Bianca Falcidieno***
Istituto di Matematica Applicata e Tecnologie Informatiche

Dr. ***Michela Spagnuolo***
Istituto di Matematica Applicata e Tecnologie Informatiche

March, 2005

Sommario

Questa tesi tratta l'analisi e la parametrizzazione di superfici rappresentate da griglie triangolari, cioè, surperfici lineari a tratti che ne costituiscono una semplice rappresentazione e che vengono comunemente utilizzate in matematica e scienze dell'informazione. Fornire rappresentazioni equivalenti e ad alto livello di una triangolazione tridimensionale $\mathcal{M}$ è di basilare importanza per poter trattare problemi computazionali e applicativi nei campi di ricerca della Geometria Computazionale, della Grafica Computerizzata, e della Modellazione di Forma. Lo scopo della tesi è di mostrare come rappresentazioni ad alto livello di una data superficie $\mathcal{M}$ possano essere utilizzate per determinare altre descrizioni ad alto livello o equivalenti di $\mathcal{M}$, e viceversa. Inoltre, questa analisi è legata allo studio di proprietà locali e globali di griglie triangolari tridimensionali sulla base dell'informazione che si vuole estrarre, anche significativa per il contesto applicativo in esame.

L'analisi locale di una griglia triangolare tridimensionale arbitraria $\mathcal{M}$ è basata su una sua segmentazione multi-scala cui è associata una parametrizzazione locale, dove l'ipotesi usuale di decomporre $\mathcal{M}$ in una famiglia di regioni omeomorfe a un disco (cioè, regioni di genere zero e con una componente di bordo) è sostituita da una segmentazione di $\mathcal{M}$ in caratteristiche di forma ciascuna delle quali di genere zero senza vincolarne il numero delle componenti di bordo. Tale scelta ed estensione è motivata dalla necessità di identificare le regioni della segmentazione con caratteristiche di forma, di ridurre la distorsione della parametrizzazione, e di fornire un miglior supporto ad applicazioni usuali della parametrizzazione quali l'approssimazione di $\mathcal{M}$ con superfici lineari e non, la mappatura di tessiture, e la compressione.

L'analisi, la caratterizzazione, e l'astrazione globale di $\mathcal{M}$ considerano gli aspetti topologici e geometrici di $\mathcal{M}$ rappresentati dalla struttura combinatoria (cioè, la connettività della griglia) e associati al relativo *embedding* in $\mathbb{R}^3$. La dualità e la regolarizzazione Laplaciana duale sono le prime caratterizzazioni di $\mathcal{M}$ presentate con lo scopo finale di una migliore comprensione delle relazioni tra connettività e geometria della griglia $\mathcal{M}$, come indicato in letteratura, e qui esteso al caso della parametrizzazione 3D. L'analisi globale di $\mathcal{M}$ è stata inoltre trattata attraverso la definizione di una funzione reale su $\mathcal{M}$ che induce un grafo di Reeb invariante rispetto a trasformazioni affini e particolarmente adatto ad applicazioni quali il confronto e il riconoscimento di forma. La teoria di Morse e il grafo di Reeb sono state inoltre utilizzate per definire un metodo nuovo e semplice per risolvere il problema della parametrizzazione globale, cioè, la ricerca di un grafo di taglio di una superficie arbitraria rappresentata da una griglia triangolare tridimensionale.

Le caratteristiche principali dell'approccio proposto rispetto allo stato dell'arte consistono nella capacità di definire una famiglia di grafi di taglio, in luogo di un unico taglio, per superfici con bordo o chiuse, che sono trattate con un unico approccio. Inoltre, ciascun grafo di taglio è regolare e il modo in cui è costruito è basato sulla procedura di taglio di superfici di genere zero utilizzata per la parametrizzazione locale di $\mathcal{M}$. Come discusso nella tesi, definire una famiglia di grafi di taglio garantisce una maggiore flessibilità e effettive semplificazioni nell'analisi, modellazione, e visualizzazione di campi scalari e vettoriali (anche dipendenti dal tempo); infatti la parametrizzazione globale di $\mathcal{M}$ permette di ridurre la "dimensione" dei problemi sopra citati attraverso la riformulazione di ciascuno di essi da una superficie 3D con genere arbitrario a un modello 2.5D (cioè, un modello di elevazione). I risultati della tesi sono rivolti principalmente ai campi di ricerca precedentemente elencati, ma alcuni di essi quali la parametrizzazione locale e globale hanno importanti legami e applicazioni relativi all'analisi del segnale (*wavelets* e compressione), all'Analsi agli Elementi Finiti (approssimazione adattiva di grigile tridimensionali), e all'analisi numerica per l'interpolazione e l'approssimazione di superfici (interpolazione polinomiale, mediante B-splines, e NURBS). Lo scopo finale della tesi è di convertire una grigia triangolare tridimensionale $\mathcal{M}$, così come un problema formulato su $\mathcal{M}$, in una che sia ad essa equivalente (griglia duale), basata sulla semantica (segmentazione in caratteristiche di forma, parametrizzazione locale), ad alto livello (famiglie di grafi di taglio, grafo di Reeb, parametrizzazione globale), e in ogni caso più semplice.

Abstract

This dissertation deals with the analysis and parameterization of surfaces represented by triangle meshes, that is, piecewise linear surfaces which enable a simple representation of 3D models commonly used in mathematics and computer science. Providing equivalent and high-level representations of a 3D triangle mesh $\mathcal{M}$ is of basic importance for approaching different computational problems and applications in the research fields of Computational Geometry, Computer Graphics, Geometry Processing, and Shape Modeling. The aim of the thesis is to show how high-level representations of a given surface $\mathcal{M}$ can be used to find other high-level or equivalent descriptions of $\mathcal{M}$ and viceversa. Furthermore, this analysis is related to the study of local and global properties of triangle meshes depending on the information that we want to capture and needed by the application context.

The local analysis of an arbitrary triangle mesh $\mathcal{M}$ is based on a multi-scale segmentation of $\mathcal{M}$ together with the induced local parameterization, where we replace the common hypothesis of decomposing $\mathcal{M}$ into a family of disc-like patches (i.e., 0-genus and one boundary component) with a feature-based segmentation of $\mathcal{M}$ into regions of 0-genus without constraining the number of boundary components of each patch. This choice and extension is motivated by the necessity of identifying surface patches with features, of reducing the parameterization distortion, and of better supporting standard applications of the parameterization such as remeshing or more generally surface approximation, texture mapping, and compression.

The global analysis, characterization, and abstraction of $\mathcal{M}$ take into account its topological and geometric aspects represented by the combinatorial structure of $\mathcal{M}$ (i.e., the mesh connectivity) with the associated embedding in $\mathbb{R}^3$. Duality and dual Laplacian smoothing are the first characterizations of $\mathcal{M}$ presented with the final aim of a better understanding of the relations between mesh connectivity and geometry, as discussed by several works in this research area, and extended in the thesis to the case of 3D parameterization. The global analysis of $\mathcal{M}$ has been also approached by defining a real function on $\mathcal{M}$ which induces a Reeb graph invariant with respect to affine transformations and best suited for applications such as shape matching and comparison. Morse theory and the Reeb graph were also used for supporting a new and simple method for solving the global parameterization problem, that is, the search of a cut graph of an arbitrary triangle mesh $\mathcal{M}$. The main characteristics of the proposed approach with respect to previous work are its capability of defining a family of cut graphs, instead of just one cut, of bordered and closed surfaces which are treated with a unique approach. Furthermore, each cut graph is smooth and the way it is built is based on the cutting procedure of 0-genus surfaces that was used for the local parameterization of $\mathcal{M}$. As discussed in the thesis, defining a family of cut graphs provides a great flexibility and effective simplifications of the analysis, modeling, and visualization of (time-depending) scalar and vector fields; in fact, the global parameterization of $\mathcal{M}$ enables to reduce the "dimension" of the above-mentioned problems by recasting each of them from a 3D surface with an arbitrary genus to a 2.5D model (i.e., an elevation model). The results of the thesis are mainly targeted to the above-mentioned research fields, but some of them such as the local and global parameterization have important links and applications related to image processing (e.g., wavelets and compression), FEMs (e.g., adaptive remeshing of 3D grids), and numerical analysis for surface interpolation and approximation (e.g., interpolation with polynomial functions, B-splines, and Non-Uniform Rational B-Splines (NURBS)). The final aim of the thesis is to convert an arbitrary triangle mesh $\mathcal{M}$, as well as a problem formalized on $\mathcal{M}$, to an equivalent (e.g., dual mesh), approximated (e.g., remeshed surface), semantic-based (e.g., feature-based segmentation and local parameterization), high-level (e.g., family of cut graphs, Reeb graph, global parameterization), and hopefully simpler, representation of $\mathcal{M}$ or formulation of the problem.

Acknowledgments

This thesis is the result of the period that I spent as a research fellow and member of the Shape Modeling Group at Imati-CNR in Genoa, joined after a fellowship of the Italian Institute of Advanced Mathematics “F. Severi” at the University of Milan. My thanks go to Bianca Falcidieno and Michela Spagnuolo who gave me the possibility of starting this long period of collaboration, supporting me through the development of my background on Shape Modeling, giving me the chance of pursuing my own ideas, and showing me the meaning of collaborative research.

At the same time, special thanks go to the members of the Shape Modelling Group; to Dr. Michela Mortara and Dr. Silvia Biasotti with whom I shared and enjoyed the challenge of writing papers on our common research activity, to my office mates dr. Marco Attene and dr. Maria Grazia Ierardi, to dr. Chiara Catalano, Antonella Galizia, Simone Marini, and Corrado Pizzi, to my swimming-pool mate dr. Daniele D’Agostino, and to the other colleagues who I met during these years at Imati-CNR.

My deep gratitude goes to my family, to Paolo and Maria, and to Fam. Beghi in Milan for the special support during the period spent in Milan, and during other important moments.

This thesis was partially supported by the EC-IST FP6 Network of Excellence “AIM@SHAPE”: *Advanced and Innovative Models and Tools for the development of Semantic-based systems for Handling, Acquiring, and Processing knowledge Embedded in multidimensional digital objects* [AIM].

Table of Contents

List of Figures

List of Tables

Introduction

This dissertation deals with computational geometry and topology, and it tackles the problem of analyzing, representing, and abstracting surfaces represented by triangle meshes, that is, piecewise linear surfaces which enable a simple representation of 3D models commonly used in mathematics and computer science. Triangulated surfaces are generated by polygonizing implicit functions, or sampling parametric surfaces, or scanning real 3D objects with optical devices. All these generation processes provide complex discrete models with arbitrary genus and curvature which are usually unsatisfactory for mathematical modeling (i.e., topological and geometric analysis), numerical simulations, and approximation. In fact, they may consist of a huge number of vertices many of them being redundant, and the vertex sampling as well as the mesh connectivity may be irregular. Therefore (see Figure 1), a basic task consists of improving an arbitrary triangulated surface $\mathcal{M}$ in terms of sampling density and distribution, regular or semi-regular connectivity, and mesh *quality* (e.g., size and shape of triangles). Other main issues to be taken into account are the relation between the accuracy and size of the approximation, the *validity* of the new representation which has to tile $\mathcal{M}$ without self-intersections and to best approximate the original shape geometry (i.e., *fidelity* of the representation).

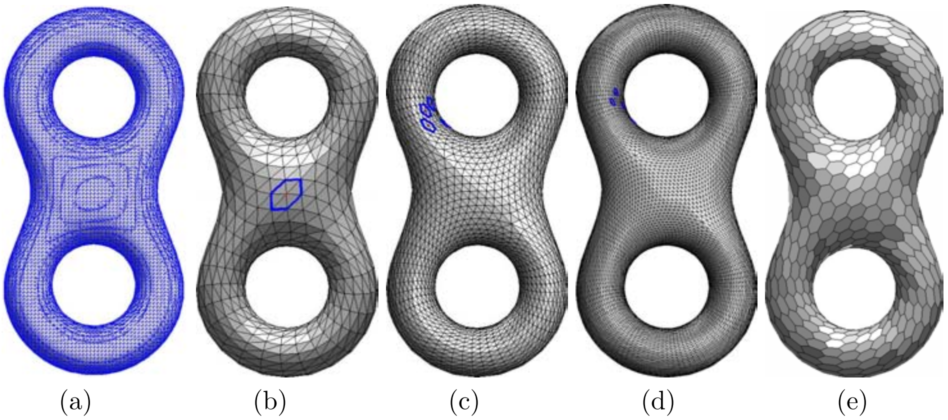


Figure 1: (a) Input triangle mesh $\mathcal{M}$ generated by polygonizing an implicit function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$, i.e. $\mathcal{M} = \{p \in \mathbb{R}^3 : f(p) = 0\}$; note the presence of tiny triangles which make unstable the convergence of Finite Element Methods. (b-d) Equivalent representations of $\mathcal{M}$ at different levels of resolution (i.e., number of vertices), (e) dual mesh.

These aims are achieved by building a set of *equivalent representations* of $\mathcal{M}$, possibly where different types of geometric models (e.g., piecewise linear, polynomial, B-splines, Non-Uniform Rational B-spline (NURBS), and implicit functions) are used to approximate $\mathcal{M}$. Providing equivalent representations of $\mathcal{M}$ is of basic importance for approaching different computational problems and applications which take as input a previously meshed surface. For instance, Finite Element Methods (FEM) require a triangle mesh with triangles whose angles are greater than a pre-defined threshold in order to guarantee the stability and convergence of approximated and discrete solvers of Partial Differential Equations (PDE's), commonly used to calculate mechanical stress or solve heat and flow equations. The categories of applications range from geometric modeling to simulation, and in the thesis we will focus our discussion on surface approximation, duality, editing, smoothing, and fairing.

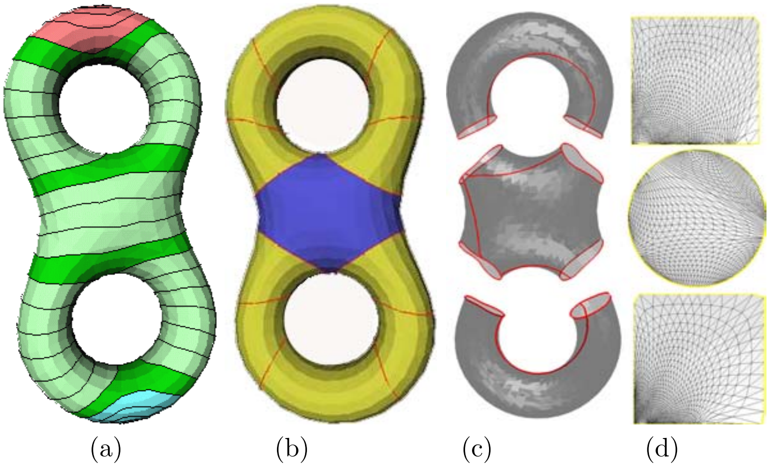


Figure 2: (a) Topology-based segmentation and (b) feature-based segmentation of the bitorus. (c-d) Local parameterization parameterization with respect to the patch decomposition shown in (b); on each patch, the red line represents the cut that has been used for its unfolding.

Beside the reduction of the complexity and optimization of $\mathcal{M}$, surface properties, such as normal vectors, curvature values, and critical points, provide only *local information* about the geometric and topological features of $\mathcal{M}$ and they lack in providing a *global* characterization of $\mathcal{M}$. Therefore, the next step consists of associating a set of *high-level representations* to $\mathcal{M}$ which extract and organize the geometric and topological information of $\mathcal{M}$ to reflect and/or to make explicit its sub-parts. The choice of these sub-parts depends on the use of the characterization of $\mathcal{M}$, and their invariance under rotations, translations, and scalings is important for studying the shape of 3D surfaces; for instance, ridges given by the extrema of the principal curvatures along their corresponding curvature lines are important view- and scale-independent features of a smooth surface. Therefore, among specific criteria used for the extraction of the high-level representations we can identify some basic properties that the final representation has to fulfill such as *invariance* with respect to a chosen set of transformations (e.g., affine transformations), *stability* to local noise and surface perturbations which may affect both the geometry and connectivity, *computational feasibility*, and *expressiveness* of the representation. These requirements for the identification of high-level shape representations lead to the necessity of descriptions based on shape segmentation. At the same time, the problem of decomposing a given surface into sub-parts is strongly intercorrelated with the problem of describing those parts and cannot be performed independently. Segmenting a 3D surface $\mathcal{M}$ into basic parts enables to describe its shape in terms of components, their characteristics and spatial arrangement; then, the final result is an abstract description of $\mathcal{M}$ that can be more useful for many purposes such as interpretation, matching, retrival, local parameterization and approximation. Following these principles, the thesis discusses two approaches to surface segmentation based on topological and geometric analysis. *Topology-based segmentation* identifies patches of $\mathcal{M}$ by studying the evolution of the iso-contours of a given function $f : \mathcal{M} \rightarrow \mathbb{R}$ and each patch is classified taking into account its genus and number of boundary components (see Figure 2(a)). *Feature-based segmentation* (see Figure 2(b)) generates patches of $\mathcal{M}$ by clustering those points which share common local properties such as curvature values, closeness with respect to the geodesic distance, or membership to a region with a specific structure (e.g., planes, generalized cone and cylinders). Other examples of high-level representations are the Reeb graph of $\mathcal{M}$ induced by a function $f : \mathcal{M} \rightarrow \mathbb{R}$ (see Figure 3(a)), which synthetically represents $\mathcal{M}$ preserving the topological properties and the main morphological characteristics, and cut graphs (see Figure 3(b)) commonly used for surface classification through the evaluation of polygonal schemes.

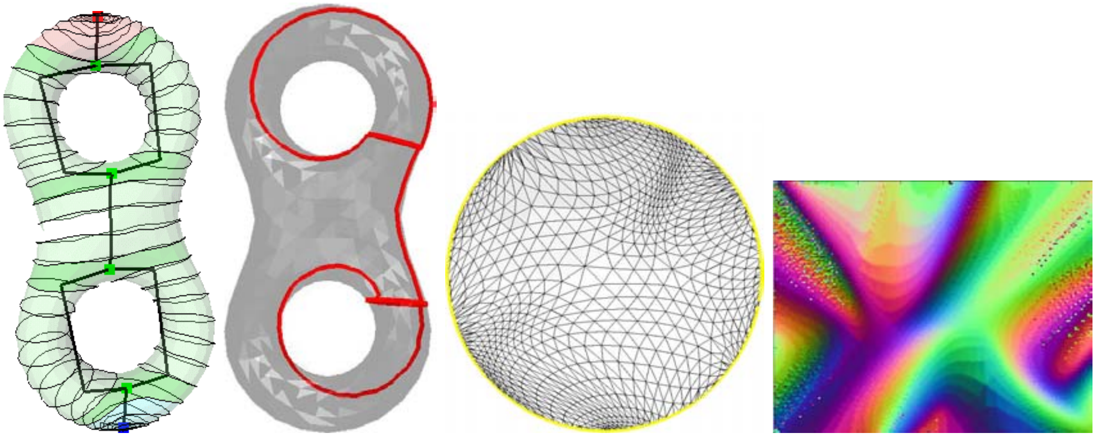


Figure 3: High-level representations of the bitorus: (a) Reeb graph induced by a harmonic function with a unique maximum (red point) and minimum (blue point), (b) cut graph γ, (c) global parameterization (and polygonal schema) of the input triangle mesh, (d) geometry image.

Extracting high-level representations of a given surface $\mathcal{M}$ is beneficial to a wealth of applications; for instance, surface classification relies on the use of topological invariants (e.g., orientability, genus, number of boundary components) which can be coded in high-level descriptors such as cut-graphs and the Reeb graph, surface recognition and interpretation are based on the identification of parts of $\mathcal{M}$ which characterize the semantic meaning of the surface with respect to the application context.

Previous considerations convey the idea that a multi-layered modeling of computational problems in the research fields of Computational Geometry, Computer Graphics, Geometry Processing, and Shape Modeling have to study the different aspects and representations of the input surface $\mathcal{M}$. The aim of the thesis is to show how high-level descriptions of a given surface $\mathcal{M}$ can be used to find other high-level or equivalent representations of $\mathcal{M}$ and viceversa. For instance, we will compute a family of cut graphs $\{\gamma\}$ of $\mathcal{M}$ by evaluating the Reeb graph R_G of $\mathcal{M}$ with respect to a mapping function $f : \mathcal{M} \rightarrow \mathbb{R}$; therefore, R_G and each cut graph in $\{\gamma\}$ (see Figure 3(a-b)) represent different high-level descriptions of $\mathcal{M}$. At the same time, the *global parameterization* $\varphi : \mathcal{M} \rightarrow \Omega \subseteq \mathbb{R}^2$ of $\mathcal{M}$ with respect to any cut graph in $\{\gamma\}$ is a simplicial isomorphism (see Figure 3(b-c)) which enables to define equivalent representations by using standard methods of surface approximation, and to improve visualization and compression techniques. The global parameterization provides a simple approach to surface approximation by using *remeshing* techniques; here, the parameter domain Ω is resampled and the new triangulation is projected back into 3D space by φ^{-1}, thus resulting in an improved version of the original model. Finding a global parameterization of the input mesh is a complex problem which is usually computationally expensive, suffers from accuracy issues, and requires restrictions on the mesh. These constraints are due to the problem of mapping a nontrivial 3D structure with an arbitrary genus and number of boundary components to a 2D parametric domain, while minimizing the induced metric distortion and the loss of information usually located by high-curvature regions of $\mathcal{M}$. Furthermore, if the parameterization is based on a segmentation of $\mathcal{M}$ (i.e., *local parameterization*) it is possible to have discontinuities of the embedding along the boundary components of adjacent patches, or to create an over-segmentation of the surface whose patches do not identify features of $\mathcal{M}$. If $\mathcal{M}$ is parameterized into a 2D square by using a stretch-minimization method, it can be represented as a *geometry image* which stores the geometry as well as any other vertex attribute (e.g., normal vectors, colors) as RGB values of a compact grid structure (see Figure 3(d)). The compactness and regularity of this data structure implicitly stores the connectivity, improves and facilitates the implementation of several algorithms such as efficient rendering with no cache indirection, texture and transparency mapping. Coding $\mathcal{M}$ as a range image shifts 3D signal processing into image (or 2D signal) analysis; then, this approach makes available techniques developed in the 2D setting to the 3D case stressing the central role of global parameterization and motivating our choice of developing new, stable, efficient, and simple methods for the evaluation of families of cut graphs of arbitrary triangle meshes.

Converting a given representation of $\mathcal{M}$ into an equivalent or high-level one has a central role in applied mathematics and also important links to new research fields where the information implicitly or explicitly stored and represented by digital surfaces (e.g, discrete surfaces, images) has to be extracted, analyzed, and compared [AIM]. In the following, we introduce the structure and the main contributions of the thesis; conclusions and future work are discussed in Chapter 5.

Overview

In **Chapter 1**, we consider basic relations between a 2-manifold triangle mesh $\mathcal{M}$ and its dual representation $\mathcal{M}'$. The achieved combinatorial properties represent the starting point for the reconstruction algorithm which maps $\mathcal{M}^{'}$ to its primal representation $\mathcal{M}$, thus defining their geometric and topological identification. This correspondence is further analyzed in order to study the influence of the information in $\mathcal{M}$ and $\mathcal{M}^{'}$ for the reconstruction process. Therefore, the first goal of the chapter consists of defining a homeomorphism between a 2-manifold triangle mesh and its dual representation with optimal (i.e., linear) computational cost and numerical stability avoiding the least-squares formulation whose solution requires $O(n_V^3)$ flops, with n_V number of vertices of $\mathcal{M}$, and it only achieves an approximated reconstruction of the input mesh. We also derive two combinatorial properties which highlight the redundancy of the geometry stored in a triangle mesh. The second goal is the definition of the "*dual Laplacian smoothing*", which combines the application to the dual mesh $\mathcal{M}^{'}$ of well-known smoothing algorithms with an inverse transformation for reconstructing the regularized triangle mesh. The use of $\mathcal{M}^{'}$ instead of $\mathcal{M}$ exploits a topological mask different from the 1-neighborhood one, related to Laplacian-based algorithms, guaranteeing good results and optimizing storage and computational requirements.

Tools for the automatic decomposition of a surface into shape features will facilitate the editing, matching, texturing, morphing, compression, and simplification of 3D shapes. Different features, such as flats, limbs, tips, pits, and various blending shapes that transition between them may be characterized in terms of local curvature and other differential properties of the surface or in terms of a global skeletal organization of the volume it encloses. However, both solutions are extremely sensitive to small perturbations in the surface smoothness and to quantization effects when they operate on triangulated surfaces. In **Chapter 2**, we propose a multi-resolution approach, which not only estimates the curvature of a vertex over neighborhoods of variable size, but also takes into account the topology of the surface in that neighborhood. The approach is based on blowing a spherical bubble at each vertex and studying how the intersection of that bubble with the surface evolves. We describe an efficient approach for computing these characteristics for a sampled set of bubble radii and for using them to identify features, based on easily formulated filters, that may capture the needs of a particular application.

Chapter 3 deals with the continuous and discrete definition and analysis of scalar fields $f : \mathcal{M} \rightarrow \mathbb{R}$ on smooth and discrete surfaces. We also discuss the definition of topological graph-like representations (i.e., the Reeb graph) which provide a powerful and synthetic sketch of the surface. Then, we present the extraction of a Reeb graph achieved by expanding topological rings on $\mathcal{M}$ starting from high-curvature regions identified by the algorithm described in Chapter 2. Among the possible applications of these concepts (e.g., modeling, morphing, matching, and recognition), in Chapter 4 we use this theory for generating an atlas decomposition of an arbitrary surface finalized at the local parameterization problem and the Reeb graph is used for solving the global parameterization of surfaces with an arbitrary genus.

Chapter 4 describes a novel approach to the local and global parameterization of triangle meshes representing 2-manifolds with an arbitrary genus. Concerning the local parameterization, decompositions of the shape based on several criteria (i.e., topology, morphology, and geometry) are computed and used to segment the shape into primitives, which define a chart decomposition of the mesh. Then, each chart is parameterized using an extension of the barycentric coordinates method. The charts are all 0-genus and can be of three types only, depending on the number of boundary components. The chart decomposition and the parameterization are used to define a shape-graph where each node represents one primitive and the arcs code the adjacency relations among the primitives. The primitives are coded together with their skeletal lines that are computed from and aligned with their parameterization. Concerning the global parameterization, given an arbitrary triangle mesh $\mathcal{M}$ our approach searches a cut graph made of the iso-contours of a fair function $f : \mathcal{M} \rightarrow \mathbb{R}$ and it works in a planar domain where geodesic curves are defined by line segments whose counterparts on $\mathcal{M}$, with respect to an appropriate diffeomorphism $\phi : \mathcal{M} \rightarrow \mathbb{R}^2$, give smooth approximations of geodesic paths. The emphasis of the proposed approach is on the definition of a simple method for finding a family of cut graphs of $\mathcal{M}$ and guided by several criteria which spread from the global parameterization (e.g., minimal length, minimization of the parameterization distortion, or interpolation of points as required by remeshing and texture mapping) to the calculation of polygonal schemes for surface classification. Finally, **Chapter 5** discusses various applications to surface deformation and approximation, and perspectives of future works.

Contribution

The contributions of the thesis related to the local analysis of an arbitrary triangle mesh $\mathcal{M}$ is represented by the multi-scale segmentation of $\mathcal{M}$ together with the induced local parameterization, which has replaced the common hypothesis of decomposing $\mathcal{M}$ into a family of disc-like patches (i.e., 0-genus and one boundary component) with a feature-based segmentation of $\mathcal{M}$ into regions of 0-genus without constraining the number of boundary components of each patch. This choice and extension is motivated by the necessity of identifying surface patches with features, of reducing the parameterization distortion, and of better supporting standard applications of the parameterization such as remeshing or more generally surface approximation, texture mapping, and compression. The proposed approach builds on a new parameterization method of a 0-genus surface $\mathcal{P}$ with k boundary components and which consists of reducing $\mathcal{P}$ to a disc-like surface through a cut γ which is evaluated in linear time, is not affected by the mesh connectivity and geometry (e.g., irregular sampling, tiny triangles, noise), and can be constrained to interpolate source points with the possibility of admitting bifurcations of γ. These elements are the most significant characteristics of the proposed framework and not shared by the previous work on local parameterization; furthermore, the generality of the cutting procedure has been used for approaching the global parameterization problem.

The contributions related to the global analysis, characterization, and abstraction of $\mathcal{M}$ take into account its topological and geometric aspects represented by the combinatorial structure of $\mathcal{M}$ (i.e., the mesh connectivity) with the associated embedding in $\mathbb{R}^3$. Duality and dual Laplacian smoothing are the first characterizations of $\mathcal{M}$ presented with the final aim of a better understanding of the relations between mesh connectivity and geometry, as discussed by several works in this research area, and extended in the thesis to the case of 3D parameterization in Chapter 4. The global analysis of $\mathcal{M}$ has been also approached by defining a real function on $\mathcal{M}$ which induces a Reeb graph invariant with respect to affine transformations and best suited for applications such as shape matching and comparison [BMM$^+$03]. Morse theory and the Reeb graph were also used for supporting a new and simple method for solving the global parameterization problem, that is, the search of a cut graph of an arbitrary triangle mesh $\mathcal{M}$. The main characteristics of the proposed approach with respect to previous work are its capability of defining a family of cut graphs, instead of just one cut, of bordered and closed surfaces which are treated with a unique approach. Furthermore, each cut graph is smooth and the way it is built is based on the cutting procedure of 0-genus surfaces that was used for the local parameterization of $\mathcal{M}$. As discussed in the thesis, defining a family of cut graphs provides a great flexibility and effective simplifications of the analysis, modeling, and visualization of (time-depending) scalar and vector fields; in fact, the global parameterization of $\mathcal{M}$ enables to reduce the "dimension" of the above-mentioned problems by recasting each of them from a 3D surface with an arbitrary genus to a 2.5D model (i.e., an elevation model). The results of the thesis are mainly targeted to the above-mentioned research fields, but some of them such as the local and global parameterization have important links and applications related to image processing (e.g., wavelets and compression), FEMs (e.g., adaptive remeshing of 3D grids), and numerical analysis for surface interpolation and approximation (e.g., interpolation with polynomial functions, B-splines, NURBS). The final aim of the thesis is to convert an arbitrary triangle mesh $\mathcal{M}$, as well as a problem formalized on $\mathcal{M}$, to an equivalent (e.g., dual mesh), approximated (e.g., remeshed surface), semantic-based (e.g., feature-based segmentation and local parameterization), high-level (e.g., family of cut graphs, Reeb graph, global parameterization), and hopefully simpler, representation of $\mathcal{M}$ or formulation of the problem.

Chapter 1

Triangle mesh duality: reconstruction and smoothing

In the first part of the chapter we introduce basic concepts related to surface topology and polyhedral surfaces which represent the input of all the methods discussed in the thesis. Then, we focus our attention on duality considering the relations between a 2-manifold triangle mesh $\mathcal{M}$ and its dual representation $\mathcal{M}'$. The achieved combinatorial properties represent the starting point for the reconstruction algorithm which maps $\mathcal{M}^{'}$ to its primal representation $\mathcal{M}$, thus defining their geometric and topological identification. This correspondence is further analyzed in order to study the influence of the information in $\mathcal{M}$ and $\mathcal{M}^{'}$ for the reconstruction process. Therefore, the first goal of the chapter consists of defining a homeomorphism between a 2-manifold triangle mesh and its dual representation with optimal (i.e., linear) computational cost and numerical stability avoiding the least-squares formulation whose solution requires $O(n_V^3)$ flops, with n_V number of vertices of $\mathcal{M}$, and it only achieves an approximated reconstruction of the input mesh. We also derive two combinatorial properties which highlight the redundancy of the geometry stored in a triangle mesh. The second goal is the definition of the "*dual Laplacian smoothing*", which combines the application to the dual mesh $\mathcal{M}^{'}$ of well-known smoothing algorithms with an inverse transformation for reconstructing the regularized triangle mesh. The use of $\mathcal{M}^{'}$ instead of $\mathcal{M}$ exploits a topological mask different from the 1-neighborhood one, related to Laplacian-based algorithms, guaranteeing good results and optimizing storage and computational requirements.

1.1 Topology of surfaces

In the following sections, we review basic concepts related to simplicial complexes, surface topology and classification; for further readings, we refer the reader to [DoC76, GP74, Mas67, Mor85]. This initial background is the minimal one required for starting the reading of the thesis; details on these and related concepts will be introduced in each chapter which uses them for the development of the presentation.

1.1.1 Simplices and simplicial complexes

We introduce basic definitions and concepts related to simplicial complexes.

Definition 1. *A finite set of points $a_0, \ldots, a_n$ in $\mathbb{R}^m$ is said to be* affinely dependent *if there exist coefficients $\alpha_0, \ldots, \alpha_n$, not all zero, such that $\sum_{j=0}^{n} \alpha_j a_j = 0$ and $\sum_{j=0}^{n} \alpha_j = 0$. Otherwise, they are called* affinely-independent.

Definition 2. *If $A \subseteq \mathbb{R}^m$, the* affine hull *of A consists of the points $x \in \mathbb{R}^m$ of the form $x = \sum_{j=0}^{n} \alpha_j a_j$, where $a_j \in A$, $\alpha_j \in \mathbb{R}$, $j = 0, \ldots, n$, $\sum_{j=0}^{n} \alpha_j = 1$.*

Let H be the affine hull of an affinely independent sequence of points $a_0, \ldots, a_n$. Every map $f : H \rightarrow \mathbb{R}^p$ with the property

$$f\left(\sum_{j=0}^{n} \alpha_j a_j\right) = \sum_{j=0}^{n} \alpha_j f(a_j),$$

where $\sum_{j=0}^{n} \alpha_j = 1$, is called an *affine transformation.* A bijective, affine transformation between two affine subspaces is called an *affine isomorphism.*

Let the points $a_0, \ldots, a_n$ of a m-dimensional Euclidean space $\mathbb{R}^m$ be an affinely independent set (see Definition 1); the *convex hull* of this set of points

$$\Delta(a_0, \ldots, a_n) := \left\{ \sum_{j=0}^{n} \alpha_j a_j : \quad \alpha_j \geq 0,\, j = 1, \ldots, n, \quad \sum_{j=1}^{n} \alpha_j = 1 \right\}$$

is called *n-dimensional simplex of vertices* $a_0, \ldots, a_n$. Therefore, the 0-dimensional simplex $\Delta(a_0)$ consists of the point a_0, the 1-dimensional simplex $\Delta(a_0, a_1)$ is the non-degenerate line segment with end-points a_0, a_1, the 2-dimensional simplex is the triangle with non-collinear vertices a_0, a_1, a_2, and finally the 3-dimensional simplex is the tetrahedron with non-co-planar vertices a_0, a_1, a_2, a_3. The empty set is considered as (-1)-dimensional simplex.

Let $H(a_0, \ldots, a_n)$ be the n-dimensional affine subspace in $\mathbb{R}^n$ uniquely defined by the set of vertices of the simplex $\Delta(a_0, \ldots, a_n)$; then,

- $\Delta(a_0, \ldots, a_n) \subseteq H(a_0, \ldots, a_n)$. The set $H(a_0, \ldots, a_n)$ is called the *carrier* of $\Delta(a_0, \ldots, a_n)$;
- $\Delta(a_0, \ldots, a_n)$ is closed in $H(a_0, \ldots, a_n)$;
- every simplex is compact;
- every n-dimensional simplex is homeomorphic to the n-dimensional closed unit ball.

For any subset $\{a_{i_0}, \ldots, a_{i_k}\}$ of an affinely independent set of points $\{a_0, \ldots, a_n\}$ in $\mathbb{R}^m$, the simplex $\Delta(a_{i_0}, \ldots, a_{i_k})$ is called *k-dimensional face* of the simplex $\Delta(a_0, \ldots, a_n)$. The $(n-1)$-dimensional simplices are called *facets.*

Definition 3. *A finite set $\mathcal{K}$ of simplices lying in the Euclidean space $\mathbb{R}^m$ is called a* simplicial complex *if the following conditions hold:*

1. *the family $\mathcal{K}$ contains all the faces of each simplex in the family;*
2. *the intersection of any pair of simplices of $\mathcal{K}$ is a common face.*

It is possible to prove that the second condition is equivalent to require that the interiors of distinct simplices in the family $\mathcal{K}$ are disjoint. If $\mathcal{K}$ is a non-empty simplicial complex, the number $\dim(\mathcal{K}) := \max\{\dim(\Delta) : \Delta \in \mathcal{K}\}$ is called *dimension* of $\mathcal{K}$, and the set $|\mathcal{K}| := \cup\{\Delta : \Delta \in \mathcal{K}\}$ is called the *underlying space* of the complex $\mathcal{K}$. Any subset X of $\mathbb{R}^m$ for which there exists a simpicial complex $\mathcal{K}$ with $|\mathcal{K}| = X$ is defined as *polyhedron*; the complex $\mathcal{K}$ is a *triangulation* of the polyhedron X (see also Section 1.1.2).

Proposition 1. *The following properties of polyhedra hold:*

- *every polyhedron is compact;*
- *the prolyhedron $|\mathcal{K}|$ is connected if and only if the complex $\mathcal{K}$ is connected;*
- *if $|\mathcal{K}| = |\mathcal{K}^{'}|$, then $dim(\mathcal{K}) = dim(\mathcal{K}^{'})$.*

In defining an n-dimensional simplex we assumed that its vertices are affinely independent, and therefore they are distinct points in $\mathbb{R}^m$. Before introducing the definition of simplicial mappings, we introduce the following notation: we use $\{a_0, \ldots, a_n\}$ if repetitions of the points are admitted and the corresponding simplex is denoted with $\Delta\{a_0, \ldots, a_n\}$. Let $\mathcal{K}$ and $\mathcal{L}$ be simplicial complexes with respective vertex sets $\mathcal{K}^\circ := \{a_0, \ldots, a_n\}$ and $\mathcal{L}^\circ$. A map $\varphi^\circ : \mathcal{K}^\circ \to \mathcal{L}^\circ$ is called a *simplicial map of the vertices* if for every set of vertices $a_{j_0}, \ldots, a_{j_n}$ which determines a simplex $\Delta(a_{j_0}, \ldots, a_{j_n}) \in \mathcal{K}$ the vertices $\varphi(a_{j_0}), \ldots, \varphi(a_{j_n})$ determine a simplex $\Delta\{\varphi(a_{j_0}), \ldots, \varphi(a_{j_n})\} \in \mathcal{L}$. A one-to-one simplicial map is called *simplicial isomorphism.*

Let $\varphi : \mathcal{K} \to \mathcal{L}$ be a simplicial map and let $\mathcal{K}^\circ := \{a_0, \ldots, a_n\}$ and $\mathcal{L}^\circ := \{b_0, \ldots, b_l\}$ be the vertex sets of $\mathcal{K}$ and $\mathcal{L}$, respectively. The map φ induces a map $|\varphi| : |\mathcal{K}| \to |\mathcal{L}|$ by using the barycentric coordinates defined by

$$|\varphi| \left(\sum_{i=0}^{n} \alpha_i a_i \right) = \sum_{j=0}^{l} \beta_j b_j,$$

where β_j equals the sum of all the α_i for which $\varphi(a_i) = b_j$, if any exists, and is otherwise zero. Therefore, we have that $\beta_j \geq 0$ for $j = 0, \ldots, l$, and $\sum_{i=0}^{k} \alpha_i = \sum_{j=0}^{l} \beta_j = 1$ (see also Section 4.5); furthermore, since φ is simplicial we have that if $x \in |\mathcal{K}|$ then $|\varphi|(x) \in |\mathcal{L}|$. The following proposition gives some basic properties of simplicial functions.

Proposition 2.

- *The composition of two simplicial maps (resp., isomorphisms) is a simplicial map (resp., isomorphism);*
- *for every simplicial map $\varphi : \mathcal{K} \to \mathcal{K}$, the induced map $|\varphi| : |\mathcal{K}| \to |\mathcal{L}|$ is continuous;*
- *if $\varphi : \mathcal{K} \to \mathcal{L}$ and $\psi : \mathcal{L} \to \mathcal{M}$, then $|\psi\varphi| = |\psi||\varphi|$. Furthermore, $|id_{\mathcal{K}}| = id_{|\mathcal{K}|}$;*
- *if φ is a simplicial isomorphism, then $|\varphi|$ is a homeomorphism.*

1.1.2 Topology of surfaces and polyhedral surfaces

In this section, we give some basic notions related to topological spaces and polyhedral surfaces.

Definition 4. *A* topological space (X, τ) *is a set X with a collection τ of subsets of X, called* open sets*, such that:*

1. *the empty set $\emptyset$ and X are open;*
2. *any union of open sets is an open set;*
3. *any finite intersection of open sets is open.*

Using the notion of *neighborhood* of each point $x \in X$ (i.e., a set containing x and open in (X, τ)) the notion of continuity, limit, etc. are defined; for more details, we refer the reader to [DoC76, GP74, Mas67, Mor85].

Let us consider a topological space made of a disjoint union P of simple polygons (i.e., homeomorphic to a disc and without holes); we suppose that an orientation of the edges of each polygon has been fixed and let $(e_1, \ldots, e_n)$ be a partition of these edges. We create the quotient space $P^{'}$ by identifying the edges e_i, $i = 1, \ldots, n$. This identification of edges corresponds to an identification on vertices of the polygons of P; in fact, two vertices are identified on the surface if and only if they are the sources or targets of two edges of P belonging to a same set e_i. Then, a *polyhedral surface* is a topological space obtained identifying the vertices and edges of a finite number of simple polygons, which become the *faces* of the polyhedral surface, in such a way that:

- the resulting space is a surface (each point has a neighborhood homeomorphic to a disc or half-disc);
- two distinct edges or vertices of a given polygon are not identified on the surface;
- the intersection of two distinct polygons is either empty, a common vertex, or a common edge.

The above conditions guarantee that the topological space $P^{'}$ (i.e., a polyhedral surface) is a surface; the viceversa also applies as stated by the following theorem [DoC76, GP74, Mas67, Mor85].

Theorem 1. *Any (compact) surface is homeomorphic to a polyhedral surface.*

Therefore, the study of surface topology can be done on polyhedral surfaces represented as a finite graph $\mathcal{G} := \mathcal{G}(V, E)$ such that:

- each face of $\mathcal{G}$ is homeomorphic to the open unit disc and incident to at least three edges;
- the boundary of each face is made of a cycle which alternates distinct vertices to distinct edges;
- the closure of two distinct faces are disjoint, or share only one vertex or edge.

1.1.3 Manifolds

Let $\mathcal{M}$ be a compact metric space and m a non-negative integer.

Definition 5. *A topological Hausdorff space $\mathcal{M}$ is a m-*dimensional topological manifold without boundary *if each point $p \in \mathcal{M}$ admits a neighborhood $U_i \subseteq \mathcal{M}$ homeomorphic to the open disc $B^m := \{x \in \mathbb{R}^m : \|x\|_2 < 1\}$ and $\mathcal{M} = \cup_{i \in \mathbb{N}} U_i$.*

Definition 6. *A topological Hausdorff space $\mathcal{M}$ is a m-*dimensional topological manifold with boundary *if each point $p \in \mathcal{M}$ admits a neighborhood $U_i \subseteq \mathcal{M}$ homeomorphic either to the open disc $B^m := \{x \in \mathbb{R}^m : \|x\|_2 < 1\}$ or to the open half-space $\mathbb{R}^{m-1} \times \{x_n \in \mathbb{R} : x_n \geq 0\}$ and $\mathcal{M} = \cup_{i \in \mathbb{N}} U_i$.*

The set of points of an m-dimensional manifold $\mathcal{M}$ for which there exists a set $U \subseteq \mathcal{M}$ homeomorphic to the m-dimensional open ball B^m such that $p \in U^\circ$ is called the *interior* $\mathcal{M}^\circ$ of $\mathcal{M}$; the complement $\partial\mathcal{M} := \mathcal{M} \backslash \mathcal{M}^\circ$ is the *boundary* of the manifold $\mathcal{M}$ and it is characterized by the following theorem.

Theorem 2. *If the boundary of an m-dimensional manifold is non-empty, then it is a $(m-1)$-dimensional manifold.*

A homeomorphism $\varphi_i : U_i \rightarrow B^m$ is associated to each open subset U_i and each pair (U_i, φ_i) is called a *map* or a *chart*; the union of charts $\{(U_i, \varphi_i)\}$ is called the *atlas* of the manifold $\mathcal{M}$; to each atlas on a manifold there is associated the concept of *transition function.* Let U_i and U_j be two arbitrary charts and $U_i \cap U_j$ be their intersection. On this intersection set, two coordinate maps $\varphi_i : U_i \cap U_j \rightarrow \varphi_i(U_i \cap U_j)$ and $\varphi_j : U_i \cap U_j \rightarrow \varphi_j(U_i \cap U_j)$ are defined; therefore, their composition $\varphi_{i,j} : \varphi_i(U_i \cap U_j) \rightarrow \varphi_j(U_i \cap U_j)$ such that $\varphi_{i,j} := \varphi_j \circ \varphi_i^{-1}$, is well defined and is called a *transition map.*

Definition 7. *A manifold $\mathcal{M}$ is called* orientable *if there exists an atlas $\{(U_i, \varphi_i)\}$ on it such that the Jacobian of all transition maps $\varphi_{i,j}$ from a chart to another one are positive for all intersecting pairs of regions. Manifolds which do not satisfy this property are called* non-orientable.

Let $f : \mathcal{M} \to \mathcal{N}$ be a function from a smooth m-manifold $\mathcal{M}$ into a smooth n-manifold $\mathcal{N}$; f is *smooth* if it is smooth when expressed in local coordinates. The *differential* df_p of f at p is the linear map

$$df_p : T_p\mathcal{M} \to T_{f(p)}\mathcal{N}$$

from the tangent space to $\mathcal{M}$ at p into the tangent space to $\mathcal{N}$ at $f(p)$, defined in local coordinates by the Jacobian matrix df of f

$$df = \begin{pmatrix} \frac{\partial y^1}{\partial x_1} & \cdots & \frac{\partial y^n}{\partial x_1} \\ \frac{\partial y^1}{\partial x_m} & \cdots & \frac{\partial y^n}{\partial x_m} \end{pmatrix}$$

where $(x^1, \ldots, x^m)$ and $(y^1, \ldots, y^n)$ are two systems of local coordinates in a neighborhood U of p and W of $f(p)$ respectively. If $(a^1, \ldots, a^m) \in T_p\mathcal{M}$ is a tangent vector to $\mathcal{M}$ at p, its image under df_p is the vector $b \in T_{f(p)}\mathcal{N}$ with components

$$b^i = \sum_{j=1}^{m} \frac{\partial y^i}{\partial x^j} a^j, \quad i = 1, \ldots, n.$$

From now on, we omit the subscript p in df_p in order to simplify the notation.

Definition 8. *Let $f : \mathcal{M} \to \mathcal{N}$ be a smooth map; a point $p \in \mathcal{M}$ is a* regular point *of f if the differential df_p is surjective. A point $q \in \mathcal{N}$ is* regular *if all points in its inverse image $f^{-1}(q) := \{p \in \mathcal{M} : f(p) = q\}$ are regular points of f. A point or a value that is not regular is called* singular *or* critical.

If p is a regular point, it follows that $m \geq n$ and the Jacobian matrix of f at p has rank n.

Theorem 3. *Let $f : \mathcal{M} \to \mathcal{N}$ be a smooth map and $q \in f(\mathcal{M})$ be a regular value of f; then, the inverse image $f^{-1}(q)$ is a smooth manifold of dimension $(m - n)$.*

For more details on the case $m = 2$, we refer the reader to Section 3.2 and 4.2.

1.1.4 Classification and decomposition of surfaces

This section introduces *topological invariants* for an arbitrary polyhedral surface $\mathcal{M}$, that is, quantities which are equal for two surfaces $\mathcal{M}$ and $\mathcal{M}^{'}$ if $\mathcal{M}$ and $\mathcal{M}^{'}$ are homeomorphic; a simple invariant of $\mathcal{M}$ is given by the number of connected components of its boundary. Let $\mathcal{M}$ a polyhedral surface with v vertices, e edges, and f faces; then, the *Euler characteristic* of $\mathcal{M}$ is defined as $\chi(\mathcal{M}) := v - e + f$. The following theorem states that the Euler characteristic and *orientability* (see Definition 7 and [Mas67]) are topological invariants.

Theorem 4. *Let $\mathcal{M}$ and $\mathcal{M}^{'}$ be two polyhedral surfaces which are homeomorphic. Then:*

- *$\mathcal{M}$ is orientable if and only if $\mathcal{M}^{'}$ is orientable;*

- $\chi(\mathcal{M}) = \chi(\mathcal{M}')$.

From the above theorem it follows that two polyhedral surfaces with different invariants are not homeomorphic; the following theorem classifies surfaces up to homeomorphism and specifies which invariants guarantee that two surfaces are homeomorphic.

Theorem 5 (Classification theorem for surfaces)**.** *Let $\mathcal{M}$ be a compact, connected, and orientable surface. There exist two unique non-negative integers g and b such that $\mathcal{M}$ is homeomorphic to a sphere to which are glued g handles and removed b pairwise disjoint disks. Furthermore, $\chi(\mathcal{M}) = 2 - 2g - b$.*

A *polygonal schema* of a connected surface $\mathcal{M}$ is a simple polygon Ω with an identification of its edges such that $\mathcal{M}$ is obtained by performing these identifications; therefore, the polygonal schemes define a simple way of building surfaces by indicating the list of the edges on Ω and the identifications that have to be performed. As detailed in Section 4.8, any graph γ embedded on $\mathcal{M}$ such that $\mathcal{M} - \gamma$ is a disc-like surface defines a polygonal schema of $\mathcal{M}$ by cutting this graph along its edges.

1.2 Polygonal meshes and duality

A *polygonal mesh* is defined by a pair $\mathcal{M} := (M, T)$ where M is a set of vertices $M := \{x_i \in \mathbb{R}^m,\ i = 1, \dots, n_V\}$, and T an abstract simplicial complex which contains the connectivity information (see Section 1.1.1). Traversing the mesh is achieved by using the relations [Man88]:

- vertex-vertex $VV(v) = (v_1, \dots, v_k)$, face-face $FF(f) = (\tilde{f}_1, \dots, \tilde{f}_m)$;
- face-vertex $VF(v) = (f_1, \dots, f_k)$, vertex-face $FV(f) = (v_1, \dots, v_q)$.

In the following of the chapter, we assume that the previous relations are consistently evaluated. A vertex v is defined as *internal* if its *1-neighborhood* $VV(v)$ is closed, i.e. v is not on the boundary of $\mathcal{M}$. Different authors have proposed optimized data structures [GS85, Man88] for efficiently representing and traversing a polygonal mesh; specializations of these techniques to triangle meshes are described in [DM02]. The duality of structures arises in different contexts such as functional/numerical analysis (e.g., dual of Hilbert spaces) and computational geometry. In the last field, one of the fundamental data structure is the Voronoi diagram [Aur91, dBKOS97, Ede87, GS85, PS85] of a discrete set of points. Its study, which has influenced different application areas such as math, computer, and natural science is strictly related to the Delaunay triangulation and their duality relations. In the sequel of the section, we briefly review the duality in the plane and its extension to 3D meshes. If $M := \{x_i\}_{i=1}^{n_V}$ is a set of n_V points in $\mathbb{R}^m$, its *Voronoi diagram* $V(M)$ is a cell complex which decomposes $\mathbb{R}^m$ into n_V cells $\{V(x_i)\}_{i=1}^{n_V}$ where $V(x_i)$ is defined as

$$V(x_i) := \{x \in \mathbb{R}^m : \|x - x_i\|_2 < \|x - x_j\|_2, j \neq i\}$$

and $\| \|_2$ denotes the Euclidean distance. This definition introduces a proximity relation among points in $\mathbb{R}^m$. In the planar case, i.e. $m = 2$, the *dual graph* $\mathcal{G}$ of $V(x)$ has a node for every cell and it has an arc between two nodes if the corresponding cells share an edge. The *Delaunay graph* of M is defined as the straight-line embedding of $\mathcal{G}$ obtained by identifying the node corresponding to the cell $V(x_i)$ with x_i and the arc connecting the nodes of $V(x_i)$ and $V(x_j)$ with the segment $\overline{x_i x_j}$. The Delaunay graph of a planar point set is a plane graph and, if M is in *general position* (i.e., no four points lie on a circle), all vertices of the Voronoi diagram have degree three. This result guarantees that all bounded faces in the Delaunay graph are triangles, thus defining the *Delaunay triangulation* of M. The extension of this theory for triangulating a set of points in $\mathbb{R}^3$ is not trivial; as a result important properties of the two dimensional Delaunay triangulation, e.g. optimal storage requirement and computational cost, partially apply to the 3D setting. Previous considerations have required the definition of new algorithms [BE92, HDD$^+$92] and 3D triangulation remains a challenging problem in Computer Graphics.

The use of several algorithms for constructing a polygonal mesh of a 3D point cloud (i.e., $m = 3$) requires to define the dual graph in a general way, taking out of consideration the method that has been used for the mesh construction. Given a polygonal mesh $\mathcal{M}$, its *dual graph* $\mathcal{G}$ has a node $v^\star$ for each face $f(v^\star)$ in $\mathcal{M}$ and it has an arc between two nodes $v^\star$ and $w^\star$ if and only if $f(v^\star)$ and $f(w^\star)$ share an edge (see Figure 1.1). In analogy with the previous definitions, the *barycenter dual graph* of $\mathcal{M}$ is defined as the straight-line embedding $\mathcal{M}^{'}$ of $\mathcal{G}$ obtained by identifying each one of its nodes with the barycenter $b_{f(v^\star)}$ of the corresponding face $f(v^\star)$, and the arc connecting the nodes $v^\star$ and $w^\star$ with the segment $\overline{b_{f(v^\star)} b_{f(w^\star)}}$. Therefore, in the dual mesh $\mathcal{M}^{'} := (B, G)$ each vertex $b_{f(v^\star)}$ corresponding to the face $f(v^\star) = (v_1, \dots, v_l)$ is computed as

$$b_{f(v^\star)} := \frac{1}{l} \sum_{i=1}^{l} x_{v_i}$$

and the connectivity G is completely defined by T. From the previous description it follows that each face of the dual mesh generally has a different number of vertices even if the input mesh is triangular, quadrilateral, etc.. Clearly, if $\mathcal{M}$ is a triangle mesh l is three.

Finally, we note that the genus of the dual mesh is equal to that of the input mesh, thus preserving its topology. This simply follows observing that the Euler characteristic $\chi(\mathcal{M}) = n_V - n_E + n_F$ is the same of $\mathcal{M}^{'}$ being $n'_V = n_F$, $n'_E = n_E$, $n'_F = n_V$, where n_V, n_E, n_F and $n^{'}_V$, $n^{'}_E$, $n^{'}_F$ are the number of vertices, edges, and faces of $\mathcal{M}$ and $\mathcal{M}^{'}$ respectively. Therefore, we can summarize this property as: *the genus of a polygonal mesh is invariant under the duality transformation.*

Overview and contributions. Recent applications to compression [LK98], smoothing [Tau02], and subdivision [ZS01] reveal an increasing attention to the correspondence between a mesh $\mathcal{M}$ and its dual representation $\mathcal{M}^{'}$. The growing interest on the primal-dual correspondence is due to a greater regularity of the dual mesh connectivity which corresponds to storage and computational optimization. The first part of the chapter analyzes in detail the topological and geometric identification between $\mathcal{M}$ and $\mathcal{M}^{'}$; more precisely, we provide a reconstruction algorithm of the geometry of $\mathcal{M}$ through that of $\mathcal{M}^{'}$, also achieving basic combinatorial properties of the 1-neighborhood of each internal vertex in $\mathcal{M}$. This correspondence results in the definition of a discrete homeomorphism between $\mathcal{M}$ and $\mathcal{M}^{'}$ whose computational cost is linear in the number of faces in $\mathcal{M}$. The stability to noise on the mesh vertices is studied underlining the correlation among adjacent neighborhoods in $\mathcal{M}$. The developed framework is used to look at the signal processing theory of triangle meshes by considering the dual mesh as noisy, and successively defining the regularized mesh through a process different from the primal-dual identification which cannot be applied due to the violation of the derived combinatorial properties. Therefore, the "*dual Laplacian smoothing*" reveals the way the regularization process affects the input mesh geometry. This approach to smoothing enables to consider a new topological mask for the mesh regularization whose effectiveness is compared with that of the Taubin's $\lambda|\mu$ algorithm [Tau95b].

The rest of the chapter is organized as follows: combinatorial relations and triangle mesh reconstruction through duality are discussed in Section 1.3, providing several considerations on the primal-dual correspondence. The dual Laplacian smoothing is analyzed in Section 1.4 underlining its relations with analogous methods and triangle mesh duality previously analyzed.

1.3 Combinatorial properties of triangle meshes

In the previous section, we have derived the invariance of the genus of $\mathcal{M}$ under the duality transformation. We are now concerned with the analysis of the geometry of $\mathcal{M}$ and $\mathcal{M}^{'}$. The most general and strictly related questions which give a deeper understanding of the relations and differences between $\mathcal{M}$ and $\mathcal{M}^{'}$ can be summarized as follows.

- Is it possible to locally characterize the 1-neighborhood structure of each vertex of $\mathcal{M}$?
- Is it possible to reconstruct the input mesh by using only the dual mesh? Which is the minimal information, if any, required for this purpose?

In the next section, we answer these questions by taking into account the case where $\mathcal{M}$ is a *2-manifold triangle mesh*; the analysis of the general problem is discussed in Section 1.3.3.

1.3.1 1-neighborhood analysis

We are going to derive two basic combinatorial properties of the 1-neighborhood structure of each internal vertex v in $\mathcal{M}$. These relations are used in the sequel for describing the linear reconstruction algorithm of $\mathcal{M}$ from $\mathcal{M}^{'}$ thus defining a complete topological and geometric identification between a mesh and its dual representation.

Proposition 3 (1-neighborhood characterization)**.** *Let $\mathcal{M}$ be a 2-manifold triangle mesh with two adjacent vertices v and w, $VF(v) = (f_1, \dots, f_k)$ the faces incident in v, and b_i the barycenter of*

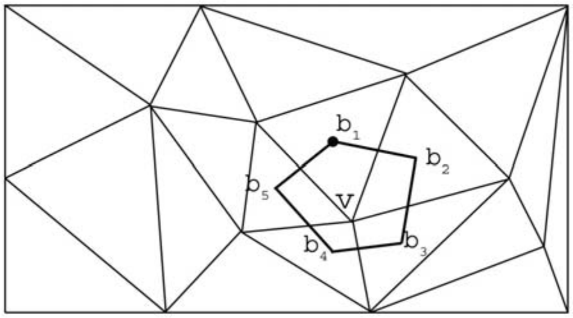


Figure 1.1: Triangle mesh and dual graph (marked line).

the face f_i, $\forall i = 1, \ldots, k$ (see Figure 1.2(a)). If v is an internal vertex, the following conditions hold:

- *if k is even,*

$$\sum_{i=1}^{k} (-1)^i b_i = 0 \tag{1.1}$$

- *if k is odd,*

$$3 \sum_{i=1}^{k} (-1)^{i+1} b_i = x_v + 2x_w \tag{1.2}$$

Proof. Considering $VV(v) = (v_1, \ldots, v_k)$, with

$$a_1 := x_{v_1} = x_w, \ a_l := x_{v_l}, \ l = 2, \ldots, k$$

the idea is to express each vertex a_l as a linear combination of $\{b_i\}_{i=1}^{l-1}$, $2 \leq l \leq k$, x_v and x_w. For each triangle f_l, we have that

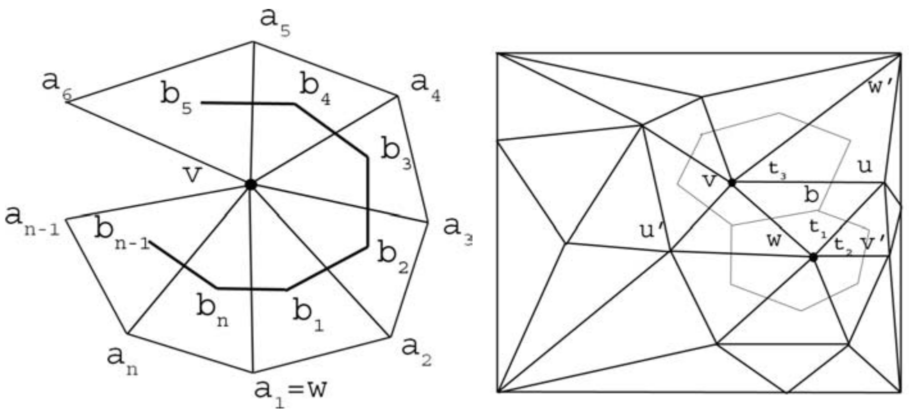
$$b_l = \frac{1}{3}(a_{l+1} + a_l + x_v) \Longleftrightarrow a_{l+1} = 3b_l - a_l - x_v.$$

Substituting in the last equality the expression of a_l in terms of b_{l-1}, a_{l-1}, x_v, and recursively applying this process we achieve that

$$a_{l+1} = \begin{cases} 3\sum_{i=1}^{l} (-1)^i b_i + a_1 & \text{if } l \text{ is even,} \\ 3\sum_{i=1}^{l} (-1)^{i+1} b_i - a_1 - x_v & \text{if } l \text{ is odd.} \end{cases}$$

The condition $a_{k+1} = a_1$ implies (1.1) if k is even, and (1.2) if k is odd. □

The interesting element in (1.1), (1.2) is that the coefficients which appear in the linear combination of the barycenters are constant and not related to their positions. Furthermore, if k is even the manifold structure of $\mathcal{M}$ ensures that $k \geq 4$; therefore, identifying b_i with the vector $(b_i - x_v)$ (1.1) gives their linear dependency relation in the vector space $\mathbb{R}^3$ only using constant coefficients.



(a) (b)

Figure 1.2: (a) 1-neighborhood of the vertex v, (b) 1-neighborhoods of v and w used for the reconstruction of the mesh geometry.

1.3.2 Triangle mesh reconstruction through duality

The first step of the reconstruction algorithm (see Figure 1.2(b)) chooses two internal vertices x_v, x_w of an edge in $\mathcal{M}$ and associated to the neighborhoods

$$VV(v) = (v_1, \dots, v_{n_1}), \quad VV(w) = (w_1, \dots, w_{n_2}).$$

Selected the vertex x_v, each one of its incident triangles $VF(v) = (f_1, \dots, f_{n_1})$, $n_1 \geq 3$, is represented by a vertex in the dual mesh $\mathcal{M}^{'}$ which is the barycenter of the related triangle in $\mathcal{M}$. Supposed to have calculated the vertices x_v, x_w and indicated with f one of the two triangles which have (u, v) as edge, the third vertex x_u of f is evaluated as

$$x_u = 3b - x_v - x_w \tag{1.3}$$

where b is the barycenter of f. After this calculation, the triangle f is marked as visited and its adjacent ones $(t_1, t_2, t_3) = FF(f)$ are considered. Using (1.3), the new vertices $u^{'}, v^{'}, w^{'}$ are calculated marking these triangles as visited. Growing from the visited faces by using their adjacent triangles and recursively applying this criterion to the non-marked faces of $\mathcal{M}$ enables to reconstruct the geometry of the input mesh with exactly n_F steps.

It remains to describe the method for evaluating the two vertices x_v and x_w which have been used for reconstructing the input mesh geometry. Without loss of generality [1] we can suppose that n_1 and n_2 are odd; applying (1.2) to x_v and x_w leads to the symmetric linear system

$$\begin{cases} 3\sum_{i=1}^{n_1}(-1)^{i+1} b_i = 2x_w + x_v \\ \\ 3\sum_{i=1}^{n_2}(-1)^{i+1} b_i^{'} = 2x_v + x_w \end{cases}$$

[1] For instance, if n_1 is even we can split f_{n_1} into two new triangles; i.e., joining its vertex v with the middle point of the edge opposite to v in f_{n_1}.

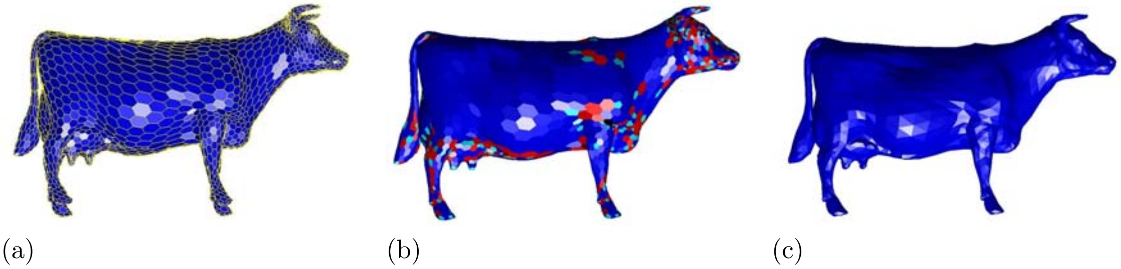


Figure 1.3: (a) Input dual mesh with 5.804 vertices and 2.904 faces, (b) dual graph colored with respect to the number of vertices in each face (see Table 1.1), (c) reconstructed triangle mesh.

Table 1.1: Face coloring in the dual graph.

Color	Number of face vertices k
yellow	$k = 1$
cyan	$2 \leq k < 6$
blue	$k = 6$
red	$7 \leq k < 10$
black	$k \geq 10$

where $VF(w) = (f_1^{'}, \ldots, f_{n_2}^{'})$ and $b_i^{'}$ is the barycenter of the face $f_i^{'}$. Since these relations are linearly independent, its unique solution is

$$\begin{cases} x_w = 2\alpha - \beta \\ x_v = 2\beta - \alpha \end{cases} \tag{1.4}$$

with $\alpha = \sum_{i=1}^{n_1}(-1)^{i+1}b_i$ and $\beta = \sum_{i=1}^{n_2}(-1)^{i+1}b_i^{'}$.

The relation (1.4) expresses these vertices as a linear combination of the barycenters of the triangles of their 1-neighborhoods; we also underline the symmetry in the expression of x_v and x_w with respect to α, β. The computational cost of the proposed algorithm is optimal because it only requires to visit all the triangles of the input mesh, and the expression (1.3) is computationally stable minimizing the numerical instability of the algorithm. Therefore, the transformation which maps $\mathcal{M}$ to $\mathcal{M}^{'}$ is linear in n_F as its inverse. An example of dual mesh and of the reconstruction process is given in Figure 1.3.

1.3.3 Considerations on the primal-dual correspondence

We present in this section several considerations on the primal-dual correspondence which is related to the dual Laplacian smoothing described in the following of the chapter. The extension of the reconstruction process from the dual mesh of a q-mesh, $q \geq 4$, cannot be directly derived from the approach previously described. In fact, supposed that $(l-1)$ vertices $(x_{v_1}, \ldots, x_{v_{l-1}})$ of a face $f = (v_1, \ldots, v_l)$ in $\mathcal{M}$ have been calculated, the last vertex x_{v_l} is evaluated using the barycenter of f

$$b_f = \frac{1}{l}\sum_{i=1}^{l} x_{v_i}$$

as

$$x_{v_l} = lb_f - \sum_{i=1}^{l-1} x_{v_i};$$

however, this information is not sufficient for finding the position of all the vertices in the adjacent faces of f (see Figure 1.4). We answer questions given in Section 1.3 in a simple way as summarized by Theorem 6.

Theorem 6. *Given a 2-manifold triangle mesh $\mathcal{M}$ with or without boundary and with at least two internal vertices, the following conditions hold:*

- *$\mathcal{M}$ and its dual mesh $\mathcal{M}^{'}$ are topologically equivalent, i.e. $\chi(\mathcal{M}) = \chi(\mathcal{M}^{'})$;*
- *$\mathcal{M}$ and $\mathcal{M}^{'}$ are geometrically equivalent, i.e. $\mathcal{M}$ (resp., $\mathcal{M}^{'}$) is reconstructed in n_F steps (resp., n_V) from its dual representation $\mathcal{M}^{'}$ (resp., $\mathcal{M}$);*
- *the vertices of $\mathcal{M}$ and $\mathcal{M}^{'}$ satisfy conditions (1.1), (1.2) given in Proposition 3.*

In Theorem 6, it has been pointed out that the dual mesh is sufficient for identifying the input triangle mesh geometry and topology without storing additional information. We want to analyze the influence of the information in $\mathcal{M}^{'}$ for the reconstruction process; equivalently, we study how the geometry of $\mathcal{M}$ is affected by changing the position of x_v and x_w. To this end, we add a noise e to each one of them considering the new points

$$\tilde{v} := x_v + e, \qquad \tilde{w} := x_w + e.$$

Denoted with $\mathcal{M}$ the triangle mesh reconstructed from $\mathcal{M}^{'}$, x_v, x_w and with $\mathcal{M}_{noise}$ the one achieved with $\mathcal{M}^{'}$, $\tilde{v}, \tilde{w}$, we want to estimate their deviation by using a norm for the error evaluation. The comparison of two triangle meshes with different geometry and connectivity has been studied in [CRS98]. Since $\mathcal{M} := (M, T)$ and $\mathcal{M}_{noise} := (\tilde{M}, T)$ share the same connectivity T, a simpler comparison among vertex positions and triangle normals is introduced using the following vectors:

$$d_v(\mathcal{M}, \mathcal{M}^{'}) := \left(\frac{\|x_i - \tilde{x}_i\|_2}{C_V}\right)_{i=1}^{n_V}, \; C_V := \max_{i=1,\dots,n_V}\{\|x_i - \tilde{x}_i\|_2\}$$

$$d_n(\mathcal{M}, \mathcal{M}^{'}) := \left(\frac{\|n_i - \tilde{n}_i\|_2}{C_N}\right)_{i=1}^{n_F}, \; C_N := \max_{i=1,\dots,n_F}\{\|n_i - \tilde{n}_i\|_2\}$$

with n_i and $\tilde{n}_i$ unit normals to the faces f_i and $f_i^{'}$.

For a better visualization, the increasing reorder of $d_v(\mathcal{M}, \mathcal{M}^{'})$ and $d_n(\mathcal{M}, \mathcal{M}^{'})$ is plotted without normalization (i.e., $C_V := C_N := 1$). As underlined in Figure 1.5(a), a small perturbation e creates a wrong reconstruction of the input triangle mesh showed in Figure 1.3(c). This phenomena is mainly due to the high correlation between vertices in $\mathcal{M}$ and $\mathcal{M}^{'}$ resulting in an error propagation

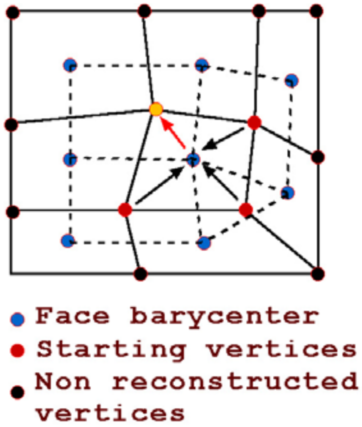


Figure 1.4: Quadrilateral mesh: dual representation (dotted line) and non-reconstructed geometry.

which grows in parallel with the visiting triangle process. This aspect is a consequence of the fact that each new vertex is calculated starting from those ones previously evaluated; indeed, after k steps (1.3) results affected by an error which is proportional to e^k. These considerations also apply if we add a noise to the vertices in $\mathcal{M}^{'}$ as underlined in Figure 1.5(b).

We associate to a mesh three matrices which code in a compact form its topology. If $VF(v) = (f_1, \dots, f_m)$ is the set of faces incident in a vertex v of $\mathcal{M}$, we construct the corresponding face-vertex matrix $I_{VF} \in M_{n_V, n_F}(\mathbb{R})$

$$I_{VF}(i,j) := \begin{cases} 1 \text{ if } f_j \in VF(v_i), \\ 0 \text{ else,} \end{cases}$$

and its normalization

$$N_{VF}(i,j) := \begin{cases} 1/m_i \text{ if } f_j \in VF(v_i), \ \#VF(v_i) = m_i, \\ 0 \text{ else.} \end{cases}$$

In a similar way, we define the normalized vertex-vertex matrix $N_{VV} \in M_{n_V, n_V}(\mathbb{R})$ as

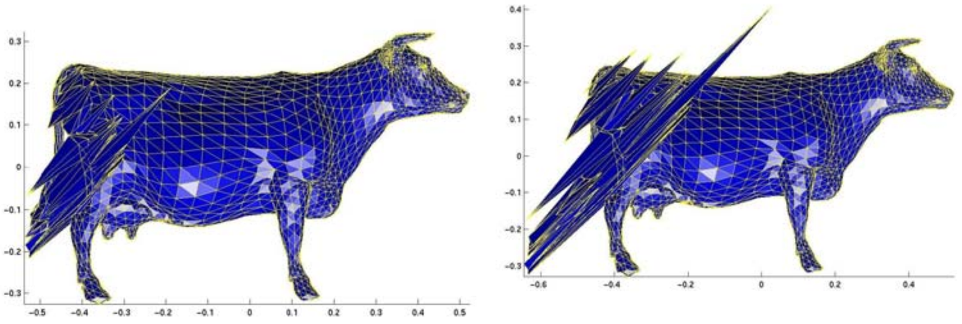
$$N_{VV}(i,j) := \begin{cases} 1/m_i \text{ if } v_j \in VV(v_i), \ \#VV(v_i) = m_i, \\ 0 \text{ else,} \end{cases}$$

and the normalized vertex-face matrix $N_{FV} \in M_{n_F, n_V}(\mathbb{R})$

$$N_{FV}(i,j) := \begin{cases} 1/m_i \text{ if } v_j \in FV(f_i), \ \#FV(f_i) = m_i, \\ 0 \text{ else.} \end{cases}$$

The properties of their spectrum have important connections with the topological characteristics (e.g., number of connected components) of the input mesh [Moh91] and with numerical properties of the Laplacian smoothing and re-sampling operator studied in [Tau02, Tau95b, TZG96] (see also Section 1.4). Using these matrices, we want to settle the numerical approach to the reconstruction process from $\mathcal{M}$ to $\mathcal{M}^{'}$, which is expressed as

$$N_{FV} M = B, \tag{1.5}$$



(a) (b)

Figure 1.5: Noise influence on the reconstruction process using: (a) noised vertices $\tilde{v}$, $\tilde{w}$ in $\mathcal{M}$, $e := 5.0 * 10^{-6}$, (b) a noised vertex in $\mathcal{M}^{'}$, $e := 5.0 * 10^{-6}$.

that is, a linear system with n_V unknowns $(x_i)_{i=1}^{n_V}$ and n_F equations. Since $n_F \approx 2n_V$, (1.5) is undetermined if $rank(N_{FV}) < n_V$, over-determined if $rank(N_{FV}) > n_V$, and it has a unique solution otherwise. In order to define a well-posed problem, (1.5) can be replaced by the *least-squares problem* [GL96]

$$\|N_{FV}\bar{M}_i - B_i\|_2 = \min_{P_i}\{\|N_{FV}M_i - B_i\|_2\},$$

or equivalently $N_{FV}^T N_{FV}\bar{M}_i = N_{FV}^T B_i$, $i = 1, 2, 3$ where M_i and B_i is the i-th component of M and B respectively. This choice produces a family of triangle meshes $\{\tilde{M}\}$

$$\tilde{M}_i := \bar{M}_i + E_i, \qquad E_i \in ker(N_{FV}), \qquad i = 1, 2, 3$$

each one represents an approximated solution of (1.5), and the computational cost is $O(n_V^3)$. With respect to the primal-dual correspondence described in Section 1.3.2, this strategy faces-up to its expensive computational cost providing a family of approximated triangle meshes $(\tilde{M}, T)$ instead of the initial mesh $\mathcal{M}$.

1.4 Dual Laplacian smoothing

In the previous section, we have focused our attention on the relation between a triangle mesh $\mathcal{M}$ and its dual representation underlining their correlation. Here, we consider applications of the dual representation for smoothing noised data sets. The key observation is that, considering the 1-neighborhood structure related to each point, $\mathcal{M}$ has a little regularity while $\mathcal{M}^{'}$ can be considered with more simplicity because each of its vertices has three links if the related triangle of $\mathcal{M}$ is internal, and one or two if it belongs to the mesh boundary. This observation is the base in [LK98] for the compression of triangle meshes; furthermore, primal-dual correspondence is partially exploited in [ZS01] for primal-dual subdivision schemes, and in [Tau02] for the definition of dual re-sampling and non-shrinking smoothing operators. Firstly, we review Laplacian and Taubin's smoothing algorithms [Tau02, Tau95b, TZG96] which are strictly related to our approach.

- **Laplacian smoothing.** Each internal vertex x_v of the input mesh is updated using its 1-neighborhood structure $VV(v) := (v_1, \ldots, v_n)$ as described by the following procedure:

  $$x_v^{(1)} := (1-\lambda)x_v + \frac{\lambda}{\sum_{i=1}^n w_i} \sum_{i=1}^n w_i x_{v_i}$$

  where $\lambda \in [0,1]$ is a positive parameter controlling the smoothing process. The weights $(w_i)_{i=1}^n$ can be chosen in different ways even if the following ones are commonly used:

  - *constant weights*: $w_i = 1,\ i = 1, \ldots, n$, i.e.

    $$x_v^{(1)} := (1-\lambda)x_v + \frac{\lambda}{n} \sum_{i=1}^n x_{v_i}; \tag{1.6}$$

  - *adaptive weights* [Tau95b]: w_i is proportional to the inverse of the distance between x_v and its neighbor x_{v_i}, i.e. $w_i := \|x_v - x_{v_i}\|_2^{-1}$. A general choice is given by $w_{ij} \geq 0$, $\sum_j w_{ij} = 1$ whose properties rely on the stochastic matrix theory [GL96, Tau00].

  We write (1.6) as

  $$M^{(1)} = [(1-\lambda)I_{n_V} + \lambda N_{VV}]M = f_\lambda(L)M$$

  where I_{n_V} is the identity matrix of order n_V, $f_\lambda(t) = (1-\lambda t)$, $L := I_{n_V} - N_{VV}$ and

  $$M = \begin{pmatrix} x_1 \\ \vdots \\ x_{n_V} \end{pmatrix} \in M_{n_V,3}(\mathbb{R}).$$

  The Laplacian smoothing reduces all non-zero frequencies of the signal corresponding to the mesh and tends to shrink its geometry. To partially solve this drawback, in [DMSB99] each smoothing iteration is combined with a mesh volume-restoring and re-scaling step.

- **Taubin's smoothing.** The solution to shrinkage proposed in [Tau95b] is based on the alternation of two scale factors of opposite signs λ, μ in the Laplacian smoothing, i.e.

  $$M_{Taubin}^{(1)} := f_\lambda(L) f_\mu(L) M$$

  where $-\mu > \lambda > 0$. Using this filter enables to suppress high frequencies while preserving the low ones. Good results are achieved by choosing the input parameters which satisfy the condition

  $$\frac{1}{\lambda} + \frac{1}{\mu} = 0.1.$$

  The application of k iteration steps gives

  $$M_{Taubin}^{(k)} = f_{\lambda,\mu}^{(k)}(L)M$$

  with $f_{\lambda,\mu}^{(k)}(t) := [f_\lambda(t) f_\mu(t)]^k$.

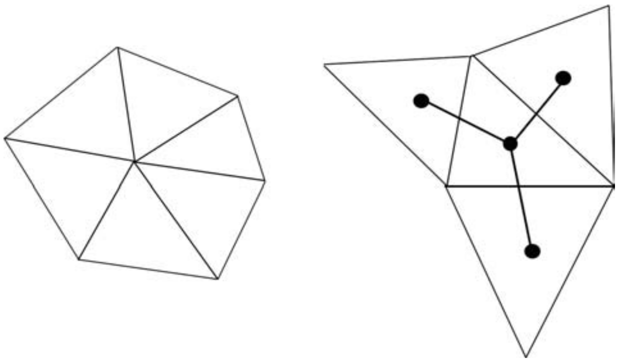


Figure 1.6: Topological masks used for the primal and dual Laplacian smoothing.

1.4.1 Dual approach to triangle mesh smoothing

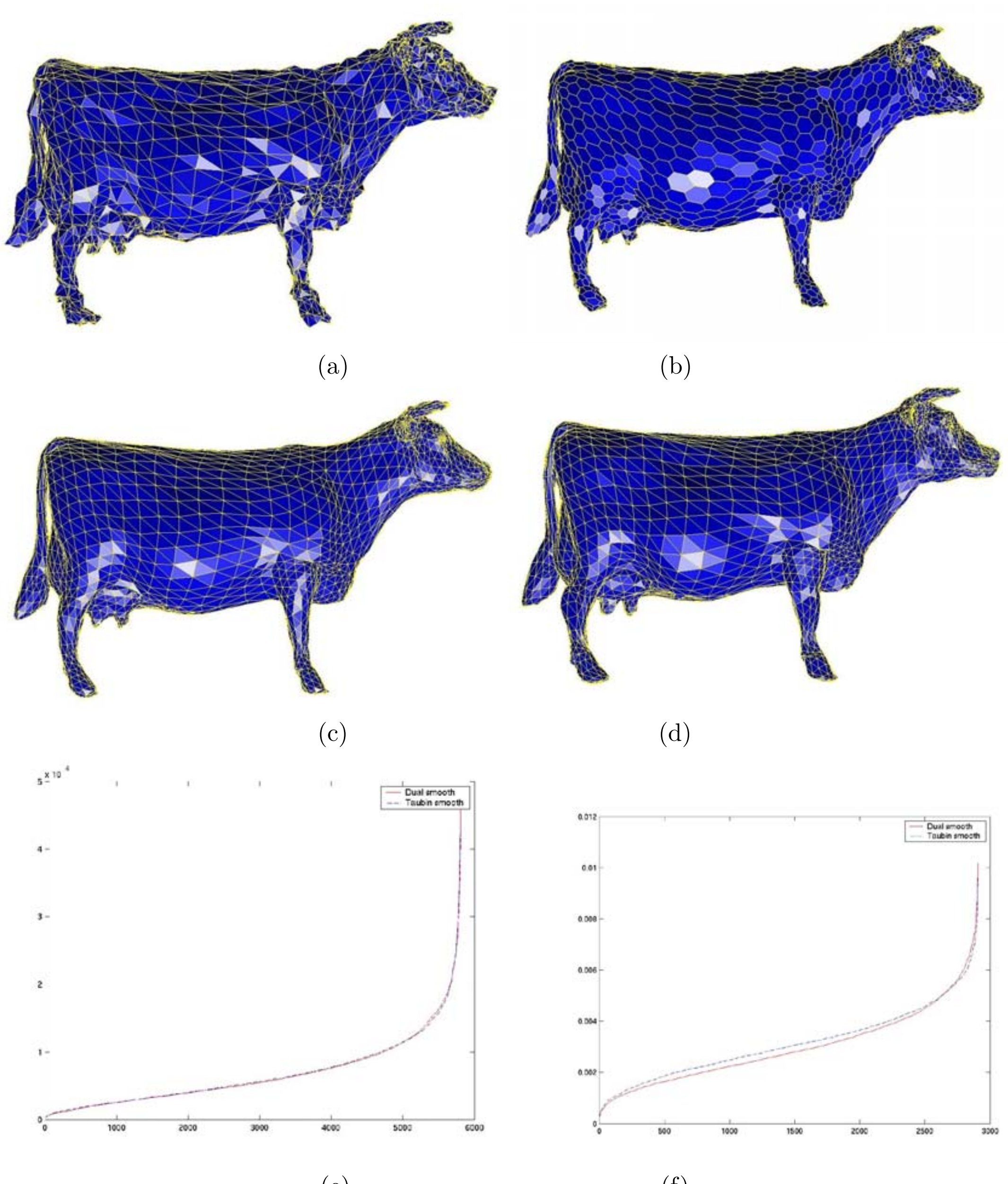
Considered a noised mesh $\mathcal{M}_{noise} := (M, T)$, the idea is to apply the Laplacian smoothing and its extensions to the dual mesh $\mathcal{M}^{'}_{noise} := (B, G)$ which is affected by noise as well as $\mathcal{M}_{noise}$. The merit of using the dual mesh instead of the input one is mainly due to the following considerations. Firstly, the normalized vertex-vertex matrix N_{FF} of $\mathcal{M}^{'}_{noise}$, which will be used for the smoothing process, has at most three non zero elements in each row. This implies the optimization of storage and computational requirements which grow with the complexity of the input mesh in terms of the number of vertices. Furthermore, the construction of the incident matrix of $\mathcal{M}^{'}_{noise}$ is simply achieved with the constant relation FF applied to $\mathcal{M}$. Secondly, the dual smoothing considers at each vertex of $\mathcal{M}$ a different topology for the regularization with respect to the 1-neighborhood structure used for the (primal) Laplacian smoothing (see Figure 1.6).

Denoted with $L^{'}$ the Laplacian matrix of $\mathcal{M}^{'}_{noise}$, and with $\mathcal{M}^{'}_{smooth}$ the smoothed dual mesh, the last step reconstructs the regularized mesh $\mathcal{M}_{smooth}$. Since the vertices of $\mathcal{M}^{'}_{smooth}$ do not satisfy (1.1) and (1.2), the considerations about the high correlation between $\mathcal{M}^{'}_{smooth}$ and $\mathcal{M}_{smooth}$ (see Section 1.3.3) highlight the impossibility of reconstructing $\mathcal{M}_{smooth}$ by using the primal-dual correspondence previously described. The solution to this problem is achieved by defining the new vertices of $\mathcal{M}_{smooth} := (M_{smooth}, T)$ as the barycenters of the faces in $\mathcal{M}^{'}_{smooth}$ which are exactly n_V. This process can be summarized as

$$M^{(k)}_{smooth} = N_{VF} \underbrace{f^{(k)}_{\lambda,\mu}(L^{'})B}_{\text{Dual smooth}} = N_{VF} f^{(k)}_{\lambda,\mu}(L^{'}) N_{FV} M \tag{1.7}$$

with $L^{'} := I_{n_F} - N_{FF}$. The previous relation expresses the regularized mesh geometry only using the information on $\mathcal{M}_{noise}$. In the following proposition, we compare (1.7) with the mesh achieved by applying the Taubin's smoothing using the same number of iterations k, and parameters $\bar{\lambda}$, $\bar{\mu}$

$$M^{(k)}_{Taubin} = f^{(k)}_{\bar{\lambda},\bar{\mu}}(L)M.$$


(a) (b)

(c) (d)

(e) (f)

Figure 1.7: (a) Input data set: num. vertices 2.904, num. triangles 5.804, (b) Laplacian smoothing applied to the dual mesh $\lambda = 0.6$, $\mu = -0.5640$, $k = 10$, (c) average reconstruction, (d) Taubin's smoothing with previous λ, μ, k, (e) error evaluation on vertices, (f) error evaluation on normals.

Proposition 4. *The asymptotic behavior of* $M^{(k)}_{smooth}$ *resembles that of* $M^{(k)}_{Taubin}$ *and its computational cost is linear in the number of vertices* n_V*.*

Proof. From the previous relation, it follows that [2]

$$\frac{\|M^{(k)}_{smooth} - M^{(k)}_{Taubin}\|_2}{\|M\|_2} \leq \|N_{FV}\|_2 \|N_{VF}\|_2 g^k(\lambda, \mu) + g^k(\bar{\lambda}, \bar{\mu})$$

with $g(x, y) := \frac{(x-y)^2}{-4xy}$. Since $|\lambda| < 1$, $|\mu| < 1$ (resp., $|\bar{\lambda}| < 1$, $|\bar{\mu}| < 1$), we have $|g(\lambda, \mu)| < 1$ (resp., $|g(\bar{\lambda}, \bar{\mu})| < 1$) thus guaranteeing that

$$\lim_{k \to +\infty} \|M^{(k)}_{smooth} - M^{(k)}_{Taubin}\|_2 = 0.$$

□

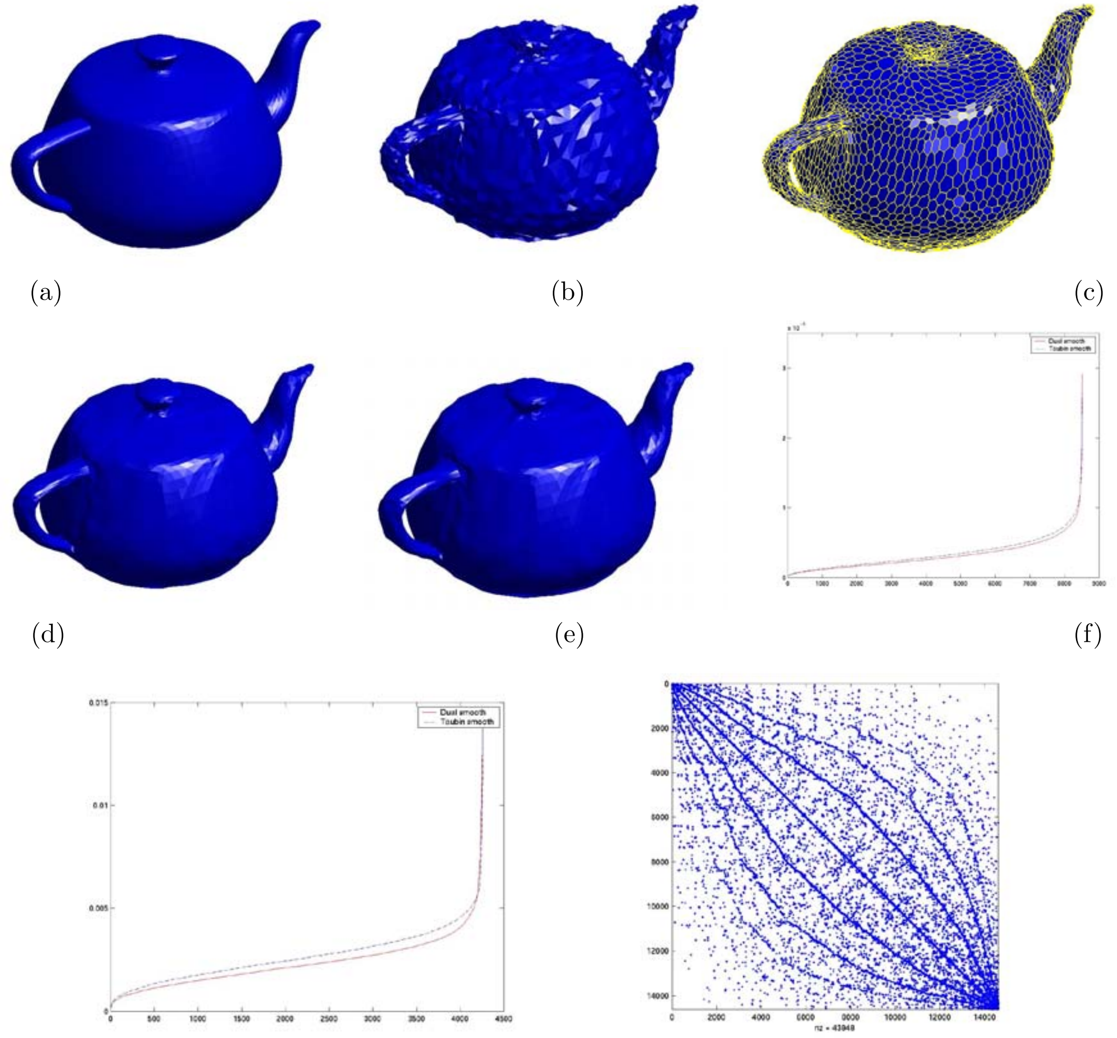
All previous considerations also apply if we consider adaptive weights instead of constant ones. Finally, we observe that we have not applied the least square approach (1.5) for reconstructing $\mathcal{M}$ from $\mathcal{M}'$ because it is computationally expensive and without evident benefits with respect to the previous choice and other examples are given in Figure 1.7, 1.8, and 1.9.

Related publications

1. G. Patanè, M. Spagnuolo. "Triangle Mesh Duality: Reconstruction and Smoothing". In *Mathematics of Surfaces 2003*, Lecture Notes in Computer Science 2768, pp. 111-128, M.J. Wilson and R.R. Martin (Eds), Springer-Verlag Berlin Herdberger.

2. G. Patanè. *Triangle mesh duality: reconstruction and smoothing.* Rapporto Tecnico N. 3/2002, Istituto di Matematica Applicata e Tecnologie Informatiche, Consiglio Nazionale delle Ricerche.

[2] The inverted parabola $f_{\lambda,\mu}(t) := (1-\lambda t)(1-\mu t)$ has its minimum at $\bar{t} := \frac{1}{2}(\frac{1}{\lambda}+\frac{1}{\mu}) \in (0, 1)$ and $f_{\lambda,\mu}(\bar{t}) = \frac{(\lambda-\mu)^2}{-4\lambda\mu}$.

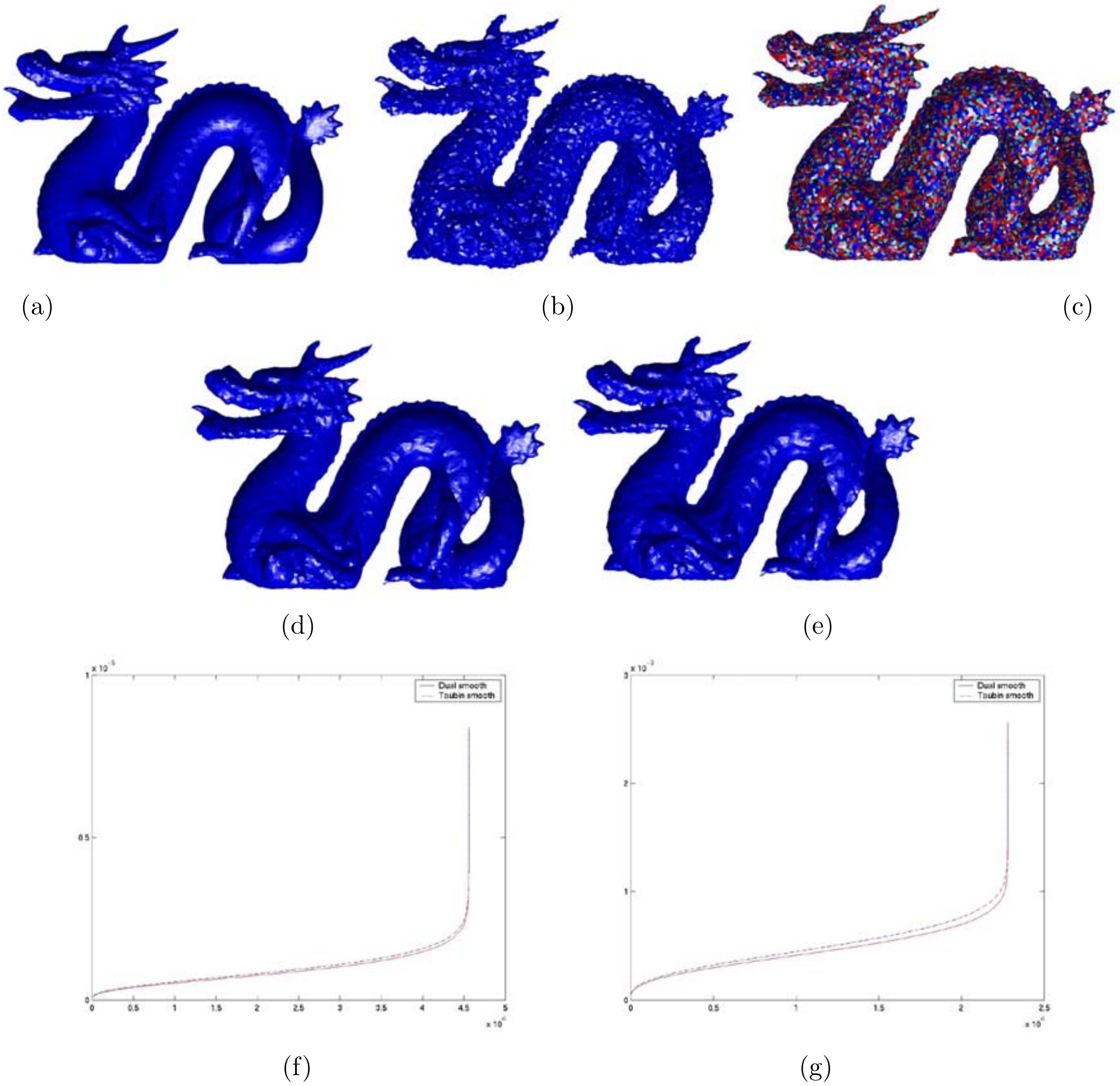


(a) (b) (c)

(d) (e) (f)

(g) (h)

Figure 1.8: (a) Input data set: num. vertices 7.308, num. triangles 14.616, (b) noised data set with normal error, (c) Laplacian smoothing applied to the dual mesh $\lambda = 0.6$, $\mu = -0.5640$, $k = 20$, (d) average reconstruction, (e) Taubin's smoothing with previous λ, μ, k, (f) error evaluation on vertices, (g) error evaluation on normals, (h) Laplacian matrix sparsity.



(a) (b) (c)

(d) (e)

(f) (g)

Figure 1.9: (a) Input data set: num. vertices 22.813, num. triangles 45.626, (b) noised data set with normal error, (c) Laplacian smoothing applied to the dual mesh $\lambda = 0.6$, $\mu = -0.5640$, $k = 10$, (d) average reconstruction, (e) Taubin's smoothing with previous λ, μ, k, (f) error evaluation on vertices, (g) error evaluation on normals.

Chapter 2

Geometric classification and decomposition of 3D surfaces

Tools for the automatic decomposition of a surface into shape features will facilitate the editing, matching, texturing, morphing, compression, and simplification of 3D shapes. Different features, such as flats, limbs, tips, pits, and various blending shapes that transition between them may be characterized in terms of local curvature and other differential properties of the surface or in terms of a global skeletal organization of the volume it encloses. Unfortunately, both solutions are extremely sensitive to small perturbations in the surface smoothness and to quantization effects when they operate on triangulated surfaces. Thus, we propose a multi-resolution approach, which not only estimates the curvature of a vertex over neighborhoods of variable size, but also takes into account the topology of the surface in that neighborhood. Our approach is based on blowing a spherical bubble at each vertex and studying how the intersection of that bubble with the surface evolves. We describe an efficient approach for computing these characteristics for a sampled set of bubble radii and for using them to identify features, based on easily formulated filters, that may capture the needs of a particular application.

2.1 Introduction

An abstract description of a shape usually combines a set of primitives, that are relevant to the specific context, and is defined in terms of their type and intrinsic shape parameters. As suggested in [Lon98], methods for shape description can be classified into two broad categories: those considering only the local properties of the boundary of the shape, and those measuring properties of the enclosed volume. Typically, boundary-based methods evaluate accurate and mathematically well-defined local characteristics, such as critical points or curvature. They may also identify specific loci on the surface, such as curvature extrema or ridges, but they generally lack in providing a global view of the shape. Furthermore, they typically work at a single resolution and thus do not organize features into a hierarchy of global and local details.

Conversely, interior-based methods, which assume that the surface is the boundary of a solid, generally provide descriptions which better highlight the global structure of the shape. Skeletons, such as the medial axis or the Reeb graph belong to this class of descriptors [ABS03, POB87, SKK91]. The great advantage of skeletons is that they provide an abstract representation by idealized lines that retain the connectivity of the original shape, thus reducing the complexity of the representation. Usually, each arc is associated with a portion of the original shape that corresponds to a feature. For example, in 2D the medial axis is constructed using the paradigm of the maximal enclosed discs, whose centers define a locus of points which describes, together with the associated radius, the width variation of the shape. The medial axis induces a decomposition of the shape into protrusion-like features, while concavities of the shape are not directly identified by the medial axis of the interior. However, the medial axis of a 3D shape is not any longer a one-dimensional graph, but it is made of surface pieces as well. Moreover, the instability of the medial axis with respect to noise has prevented its use in many application areas. Approaches to construct and store the medial axis at different scales have been also proposed, which implicitly address the problem of noise reduction as well [POB87, DLN87, Ogn94].

Another notable example of topology-driven skeleton is given by the Reeb graph [SKK91, ABS03]. The Reeb graph is a topological structure which codes a given shape by storing the evolution of criticalities of a mapping function defined on the boundary surface. In particular, when the height function with respect to a predefined direction n is chosen, the Reeb graph describes the evolution of the contours obtained by intersecting the shape with planes orthogonal to n. The decomposition induced by the Reeb graph corresponds to a segmentation of the solid into slices and the corresponding branches of the Reeb graph identify the connected components of the surface. The description obtained using a Reeb graph approach is suitable for matching purposes especially if the mapping function is chosen in order to provide invariance under affine transformations. Such orientation-independent approaches have been proposed in [HSKK01, MP02a, MP02b]; however, they are computationally intensive and offer little control over the scale at which the shape is analyzed. For more details on these techniques, we refer the reader to Chapter 3.

Overview and contributions. Shape analysis and coding are challenging problems in Computer Vision and Graphics. An ideal shape description should be able to capture and compute the main features of a given shape and organize them into an abstract representation which can be used to automate processes such as matching, retrieval, or comparison of shapes. We have tackled the problem in the context of 3D surfaces represented by triangular meshes, having in mind that a good shape description should be able to distinguish between global and local features and should be based on geometric properties of the shape which are invariant under rotation, translation, and scaling. To characterize a shape we have used the paradigm of *blowing bubbles*: a set of spheres of increasing radius R_i, $i = 1, \ldots, n$ is drawn, whose centers are at each vertex of the mesh, and whose radius represents the scale at which the shape is analyzed. The number of connected components of the intersection curve between each bubble and the surface gives a first qualitative

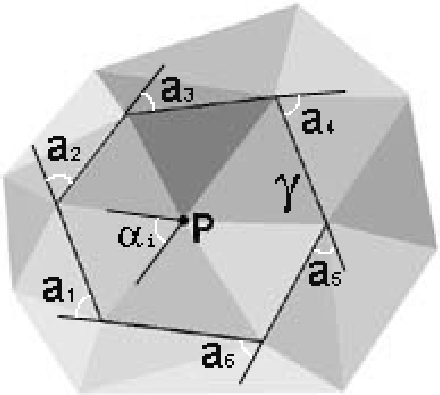


Figure 2.1: $Star(p)$ is enclosed by a curve γ. The exterior angles $\{a_i\}$ and interior angles $\{\alpha_i\}$ are shown.

characterization of the shape in a 3D neighborhood of each vertex. Then, the evolution of the ratio of the length of these components to the radius of the spheres is used to refine the classification and detect specific features such as sharp protrusions or wells, mounts or dips, blends or branching parts. For example, a point on a thin limb will generate an intersection curve made of only one component, which will rapidly split into two connected components as soon as the radius of the sphere exceeds the limb size. Conversely, the tip of a protrusion will always generate only one connected component, and the ratio between the intersection curve length and the sphere radius can be used to evaluate how curved is the surface at the protrusion tip.

The description achieved provides an insight on the presence of features together with their morphological type, persistence at scale variation, amplitude and/or size. The algorithm for the decomposition is concise and simple, and the decomposition is independent on the orientation of the surface in space and equally distributed in all directions. The number of radii and the interval given by the minimum and maximum radius is related to the spectrum of sizes of the features of interest. Moreover, the multi-scale approach and the chosen descriptors reduce the influence of noise on the shape evaluation, it gives us more flexibility to formulate the filters for shape analysis, and it captures the more representative properties in a more detailed description. In this chapter, we focus on the method adopted for the segmentation, while applications to skeleton extraction and local parameterization are given in Chapter 3 and 4 respectively.

The chapter is organized as follows: in Section 2.2, previous work and the theoretical background relevant for the described method is briefly reviewed. The approach to shape classification is presented in Section 2.3. The algorithm details are described in Section 2.4, the mesh decomposition strategy is discussed in Section 2.5, and Section 2.6 includes critical considerations and remarks. Then, a feature-based segmentation is detailed in Section 2.7 and 2.8; links to further applications discussed in the thesis are given in Section 2.9.

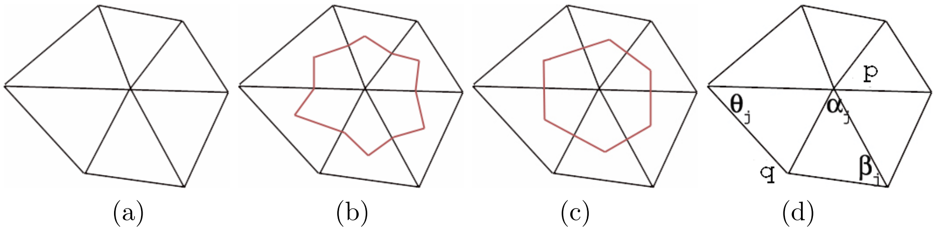


Figure 2.2: (a) 1-star neighborhood, (b) barycentric neighborhood, (c) Voronoi cell, (d) angles of the triangles belonging to the 1-star of p.

2.2 Theoretical background

This section provides definitions and concepts [GP74, Lip69, Mor85] useful for describing our approach. Let $\varphi : U \subseteq \mathbb{R}^2 \rightarrow \mathbb{R}^3$ be a C^2-parameterization of the surface $\mathcal{M} := \{\varphi(u,v) : (u,v) \in U\}$ (see also Section 4.2); the classification of local properties of $\mathcal{M}$ is traditionally based on the study of its *mean* and *Gaussian* curvature. More precisely, chosen a point $p \in \mathcal{M}$ we locally approximate the surface by its tangent plane with normal vector n. For every unit vector $e_\theta := (\cos\theta, \sin\theta)$ in the tangent plane, the normal curvature $\kappa^N(\theta)$ at p is defined as the curvature (at p) of the curve that belongs to the surface $\mathcal{M}$ and to the plane which interpolates p and contains the vectors n and e_θ. The two *principal curvatures* κ_1 (*maxima*), κ_2 (*minima*) of $\mathcal{M}$, with their associated orthogonal directions e_{θ_1}, e_{θ_2} are the extrema of all the normal curvatures $\{\kappa^N(\theta) : \theta \in [0, 2\pi)\}$. Using the identity $\kappa^N(\theta) = \kappa_1 \cos^2\theta + \kappa_2 \sin^2\theta$, $\theta \in [0, 2\pi)$, the *mean-curvature* $\overline{\kappa}$ is defined as the average of the normal curvatures

$$\overline{\kappa} := \frac{1}{2\pi}\int_0^{2\pi} \kappa^N(\theta) d\theta = \frac{1}{2}(\kappa_1 + \kappa_2),$$

and the *Gaussian curvature* is set as $\kappa_G := \kappa_1\kappa_2$. Previous definitions formalize the relation between the surface shape and its position with respect to the tangent plane. For elliptic-shaped surfaces, the centers of curvature of all the normal sections will lie on the same side of the tangent plane with positive values for the minimum and maximum of curvature. For hyperbolic-shaped surfaces, the centers of curvature will move from one side of the surface to the other, with a negative minimum value and a positive maximum value assumed at opposite sides with respect to the tangent plane. For parabolic-shaped surfaces, one of the principal directions will have curvature equal to zero, that is, along that direction the normal section will be a straight line; this is the case of ruled surfaces which are also said to have no double curvature. The planar case is obvious.

The mean-curvature provides a link between surface area minimization and mean curvature flow as expressed by the relation

$$\overline{\kappa} n = \frac{1}{2} \lim_{diam(\mathcal{N}) \rightarrow 0} \frac{\nabla \mathcal{N}}{\mathcal{N}},$$

where $\mathcal{N}$ is an infinitesimal area around the point p on $\mathcal{M}$, $diam(\mathcal{N})$ is its diameter, and ∇ is the

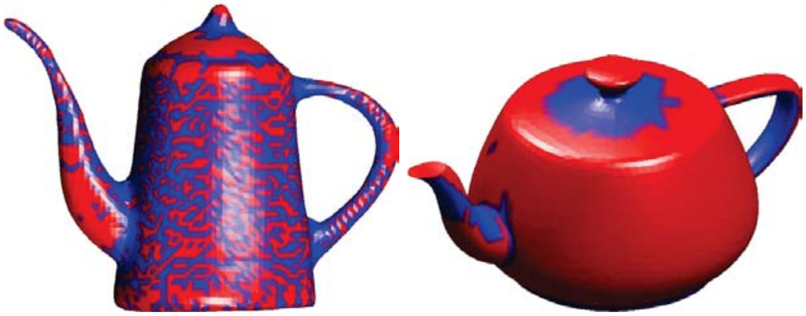


Figure 2.3: Discrete Gaussian curvature and sensibility to local noise: red and blue vertices represent elliptic and hyperbolic points.

gradient with respect to p. The Gaussian curvature is expressed by the limit

$$\kappa_G = \lim_{diam(\mathcal{N})\to 0} \frac{\mathcal{N}^{\mathcal{G}}}{\mathcal{N}}$$

where $\mathcal{N}^{\mathcal{G}}$ is the area of the Gauss map (i.e., its spherical image) associated to the infinitesimal patch $\mathcal{N}$; therefore, it represents a measurement at any point p of $\mathcal{M}$ which is the excess per unit area of a small patch of the surface. Given a closed curve γ on a surface $\mathcal{M}$, let T_γ be the total turning that the unit tangent t undergoes when it is carried along γ, defined as the sum of the local turnings, i.e. *exterior angles* [Lip69] (see Figure 2.1). Then, the quantity $I_\gamma = 2\pi - T_\gamma$ is called the *angle excess* of the curve γ and it is related to the curvature of $\mathcal{M}$ within γ, as described by the Gauss-Bonnet formula.

Theorem 7 (Gauss-Bonnet Formula)**.** *Let γ be a curvilinear polygon of class C^2 on a surface patch of class C^k, $k \geq 3$. Suppose γ has a positive orientation and its interior on the patch is simply connected. Then,*

$$\int_\gamma \kappa_g ds + \iint_\Sigma \kappa_G dS = 2\pi - \sum_i \alpha_i = I_\gamma \qquad (2.1)$$

where κ_g is the geodesic curvature along γ, Σ is the union of γ and its interior, $\{\alpha_i\}$ are the exterior angles of γ, ds and dS are the curve and surface element respectively.

Among the properties of the angle excess, the following ones have a particular interest for our approach:

- I_γ is independent of the chosen starting point on γ;
- I_γ is additive;
- for any topological disc on an arbitrary surface at p, the angle excess around the boundary is equal to the total curvature of the interior.

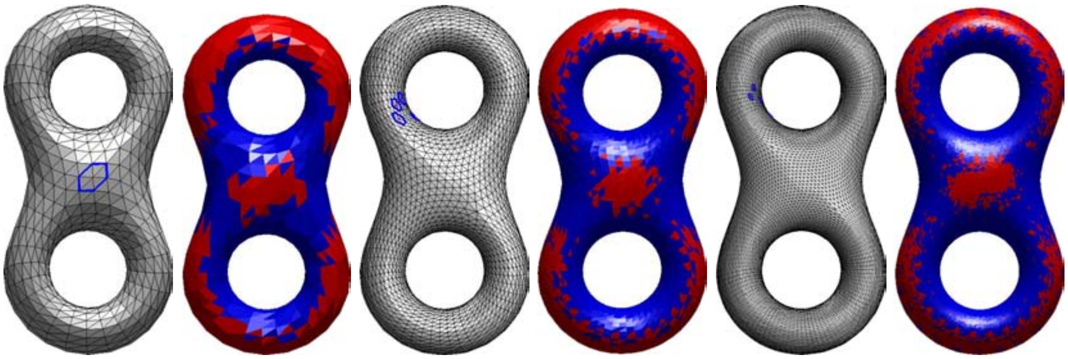


Figure 2.4: Gaussian curvature evaluation on the bitorus at different levels of resolution.

The definition of the curvature at each point of a triangulation is not trivial because a triangular mesh is parameterized by a piecewise continuous function [FS93, GW92]. These methods can be divided into two main groups: *continuity-based* and *property-based* algorithms. The first ones are developed transforming the discrete case to the continuous one by using a local fitting of the surface which enables to apply standard definitions. For example, in [Ham93] an approximation of the curvature is derived at each vertex by applying the continuous definition to a least-squares paraboloid fitting its neighboring vertices, while in [Tau95a] it is evaluated by estimating its tensor curvature. The second class of algorithms defines equivalent descriptors starting from basic properties of continuous operators but directly applied to the discrete settings. In this case, the reformulation of the mean and Gaussian curvature for C^0 surfaces uses a neighborhood of each internal vertex p of the triangle mesh $\mathcal{M}$; among the range of possibilities, common choices (see Figure 2.2(a-c)) are the *1-ring neighborhood* (i.e., the set of triangles incident to p), the *barycentric neighborhood*, and the *Voronoi cell*. For the barycentric neighborhood (resp., Voronoi cell), the patch around p on $\mathcal{M}$ is achieved by connecting the barycenter (resp., circumcenter) of each face t incident to p to the midpoints of the edges of t emanating from p. The methods proposed in [DMSB02, PS99] are based on the Laplace-Beltrami operator, the Gauss map, and the Gauss-Bonnet theorem guaranteeing the validity of differential properties such as area minimization and mean curvature flow [GP74]. Indicated with $\mathcal{N}$ one of the previous neighborhoods of p on an arbitrary triangle mesh $\mathcal{M}$, it is possible to prove that the mean curvature at p can be evaluated as follows

$$\overline{\kappa}n = \frac{1}{A(\mathcal{N})} \sum_{q \in N(p)} (\cot\theta_j + \cot\beta_j)(q-p),$$

where $A(\mathcal{N})$ is the area of $\mathcal{N}$, f is the number of faces incident to p, and the angles θ_j, β_j are depicted in Figure 2.2(d).

The angle excess can be used to evaluate the Gaussian curvature at mesh vertices [FS93, PS99]; in fact, the boundary of the Voronoi cell $\mathcal{N}$ (see Figure 2.1) defines a closed path on the mesh to which we may apply the Gauss-Bonnet formula (2.1). Since the geodesic curvature along the boundary is obviously zero (edges are straight), the total curvature at p is simply quantified by the sum of the exterior angles. To understand better the geometry of the situation, we can imagine

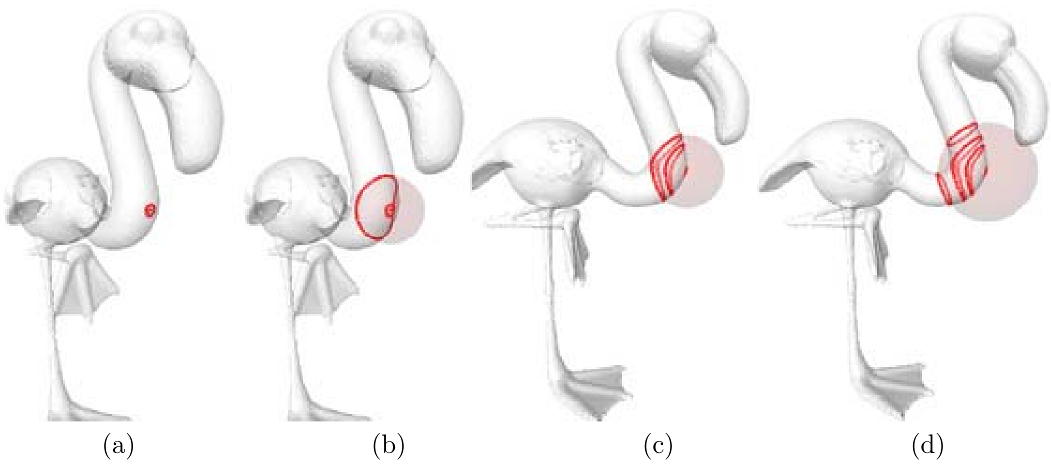


Figure 2.5: Evolution of the intersection curves between the input surface and a set of spheres with the same center and increasing radius.

to locally cut $\mathcal{N}$ along any of the edges incident in p, and to develop $\mathcal{N}$ onto the plane without shrinking the surface. The sum of the exterior angles corresponds to the sum of the angles at p in $\mathcal{N}$. This result is consistent with the intrinsic nature of the Gaussian curvature since the angle excess only depends on the angles, that is, this value does not change if the mesh is deformed preserving the distance among points. Also, the computation of the angle excess can be performed without resorting to any coordinate system, as the angles may be obtained using only the edge length and not the vertex coordinates. The discrete Gaussian curvature at a vertex p of the mesh can be therefore evaluated by

$$\kappa_G = \frac{2\pi - \sum_{i=1}^{f} \alpha_i}{A(\mathcal{N})},$$

with f number of triangles incident in p; that is, the local angle excess in p weighted by the area $A(\mathcal{N})$ of a small patch of surface around p given by $\mathcal{N}$ (see Figure 2.2(d) and 2.1).

The above-mentioned approaches are usually sensitive to noise and, even if a multi-resolution setting is chosen, they usually require smoothness conditions on the input mesh. Furthermore, the smoothing process used to get stable and uniform curvature estimations introduces a deficiency in the magnitude evaluation and, consequently, difficulties in the accurate distinction between planar patches and curved surfaces with low curvature. In Figure 2.3 and 2.4, the Gaussian curvature is shown; it can be seen how the results are effected by small local undulations.

2.3 Geometric and topological classification

The approach proposed here for describing a 3D shape integrates boundary and interior information of the shape finalized at defining a complete multi-scale vertex classification. The link between closed paths and curvature has suggested to specialize its study to the family of closed paths built by intersecting the surface with spheres centered in each of its points. The study of

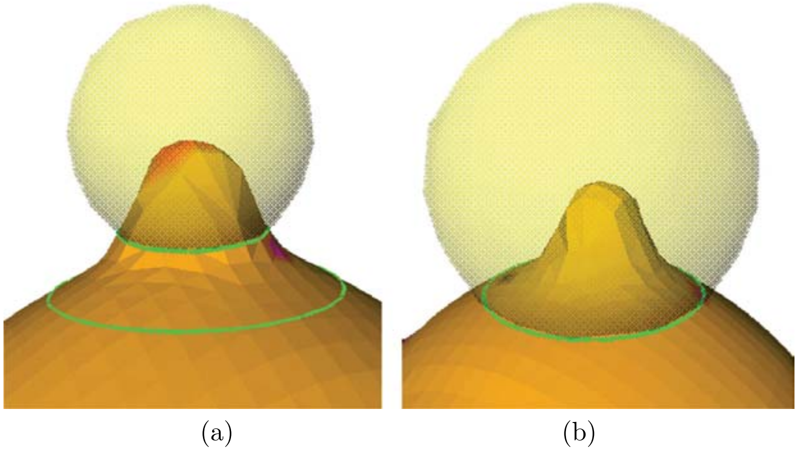


Figure 2.6: A tip detected at a small radius (a) will be characterized as a mount at a larger radius (b).

the evolution of these curves and the geometric characterization of the mesh areas intersected by the spheres are the core of the proposed method. The topology of the intersection curves changes according to the surface shape: in Figure 2.5(a-c), the highlighted sphere intersects the surface only at one curve, while in (d) the boundary of the intersection area splits into two connected components. This is likely to happen, for example, near handles and branches, or around deep pits. Therefore, the variation in the boundary suggests that the vertex is located on a feature, which becomes relevant at the scale, or radius, at which the change occurs.

Given a set of radii $R_i, i = 1, \ldots, n$, each vertex of the mesh is classified with a n-dimensional vector of morphological labels, each corresponding to its type at the related scale. Shape features of the mesh are then identified by connected regions of vertices with the same label at a given scale, and the geometric parameters computed to assign the label will characterize the feature. For example, a tip and a mount are both characterized by one intersection curve, but they can be distinguished measuring the curvature induced by the intersection curve on the surface (see Figure 2.6).

Features which are identified by two intersection curves are further characterized by measuring the relative curve length and by checking if they define a volume which is inside or outside the shape (see Figure 2.7). The vector of labels and corresponding geometric parameters, together with the persistence of type through the scale values, are used to distinguish between global and local features with respect to the scale range.

Given a point p, a scale R_i and the corresponding intersection curve $\gamma(p, R_i)$, the point p is classified according to the number of connected components of $\gamma(p, R_i)$, to the curvature value if $\gamma(p, R_i)$ has only one component, or to the relative length if $\gamma(p, R_i)$ has two components, and according to a concavity/convexity check in all cases.

The classification criteria lead to a complete characterization of vertices, which expresses both

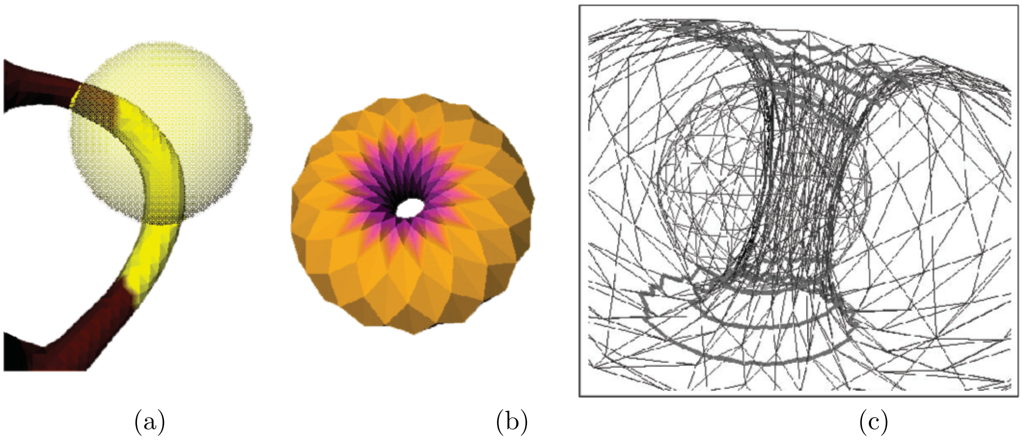


Figure 2.7: A handle (a) is distinguished from a through hole (b), (c).

geometric and morphological properties of the surface. In the following, the main steps of the classification procedure are detailed and the descriptors used to refine each class are introduced. We distinguish between geometric descriptors, which are the surface curvature and the relative length of the intersection curves, and the so-called *status* descriptors which distinguish between concave/convex or empty/full features.

2.3.1 Classification based on intersections

Given a 3D mesh Σ and a set of radii $R_i, i = 1, \ldots, n$, let $S(p, R_i)$ be the sphere of radius R_i and center p, and $\gamma(p, R_i)$ the boundary of the region of Σ containing p delimited by the intersection curves between the mesh and $S(p, R_i)$. Other regions of intersection might occur, but only the one containing p is taken into account. The first morphological characterization of the surface at a vertex p at scale R_i is given by the number of connected components of $\gamma(p, R_i)$. We consider the following cases:

- 1 *component*: the surface around p can be considered topologically equivalent to a plane (see Figure 2.8(a));
- 2 *components*: the surface around p is tubular-shaped (see Figure 2.8(b));
- *3 or more components*: in a neighborhood of p a branching of the surface occurs (see Figure 2.8(c)).

In topological terms, two components identify a handle in the surface, three or more components highlight a split. If $\gamma(p, R_i)$ is made by one component, the angles excess is computed and the vertex is classified as sharp, rounded, or blend (see Section 2.3.2). If $\gamma(p, R_i)$ is made by two components, their lengths are used to distinguish between conic and cylindrical shapes. For branching parts, no further geometric parameters are computed (see Table 2.1).

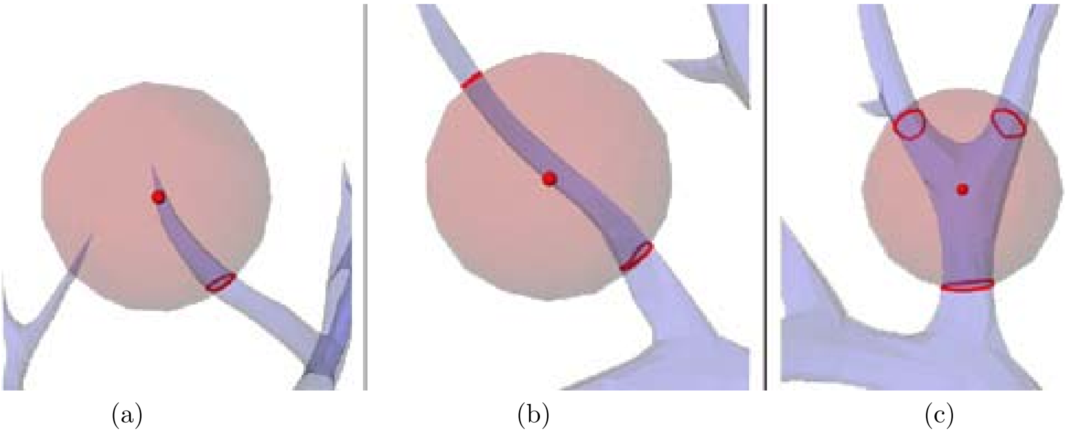


Figure 2.8: Intersection between the input surface and spheres of the same radius R centered in different vertices of Σ: the number of intersection curves changes according with the local shape of Σ around v. In (a) (resp., (b) and (c)) the vertex v, at the level of detail R is classified equivalent to a disc (resp., tubular-shaped and branch).

To run the process, a set of radii must be selected for the computation of the intersections. The maximum and minimum radii (R_{max} and R_{min} respectively) determine an interval which is uniformly sampled according to the number of radii the user wants to use. This step produces the values of $R_i, i = 2, \ldots, n-1$ ($R_1 := R_{min}$ and $R_n := R_{max}$). Both R_{min} and R_{max} can be defined by the user by means of a slider in the GUI; otherwise, they are automatically set proportionally to the size of the surface. In this latter case, R_{min} is set to the minimum edge length and R_{max} to the half of the diagonal bounding box of the surface. The choice of the radius range obviously influences the classification results because it defines the number and sizes of neighborhoods used to analyze the surface shape around each point. In this sense, a-priori knowledge about the feature size of interest surely improves the decomposition.

2.3.2 Curvature characterization

As described in Section 2.2, when $\gamma(p, R_i)$ has only one boundary component, the curvature at the point p, at scale R_i, is the angle excess of $\gamma(p, R_i)$. Instead of using the angle excess, we use the length of $\gamma(p, R_i)$ divided by the radius R_i, i.e. $L_{\gamma(p,R_i)} = \mathit{length}(\gamma(p, R_i))/R_i$. Note that this value has the dimension of an angle and it always assumes a positive value. Since we want to characterize the curvature of a surface, vertices will be labeled as sharp, rounded, or blend points according to their approximated curvature values by establishing some thresholds on the interval $[0, +\infty)$. We can distinguish the following cases (see Figure 2.9):

- *sharp vertices*: let us consider a conic surface, with spike point p and $\alpha \in (0, \pi/2]$ the half amplitude of the cone. Intersecting the cone with a sphere centered in p and with radius R_i generates a circular curve of length $2\pi R_i \sin(\alpha)$ with $L_\gamma(p, R_i) = 2\pi \sin(\alpha)$, which is an increasing function of $\alpha \in (0, \pi/2]$: the lower the value of α, the more the surface around

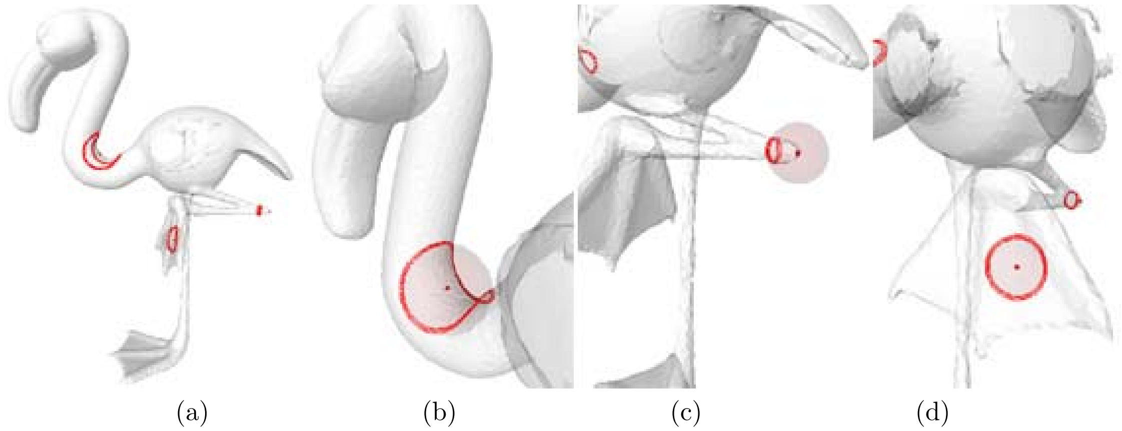


Figure 2.9: Several cases of one intersection curve: note the relation between the intersection curve length and the curvature of the surface in the neighborhood of the center of the sphere. (a) Vertex classification: (b) blend, (c) sharp, (d) planar.

p tends to a cone shaped point. Intuitively, we consider p a sharp vertex if $\alpha \leq \pi/4$ and consequently the curvature threshold is set to $T_s = \sqrt{2}\pi$.

- *rounded/blend*: to distinguish between these two situations, we observe that the surface is *rounded* in a neighborhood of a point if its curvature is positive while a *blend* occurs when the surface is hyperbolic, that is, its curvature is negative. The limit case between the two is obviously a *flat* point and, considering that we use $L_{\gamma(p,R_i)}$ to evaluate the curvature, it is clear that setting the threshold to $T_b = 2\pi$ discriminates between *rounded* and *blend*. The intersection between the sphere and a plane, indeed, results in an intersection curve whose length is equal to $2\pi R_i$.

Summarizing (see Figure 2.10), the characterization of a point p at scale R_i is set as follows:

- $0 \leq L_\gamma(p, R_i) \leq \sqrt{2}\pi$: p is *sharp*;
- $\sqrt{2}\pi < L_\gamma(p, R_i) \leq 2\pi$: p is *rounded*;
- $L_\gamma(p, R_i) > 2\pi$: p is *blend*.

2.3.3 Relative length characterization

We now consider the case of two connected components in the intersection curve $\gamma(p, R_i)$. As mentioned above, this means that p lies on a region of the surface which appears as an elongated shape at that scale, like a tubular protrusion or a handle in the surface. We can specialize this remark as follows: if the length of the two intersection components is nearly the same, the shape at the scale R_i can be approximately considered cylindrical; if one is much longer than the other, it means that the shape may be seen as a conic part (see Figure 2.11). Let γ_1 and γ_2 be the two

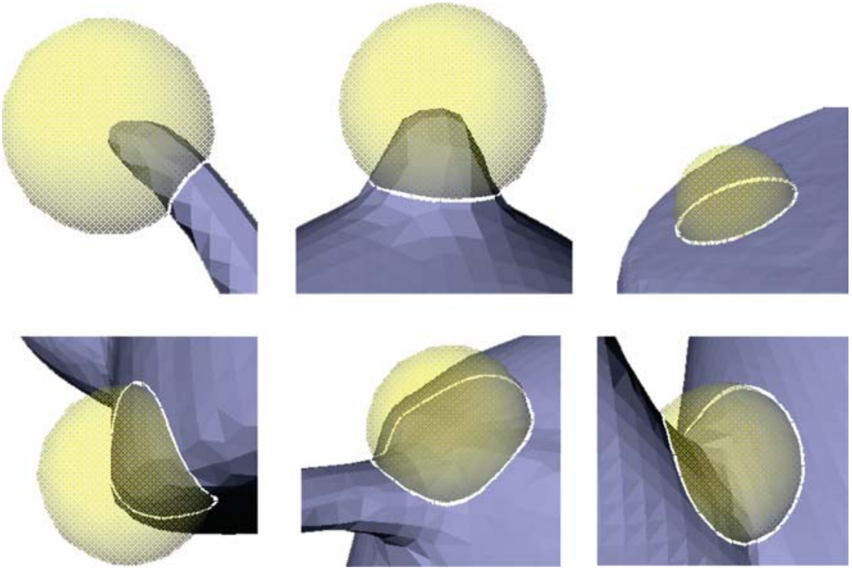


Figure 2.10: Several cases of one intersection curve: note the relation between the intersection curve length and the curvature of the surface in the neighborhood of the center of the sphere. (a) Vertex classification: (b) blend, (c) sharp, (d) planar.

intersection components, and l_1, l_2 their lengths with $l_1 \geq l_2$. The shape is considered conical if $l_1 \geq 2l_2$, cylindrical otherwise. The related threshold is $T_c = 1/2$ which guarantees that the amount l_2/l_1 (belonging to $[0, 1]$) uniquely determines whether the local shape of the surface around p is cylindrical or conic.

2.3.4 Status characterization

The extraction of morphological features on a surface is based on different operators, each related to a different aspect or property of the related feature. Up to now, we have seen how to distinguish between sharp or rounded, tubular or branching and so on, while the *status characterization* allows us to distinguish between convex/concave or, more generally, full/empty features. For instance, to discriminate between convex and concave vertices would lead to refine the classification of a *sharp* point as a tip or a pit, a *rounded* point as a mount or a dip. Obviously, the distinction between convex and concave does not make sense for blend points. For vertices on tubular features, that is with two or more connected components in $\gamma(p, R_i)$, it is checked if the surface intersected by the sphere encloses a volume which is inside or outside the surface (see Figure 2.7). In this case, the status characterization allows to distinguish a handle from a deep tubular depression of the surface.

Let us consider the status characterization in the case of one component first. As for curvature computation, concavity/convexity evaluation at a vertex p of a triangular mesh is highly affected by local noise in the $Star(p)$. The local geometry is depicted in Figure 2.12(a): a given edge e

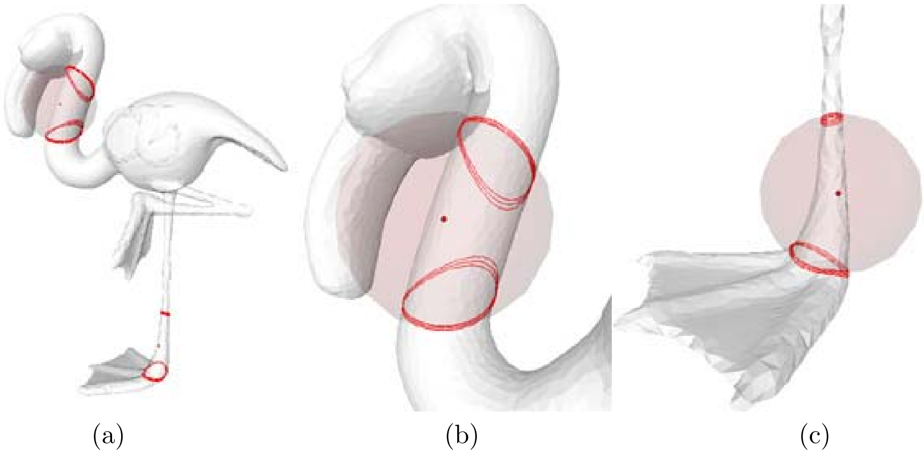
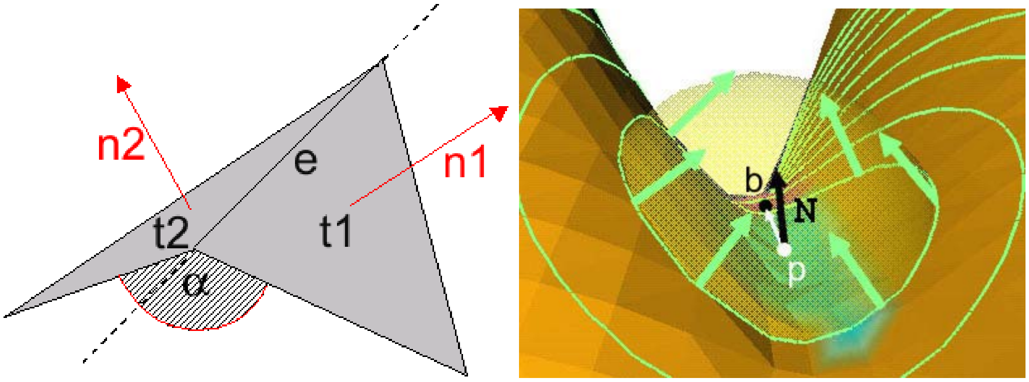


Figure 2.11: (a) Blend vertices classified as (b) cylindrical and (c) conical.

shared by triangles t_1, t_2 of a mesh is convex (resp., concave) if the angle formed by t_1, t_2, inside the surface, is less (resp., more) than π.

Consequently, a given vertex p is defined strictly convex (resp., strictly concave) if all the incident edges in p are convex (resp., concave). Because in most cases the incident edges in p are both convex and concave, the previous classification cannot be applied. Note that even a slight modification of the point coordinates can produce a complete different classification. For these reasons, the method adopted for assigning a *convex* or *concave* label to a vertex p at scale R_i again uses the intersection between the mesh and the sphere. In the case of one connected component of the intersection curve, the center of mass b of γ and the average normal N of the intersected triangles are computed. The vertex p is considered concave (convex) at scale R_i, if p lies below (resp., above) γ, that is $< N, b - p > 0$ (resp., < 0), where '$<\, , \,>$' denotes the inner product. We refer to Figure 2.12(b) for an easier understanding.

Suppose now that $\gamma(p, R_i)$ has two intersection components. Again we can distinguish between the case in which the local shape is a tubular protrusion or a tubular well of the surface (see Figure 2.7(a), (b)), in analogy with the property of convex/concave mentioned above for points generating one intersection curve. If the number of connected components of the intersection curve is two, as in Figure 2.13(a), we consider the orientation of each component of $\gamma(p, R_i)$ as naturally induced by the surface orientation (see Figure 2.13(b)). If p lies on a tubular protrusion of the surface, it happens that the normal vector of the average plane related to each connected component of $\gamma(p, R_i)$ is directed towards p (see Figure 2.13(c)) according to the right-hand rule; if p appears on a tubular depression of the surface, the vectors have opposite directions. This statement holds for three or more connected components too, thus it is possible to discriminate between a branch on the outer surface or a splitting cavity.



(a) (b)

Figure 2.12: (a) Edge concavity or convexity criterion and (b) configuration of the intersection curve normals around a concave point.

2.4 Algorithm and implementation details

This section deals with the implementation details of the algorithm used for the sphere-mesh intersection, which is the core of the shape characterization. Throughout the discussion we refer to the pseudo-code given at the end of the section. The mesh is encoded by means of a triangle-based data structure which stores:

- for each triangle, its three vertices and its three adjacent triangles, which represent the Triangle-Vertex (TV) and Triangle-Triangle (TT) relations, respectively
- for each vertex, its coordinates and one (arbitrary) of its incident triangles (VT^*).

The Vertex-Triangle relation (VT) associating to a vertex all its incident triangles is necessary to navigate the mesh, but the memory space required can be strongly optimized by coding just one of those triangles per vertex (partial Vertex-Triangle relation or VT^*). The total VT relation can be retrieved in linear time by iteratively applying the TT relation starting from the stored triangle. A scheme of the data structure is given in Figure 2.14. The storage of this data structure requires $3n_V * sizeof(float) + 6 * n_T * sizeof(int) + n_V * sizeof(int)$, since vertices and triangles occur in the TT, TV, and VT^* structures as integers.

For each vertex v the computation of the intersection curves between the mesh and a set of n spheres centered in v with increasing radii $R_1,\ldots,\ R_n$ is computed as follows:

- one of the triangles incident in v is inserted in the queue Q (this operation takes constant time if we have the $VT^\star$ data structure) and it is marked as visited;
- a triangle t is extracted from Q and the main loop is repeated until Q is empty;

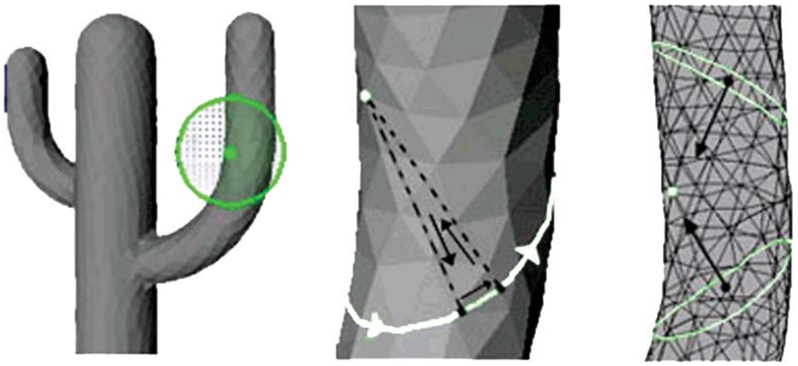


Figure 2.13: Situation in which two orientation curves are generated: (a), (b) curve orientation derived by triangle orientation, (c) average normal of intersecting curves.

- the algorithm checks if the spheres intersect t: for each radius R_i, if at least two vertices p, q of t, distinct from v, satisfy the conditions $\|p - v\|_2 \leq R_i$, $\|q - v\|_2 \geq R_i$, t is intersected by the sphere $S(v, R_i)$. This operation takes constant time. In this case, t is considered as a seed triangle for tracing the whole line of intersection whose continuation is searched in the triangles adjacent to t. The function *intersection* is thus invoked to complete the intersection curve starting from t and moving on its neighbors intersected by $S(v, R_i)$. The curve $t \cap S(v, R_i)$ is calculated considering the intersection points between the sphere and each edge of t. More precisely, given an edge $[a, b]$ of t, we can parameterize it as $u(s) := sa + (1 - s)b$, $0 \leq s \leq 1$; the intersection points, if any, are located as $u(s_0)$, $u(s_1)$ with s_0, s_1 solutions in $[0, 1]$ of the equation $\|u(s) - v\|_2^2 = R^2$ of degree two in the unknown s. As shown in Figure 2.15, the length of $\gamma(v, R_i)$ is the sum of the composing arc lengths, each one belonging to an intersecting triangle, and given by $R_i\theta$ where θ is the angle $\widehat{apb}$, a, b being the intersection points. The call of this function increases the number of connected components of the intersection line for a given radius. Moreover, the neighbors of the intersection triangles traversed but not marked, that is, those ones which lay outside $S(v, R_i)$ but inside $S(v, R_n)$, are inserted in Q;
- If t is not an intersection triangle and it lies inside the sphere of maximum radius $S(v, R_n)$, its neighbor triangles (if not marked) are inserted in Q; otherwise, it is simply discarded.

The construction of a connected component of the intersection line may take as many constant operations as the number of intersection triangles, i.e. $O(n_T)$ in the worst case. However, in this implementation each triangle is visited only once marking triangles when they are inserted in Q avoids to consider them more than once, and the intersection triangles traversed during the execution of *intersection* are not stored in Q. Therefore, the main loop takes $O(n_T)$, that is $O(n_V)$; doing this operation on the whole mesh takes $O(n_V^2)$ time. Note that if the step between the radii is small with respect to the average edge length, a triangle can be easily intersected by more than one sphere, and the function *intersection* could be invoked on the same triangle as many times

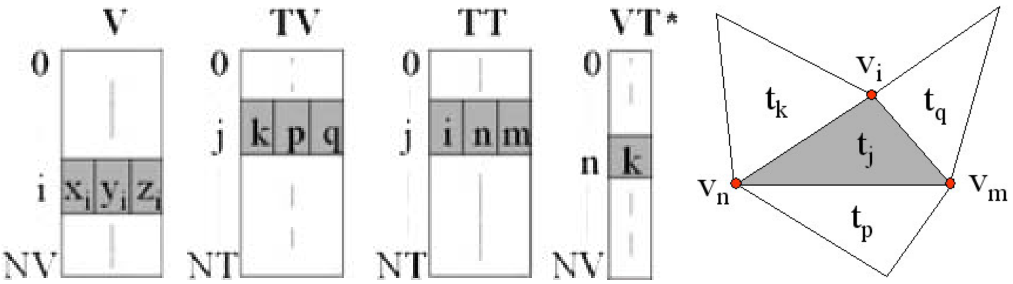


Figure 2.14: Data structure organization: the information for the vertex v_i and the triangle t_j are shown.

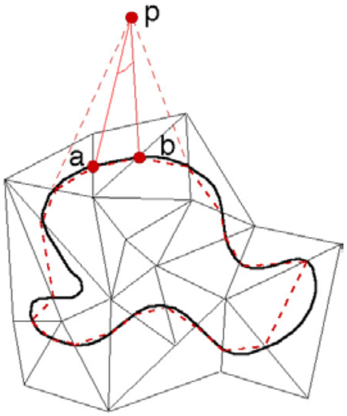


Figure 2.15: Approximated length of the intersection paths.

as the number of radii. Anyway, once the number n of radii is chosen this is a constant value, so that the loop (L) does not increase the complexity.

```
main(){
$Q = \emptyset$;
$\sharp$(connected comp. for $R_i$):=0;
for all $v \in V$ {
        $Q \longleftarrow TV^\star(v)$;
        // main loop
        while( $Q \neq \emptyset$ ) {
                $t$ = first element removed from $Q$;
                (L) for $(R_i = R_1, \ldots, R_n)$
                        if ( $\exists\ v_l, v_m \in TV(t)\ :\ \|v - v_l\|_2 \leq R_i$
                              & $\|v - v_m\|_2 \geq R_i$ ){
                              $intersection(t, R_i, v)$;
                              $\sharp$ (connec. comp. for $R_i$) ++;}
        }
for all $t_i \in TT(t)$
        if( $t_i$ is not marked & $(\exists v_l \in TV(t_i) : \|v - v_l\|_2 \leq R_n)$ )
            $Q \longleftarrow t_i$;
}

intersection($t$,$R_i$,$v$) {
$L_i$= intersectionLength($t$,$R_i$,$v$);
$t_{next} = t$;
do {
    $t_{next} = t_i \in TT[t_{next}] : t_i \cap S(v, R_i) = \emptyset$
    $L_i$ += intersectionLength($t_{next}$,$R_i$);
    if( $t_i$ is not marked & $(\exists v_l \in TV(j,i) : \|v - v_l\| \leq R_{max})$ )
        $Q \longleftarrow t_i$;
        } while($t_{next} \neq t$);
}
```

2.5 Mesh decomposition

The focus of this section is the integration of the different characterizations, described in Section 2.3, to achieve a unique segmentation of the input mesh into morphological features represented by closed regions with uniform properties. In Table 2.1, a summary of the labels assigned to vertices, for a given scale R_i, is shown. After the label assignment, the input mesh is decomposed into patches using a region-growing procedure, and each patch corresponds to a shape feature relevant at scale R_i.

The morphological classification associates a vector of feature labels to each vertex, and each label describes the vertex at the corresponding scale. Selecting the scale of interest, the surface can be rendered using a color-coding of the feature labels. The achieved decomposition is an affine-invariant segmentation into disjoint, non-empty subsets which code the geometry and shape evolution through scale changes. In Figure 2.16, an example is shown; the different views display the mesh decomposition at different scales, and the colors are those give in Table 2.1.

The tools defined are used to analyze a shape at different scales, but they can also be used to derive information about the *persistence* of a shape feature across the scale range. It is also possible to define a basic query language which allows to extract features defined by the user as relations among morphological labels at different scales. For example, there has been defined a coarse feature-based query language, which allows the user to submit a query like "which are the vertices whose feature type is TIP at scale R_3 and MOUNT at scale R_6?". To this purpose, a query vector with wild card v_q, where $v_q[i]$ specifies the requested feature type at scale R_i, and the

Table 2.1: Morphological feature characterization.

Label	Feature	Color	#∩	Geometric	Status
T	TIP	red	1	$L/R_i \leq T_s$	convex
P	PIT	blue	1	$L/Ri \leq T_s$	concave
M	MOUNT	orange	1	$T_s < L/Ri \leq T_b$	convex
D	DIP	cyan	1	$T_s < L/R_i \leq T_b$	concave
B	BLEND	pink	1	$L/R_i > T_b$	–
L	LIMB	yellow	2	$L_2/L_1 \geq T_c$	full
W	WELL	violet	2	$L_2/L_1 \geq T_c$	empty
J	JOINT	brown	2	$L_2/L_1 < T_c$	full
F	FUNNEL	gray	2	$L_2/L_1 < T_c$	empty
S	SPLIT	green	≥ 3	–	full
H	HOLLOW-Y	black	≥ 3	–	empty

AND/OR Boolean operators are used. For instance, suppose we used a set of ten radii; the vector [*,*,T,*,*,M,*,*,*,*] with the AND operator specify the previous query. Here, feature labels are those in Table 2.1, and the symbol '*' means that, at that scale, each feature type is allowed.

The combination of morphological labels with logical operators enables the construction of a high-level language for shape interrogation guaranteeing a multi-task model. In fact, the user is able to extract a single shape element using a single query, to combine them and, as discussed in Chapter 4, to locally modify the geometry, by using other surface patches, or modifying the topology by changing the structure of the adjacency graph which codes the shape decomposition.

The results obtained (see Figure 2.17) suggest that further improvements of the query language will allow the extraction of higher-level features, like handles or main body of a given surface. Tubular components can be extracted choosing LIMB OR JOINT vertices; among them, handles correspond to cycles in the region adjacency graph, and protrusions are adjacent to TIP or MOUNT zones. Selecting points which assume TIP OR MOUNT OR DIP OR BLEND features at most levels of detail identify the main body of the surface, and so on. Using the described language, a mesh can be analyzed in a rather flexible way.

Instead, if we are willing to extract a global shape description which takes into account the whole range of scales into a single decomposition of the mesh, the following voting algorithm for persistence can be used. First, the points are classified according to the intersection connectivity, that is, according to the number of single, double, or multiple components in the intersection boundary, considering the whole range of scales. The mesh is therefore segmented in parts which are characterized by having either almost always one intersection, or two, or more. This step provides a first insight on features which are persistently protrusion-like, handle-like or branch-like, without distinguishing if they are convex/concave or full/empty. In Figure 2.18(a), the result of this segmentation is shown, where the blue parts are composed by vertices having only one intersection for more than the 75% of scales, the red are those having two intersections for the same threshold. Finally, the grey areas are those corresponding to shape transitions where both

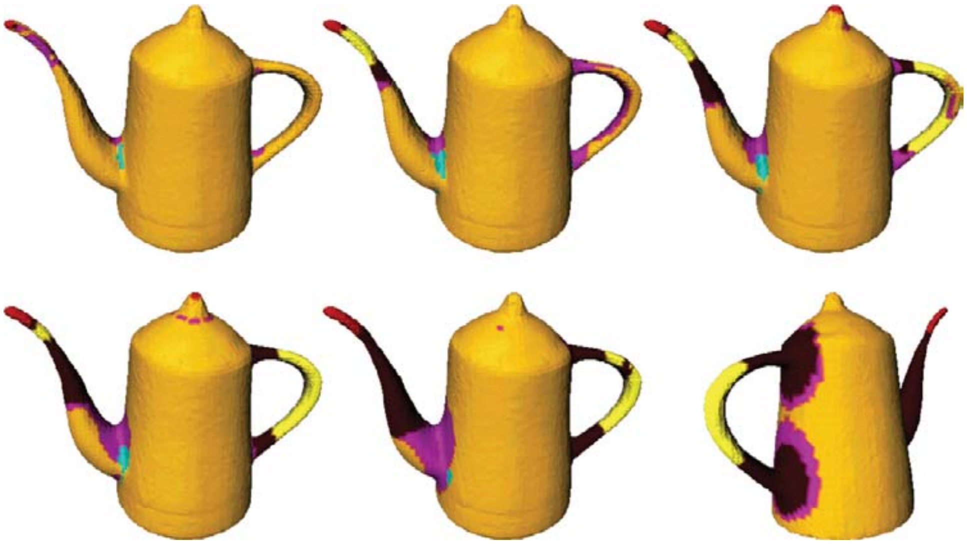


Figure 2.16: Shape segmentation on the pot at different scales.

one and two intersections occur approximately in the same percentage.

Within each of the resulting parts, a further classification can be done considering the related geometric characterization. For example, in Figure 2.18(b) the shape vertices are colored with different blue saturations depending on the curvature; a darker blue corresponds to a higher curvature. An analogous criterion is applied to vertices with two intersections using their relative length where a lighter red is related to cylindrical-like features. With regard to the choice of the threshold for the persistence analysis, the value has to be tuned according to the number of radii used. In our experiments, we used an average of ten radii and the value 75% provided good results. A threshold bigger than 75% has also been used, and it results in a stricter selection of patches with one or two intersections; this choice generally enlarges transition areas.

2.6 Method insights

In this section, results and properties of the shape characterization are discussed, focusing on the comparison between the multi-scale and the persistence analysis. Examples of an application of this method for skeleton extraction are also given in Chapter 3.

First of all, let us discuss the behavior of the curvature evaluation using different scales. The curvature is analyzed in a neighborhood whose size depends on R_i: for small values of R_i, such as the average length of the edges incident in p, the curvature approximation resembles the discrete curvature estimation proposed in [DMSB02] and shows similar problems while, for increasing values, it becomes more stable to noise. We have tested the behavior of the curvature on smooth and rough surfaces, as follows.

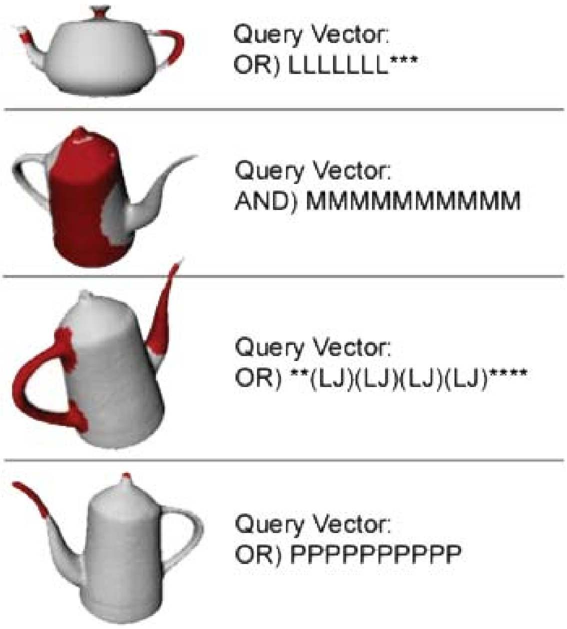


Figure 2.17: Queries with matched points are depicted in red. The use of AND, OR operators among the scales is specified before the label vector. The round parenthesis between labels work as OR between feature at the same scale.

The first test surface used is a torus and the multi-resolution curvature has been compared using two radii: R_i equal to the minimum edge of the input mesh and $2R_i$. Curvature values achieved with R_i corresponds to the theoretical point classification into parabolic, hyperbolic, and elliptic regions on the torus (see Figure 2.19(a)). Increasing the radius to $2R_i$, results in a similar classification where parabolic points, identified as the boundary between hyperbolic and elliptic ones, are shifted with respect to their theoretical position represented by the green line (see Figure 2.19(b), (c)).

The shape decomposition has been tested for stability to noise also on a rough surface. In Figure 2.20(a) and (c), the results obtained on the rabbit and on the same data set with added noise are compared. The main features, like tips and limbs, are preserved and the influence of the thresholds involved is lessened by the persistence analysis (see Figure 2.20(b) and (d)).

The multi-scale decomposition depends on the chosen set of radii, and if a too small radius leads to noise problems on the other hand, a too large radius can give a meaningless result. Small radii can be used to determine detail features while bigger ones are able to capture global characteristics of the surface. From these considerations, it follows that the choice of R_i is related to the scale of the features which have to be extracted, and the use of a set of increasing radii is suitable for performing a multi-scale analysis of the shape.

The multi-scale analysis can be used also for a generic segmentation of the surface according to the curvature evaluation, by grouping those points which share a similar curvature value with respect

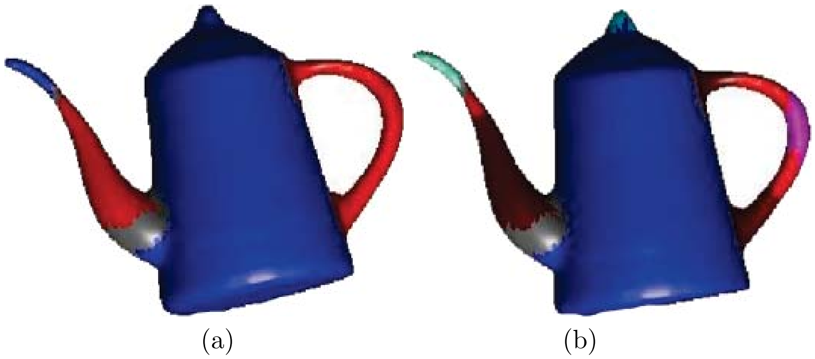


Figure 2.18: (a) Persistence analysis of the number of intersection curves and (b) its refinement using geometric information.

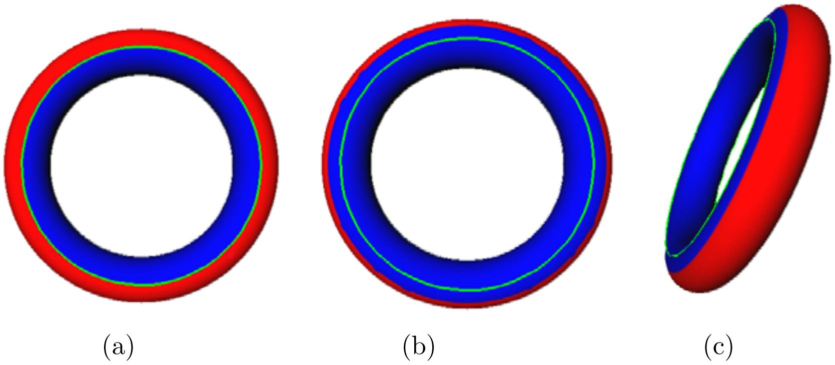


Figure 2.19: The point classification corresponding to R_i chosen as the minimum edge: (a) red and blue vertices locate elliptic and hyperbolic points while the green line visualizes the theoretical parabolic line. The results obtained with radius $2R_i$ are shown in (b), (c).

to a given threshold ϵ and a radius R_i, i.e.

$$p, q \,\text{belong to the same patch} \Longleftrightarrow |L_\gamma(p, R_i) - L_\gamma(q, R_i)| \leq \epsilon.$$

This approach is commonly applied by segmentation methods based on curvature. Depending on the type and complexity of the shape to be analyzed, the simple curvature evaluation may give better results than the refined segmentation. The segmentation described by the labels in Table 2.1 corresponds, indeed, to a decomposition into protrusion-like and handle-like features which might be not relevant for certain shapes. As shown in Figure 2.21, shapes composed by detailed and unstructured features are better analyzed with the multi-scale curvature evaluation, because in such case a decomposition into features, such as limbs and tips, is not meaningful. Future developments will mainly focus on the specification of the feature adjacency graph and on the study of its evolution within the scale range required.

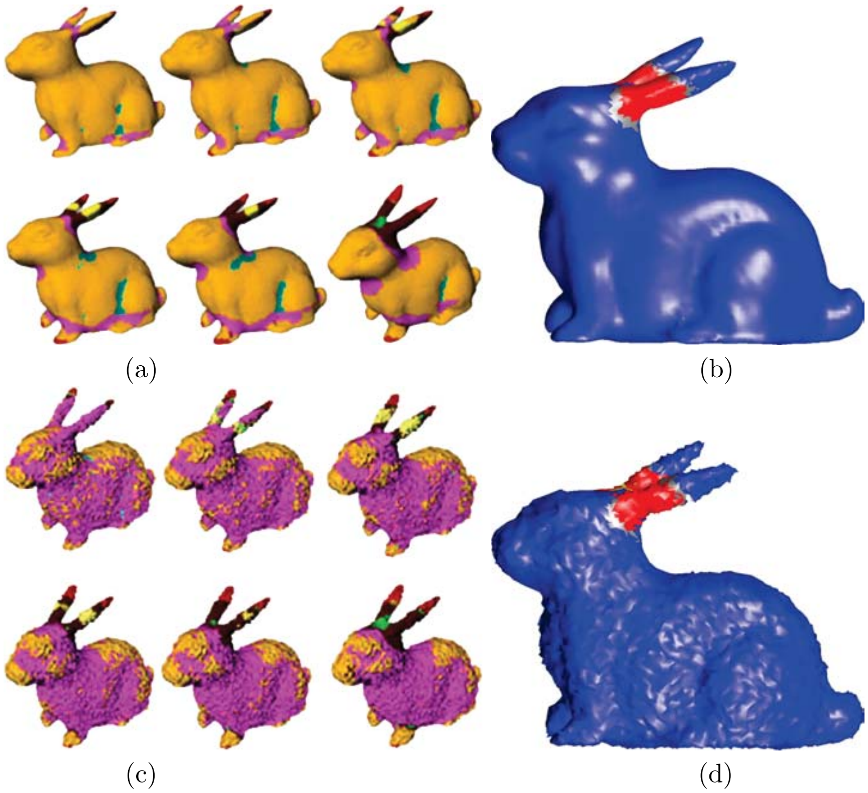


Figure 2.20: (a), (b) Shape segmentation and persistence analysis on the original rabbit, and (c), (d) on the model with added noise. Achieved segmentation based on persistence analysis are nearly identical.

2.7 Feature-based segmentation

The evolution of intersection curves produced by blowing bubbles at mesh vertices has been proved to be a good approach to characterize a shape using meaningful shape features. Increasing the radius of the bubbles produces an easy and efficient multi-scale analysis of the shape, which can be effectively used to produce a set of specific and flexible tools for shape analysis. The resulting description is an affine-invariant segmentation of disjoint, non-empty subsets and equally distributed in all directions, which codes the geometry and shape evolution through scale changes. Finally, the decomposition is computationally affordable and consistent with previous work on these topics.

In the following, we specialize this shape classification method for detecting tubular features of 3D surfaces represented by a triangle mesh. The *Plumber* algorithm segments a surface into connected components that are either body parts or elongated features, that is, handle-like and protrusion-like features, together with their concave counterparts, i.e. narrow tunnels and wells. The segmentation can be done at single or multi-scale, and produces a shape graph which codes how the tubular components are attached to the main body parts. Moreover, each tubular feature

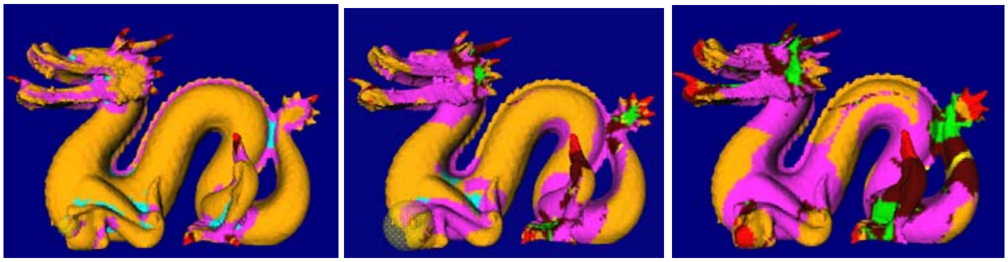


Figure 2.21: Feature decomposition on the dragon at three different scales.

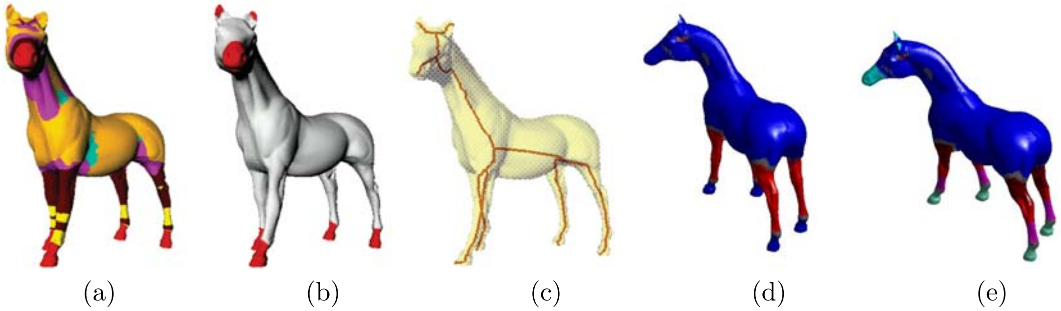


Figure 2.22: Global framework: (a) curvature estimation on the horse with different radii, (b) peak regions are extracted with a query, (c) regions selected in (b) are used as seed points for extracting the skeleton, (d), coarse persistence analysis, (e) refined persistence analysis.

is represented by its skeletal line and an average cross-section radius.

Given a two-manifold closed surface represented by a triangle mesh, *Plumber* automatically extracts the features that can be described as generalized cylinders or cones; we call these features, together with their concave counterparts, i.e. narrow tunnels and wells, *tubular features.* The *Plumber* approach classifies the vertices of a given triangle mesh according to their curvature and shape behavior in neighborhoods of increasing size (see Figures 2.23, 2.24). Seed vertices are located on tubular features, and clustered to form candidate seed regions which are then used to compute the first reliable tube section, called the *medial loop*, which is ensured to be around each candidate tube and which works as a generator of the feature. Then, the medial loop is moved in both directions on the surface, by using spheres placed at the barycenters of the new medial loops, until the tube is completely swept. The size of the tube is related to the radius of the sphere, and the stop criterion is given by the abrupt variation of the medial loops' lenght. The tube detection is devised in order to work in a multi-scale fashion, where small tubes are detected at first and larger ones at following steps. After the surface segmentation, a geometric representation of each tubular feature is constructed by computing its skeletal line. The configuration of each feature, whose section and length can arbitrarily vary, and its attachments to the body are hierarchically coded in a shape graph.

Different application fields make the surface segmentation an important task. For instance, while tubular structures can be quite easily defined during design processes their automatic extraction

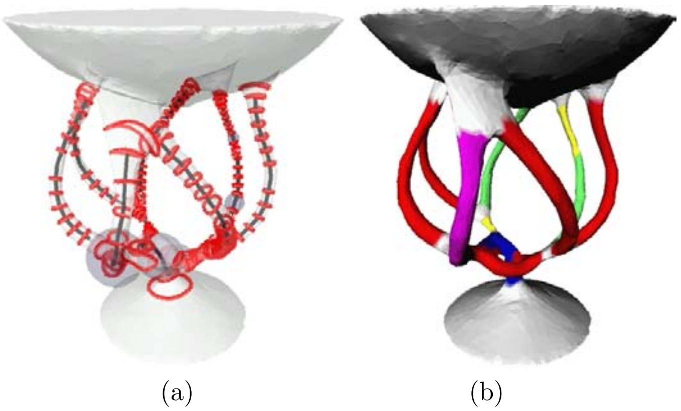


Figure 2.23: Tubular features recognized by *Plumber* on a complex model: (a) tube axis and loops, (b) tubes colored with respect to their scale.

from 3D meshes is not a trivial task. We believe that a variety of applications, especially shape recognition and analysis, will benefit if tubular features are identified and abstracted to a centerline and a set of sections. These abstract models, may facilitate the measurements of changes over time in medical applications (e.g., calcification process), or detect abnormalities such as unnatural narrowing or ballooning. Finally, reliable cylindrical models are essential for proper design of prosthetic tubular structures.

The basic idea of *Plumber* consists of describing a shape by using both local point-wise, and global region-wise measures for shape decomposition and skeleton extraction; in the following, we review the state of the art on those concepts used in the chapter.

In [KT03], the segmentation method is defined as a fuzzy clustering of vertices where the probability that a face belongs to a patch depends on its distance from the other faces of the patch. The advantage of the method is the avoidance of over-segmentation, and that boundaries between adjacent regions are not jagged. The results show that the segmentation is meaningful, in the sense that the extracted components locate the main natural features of the surfaces. In [LTH01], tubular parts are identified using a sweeping techniques along the arcs of the skeleton which is constructed by joining the edges remaining after an edge collapse process on the whole mesh. These edges are linked in a tree structure, and it is used as a support for the sweeping process where the mesh is intersected by a set of planes and tubes are identified by looking at the geometry of the cross-sections.

The main difference between *Plumber* and segmentation methods previously discussed is that we extract building primitives of the surface with a specific structure, i.e. generalized cones and cylinders, and not only related to a curvature and concavity analysis [MPS$^+$04a, KT03]. Furthermore, while skeletal representations do not provide a scale-based decomposition of the shape and are usually unstable with respect to wripples or wrinkles, *Plumber* differentiates tubular features of different dimension.

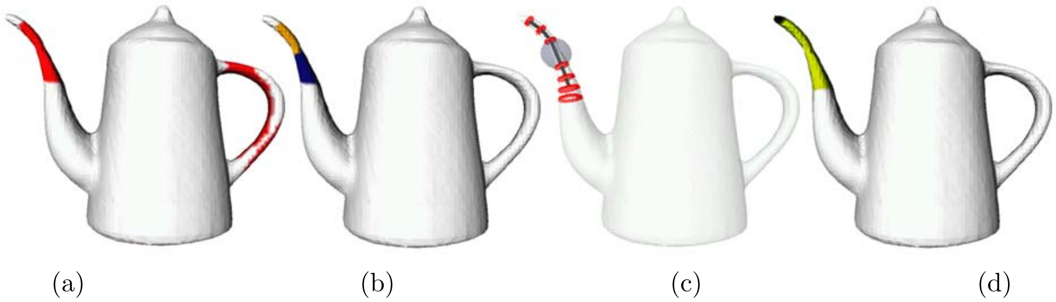


Figure 2.24: *Plumber* method: (a) identification of limb vertices, (b) extraction of their connected components and medial loop, (c) iteration, (d) tube and a cap (black) found at this scale.

2.8 The *Plumber* method

Intuitively, ideal tubes are identified by parts of the shape whose intersection with a sphere of appropriate radius produces two intersection curves. The section of the tube and its axis can be arbitrarily shaped, and the size of the tube is kept as a constraint during the identification process. Chosen a level of detail R, *Plumber* performs the following steps:

1. identify *limb-regions* associated with at least two loops on M (see Figure 2.24(a)). This classification is achieved by using the approach described in the first part of this chapter;
2. shrink each of the two selected boundary components along the surface to its medial-loop, whose points are nearly equidistant from the two border loops (see Figure 2.24(b));
3. expand-back the medial-loop by sweeping the extent of the shape in both directions. More precisely, at each iteration we place a sphere of radius R in the barycenter of the new medial loops. If the intersection between the sphere and the surface generates two loops, mesh vertices inside the sphere are marked as visited;
4. the procedure is iterated in both directions until:
   - no more loops are found, or more than one loop is found on not-visited regions;
   - the new loop lies on triangles that are already part of another tube, or the length of the new loop exceeds a pre-defined threshold.
5. the tube skeleton is extracted by joining the loops' barycenters.

The previous steps are detailed in the following paragraphs.

Vertex classification. Intersecting the mesh with a sphere with radius R_i allows to identify limb-vertices if they lay on a tube of diameter R_i or smaller. At each vertex $p \in M$, we consider three spheres of radius $R_i - \epsilon$, R_i, and $R_i + \epsilon$ with ϵ given threshold proportional to the minimum

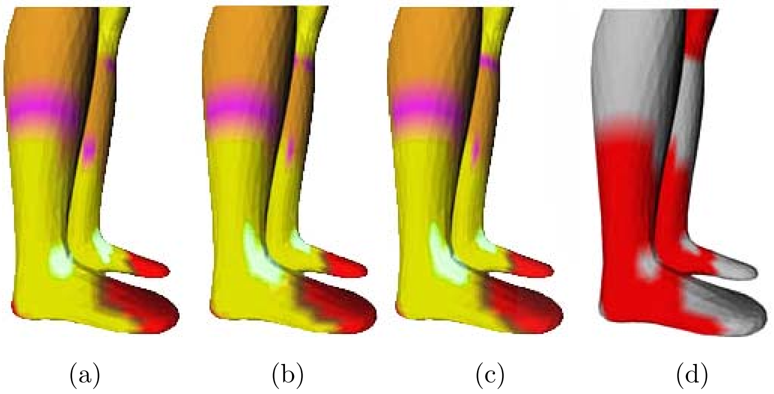


Figure 2.25: In yellow limb-vertices found at scale $R - \epsilon$ (a), R (b), and $R + \epsilon$ (c). All the limb vertices are depicted in (d).

edge in the triangulation. We consider limb vertices those ones whose curve γ_i has two or more boundary components (see Figure 2.25). This classification improves stability for identifying tubes of arbitrary cross section where isolated limb-vertices could appear; a stricter choice consists of considering as limb vertices those ones classified with the same label at all the three scales.

Identification of tube candidates from limb vertices. The second step is the identification of the maximal edge-connected components of limb-vertices, using a depth-first search. Note that while the analysis of the evolution of γ_i produces a vertex-oriented classification of M, regions composed by limb-vertices are not guaranteed to be tube-shaped as a whole (see Figure 2.24(a), on the handle). In particular, limb regions may have not two boundary curves. Therefore, the next step defines a criterion for judging if a limb-region is a good candidate for the tube identification; that is, the limb-region is *around* the tube. For instance, in the case that the tube section is ellipsoidal and its size is greater than the chosen scale, it may happen that the spheres used to classify the vertices produce only one intersection curve on one side of the tube, and two on the other side (see Figure 2.26), thus giving rise to a limb-region not surrounding the tube. Therefore, the region is not tube-shaped at the given scale and it has to be discarded; it will be found at a larger scale.

Medial loop generation. Seed tubular regions are used to construct a medial loop around each candidate tube that will be used for the tube identification and its centerline construction.

Since we have already detected all the candidate tubular regions, a seed point for each tube is selected; for instance, we could choose the centroid of each region, i.e. the point with maximum distance from the region boundary, and then generate the loop with one of the methods proposed in [VF02, GW01, LPVV01]. Instead, *Plumber* relies on the limb-region boundaries which are loops surrounding the tube. The idea is to find the medial loop by moving the boundary loops in the middle of the limb-region; to this end, we perform a morphological shrink by simultaneously

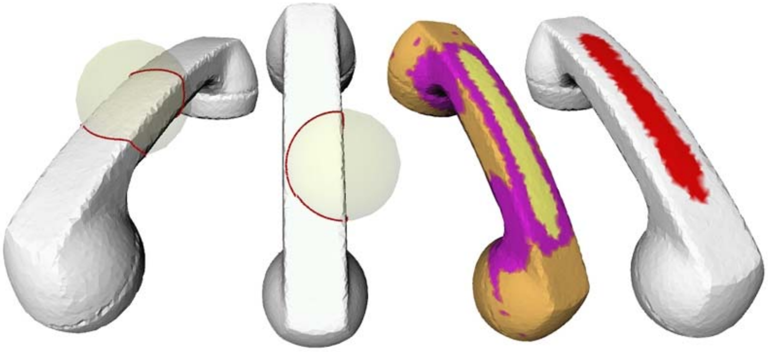


Figure 2.26: Example of limb-regions (in yellow) whose vertices on γ have one boundary component.

invading the component from its two boundary components.

Firstly, the two boundary components of the limb-region (the two of greater length if the region has three or more border loops, as in Figure 2.27(e)) are computed. Let R and L (for "right" and "left" respectively) be the two boundary components of the tube; at first, each vertex p on L is associated with the couple $(0, +\infty)$ that indicates that p has distance 0 (resp., $+\infty$) from the boundary L (resp., R). The same initialization applies to R.

The distance of a vertex p from one boundary is computed as the shortest edge path connecting p with a boundary vertex. Then, the distance values of all points are updated, propagating from L and R towards the interior of the region. The distance propagation from L will update the first value of the distance vector, while that from L will affect the second field; at the end of the process, vertices are classified as nearer to L or R (see Figure 2.28). Edges connecting vertices of different type are cut by the medial loop we are looking for; that is, we join the mid-points of those "medial edges" to produce a medial loop. This construction achieves two good effects with respect to other methods [Kar99, VF02]: the medial loop is guaranteed to be non-trivial and inside the region. The non-minimality of its length does not affect the growth of the tube and the construction of the skeleton. In the case of three or more boundary components, the choice of starting from the two loops of greater length is to guarantee a stronger reliability to the tube extraction with respect to smaller intersection curves which may be due to local undulations of the shape.

Loop expansion and skeleton construction. The loop expansion is controlled by a verification procedure which, at each step, extends the center-line and at the same time ensures that the surface is tubular around it. A first medial sphere is drawn, whose center p is the barycenter of the medial loop, and whose radius is R. If $M \cap S(p, R)$ has not two boundary components, the growing stops and the candidate tube is discarded. Otherwise, a new sphere with the same radius is centered in the barycenter of the two loops; the process is then split into two parts, trying to grow the tube in both directions. Now we focus on the sphere moving in one of the two directions,

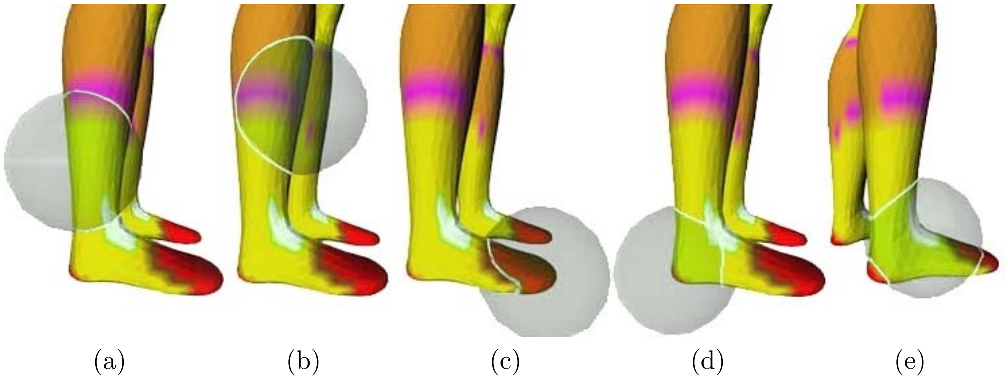


Figure 2.27: A tubular region affected by small features, like the neel. The configuration of the sphere/mesh intersection is depicted, with spheres centered in vertices of different feature types: (a) limb, (b) blend, (c), (d) tip, (e) split.

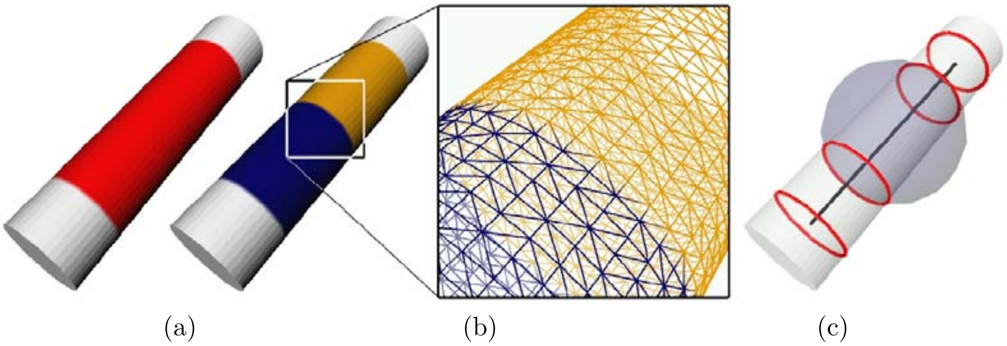


Figure 2.28: (a) Limb vertices, (b) connected component of the limb vertices with two boundary components, and medial loop (marked curve), (c) medial sphere centered in the barycenter of the medial loop, and tube growing.

since the other case is symmetric.

At each iteration, the sphere rolls to the barycenter of the next loop and the triangles laying completely or partially inside the sphere are marked as belonging to that tube. Then, the intersection between the sphere in the new position and the mesh is again computed, taking into account only the intersection curves through non visited triangles (all the spheres except the medial one have always a “backward” loop, passing on the already marked triangles). During the loop expansion, the following cases may arise:

- no intersection curves are found. This is the case of a tubular protrusion terminating in a tip; visited triangles locate a *cap* (see Figure 2.29(a), in the square);
- the intersection curve consists of one loop (see Figure 2.29(a)). If its length is less than a pre-defined threshold, the size of the tube section is not varying too much; the loop becomes a new cross section and its barycenter contributes to the skeleton as a new node. Otherwise

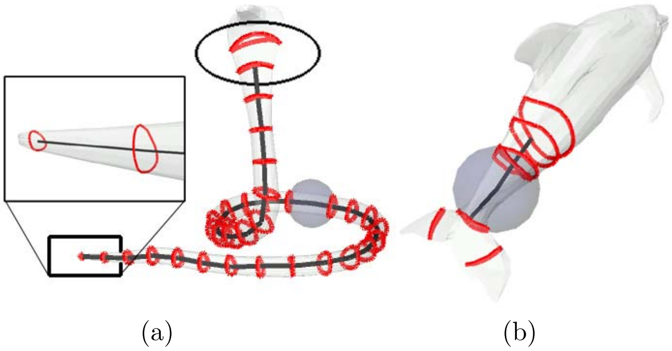


Figure 2.29: (a) No new loop is found on the snake tail (in the box), and a loop discarded after the length check on the head (in the oval). (b) A branching occurs on the dolphin tail.

(see Figure 2.29(b), in the oval), the growth stops.

- the intersection counts two, or more loops; that is, a bifurcation occurs (see Figure 2.29(b)). The growing of the tube in this direction stops and the last visited triangles are unmarked.

Finally, the barycenters of the medial loops are joined to define the tube skeleton.

Multi-scale analysis. The extraction of tubes at scales $R_1, \ldots, R_n$ adopts a fine-to coarse strategy, marking triangles as visited while the tube grows and which are not taken into account during the following steps (see Figure 2.30). Analogously, the medial loop computation simply does not take into account smaller tube vertices, propagating distance values only on not-visited vertices (Figure 2.30(d)). Decisions are taken when the loop passes partially on not-visited and tube triangles. For example, in the case depicted in Figure 2.30(e), the two smaller loops fall on tube triangles, and are not counted; therefore, this is the case of two intersection loops and not that of a branching. The tube is grown and the result of the two iteration steps is shown in Figure 2.30(f). This set of radii is selected by the user or assigned by uniformly sampling the interval from the minimum edge length in M to that of the diagonal of its bounding box.

At the end of the whole process, tubes are labeled with respect to the scale at which they were found. The connected components of the shape parts which are not classified as tubes or caps define *body* parts of the surface, and the resulting decomposition is coded in a tube-body connectivity graph which represents the spatial arrangement of the tubular features onto bodies. A smooth transition of radii ensures a meaningful growth of the tube at a scale R_i, while discarding smaller features and analyzed at the previous levels of detail $R_j, j = 1, \ldots, i-1$.

Strict/non strict mode. Together with the size of the sphere rolling over the centerline, the other parameter to be fixed in the tube growing step is the threshold in the loop length check. To this end, we stop the growing when the tube becomes too large, i.e. the length of the intersection loops varies too much.

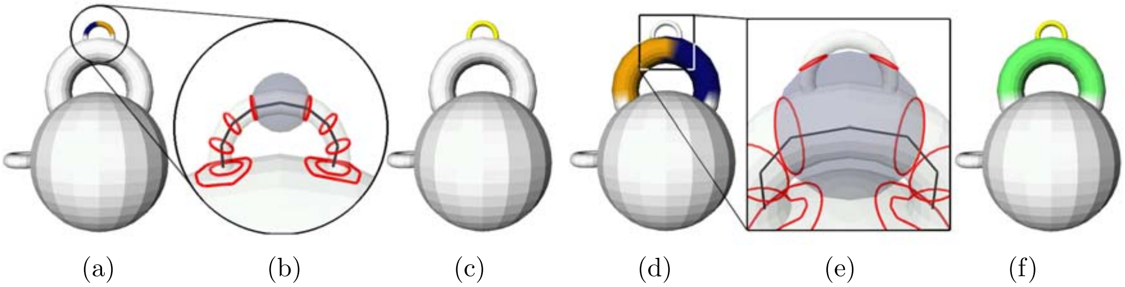


Figure 2.30: Iteration of *Plumber* at increasing scales.

If we consider a natural surface, we probably do not want to decompose natural limbs into pieces; on the other hand, in the case of a manufactured model, we may want to be precise with respect to the tube size, eventually splitting a tube into components of (even slightly) different sections. For this reason, two alternatives are available (see Figure 2.31):

- a *strict* mode, useful in CAD and medical applications;
- a *non-strict* mode, for other applications where a continuous variation of the tube size does not require to split the tube.

In the strict mode, each time a new loop is generated its length is compared with that of the intersection loops associated to the medial sphere at the beginning of the process, and not with the length of the medial loop tube section which can be non minimal and misleading. Other choices were also taken into account, such as average, minimum, and maximum tube length; the one adopted is a compromise between the required strictness and robustness. In the non strict mode, a loop is accepted if its length is less or equal to twice the length of the previous loop. In both cases, the user can select values on the base of a-priori information or specific needs.

Shape graph. Throughout the previous paragraphs we have detailed a method for identifying and classifying tubes of different size and bodies achieving a segmentation of the input surface. We enrich this geometric classification with an explicit representation of the structure of the model which codes the relations between primitives in a hierarchy of tubes and bodies. This structured representation is a *shape-graph* whose nodes are the extracted primitive shapes, while the arcs code the adjacency relation among the previous ones, i.e. their relative position and orientation. Each node is a *tube*, whose labels are the medium radius and the axis length, a *body*, whose labels are the number of holes and the approximate volume, or a *cap*, whose labels are the basis section, the axis length and the curvature extrema. Each arc between two adjacent nodes falls into one of these cases (see Figure 2.32 and 2.33): tube-body, tube-tube, cap-tube. The tube-body or tube-tube adjacency is called H-junction (i.e., *handle-junction*) if both boundaries of the tube lay on the same body or tube respectively; in this case, the arc is a loop and the tube locates a handle on the input surface. In the case that only one boundary of the tube belongs to the tube-body the adjacency is called a T-junction.

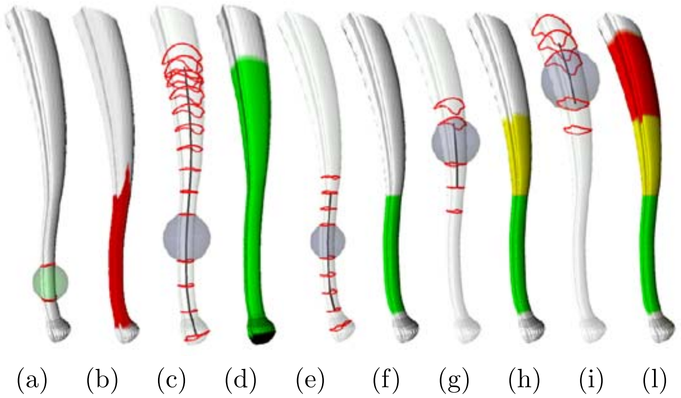


Figure 2.31: (a) Initial level of detail, (b) limb-region, (c) tube growing in non-strict mode, and (d) tube extraction. In (e), tube growing from the same limb-vertices in strict mode, (f) the extracted tube, (g-i) next iterations, (l) achieved segmentation at the chosen scale.

Computational complexity. The predominant cost of the method is represented by the initial surface characterization [MPS$^+$04a] to detect limb vertices, which is $O(n_V^2)$ with n_V number of mesh vertices. The following tube extraction procedure is much faster. The clustering of limb vertices into regions is treated triangle-wise; starting from a first seed triangle having three limb vertices, the region is constructed adding neighboring limb triangles through a breadth-first search. The boundary computation of a region is linear in the number of vertices of the region: all the vertices are visited, and when a seed boundary vertex is found, the boundary loop which it belongs to is computed moving by adjacency.

The medial loop computation is in the worst case very expensive: the problem of computing the minimal distance between two vertices can be solved by the Dijkstra's algorithm in $O(nlog(n))$, where n is the region cardinality, but in our case the minimum distance from all the boundary points takes $O(n^2log(n))$. In practice boundary vertices are much less than n, about $n^{1/2}$, thus reducing time complexity in the average case. The tube growing procedure consists at each step in a triangle visit, starting from those laying on the previous medial loop, until a triangle intersected by the sphere is found. Each triangle inside the sphere is visited once, and the computation of the intersection curve itself is linear in the number of intersected triangles, determined by adjacency. So each tube is grown in linear time with respect to the number of triangles it includes. Timings are reported in Table 2.2.

2.9 Applications and Conclusions

The *Plumber* algorithm provides a multi-scale method to decompose a complex shape into its tubular features and bodies. The segmentation considers as bodies those regions that are not tubular shaped; therefore, a sub-classification of these primitives is necessary. Main difficult tasks are their identification, general configuration and the identification of a basic shape for the

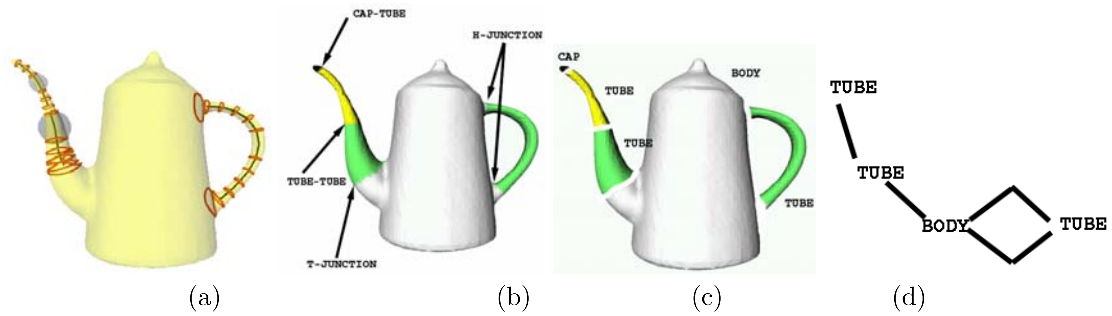


Figure 2.32: (a) Centerlines on a tea-pot with respect to two levels of detail, (b), (c) segmentation of the tea pot into cap, body, tubes and adjacency relations, (d) shape graph.

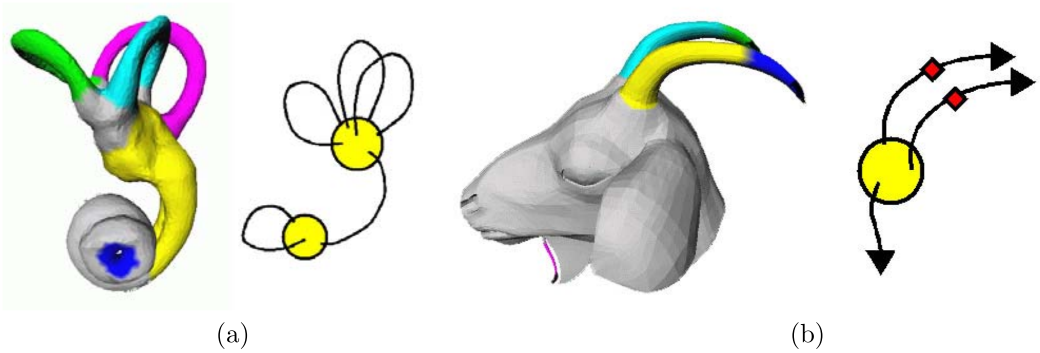


Figure 2.33: Surface decomposition and shape-graph: (a) decomposition of the inner part of the ear into four tubular patches codes as H-junction; (b) the corns of the goat are coded in the shape graph as T-junction.

abstraction.

The interpretation and categorization of tubular features has the drawback of introducing heuristic thresholds to make decisions on the tube size, or for distinguishing branching parts from complicated configurations of tubes as in Figure 2.23 and 2.30. The reduction of the influence of these parameters and the abstraction of tubular features with generalized cylinder and cones for collision detection applications are the further improvements of *Plumber*.

The usefulness of the description obtained goes beyond the problem of shape decomposition or applications to virtual humans (see Figure 2.34); in Chapter 3 and Chapter 4 it serves as a basis and for building effective tools for shape processing and editing. More precisely, in Chapter 3 the multi-scale curvature evaluation is used to find the seed points for the construction of an affine-invariant skeleton which represents the input for a wide class of applications, such as matching and topological characterization of 3D-shapes; Figure 2.22 shows all the steps of the construction. This segmentation is also exploited in Chapter 4 for defining a *feature-based local parameterization* of surface with an arbitrary genus.

Table 2.2: *Plumber* timings (m:s:ms) performed on an Athlon 1000 MHZ.

Model	n_V	Vert. classif.	Medial loop	Tube grow
Cylinder	4038	00.26.84	00.15.45	$< 1sec.$
Pot (1 iter.)	14616	00.08.47	00.03.24	$< 1sec.$
Pot (2 iter.)	–	00.13.45	00.13.50	$< 1sec.$
Schale	10892	00.19.65	02.25.96	$< 1sec.$

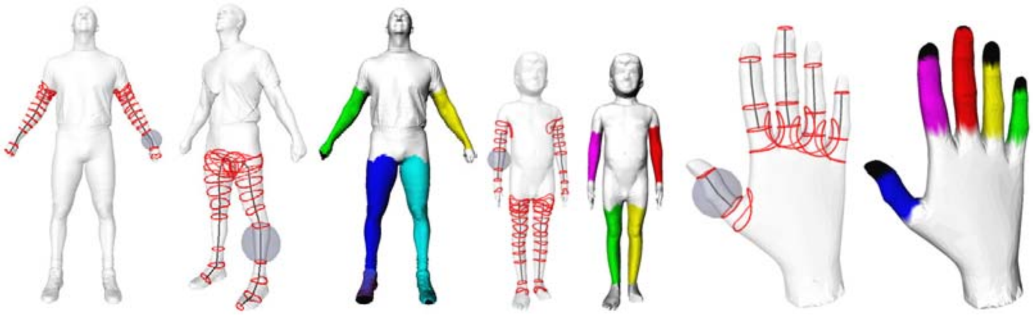


Figure 2.34: Segmentation into tubular features, and related skeletal lines, of human bodies with respect to different levels of detail.

Related publications

1. M. Mortara, G. Patanè, M. Spagnuolo, B. Falcidieno, J. Rossignac. “Blowing bubbles for multi-scale analysis and decomposition of triangle meshes”. In *Algorithmica*, Special Issue on Shape Algorithmics, Volume 38, Num. 2, 2003, pp. 227-248, Springer-Verlag.

2. M. Mortara, G. Patanè, M. Spagnuolo, B. Falcidieno, J. Rossignac. “Plumber: a method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies”. In *ACM Symposium on Solid Modeling and Application 2004*, Genova, Italy, 9-11 June 2004, 339-344.

3. M. Mortara, G. Patanè. “Multiscale Curvature Estimation for Segmenting a Triangle Mesh into Shape Features”. In *Proceedings of Eurographics Italian Chapter*, Università degli Studi di Milano, 25-26 September 2003.

4. M. Mortara, G. Patanè, B. Falcidieno, M. Spagnuolo, J. Rossignac. *Plumber, the tube finder.* Rapporto Tecnico N. 1/2003, Istituto di Matematica Applicata e Tecnologie Informatiche, Consiglio Nazionale delle Ricerche.

5. B. Falcidieno, M. Mortara, G. Patanè, J. Rossignac, M. Spagnuolo. *Tailor: understanding 3D shapes using curvature.* Rapporto Tecnico N. 9/2001, Istituto per la Matematica Applicata, Consiglio Nazionale delle Ricerche.

Chapter 3

Topological analysis of 3D surfaces

This chapter deals with the continuous and discrete definition and analysis of scalar fields $f : \mathcal{M} \rightarrow \mathbb{R}$ on smooth and discrete surfaces. We discuss the definition of topological graph-like representations (i.e., the Reeb graph) which provide a powerful and synthetic sketch of the surface. Then, we present the extraction of a Reeb graph achieved by expanding topological rings on $\mathcal{M}$ starting from high-curvature regions identified by the algorithm described in Chapter 2. Among the wide range of applications of these concepts (e.g., modeling, morphing, matching, and recognition), in Chapter 4 we use this theory for generating an atlas decomposition of an arbitrary surface finalized at the local parameterization problem and the Reeb graph is used for solving the global parameterization of surfaces with an arbitrary genus.

3.1 Introduction

A possible approach to the problem of shape classification and understanding is to represent shape properties of a given surface through shape descriptors which are useful to detect those characteristics which are invariant to position, orientation, and local noise. Among all, descriptors based on geometry and topology are suitable for dealing with the definition of basic models for representing and generating shapes. For these reasons, an increasing attention has been focusing on the formalization of shape as a combination of geometry, topology, and semantic in order to define tools for assessing the similarity among different models. Shape interpretation is especially relevant for the perception of complex forms, where the ability of varying the level of descriptive abstraction is the key for recognizing and classifying highly complex shapes through a multi-resolution framework which abstracts the surface at different levels of detail. The aim of skeleton extraction is to take out and convert shape characteristics and properties of the surface into a compact representation. One of the best known shape descriptor is the Medial Axis introduced in Section 2.1. From a mathematical point of view, classical tools as Morse theory, homotopy, and homology are appropriate to deal with topological questions in Computer Graphics applications.

A high level representation of $\mathcal{M}$, and strictly related to our method, is provided by the Reeb graph [Ree46] which is defined by coding the evolution of the level sets $M_\alpha := \{x \in \mathcal{M} : f(x) = \alpha\}$, $\alpha \in \mathbb{R}$, on $\mathcal{M}$ with respect to a given smooth function $f : \mathcal{M} \to \mathbb{R}$. When the height function is chosen, the graph results in an intuitive representation but it does not distinguish between structural and detail features. Furthermore, the description is not affine-invariant depending on the direction of the chosen height function and providing different graphs according to the position of the surface in space.

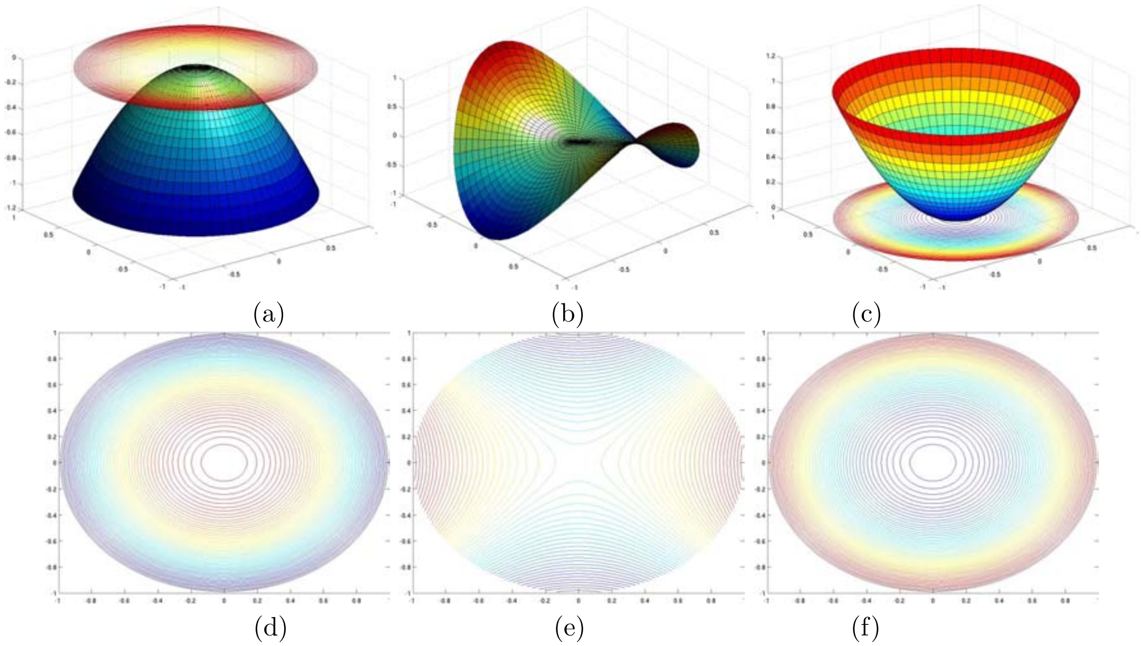
Overview and contributions. The main contribution of this chapter is the extraction of a Reeb graph R_G of a surface triangulation $\mathcal{M}$ which is invariant with respect to affine transformations. Using R_G as an alternative representation of $\mathcal{M}$ is of basic importance for solving several and different problems which spread from recognition to matching [BMM$^+$03] and parameterization [PSF04]. The following discussion provides the background exploited in Chapter 4 for studying the local and global parameterization of surfaces represented by triangle meshes. More precisely, the Reeb graph R_G induced by an arbitrary scalar field f on $\mathcal{M}$ will be used for defining a decomposition of $\mathcal{M}$ into 0-genus charts each one is parameterized with respect to a smooth cut which reduces it to have one boundary component (see Section 4.6). The global parameterization is based on the link between the combinatorial structure of R_G and the topology of $\mathcal{M}$; in fact, the cycles of R_G will be used for identifying the topological handles of $\mathcal{M}$ and the associated iso-contours locate the meridian curves along which the surface is cut for reducing its genus to zero (see Section 4.8).

The chapter is organized as follows: in Section 3.2, we introduce definitions and properties of Morse Theory and the Reeb graph detailing its discrete counterpart for continuous functions defined on arbitrary triangle meshes [Bia04]. Then, in Section 3.3 we present an affine-invariant skeletal representation of triangle meshes starting from high-curvature regions of the input surface [MP02a, MP02b, MPS$^+$04a], extracted using the method detailed in Chapter 2. In Section 3.4, we prove that this skeleton is a Reeb graph with respect to the topological distance from global features defined by curvature extrema. The properties of the proposed approach and a comparison with the Reeb graph induced by the set of functions which is commonly used is provided in Section 3.5.

3.2 Critical points and Morse functions

Let $f : \mathcal{M} \to \mathbb{R}$ be a smooth function defined on a manifold $\mathcal{M}$ (see Section 1.1.3); if $p \in \mathcal{M}$ is a critical point of f (i.e., $\nabla f(p) = 0$), we can consider the second differential of f

$$d^2 f = \left(\frac{\partial^2 f}{\partial x^i \partial x^j} \right)_{i,j}$$



(a) (b) (c)

(d) (e) (f)

Figure 3.1: (a) Maximum, (b) saddle, and (c) minimum. (d-f) Associated iso-contours with respect to the height function.

which defines the *symmetric bilinear form* on $T_p\mathcal{M} \times T_p\mathcal{M}$,

$$d^2f(a,b) = \sum_{i,j} \frac{\partial^2 f}{\partial x^i \partial x^j} a^i b^j$$

where $a := (a^1, \ldots, a^n)$ and $b := (b_1, \ldots, b^n)$ are tangent vectors in $T_p\mathcal{M}$.

Let V be a vector space of finite dimension, $B : V \times V \to \mathbb{R}$ a symmetric bilinear form, and $Q : V \to \mathbb{R}$, $Q(v) := B(v,v)$, $v \in V$, the corresponding quadratic form. The *index* of B is the dimension of the maximal linear space $W \subseteq V$ where B is *negative definite*, that is, $B(v,v) = Q(v) < 0$, for all $v \in W$, $v \neq 0$. The *nullity* of B is the dimension of the nullspace of B defined as Ker(B):=$\{w \in V : B(v,w) = 0, \forall v \in V\}$; B is defined *degenerate* or *singular* if its nullity is greater than zero. Since B and Q are uniquely determined by their system of eigenvalues and eigenvectors, the index (resp., nullity) of Q corresponds to the number of negative (resp., null) eigenvalues.

Definition 9. *A critical point p of f is* non-degenerate *if d^2f is non-degenerate at p, that is, det$(d^2f) \neq 0$ at p. The* index *(resp.,* nullity*) of f at p is the index (resp., nullity) of d^2f at p.*

Definition 10. *A smooth function $f : \mathcal{M} \to \mathbb{R}$ is* Morse *if all its critical points are non-degenerate.*

It is possible to prove that non-degenerate critical points are isolated, that is, it does not exist a sequence of non-degenerate critical points of f which converges to these points; therefore, surfaces with plateaux or volcano rims do not comply with the definition of Morse functions [Mil63].

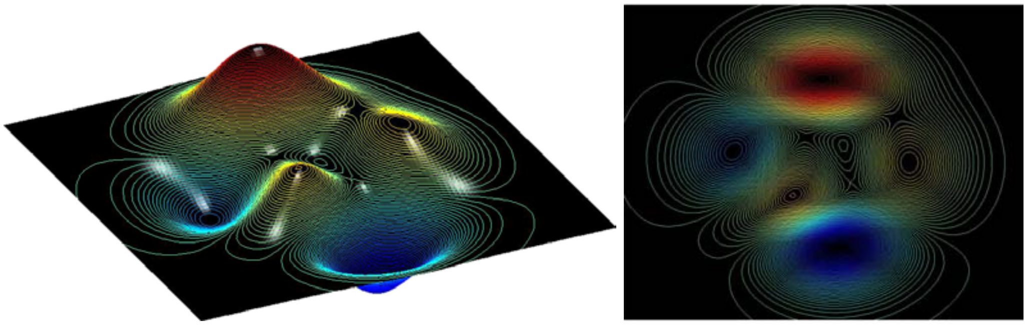


Figure 3.2: Evolution of the iso-contours with respect to the height function.

The characterization of non-degenerate critical points is given by the Morse Lemma which locally characterizes f as a diagonalized quadratic function on appropriate neighborhoods of these vertices.

Lemma 1 (Morse Lemma)**.** *If p is a non-degenerate critical point of a function f on a manifold $\mathcal{M}$, there exists an open neighborhood of p in $\mathcal{M}$ and a set of local coordinates $(x^1, \ldots, x^n)$ such that f has the form*

$$f(x) = f(p) - (x^1) - \ldots - (x^\lambda)^2 + (x^{\lambda+1})^2 + \ldots + (x^n)^2$$

where λ is the index of f at p.

We now consider the case $m = 2$. If $\alpha \in \mathbb{R}$ is a regular value of f (see Definition 8), the *level curve* (or *iso-contour*)

$$C_\alpha := f^{-1}(\alpha) = \{x \in \mathcal{M} : f(x) = \alpha\}$$

is a smooth submanifold of $\mathcal{M}$ (see Theorem 3 at page 13). If we consider

$$\mathcal{M}_\alpha := f^{-1}((-\infty, \alpha]) = \{x \in \mathcal{M} : f(x) \leq \alpha\},$$

then $\mathcal{M}_\alpha$ is a two-manifold with boundary C_α. The analysis of the changes of the level curves C_α when α varies in $\mathbb{R}$ is detailed in the following proposition.

Proposition 5. *Let f be a smooth function on a compact surface $\mathcal{M}$, and assume that the interval $[\alpha, \beta]$, $\alpha < \beta$, does not contain critical values of f. Then,*

- *the level sets C_α and C_β are diffeomorphic;*
- *$\mathcal{M}_\alpha$ and $\mathcal{M}_\beta$ are diffeomorphic as two-manifolds with boundary.*

We analyze the evolution of the level curves of f near a non-degenerate critical point p of f; without loss of generality, we assume that $f(p) = 0$. Using the Morse Lemma 1, there exists a neighborhood U of p and a system of local coordinates (x, y) at p such that:

- the index of f at p is two, that is, $f = -x^2 - y^2$; in this case, p is a *maximum* (see Figure 3.1(a, d)). If ϵ is a small negative regular value of f, the level set $f^{-1}(\epsilon)$ has a connected component near p which is a circle with center p in the Morse coordinates x and y;

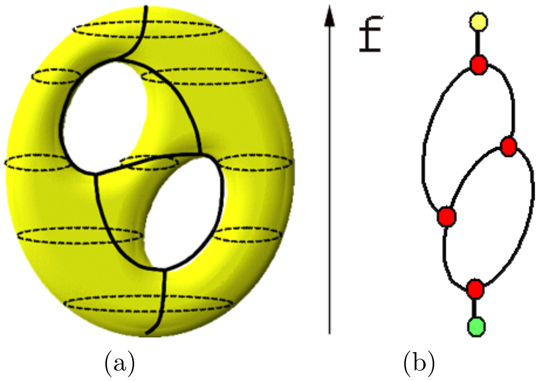


Figure 3.3: Surface of genus two and iso-contours with respect to the height function f, (b) Reeb graph where the red nodes are related to saddle points and the yellow and green node corresponds to a maximum and minimum respectively.

- the index of f at p is one, that is, $f = -x^2 + y^2$ or $f = x^2 - y^2$; in this case, p is a *saddle* (see Figure 3.1(b, e)). Let us take into account the case $f = -x^2 + y^2$; analog considerations apply to $f = x^2 - y^2$. On U, we consider the three level curves $C_{-\epsilon} := f^{-1}(-\epsilon) = \{(x, y) \in U : -x^2 + y^2 = -\epsilon\}$, $C_0 := f^{-1}(0) = \{(x, y) \in U : x = \pm y\}$, and $C_\epsilon := f^{-1}(\epsilon) = \{(x, y) \in U : -x^2 + y^2 = \epsilon\}$. Therefore, $C_{-\epsilon}$ and C_ϵ are the union of hyperbolas, while C_0 is the union of two lines $x = \pm y$;
- the index of f at p is zero, that is, $f = x^2 + y^2$; in this case, p is a *minimum* (see Figure 3.1(c, f)). If ϵ is a small positive regular value of f, the level set $f^{-1}(\epsilon)$ has a connected component near p which is a circle with center p in the Morse coordinates x and y.

Theorem 8 (Morse)**.** *Let $\mathcal{M}$ be a smooth manifold. Morse functions are everywhere dense in the space of all smooth functions on $\mathcal{M}$.*

From the above theorem, it follows that any smooth function on $\mathcal{M}$ can be converted to a Morse function as a result of a perturbation which splits degenerate critical points into non-degenerate singularities. A Morse function $f \rightarrow \mathbb{R}$ is *simple* if its critical points have different values; otherwise, it is called *complicated*. Using Theorem 8, we can convert a complicated Morse function into a simple one by moving the critical points which belong to the same iso-contour to closely spaced contour levels. However, this construction is not always possible if, for example, we want to maintain symmetries on the distribution of the critical points.

Let us now suppose that $\mathcal{M}$ is a compact surface and $f : \mathcal{M} \rightarrow \mathbb{R}$ is a Morse function. If f is simple, the Morse Lemma implies that for every regular value $\alpha \in \mathbb{R}$ the iso-contour $f^{-1}(\alpha)$ has the local structure of an isolated point at a maximum or minimum, a transversal crossing of two curves at a saddle, and a smooth curve at other points. As α varies on $\mathbb{R}$ and reaches a critical value, the corresponding level sets merge, split, appear, or disappear and their evolution (see Figure 3.2) can be converted into a graph representation as detailed by the following definition.

Definition 11. *Let $\mathcal{M}$ be a 2-manifold and $f : \mathcal{M} \to \mathbb{R}$ be a smooth mapping function. The* Reeb graph R_G *of $\mathcal{M}$ with respect to f is the quotient space of $\mathcal{M} \times \mathbb{R}$ defined by the relation " $\sim$ " with*

$$(p, f(p)) \sim (q, f(q)) \leftrightarrow f(p) = f(q)$$

and p, q belong to the same connected component of $f^{-1}(f(p))$.

Therefore, the Reeb graph is a linear skeleton which consists of two types of elements (see Figure 3.3):

1. *arcs* coming from the connected components related to the part of the surface $\mathcal{M}$ which is located between two critical values. All arcs are *oriented* according to increasing of f;
2. *nodes* representing critical points. To each node we associate a *label* depending on its classification as maximum, minimum, saddle, or regular point.

For a closed surface $\mathcal{M}$, the relation between the critical points of $(\mathcal{M}, f)$ and $\chi(\mathcal{M})$ (or equivalently, its genus g) is given by the condition

$$\chi(\mathcal{M}) = \text{minima} - \text{saddles} + \text{maxima}, \tag{3.1}$$

and the following result provides the link between the topology of $\mathcal{M}$ and the combinatorial structure of its Reeb graph [CMEH+03].

Proposition 6. *Let $\mathcal{M}$ be a connected, orientable 2-manifold of genus g, $f : \mathcal{M} \to \mathbb{R}$ a Morse function, and R_G the Reeb graph of $\mathcal{M}$ with respect to f. Then,*

- *if $\mathcal{M}$ is closed, R_G has g loops;*
- *if $\mathcal{M}$ has $b \geq 1$ boundary components, R_G has k loops with $g \leq k \leq 2g + b - 1$.*

On the set $\mathcal{Q}$ of pairs $(\mathcal{M}, f)$ such that $\mathcal{M}$ is an orientable, compact manifold and $f : \mathcal{M} \to \mathbb{R}$ is a Morse function, we consider the equivalence relation $\mathcal{R}$ defined as: $(\mathcal{M}, f)\mathcal{R}(\mathcal{N}, g)$ if and only if there exists a diffeomorphism $\varphi : \mathcal{M} \to \mathcal{N}$ which maps iso-contours of f on $\mathcal{M}$ to iso-contours of g on $\mathcal{N}$. In this case, $(\mathcal{M}, f)$ and $(\mathcal{N}, g)$ are called *fiber-equivalent*.

Definition 12. *Given $(\mathcal{M}, f)$ and $(\mathcal{N}, g)$ in $\mathcal{Q}$, the associate Reeb graphs $R_{\mathcal{M}} := \mathcal{M}/ \sim_f$ and $R_{\mathcal{N}} := \mathcal{N}/ \sim_g$ are* isomorphic *if there exists a homeomorphism between them that:*

- *preserves the node classification;*
- *either preserves the orientation of the edges or changes it on all the edges simultaneously.*

As stated by the following proposition, the Reeb graph $R_{\mathcal{M}}$ of $(\mathcal{M}, f)$ uniquely defines the equivalence class of $(\mathcal{M}, f)$.

Proposition 7. *Let $(\mathcal{M}, f)$, $(\mathcal{N}, g)$ be in $\mathcal{Q}$ with f and g simple Morse functions. Then, $(\mathcal{M}, f)\mathcal{R}(\mathcal{N}, g)$ if and only if $R_{\mathcal{M}}$ and $R_{\mathcal{N}}$ are isomorphic. In particular, $\mathcal{M}$ and $\mathcal{N}$ are diffeomorphic.*

3.2.1 Discrete Morse Theory and Reeb graph

Above this point, it has been shown that if f is at least $C^2(\mathcal{M})$ Morse theory guarantees that the topological changes of the iso-contours occur only at the critical values of f, and the Reeb quotient space is represented by a graph whose nodes correspond to critical values and whose arcs code the evolution of contours. In this section, we consider the discretization of Morse theory when f is sampled on the vertices of a triangle mesh $\mathcal{M}$ and extended to the faces of $\mathcal{M}$ by linear interpolation (see Section 4.5 at page 96, for more details). Common approaches to the characterization of discrete surfaces use local point-wise criteria to detect and classify critical points, by simulating the concept of critical point using the connectivity of the mesh as underlying topological space [Ban67, EHNP03]. Two drawbacks can be identified: first, these methods rely on the hypothesis that f assumes a different value on all edge-adjacent vertices; second, the number of the detected critical points is usually very high and pruning or simplification steps are necessary to make the resulting structures meaningful.

Following the approach proposed in [Bia04], another solution is to observe that the contour levels decompose $\mathcal{M}$ into a set of regions, whose boundaries contain a complete information for detecting critical areas and for classifying them as maximum, minimum, and saddle areas. For example, if a contour does not contain any other contour and the value of the function on it is higher than the successive one, then the contour identifies a local maximum area for the chosen function f. Let $f : \mathcal{M} \to \mathbb{R}$ be a real mapping function defined on a triangle mesh $\mathcal{M}$, $[f_{min}, f_{max}]$ an interval containing $f(\mathcal{M})$, and $f_{min} =: f_0 < f_1 < \cdots < f_n < f_{max} := f_{n+1}$ the distribution of the values of the contour levels $C_f(\mathcal{M})$, which are supposed to be all non degenerate contours [1]. In addition, let $I = \{(f_i, f_{i+1}), i = 0, \dots, n\} \cup \{f_{min} = f_0, f_1, \dots, f_n, f_{max} = f_{n+1}\}$ be the partition of the interval $[f_{min}, f_{max}]$ provided by the set of the $n+1$ open interior parts of $[f_{min}, f_1, \dots, f_n, f_{max}]$ and the function values of the contour levels. Starting from Definition 11, we define its counterpart in the discrete setting as defined below.

Definition 13. *The* Extended Reeb equivalence *between two points $p, q \in \mathcal{M}$ is given by the following conditions:*

1. *$f(p)$ and $f(q)$ belong to the same element of $t \in I$;*
2. *p, q belong to the same connected component of $f^{-1}(f(t))$, $t \in I$.*

Therefore, by applying the notion of the quotient relation, it follows that all points belonging to a region R are Reeb-equivalent in the extended sense and they may therefore collapse into the same point of the quotient space. The quotient space obtained from such a relation is a discrete space, which is called *Extended Reeb* quotient space (ER). The characterization criterion depends on the frequency of the slicing process; however, holes contained within two consecutive slices are detected by using the Euler formula which enables to locally update the structure of the Extended Reeb graph while keeping its topological correctness.

[1] A contour is *degenerate* if it is a connected component of an iso-contour of f related to a critical value; in this case, it is always possible to slightly change that value and get a non degenerate contour.

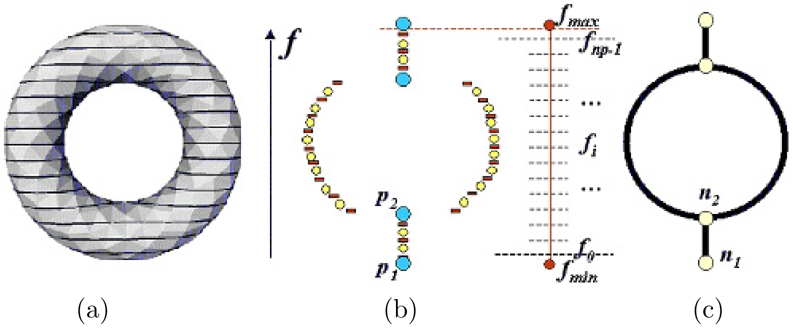


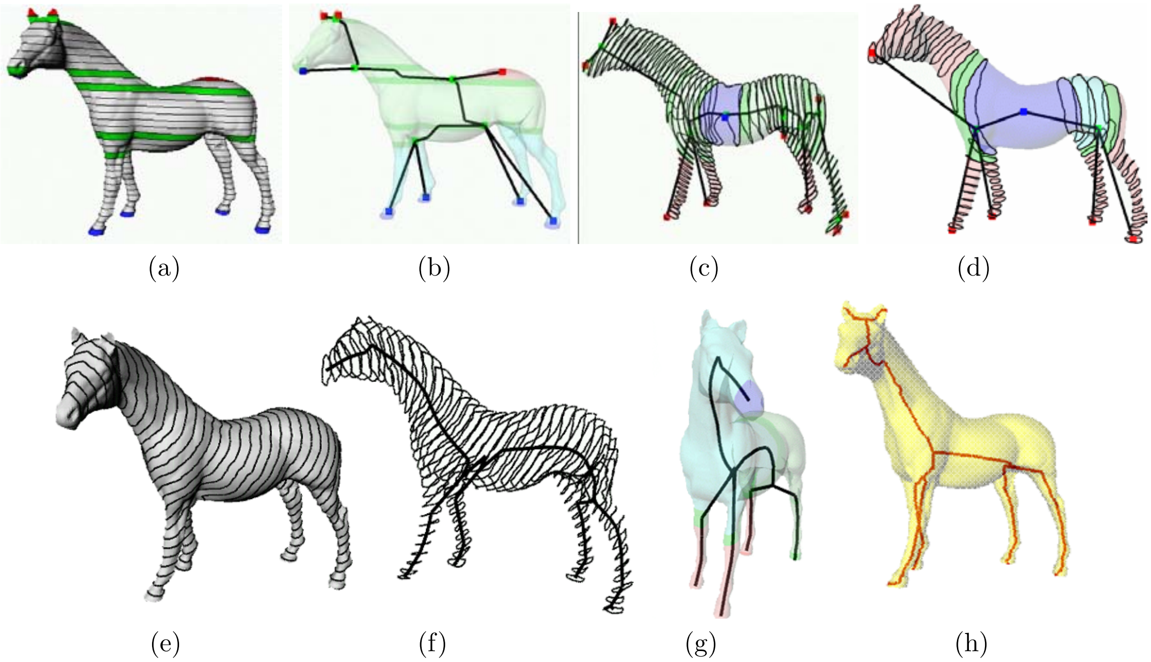
Figure 3.4: (a-b) The quotient space induced by f, and (c) its graph-like representation. Connecting points are depicted by using red rectangles while normal points are shown as circles; circles represent the quotient of a region, while rectangles are images of contour levels.

The construction of the Extended Reeb graph [Bia04, ABS03] is based on a slicing strategy that traces the iso-contours of f on $\mathcal{M}$ while inserting them in the mesh as constraints; in this way, studying the properties of the resulting mesh strips delimited by the iso-contours corresponding to consecutive iso-values guarantees that the holes of $\mathcal{M}$ are correctly represented in the associated Reeb graph. The idea is to locate *critical areas* within which at least one critical point occurs and identify, starting from them, the smallest area on $\mathcal{M}$ whose behavior is topologically equivalent to the critical point (i.e., the *influence zone* of the critical point). The number of connected components in the boundary of these regions and the corresponding iso-values of f allow to distinguish between *regular* and *critical* regions, the latter containing critical points of the function f. More precisely, if a region has one or more than two boundary components, then it is classified as critical; otherwise, if the two boundary components are related to two different iso-levels, the region is regular.

Critical regions with only one boundary component contain a critical point which is either a minimum or a maximum, and they are distinguished by checking the values of f outside the region. A region with two boundary components with the same iso-value identifies either a *degenerate* critical area of f, or a handle-like maximum or minimum critical point. The distinction between the two cases is done by checking the behavior of the surface across the region boundary:

- if all the connected components have ascending directions, then the area corresponds to a minimum;
- if all the connected components have descending directions, then the area corresponds to a maximum;
- otherwise, it is a saddle.

For maxima and minima, it is possible to distinguish between degenerate and handle-like situations by considering the inclusion relations of the corresponding iso-contours. In particular, two configurations can happen: if one boundary component contains all the others, then the region


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

(e) (f) (g) (h)

Figure 3.5: (a-b) Iso-contours and the Reeb graph of the input surface with respect to the height function, (c) the distance from the barycenter, (d) the geodesic distance, (e-g) the geodesic distance from one point, and (h) the proposed method. Red regions are related to maxima, blue areas locate minima, and green regions identify saddles zones.

represents a degenerate area otherwise it is a handle-like critical area. The influence zones of saddle points may be also regarded as the area on $\mathcal{M}$ where the topological change of the surface occurs, when the surface is analyzed at the chosen frequency. Note that all critical areas may contain degenerate critical points, i.e. non-isolated ones, and that influence zones may contain more than one isolated critical point corresponding at the same iso-level, that is, they may locate non-simple critical points.

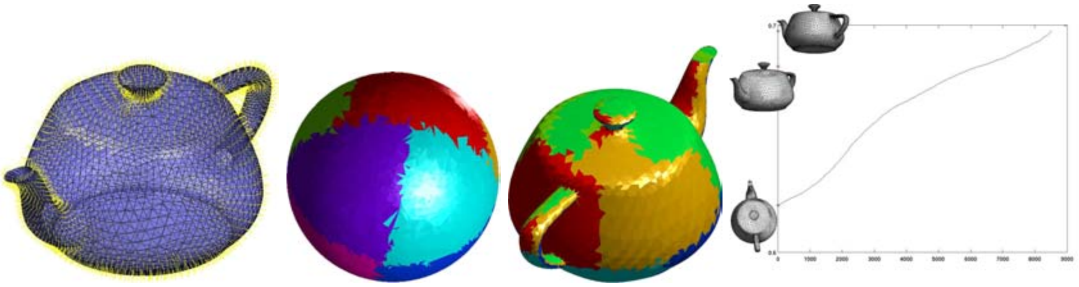
During the ER graph construction, the number of boundary components and the genus of the current regions are updated; after this step, if a region contains a hole (i.e., its genus is equal or greater than one), then the slicing is insufficiently sampled and a more dense distribution is used. The coding of the critical areas in the ER graph still verifies the Morse relation (3.1); since the genus of each region is zero we are guaranteed that the Euler characteristic of each slice is two and therefore the contribution of each critical area is given by $(2 - \tilde{b})$, where $\tilde{b}$ is the number of its boundary components.

To represent the ER quotient space as a graph, the classes which are defined by points on contours are represented by *connecting points*, while all other classes are represented by normal points, simply called points (see Figure 3.4(a-b)). Connecting points are representative of contours and normal points are representative of regions. A point p representing a region R is adjacent through a connecting point to another point q representing another region R' in the quotient space, and a normal point is adjacent to as many connecting points as the number of connected components of the border of the associated region. From this point of view, the image of a regular region of $\mathcal{M}$ in the ER quotient space is adjacent only to two connecting points. Therefore, the connectivity changes of the graph representation are concentrated in the image of the critical areas, and they are equivalent to the standard Reeb graph representation which can be easily derived by merging the intermediate nodes representing regular areas into a single arc; in Figure 3.4(b), normal points that correspond to critical areas are depicted in orange. After this merging step, the ER graph simply consists of nodes representing critical areas and the associated connecting arcs. In Figure 3.4, an example of the Extended Reeb quotient space is shown for a torus with respect to the height function f; in particular, the sequence of points between p_1 and p_2 of the quotient space ER in (b) represents an arc between the nodes n_1 and n_2 in (c).

3.2.2 Choosing the function f

The choice of each function f highlights one shape aspect discarding others; for preferring a particular function f with respect to another one, a constraint on its choice is to be a good compromise between calculation, invariance with respect to an appropriate set of transformations (e.g., affine transformations), and description efficacy. Furthermore, we can distinguish two classes of mapping functions: those ones which are defined in the continuous case and then applied to the discrete context (e.g., the height function, the Euclidean distance from the surface barycenter), and maps directly defined on the input triangle mesh (e.g., discrete harmonic scalar fields). In this last case, an approximation of the function values on the edges and faces is achieved by linearly interpolating the values of f on the vertices, and then used for the contour level calculation [Bia04, PCM02]. While the properties of the segmentation are completely independent of the chosen mapping function f, different f functions generate different decompositions and this allows to adapt the decomposition to the characteristics of the specific shape. We briefly review several choices [BMMP03] for the construction of the ER graph to give an idea of the kind of graph they produce. The use of the decomposition for local parameterization is discussed in Section 4.6.

Reeb graph with respect to the height function. If the map f is the height function ($\forall p = (x, y, z) \in \mathbb{R}^3$, $f(x, y, z) = z$), its critical points are peaks, pits, and passes and the equivalence classes correspond to the intersection of the triangle mesh with a set of planes orthogonal to the direction $e_3 := [0, 0, 1]$. The height function with respect to e_3 is best suited for digital terrain models where associating to each point its elevation value is a natural choice. Since the height function with respect to the vector e_3 can be non-optimal with respect to the shape of an arbitrary input surface (see Figure 3.5(a-b)), we describe an automatic method for selecting the direction of the height function and based on a clustering of the normal vectors of $\mathcal{M}$ [Pat04]. To this end, a face f_i of the input triangle mesh $\mathcal{M}$ is called *degenerate* with respect to the view direction v if its projection along v is a line segment, that is, the plane containing f_i is parallel to v. Therefore, we define the best view direction w for $\mathcal{M}$ as the solution of the following problem (see Figure 3.6)



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

Figure 3.6: (a) Normal vectors of the faces of the teapot, (b) clustered regions on the unit sphere, and (c) induced segmentation (the faces belonging to the same cluster are represented with the same color). (d) Optimal (on the bottom) and worst view (on the top) of the input surface given by the eigenvectors of the matrix A.

$$\min_{v\in S^2}\left\{\hat{d}(v) := \frac{1}{n}\sum_{i=1}^{n_F}\hat{d}(f_i,v)\right\}$$

where $\hat{d}(f_i,v) := 1 - | < n_i, v > |^2 \in [0,1]$ is the "degeneracy" of the face f_i with respect to the view direction v, n_i is the unit normal of the face f_i, n_F is the number of faces, and $S^2 := \{v \in \mathbb{R}^3 : \|v\|_2 = 1\}$ is the unit sphere. Therefore, we are searching the direction w which minimizes the degeneracy of the faces, that is, the one which minimizes the alignment of the normal vectors of $\mathcal{M}$ with respect to w. Since the function $\hat{d}$ is continuous and S^2 is compact, a global minimum and maximum exist. For finding them, we use the Lagrange multipliers and we search the critical points of the function $F_\lambda(v) := \hat{d}(v) + \lambda(\|v\|_2 - 1)$, $v := (x,y,z) \in S^2$, which are achieved by solving the following system

$$\begin{cases} \partial_x F_\lambda = 0 & \text{(a)} \\ \partial_y F_\lambda = 0 & \text{(b)} \\ \partial_z F_\lambda = 0 & \text{(c)} \\ \partial_\lambda F_\lambda = 0. & \end{cases}$$

If $n_i := (n_{i,x}, n_{i,y}, n_{i,z})$, we have that

$$\partial_x F_\lambda(v) = 2\left[-\frac{1}{n_F}\sum_{i=1}^{n_F} n_{i,x} < n_i, v > + \lambda x\right]$$

and therefore omitting the constant factor 2

$$\partial_x F_\lambda(v) = -\frac{1}{n_F}\left[\sum_{i=1}^{n_F} n_{i,x}^2 x + \sum_{i=1}^{n_F} n_{i,x}n_{i,y}y + \sum_{i=1}^{n_F} n_{i,x}n_{i,z}z\right] + \lambda x.$$

Since analogous relations hold for $\partial_y F_\lambda$ and $\partial_z F_\lambda$, the conditions (a+b+c) are equivalent to the linear system

$$Av = \lambda v$$

where A is the symmetric 3×3 matrix

$$A := \frac{1}{n_F} \begin{pmatrix} \sum_{i=1}^{n_F} n_{i,x}^2 & \sum_{i=1}^{n_F} n_{i,x} n_{i,y} & \sum_{i=1}^{n_F} n_{i,x} n_{i,z} \\ -- & \sum_{i=1}^{n_F} n_{i,y}^2 & \sum_{i=1}^{n_F} n_{i,y} n_{i,z} \\ -- & -- & \sum_{i=1}^{n_F} n_{i,z}^2 \end{pmatrix}.$$

From the previous relation, it follows that we need to analyze the eigenvalues $0 \leq \lambda_1 \leq \lambda_2 \leq \lambda_3$ and eigenvectors v_1, v_2, v_3 of A with $Av_i = \lambda_i v_i$; if $\lambda_i = 0$, the corresponding solution $v_i = 0$ is not meaningful because we require that $\|v_i\|_2 = 1$. Therefore, the unitary eigenvectors corresponding to the non-zero eigenvalues are solutions of our problem and among them we select the one that minimizes the function $\hat{d}$. We note that the above procedure is equivalent to consider the *principal component analysis* [Jol86] of the set of unit normals of $\mathcal{M}$. More precisely, the PCA on the set $\{n_i : i = 1, \ldots, n_F\}$ considers the spectrum of the matrix $G^T G$,

$$G := \begin{pmatrix} n_1 \\ \ldots \\ n_{n_F} \end{pmatrix};$$

therefore, we have that $A = \frac{1}{n_F} G^T G$. Analogous considerations apply to the weighted problem with

$$\hat{d}(v) := \frac{1}{\sum_{i=1}^{n_F} w_i} \sum_{i=1}^{n_F} w_i \hat{d}(f_i, v)$$

and $w_i \geq 0$, $i = 1, \ldots, n_F$.

Reeb graph with respect to the distance from the barycenter. The distance function of the surface vertices from a given point p of the Euclidean space, $f_p(x) := \|p - x\|_2$, $x \in \mathcal{M}$, represents a class of Morse functions. The point p could belong to the mesh or not, even though a reasonable choice is the barycenter of the surface which is easily calculated and, due to its linear dependence on all the vertices, stable to small perturbations. In this case, the iso-contours of f correspond to the intersection of $\mathcal{M}$ with a set of spheres centered in the barycenter p of $\mathcal{M}$ and with different radii; therefore, it is possible to recognize protrusions and hollows with respect to p by analyzing the maxima and minima of the function f (see Figure 3.5(c)).

Reeb graph with respect to the geodesic distance. In [HSKK01], a map based on the geodesic distance from a source point $p \in \mathcal{M}$ has been defined. The determination of p is not simple and the solution proposed by the authors is to define, for every point $x \in \mathcal{M}$, $h(x)$ as the sum of all geodesic distances $g(x, p)$ from x to p when p varies on the input manifold (see Figure 3.5(d)), that is,

$$h(x) := \int_{\mathcal{M}} g(x, p) dp.$$

This function is not invariant to scaling of the surface and it is replaced by its normal representation defined as

$$\forall x \in \mathcal{M}, \qquad h_n(x) := \frac{h(x) - \min_{x \in \mathcal{M}} \{h(x)\}}{\max_{x \in \mathcal{M}} \{h(x)\}}.$$

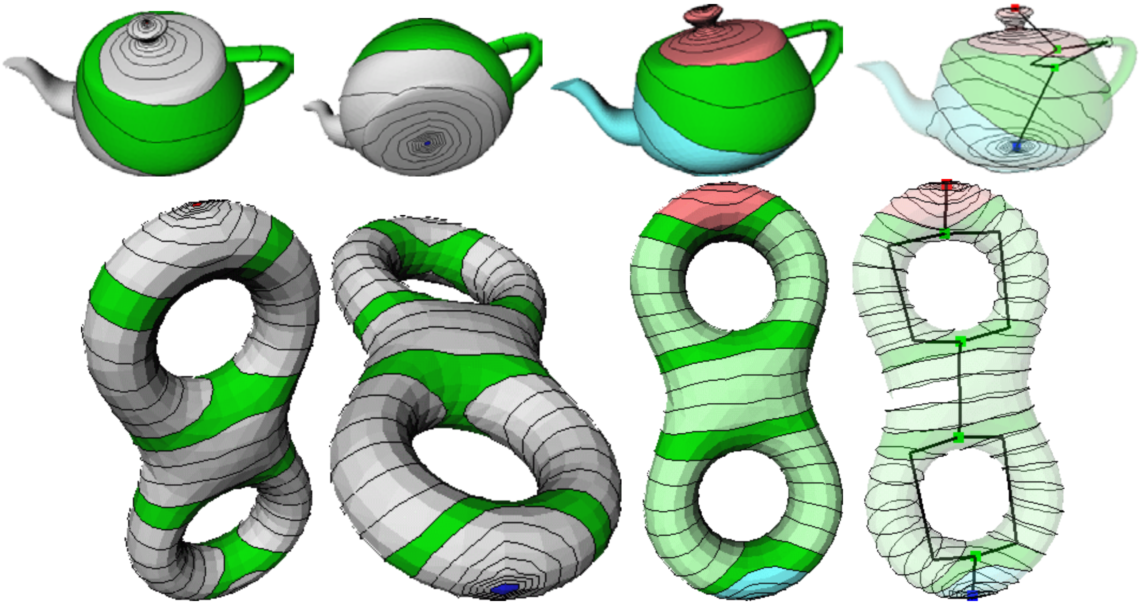


Figure 3.7: Iso-contours and Reeb graph associated to the harmonic scalar field $f : \mathcal{M} \to \mathbb{R}$ achieved by using as points for the Dirichlet boundary conditions the marked vertices; the blue node locates the minimum and the red one the maximum of f.

Reeb graph with respect to the geodesic distance from one point. The approach to shape description described in [VL99, WDSB00] deals with the construction of centerlines from unorganized point sets and later developed for polyhedral surfaces in [LV99]. This graph is "essentially" a tree whose edges connect the barycenters of the level sets of the function given by the shortest approximated geodesic distance from a source point. The resulting skeletal curve depends on the appropriate choice of the order of the neighborhood graph, on the location of the source point, and on the number k of level sets which slice the surface. To choose the seed point an heuristic is used which seems to work well on elongated tubular shapes such as blood vessels and bones; furthermore, examples are given to show that on these kinds of input data the centerlines obtained with different source points are nearly the same. However, the choice of only one seed point determines a privileged slicing direction; this element and a wrong choice of k can lead to the loss of some features if the surface is not strictly cylindrical (such as the ear of the horse in Figure 3.5(e-g)).

Harmonic fields on arbitrary triangle meshes. In this paragraph, we define a method for constructing *harmonic scalar fields* over arbitrary surface triangulations [NGH04]; the specialization of this approach to the planar parameterization of disc-like surfaces (i.e., 0-genus and one boundary component) is discussed in Section 4.5.2. Let $\mathcal{M}$ be an arbitrary surface triangulation, i.e. arbitrary genus and boundary components, and $X := \{x_i : i = 1, \ldots, n_V\}$ be the set of vertices of $\mathcal{M}$. We want to define a scalar function $f : \mathcal{M} \to \mathbb{R}$ satisfying the Laplace equation

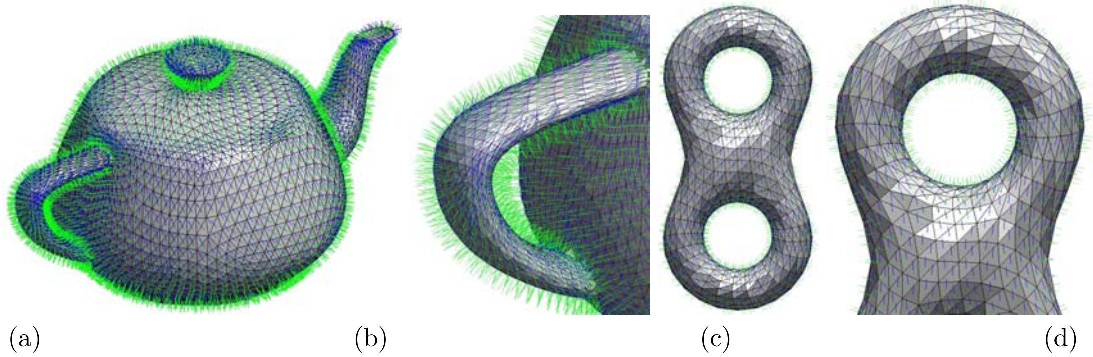


Figure 3.8: Orthogonal and tangent vector field associated to the harmonic field shown in Figure 3.7 on (a-b) the teapot, (c-d) the bitorus.

with Dirichlet boundary conditions, that is,

$$\begin{cases} \Delta f = 0 & \text{in } \mathcal{M}, \\ f = f_0 & \text{in } \mathcal{C} \subseteq \mathcal{M}. \end{cases}$$

Using barycentric coordinates (see also Section 4.5 and equation (4.8)) and the fact that $\mathcal{M}$ is a triangle mesh, it is enough to define the values of f on the set X and then to extend them from X to $\mathcal{M}$ through linear interpolation over each triangle of $\mathcal{M}$; that is, if $x \in \Delta := [x_{i_1}, x_{i_2}, x_{i_3}] \in \mathcal{M}, x = \lambda_1 x_{i_1} + \lambda_2 x_{i_2} + \lambda_3 x_{i_3}$, with barycentric coordinates $\lambda_1, \lambda_2, \lambda_3$, then

$$f(x) := \lambda_1 f(x_{i_1}) + \lambda_2 f(x_{i_2}) + \lambda_3 f(x_{i_3}).$$

Let $C \subseteq V$ be a set of indices and $\{\alpha_i\}_{i \in C} \subseteq \mathbb{R}$ arbitrary real values; then, the values $\{f(x_i)\}_{i \in V \setminus C}$ are defined as the solution of the linear system of equations

$$\begin{cases} \sum_{j \in N(i)} w_{ij}(f(x_j) - f(x_i)) = f(x_i) & i \in V, \\ f(x_i) = \alpha_i, & i \in C. \end{cases} \tag{3.2}$$

with coefficients

$$w_{i,j} := \cot(\alpha_{ij}) + \cot(\beta_{ij}),$$

where α_{ij} and β_{ij} are the angles of the 1-star of x_i opposite to the edge (i, j) (see also Section 4.5.2 and equation (4.20) at page 103). The main properties of the scalar field f over $\mathcal{M}$ are:

1. f has no local extrema other than at constrained vertices;
2. if all the constrained minima $\{\alpha_i : i \in C_{\min} \subseteq C\}$ are assigned the same global minimum value $\alpha_{\min}$ and all the constrained maxima $\{\alpha_i : i \in C_{\max} \subseteq C\}$ are assigned to the same global maximum $\alpha_{\max}$ then all the constraints are extrema of the resulting scalar field f.

In Figure 3.7, the iso-contours of harmonic scalar fields over different triangulated manifolds are shown.

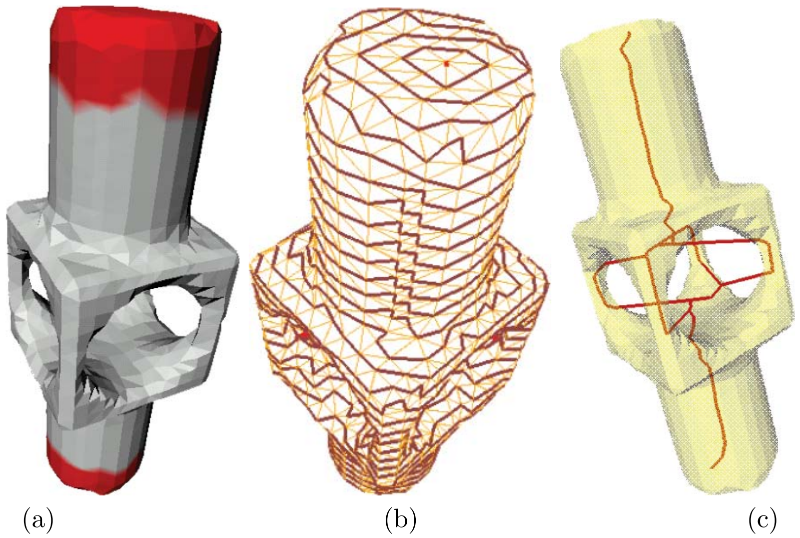


Figure 3.9: Overview of the main steps of the proposed approach: (a) high-curvature regions, (b) topological ring expansion, (c) skeleton.

3.2.3 Vector fields associated to scalar fields

Let us assume that a scalar field f on $\mathcal{M}$ has been assigned; we want to define the *orthogonal* $g_1 := \nabla f$ and *tangent* $g_2 := \mathcal{R}f$ *vector field to* f, where $\mathcal{R}$ is the counter-clockwise rotation by $\pi/2$ in the local tangent plane. To this end, we consider the triangle $\Delta := [x_i, x_j, x_k] \in \mathcal{M}$ with unit normal vector

$$n := \frac{\overline{N}}{\|\overline{N}\|_2}, \quad \overline{N} := (x_j - x_i) \wedge (x_k - x_i).$$

Then, the gradient vector g_1 at the vertices of the triangle Δ is the solution of the 3×3 linear system

$$\left[\begin{array}{l} x_j - x_i \\ x_k - x_j \\ n \end{array}\right] g_1 = \left[\begin{array}{l} f(x_j) - f(x_i) \\ f(x_k) - f(x_j) \\ 0 \end{array}\right].$$

An example is given in Figure 3.8.

3.3 Reeb graph from curvature extrema

In the following sections, we describe a new approach to the construction of a Reeb graph of $\mathcal{M}$ which takes into account high-curvature regions and topological rings on $\mathcal{M}$. A surface can be seen as made essentially of a main body with several protrusions: for instance, the overall shape of a man consists of the torso from which head, arms, and legs depart. From this point of view, the main features of a 3D surface are its protrusions and these are detected by curvature

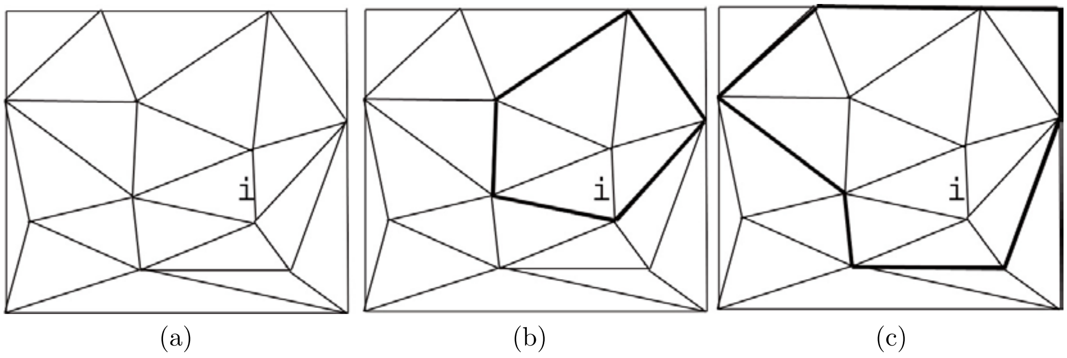


Figure 3.10: Local neighborhood system on a triangular mesh: (a) input triangle mesh, topological rings of order 1 in (b), and 2 in (c).

extrema. Following these considerations, we are going to construct the skeleton of a 3D surface as a graph which, starting from high curvature regions, grows on the surface towards the main body according to the mutual adjacency among features. The proposed construction of the skeleton of a 3D surface represented by a triangle mesh is made in three steps.

1. A curvature estimation is performed on the mesh and several zones of high curvature, which identify the surface features, are extracted (see Figure 3.9(a)). The procedure of curvature estimation is discussed in Chapter 2 and it detects regions which identify "global" curvature features and it creates a hierarchy of such elements with respect to their scale. For each high curvature region Γ_i, a representative vertex p_i is selected. For a more pleasant visualization p_i is chosen as the centroid of Γ_i and computed as the farthest vertex from the region boundary; anyway, we underline that, from the topological point of view, the arbitrary choice of p_i is irrelevant. Each vertex in the set $\{p_i\}$ represents the terminal node of the graph.

2. Starting simultaneously from the centroids of the high curvature regions, topological rings consisting of vertices which have the same topological distance (i.e., minimum number of edges) from the nearest centroid are computed, growing of one edge at a time, until the surface is covered (see Figure 3.9(b)).

3. The centroid of each high curvature region becomes the terminal node of the graph, while points of split or collision among topological rings during the expansion phase individuate its branching nodes. The centroids of consecutive rings are connected to form an edge of the skeleton (see Figure 3.9(c)).

Mesh connectivity and topology. As detailed in Section 1.1.1, we represent a triangle mesh $\mathcal{M}$ as the pair $\mathcal{M} := (M, T)$ where $M := \{x_i \in \mathbb{R}^3,\ i = 1, \ldots, n_V\}$ is a list of n_V vertices and T is an *abstract simplicial complex* [ES92] which contains the adjacency information whose subsets come in three types: vertices $\{i\}$, edges $\{i, j\}$, and faces $\{i, j, k\}$. Using T, we can construct the

1-neighborhood structure as

$$\{N(i) : i = 1, \ldots, n_V\}$$

with

$$N(i) := \{j \in \{1, \ldots, n_V\} : (i, j) \in T\}.$$

The previous relation assigns each vertex i the set of its 1-neighborhoods, that is, the vertices j such that (i, j) is an edge of the triangulation T. The size of the neighborhood structure can be recursively enlarged by defining a *n-neighborhood* as

$$\begin{array}{l} N(i_i, \ldots, i_k) := \bigcup_{l=1}^{k} N(i_l), \\ N^0(i) := \{i\}, \\ N^1(i) := N(i), \\ N^n(i) := N^1(N^{n-1}(i)),\ n \geq 2. \end{array}$$

Therefore, given a vertex $i \in \{1, \ldots, n_V\}$ we can define its *local neighborhood system* as

$$B_i := \{T(N^k(i)) : k = 0, \ldots\}$$

with

$$T(N^k(i)) := \bigcup_{l,p,q \in N^k(i), \{l,p,q\} \in F} T(l, p, q)$$

and $T(l, p, q)$ triangle with vertices x_l, x_p, x_q. Finally, we refer to the border of $T(N^k(i))$, i.e. $\partial T(N^k(i))$, as *topological ring of order k for i* (see Figure 3.10). From the previous definitions the following conditions hold: $\forall i \in \{1, \ldots, n_V\}$

- $x_i \in T(N^k(i)), \qquad k = 0, \ldots$
- $T(N^k(i)) \subseteq T(N^{k+1}(i)), \qquad k = 0, \ldots$
- if $\mathcal{M}$ is connected, then
$$\mathcal{M} = \bigcup_{k=1,\ldots} T(N^k(i)).$$

Topological expansion. Above this point, we have segmented the mesh into significant/non significant regions, according to the curvature value at each vertex. Since high-curvature regions identify the extremities of the surface protrusions, the skeleton of a surface characterized in this way has their representative points as terminal nodes. Therefore, for each region a representative vertex is chosen as the farthest one, with respect to the topological distance, to the boundary. Starting from the representative vertices we can construct a skeleton of the surface using a topological expansion approach, based on the idea of topological rings growing from representative vertices and traversing the mesh, until rings split, collide with others, or can be expanded no more (see Figure 3.11). Starting at the same time from all the representative vertices, the topological rings grow one step at a time until:

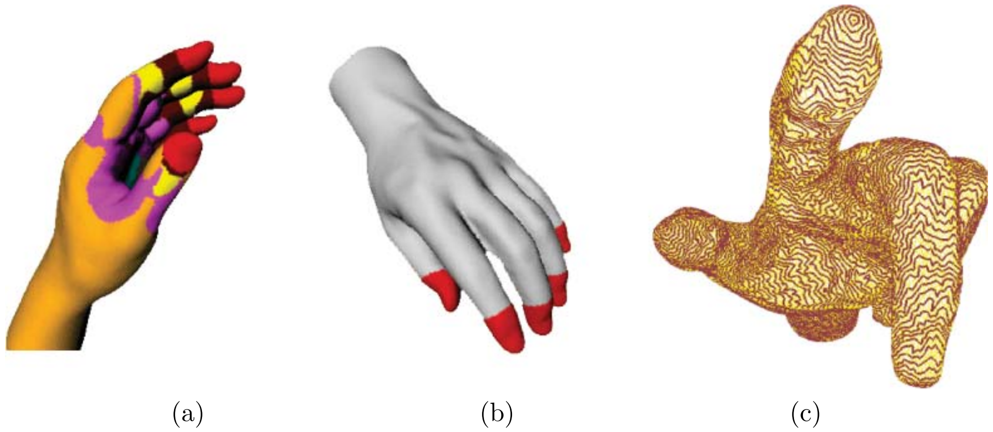


Figure 3.11: (a) Vertex classification on the hand, (b) high curvature regions, (c) topological rings achieved by using the red areas depicted in (b) as seed regions.

- rings belonging to different representative vertices collide: a union occurs;
- a ring intersects itself: a split occurs;
- a ring can be expanded no more: the ring terminates.

Union. When expanding a topological ring we meet a vertex already belonging to the topological ring (of the same order) of another representative vertex, the two rings intersect. From the point of view of the surface, this means that we have found a branching zone of the surface where two distinct protrusions depart. In the same way, the corresponding skeleton will have a branch with two edges joining. The rings of each representative vertex, which determined the collision, can be expanded no more and the construction of their ring sets is completed. A new topological ring is created for the intersection vertex; in this case, unlike what happens for representative vertices, the first topological ring coded is the union of the two rings colliding (see Figure 3.12(a)).

Split. It can also happen that a ring intersects itself in one or more of its vertices; this is the situation of rings expanding near handles, through holes of the surface or where a new protrusion starts. In this case, the intersecting ring splits in two parts and its expansion stops in favor of two new rings derived from the split. Like the union case, the adjacency between the father vertex and the two children is set (see Figure 3.12(b)).

Termination. After a finite number of steps, the splitting and joining rings will cover the whole surface. A ring terminates when the next step would produce a non valid boundary, that is, with less than three vertices. When a ring terminates, it means that there were no more significant features in the surface. Therefore, branches of the skeleton will be not produced in the case of a ring which terminates without having a union or a split (see Figure 3.12(c)).

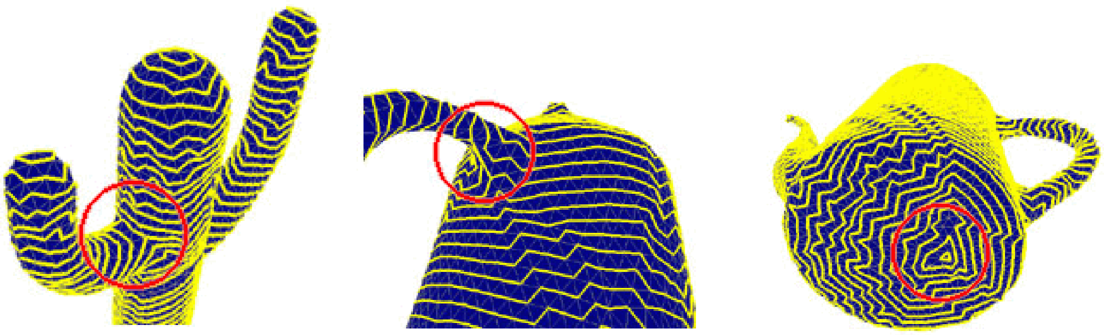

(a) (b) (c)

Figure 3.12: Example of (a) union, (b) split, (c) termination.

Graph visualization. When all the ring sets of the representative vertices and the ones created during their expansion are terminated, the algorithm can draw the skeleton as the adjacency graph encoded during the expansion phase:

- each representative vertex gives a *terminal node*;
- each intersection vertex (occurring in a union or a split configuration) gives a *branching node*;
- the topological rings, belonging to the ring set of a node, give an arc which goes out from that node[2]. In particular, an arc is drawn by joining the centers of mass of all its topological rings.

In Figure 3.13 the skeleton of a woman body has been extracted by using high curvature regions located by red areas.

3.4 Graph as quotient space

In this section, we are concerned with the formulation of the graph construction for triangular meshes and manifolds demonstrating that it is the quotient space of the input surface $\mathcal{M}$ with respect to an equivalence relation "$\sim$". The proposed construction of G as $\mathcal{M}/\sim$ given in Proposition 8 enables to:

- verify that G depends only on the topology of $\mathcal{M}$ and on a finite set of representative points $\{p_1, \ldots, p_n\}$ in $\mathcal{M}$ of high-curvature values;
- prove that G is affine-invariant: this characteristic represents one of the main properties of G;

[2] Note that when a split or a union occurs, there can be two arcs going off the same node: in this case two distinct ring sets are created for the same node. So each ring set always defines one node and one arc.

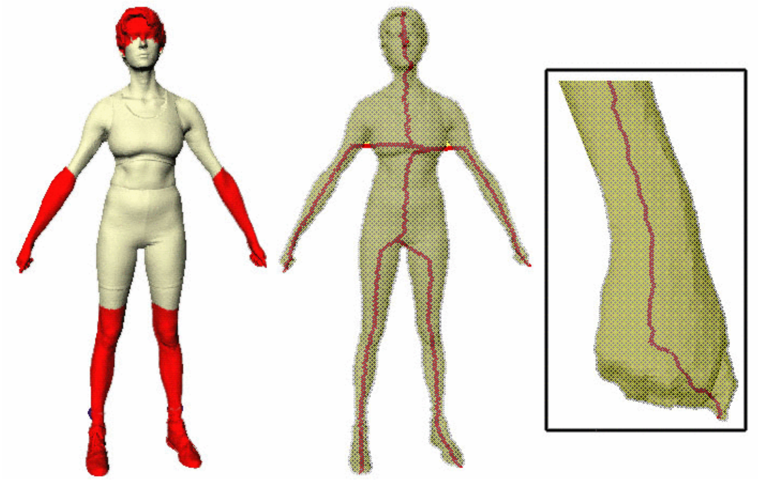


Figure 3.13: Example of skeleton on the woman body.

- extract the information on G (e.g., compactness, connectivity, etc.) starting from $\mathcal{M}$ and exploiting the properties of the quotient space.

Proposition 8. *The graph G previously defined is a Reeb graph.*

Proof. We have to define a function $f : \mathcal{M} \to \mathbb{R}$ such that $G = \mathcal{M}/\sim$ with $\sim$ equivalence relation induced by f and given in Definition 11. Selected a point p_i in $\mathcal{M}$ we introduce the function

$$\begin{array}{ll} f_{p_i} : \mathcal{M} & \longrightarrow \mathbb{N} \\ \quad x & \mapsto f_{p_i}(x) := \min\{k : x \in T(N^k(i))\}, \end{array}$$

i.e., $f_{p_i}(x)$ is the minimal *topological distance* between p_i and x. We can extend in a simple way the previous function to a finite set of vertices $\{p_1, \dots, p_n\}$ as

$$\begin{array}{ll} f : \mathcal{M} & \longrightarrow \mathbb{N} \\ \quad x & \mapsto f(x) := \min_{k=1,\dots,n}\{f_{p_k}(x)\}, \end{array}$$

i.e., f assigns to x its minimal topological distance with respect to the case of more than one vertex $(n = 1, f = f_{n_1})$. Starting from f and $\mathcal{M}$ we are able to construct the relation $\sim$ as follows (see Figure 3.14):

$$p, q \in \mathcal{M}, \;\; p \sim q \;\text{ iff }\; f^{-1}(f(p)) \cap f^{-1}(f(q)) \neq \emptyset. \tag{3.3}$$

First of all, (3.3) implies that if $f(p) \neq f(q)$ then $p \not\sim q$; in fact, we have

$$f^{-1}(f(p)) \cap f^{-1}(f(q)) = f^{-1}(f(p) \cap f(q)) = f^{-1}(\emptyset) = \emptyset.$$

Therefore, necessary condition for $p \sim q$ is that $f(p) = f(q)$, that is, p and q have the same topological distance from the selected set of vertices $\{p_1, \dots, p_n\}$. Secondly, two points p and q are identified by $\sim$ if and only if they have non-disjoint topological rings (i.e., $f^{-1}(f(p)) \cap f^{-1}(f(q)) \neq \emptyset$) of the same order (i.e., $f(p) = f(q)$). These conditions guarantee that $G = \mathcal{M}/\sim$. □

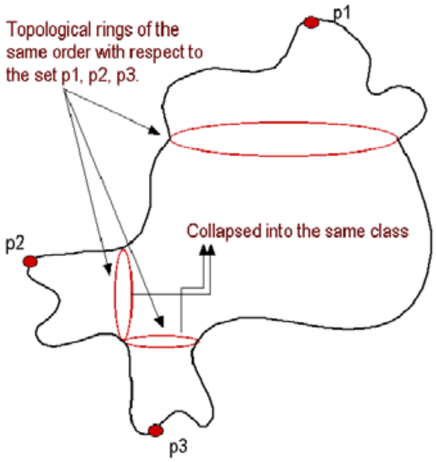

Topological rings of the same order with respect to the set p1, p2, p3.
p1
Collapsed into the same class
p2
p3

Figure 3.14: Example of topological rings on a manifold.

Using the definition of G as quotient space, we derive a set of properties of G; more precisely, from the previous relations it follows that:

$$G = \mathcal{M}/\sim = \{[x] : x \in \mathcal{M}\}$$

where the class of x is represented by the set

$$[x] := \{y \in \mathcal{M} : y \sim x\}.$$

Furthermore, the following conditions hold:

- $[x] \neq \emptyset, \forall x \in \mathcal{M}$ (i.e., every class is not empty);
- $x \sim y \Rightarrow [x] \cap [y] = \emptyset$ (i.e., two vertices which satisfy the relation $\sim$ have non-empty intersection. Generally, the inverse condition is not true because $\sim$ is not transitive.);
- $\bigcup_{x \in \mathcal{M}} [x] = \mathcal{M}$ (i.e., $\{[x]\}_{x \in \mathcal{M}}$ represents a cover of $\mathcal{M}$).

Graph definition for manifolds. We extend the previous model to a compact manifold without boundary embedded in $\mathbb{R}^3$ with the Euclidean topology underlining the general application of our model for the extraction of an affine-invariant shape description. In the following, we review definitions and concepts on topology introducing the basic notions and referring to [ES92] for further readings. The structure of the section reflects that of the previous one facilitating the parallelism between the continuous case study and the discrete one which has been used for the implementation of the algorithm. Given a topological space (X, τ), we define as

- *induced topology of X in $S \subseteq X$*: the topology τ_S defined as

$$\tau_S := \{A \cap S : A \in \tau\};$$

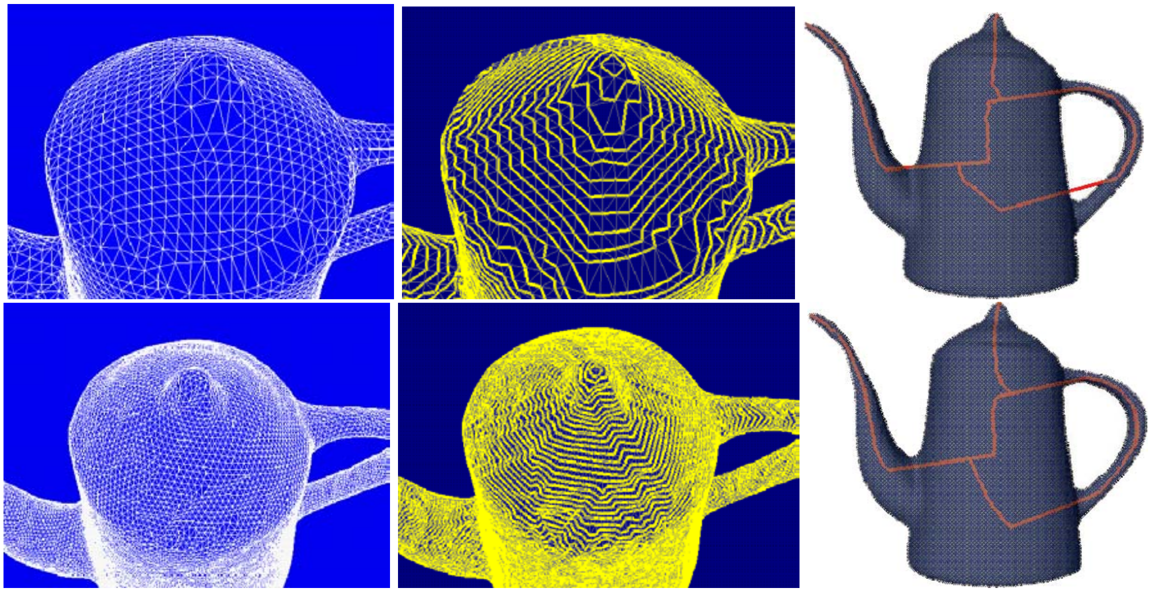


Figure 3.15: Input mesh, topological rings, and skeleton on the same surface at two level of resolution.

- *local base of p in X*: the family B_p consisting of neighborhoods of the point p such that for every neighborhood U of p there is a set $V \in B_p$ such that $V \subseteq U$;
- *boundary of $A \subseteq X$*: $\partial A := \overline{A} \cap \overline{X - A}$, that is, the intersection between the closure of the set A and its complement $(X - A)$ in X.

Given a point $p \in \mathcal{M}$ we define, for every $R > 0$, the open ball of center p and radius R as

$$B(p, R) := \{x \in \mathbb{R}^3 : \|x - p\|_2 < R\}$$

and with $U(p, R)$ the connected component of p in $\mathcal{M} \cap B(p, R)$. Therefore, we can associate to each point $p \in \mathcal{M}$ the family of neighborhoods $\{U(p, R)\}_{R>0}$, with $U(p, 0) := \{p\}$, centered in p. From the previous relations we can derive the following properties which will be used for constructing the graph of $\mathcal{M}$:

- $\{U(p, R)\}_{R>0}$ is a local base of the space $\mathcal{M}$ at the point p with respect to the topology τ_S induced by the Euclidean topology τ in $\mathcal{M}$. This property follows using the definition of τ_S and the fact that the set $\{B(p, R)\}_{R>0}$ is a local base of p in $(\mathbb{R}^3, \tau)$;
- $R_1 < R_2 \Rightarrow U(p, R_1) \subseteq U(p, R_2)$: in fact, $R_1 < R_2 \Rightarrow B(p, R_1) \subseteq B(p, R_2) \Rightarrow \mathcal{M} \cap B(p, R_1) \subseteq \mathcal{M} \cap B(p, R_2) \Rightarrow$ the connected component of p in $\mathcal{M} \cap B(p, R_1)$ is a subset of that in $\mathcal{M} \cap B(p, R_2)$.

Introduced the counterparts of the concepts defined for the triangular mesh, we can extend the previous functions as described in the following. Chosen a point $p \in \mathcal{M}$, we define the map

$$\begin{array}{cccc} f_p: & \mathbb{R}^3 & \longrightarrow & \mathbb{R} \\ & x & \mapsto & f_p(x) := \|p - x\|_2 \end{array}$$

and then, fixed a set of points $P := \{p_1, \dots, p_n\} \subseteq \mathcal{M}$, we construct its extension to the set P as

$$\begin{array}{cccc} f: & \mathcal{M} & \longrightarrow & \mathbb{R} \\ & x & \mapsto & f(x) := \min_{k=1,\dots,n}\{f_{p_k}(x)\}. \end{array}$$

The function f is continuous because it is the composition of the continuous maps

$$\begin{array}{cccc} (f_{p_1}, \dots, f_{p_n}): & \mathcal{M} & \longrightarrow & \mathbb{R}^n \\ & x & \mapsto & (f_{p_1}(x), \dots, f_{p_n}(x)) \end{array}$$

and

$$\begin{array}{cccc} \min: & \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ & (x_1, \dots, x_n) & \mapsto & \min_{i=1,\dots,n}\{x_i\}. \end{array}$$

Therefore, in analogy with the previous section we can introduce the relation $\sim$ as: $p, q \in \mathcal{M}, p \sim q$ if and only if there exists $R > 0$ such that $f(p) = f(q)$ and p, q belong to the same connected component of $f^{-1}(f(p))$. In other words, necessary and sufficient condition for $p \sim q$ is that p and q have the same topological distance from the selected set of points $\{p_1, \dots, p_n\}$ and they belong to the connected component of the pre-image of their (common) value $f(p)$. The relation $\sim$ is symmetric, reflexive, and transitive because it is the intersection of two equivalence relations (i.e., function equality and membership to the same connected component). Using the properties of the quotient space, we deduce that $\sim$ induces in $\mathcal{M}$ a decomposition into a family of non-empty, disjoint topological classes. If the input surface $\mathcal{M}$ is compact/connected then G is compact/connected; anyway, the canonical projection

$$\begin{array}{cccc} \pi: & \mathcal{M} & \longrightarrow & G = \mathcal{M}/\sim \\ & x & \mapsto & [x] \end{array}$$

is continuous with respect to the quotient space topology (i.e. $A \subseteq G$ is open if $\pi^{-1}(A)$ is open in $(\mathcal{M}, \tau)$). Now we want to investigate the properties of $\sim$:

- reflexive: $\forall p \in \mathcal{M}, p \sim p$;
- symmetric: $\forall p, q \in \mathcal{M}, p \sim q \Rightarrow q \sim p$;
- transitive: $\forall p, q, r \in \mathcal{M}, p \sim q, q \sim r \not\Rightarrow p \sim r$. In fact, the intersection of sets is not transitive on $P(X) := \{X : X \subseteq \mathcal{M}\}$.

The previous construction can be reduced to define the function f such that $Im(f) := \{f(x) : x \in \mathcal{M}\} \subseteq \mathbb{N}$ choosing, for a given point $p \in \mathcal{M}$, its local base as $B_p = \{\mathcal{M} \cap B(p, kn)\}_{n \in \mathbb{N}}$, with $k \in \mathbb{R}^+$ step, and defining for every $x \in \mathcal{M}$ $f(x) := \min_{k=1,\dots,n}\{f_{p_k}(x)\}$ where $f_{p_i}(x) := \min\{n : x \in B(p_i, kn)\}$.

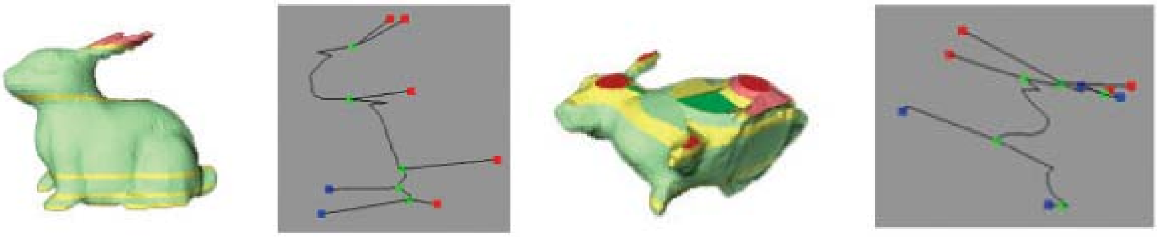


Figure 3.16: Reeb Graph on the rabbit with respect to the height function related to different directions.

3.5 Conclusions

We present the main properties of the graph G and a comparison with respect to other approaches. The complexity of the proposed graph, in terms of number of nodes and branches, depends on the shape of the input surface and on the set of points $\{p_i\}_{i=1}^{n}$ that have been selected by the multi-scale curvature evaluation. The construction of G is guided, in the first step, by the topology of the mesh through the connectivity relations in T and, secondly, by the geometry of $\mathcal{M}$ which influences the chosen representative point p for its equivalence class $[p]$. The only requirement for an optimal construction of the graph deals with a uniform mesh finalized at having a topological distance between points to which corresponds a medium step on the mesh (see Figure 3.15). The graph structure is not incremental in the sense that if we add a point p_{n+1} to the set $\{p_1, \ldots, p_n\}$, which is used for the construction of G, the new graph $\tilde{G}$ defined by $\{p_1, \ldots, p_n, p_{n+1}\}$ is not achieved by adding to G the new branch starting from p_{n+1}. The proposed graph is affine-invariant with respect to translation, rotation, and scaling because the function f which has been used for defining the relation $\sim$ and the quotient space G does not rely on a local system of coordinates in which the surface is embedded as it happens in the case of the Reeb graph constructed, for example, using the height function (see Figure 3.16). However, we stress that both graphs are built in the same way by choosing only two different functions; our graph (resp., the Reeb graph) codes the surface shape by analyzing the evolution of topological arcs (resp., contours) obtained by the mesh topology (resp., intersecting the surface with a family of planes orthogonal to a chosen direction). On the other hand, if the curvature evaluation process does not recognize at least one feature region (e.g., surfaces with constant curvature values such as spheres), our approach is not useful to extract a description of the shape; on the contrary, the height function always guarantees to get a result. The Reeb graph with respect to an arbitrary function f and the related shape decomposition are used in Chapter 4 for defining a new approach to the local and global parameterization of triangulated 2-manifolds with an arbitrary genus. Other examples are given in Figure 3.17 and Table 3.1 gives the list of computational costs.

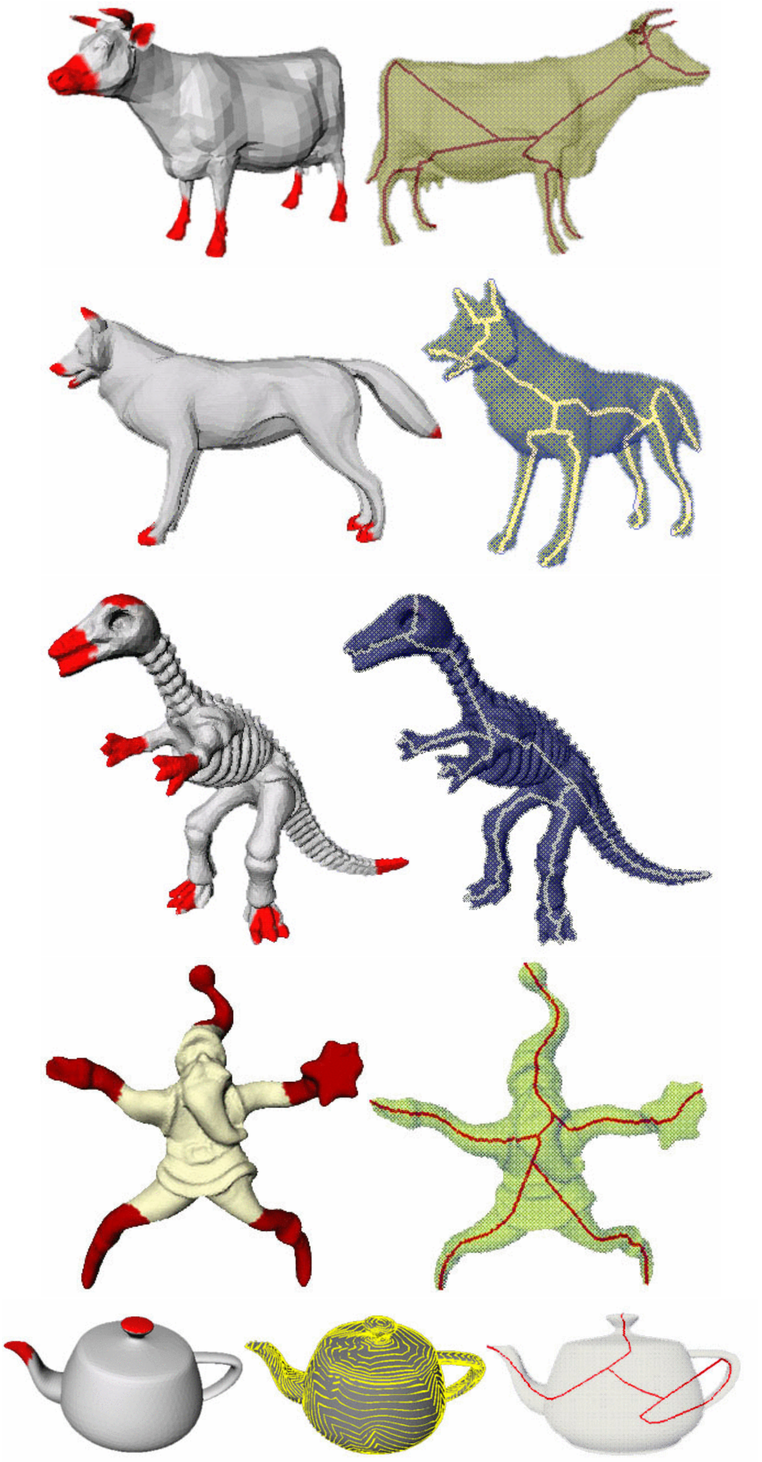


Figure 3.17: Example of skeletons of different surfaces.

Table 3.1: The computational cost for the extraction of the medial axis transformation depends on the number of line elements, n_s, the number of steps taken along each seam s, and the number of boundary entities n of the MAT (details can be found in [SPB96]). The computational costs related to the evaluation of f (worst and average case) and to the extraction of the induced Reeb graph are expressed in terms of the number b of triangle bases used for the evaluation of f, (in practice, b is less than 150), n is the number of vertices of the input triangle mesh, $k_{\max}$ is the size of the maximum neighborhood of each vertex used to approximate the curvature.

Method/f	Graph Constr.	Funct. Eval. (*worst*)	Funct. Eval. (*average*)
MAT	$O((n+s)n_s \log n)$	-	-
Height	$O((n+k)\log(n+k))$	$O(1)$	$O(1)$
Dist. from bar.	$O((n+k)\log(n+k)$	$O(n)$	$O(n)$
Int. geod. dist.	$O(n+k)$	$O(n^2 \log n)$	$O(bn \log n)$
Dist. source point	$O(n)$	$O(n \log n)$	$O(n \log n)$
Top. dist. curv. extr.	$O(n)$	$O(n^2)$	$O(nk_{\max})$

Related publications

1. S. Biasotti, S. Marini, M. Mortara, G. Patanè, B. Falcidieno, M. Spagnuolo. “3D Shape Matching through Topological Structures”. In *Discrete Geometry for Computer Imagery*, Lecture Notes in Computer Science, Volume 2886, pp. 194-204, Springer-Verlag, Naples, Italy, 19-21 November 2003.

2. S. Biasotti, S. Marini, M. Mortara, G. Patanè. “An overview on properties and efficacy of skeletons in Shape Modelling”. In *Proceedings of Shape Modeling International and Applications 2002*, pp. 245-254, IEEE Computer Society Press, Seoul National University, Korea, 12-16 May 2003.

3. M. Mortara, G. Patanè. “Shape covering for skeleton extraction”. In *International Journal of Shape Modeling*, Special Issue of “International Conference on Shape Modeling and Applications 2002”, Vol. 8, Num. 2, pp. 139-158, World Scientific, December 2002.

4. M. Mortara, G. Patanè. “Affine-invariant skeleton of 3D-shapes”. In *Proceedings of Shape Modeling International and Applications 2002*, pp. 245-252, IEEE Computer Society Press, Banff, Alberta, Canada, 17-22 May 2002.

5. S. Biasotti, M. Mortara, G. Patanè. “A topology-based approach to shape modelling”. In *Proceedings of Eurographics Italian Chapter*, Università degli Studi di Milano, 11-12 July 2002.

6. S. Biasotti, B. Falcidieno, M. Mortara, G. Patanè, M. Spagnuolo. “Computational topology techniques for shape modeling”. In *ERCIM-News*, pp. 23-24, July 2002.

7. M. Mortara, G. Patanè. *Affine-invariant skeleton of 3D shapes.* Rapporto Tecnico N. 12/2001, Istituto per la Matematica Applicata, Consiglio Nazionale delle Ricerche.

Chapter 4

Local and global parameterization of surfaces with arbitrary genus

This chapter describes a novel approach to the local and global parameterization of triangle meshes representing 2-manifolds with an arbitrary genus. Concerning the local parameterization, decompositions of the shape based on several criteria (i.e., topology, morphology, and geometry) are computed and used to segment the shape into primitives, which define a chart decomposition of the mesh. Then, each chart is parameterized using an extension of the barycentric coordinates method. The charts are all 0-genus and can be of three types only, depending on the number of boundary components. The chart decomposition and the parameterization are used to define a shape-graph where each node represents one primitive and the arcs code the adjacency relations among the primitives. The primitives are coded together with their skeletal lines that are computed from and aligned with their parameterization. The surface decomposition is used for extracting a skeleton aligned with the local parameterization which is exploited for shape deformations which are free of artifacts. The application of the parameterization approach to remeshing (i.e., piecewise linear interpolation and approximation) guarantees that extraordinary vertices are localized only where two patches share a boundary and they are not scattered on the whole surface. Concerning the global parameterization, given an arbitrary triangle mesh $\mathcal{M}$ our approach searches a cut graph made of the iso-contours of a fair function $f : \mathcal{M} \rightarrow \mathbb{R}$ and it works in a planar domain where geodesic curves are defined by line segments whose counterparts on $\mathcal{M}$, with respect to an appropriate diffeomorphism $\phi : \mathcal{M} \rightarrow \mathbb{R}^2$, give smooth approximations of geodesic paths. The emphasis of the proposed approach is on the definition of a simple method for finding a family of cut graphs of $\mathcal{M}$ and guided by several criteria which spread from the global parameterization (e.g., minimal length, minimization of the parameterization distortion, or interpolation of points as required by remeshing and texture mapping) to the calculation of polygonal schemes for surface classification.

4.1 Introduction

The parameterization of 2D manifold triangle meshes has been largely analyzed in Computer Graphics due both to its theoretical importance for surface classification and impact on applications such as texture mapping, compression, and animation. While the parameterization of manifolds homeomorphic to a disc is well understood, the case of surfaces with an arbitrary genus is still a challenging research topic. This problem is usually approached in two different ways: either the surface is decomposed into a family of disc-like patches (i.e., *local parameterization*, or atlas generation), or a cut is defined on the surface which is successively unfolded onto a planar domain (i.e., *global parameterization*).

Existing methods of local parameterization are mainly driven by considerations concerning texture mapping and compression as primary application of the parameterization results; therefore, the patches are generally not related to features of the surface and the number of generated patches is mainly driven by error or distortion metrics. These side-effects are the starting point of our work and motivate the necessity of studying more general methods for local parameterization which extend the state of the art in the field, maintaining simplicity, robustness, and range of applicability. We approach the problem by discussing the properties of shape segmentation methods from a general perspective, focusing on a minimal set of constraints that the segmentation should satisfy in order to be a valid input for the proposed framework. The final result is a general framework which defines a family of local parameterizations guided by different criteria such as the shape of the input surface, its topology, or any other property which guarantees that each patch has 0-genus and by the aim of building a centerline skeleton, aligned with the parameterization, and which can be used for deformations in a simple manner.

Alternatively, an arbitrary surface can be unfolded into a single planar domain, by cutting the surface along a connected path. Using a single patch allows to apply *regular* remeshing and to achieve high compression rates; however, it can create parameterizations with greater distortion and a less uniform sampling with respect to the use of multiple local charts. This phenomenon is mainly due to the difficulty of finding a good cut and of controlling its alignment with sharp features (e.g., high-curvature lines); furthermore, the construction of the cut can create complicated and noisy line networks for unfolding surfaces of high genus. The proposed approach is based on the characterization of a given surface $\mathcal{M}$ provided by the Morse Theory and the Reeb graph as described in Chapter 3. More precisely, the Reeb graph of $\mathcal{M}$ with respect to a function $f : \mathcal{M} \rightarrow \mathbb{R}$ codes $\mathcal{M}$ as a graph whose cycles are used to identify the topological handles of $\mathcal{M}$ and the associated iso-contours correspond to the meridian curves along which the surface has to be cut for reducing its genus to zero.

4.1.1 Overview and contributions

The local parameterization problem is approached by defining an atlas generation method for 2-manifold triangle meshes of arbitrary genus, which produces a *canonical* decomposition of the shape, a *minimal* number of charts, a *flexible* decomposition schema able to capture the most interesting shape features, and a coding of the parameterization which is *useful* for applications such as editing and deformation. The crux of the proposed method is the exploitation of the information about both the surface topology for optimizing its decomposition and cutting, and the surface geometry for minimizing the parameterization distortion. The use of topological information provides evident benefits in all subsequent stages of the parameterization and its applications. The proposed approach is applied to remeshing, and produces remeshed shapes whose extraordinary vertices (i.e., with valence different than six) are localized on *smooth boundaries* where adjacent patches join, and they are not scattered on the surface as in previous work.

Segmentation methods. The only assumption on the surface decomposition used as a basis for the local parameterization is that each patch has 0-genus, without constraining the number of boundaries. This *general* requirement provides a higher degree of flexibility for the choice of the segmentation method and for the identification of patches which represent *features* of the input shape. With respect to methods based on mesh simplification, a feature-based segmentation could produce better results from the point of view of the parameterization: patch boundaries could be drawn at lines of high curvature, symmetries of the input surface could be exploited, or patches could be defined to identify topological primitives.

The concept of feature depends on the application context. For instance, in CAD, features of a shape are slots, pockets through holes, and many more. In our general case, features represent regions of interest with respect to the geometric and topological aspects. Each patch could represent a *geometric* or *morphological feature*, for instance a protrusion of the shape, a *topological feature*, such as a part of the surface whose level sets are homeomorphic, or also a patch blending two features. Let $\mathcal{J}$ be the family of segmentation methods which decompose a given surface $\mathcal{M}$ into 0-genus patches (see Section 4.6 for the formal definition). In the chapter, we review different methods which belong to $\mathcal{J}$ and we show how the use of these primitives enables to segment $\mathcal{M}$ in a natural way, both in terms of the geometry of the primitives and their adjacency relations, thus proving the flexibility of the proposed framework and how different decomposition strategies provide several inputs for the local parameterization. The selected segmentation methods can be applied to closed or bordered meshes and produce smooth patch boundaries. In particular, we will discuss *topology*-based segmentation which makes use of the Reeb graph [PSF04, SKK91] or of dynamic structures [DGG03], *morphology*-based segmentation which is related to a multi-scale shape analysis [MPS$^+$04b], and *geometry*-based segmentation defined by a clustering among vertices [KT03].

Patch classification and parameterization. For defining a local parameterization based on any segmentation method in $\mathcal{J}$, we need a simple method for parameterizing patches of 0-genus with an arbitrary number of boundaries. Independently of the specific segmentation used to define the charts, we classify the patches according to the number of their boundary components into three types: generalized cones, cylinders, and bodies. Then, each patch type is parameterized using an extension [PSF04] of the barycentric coordinates method [Flo97, Tut63]. More precisely, we reduce this problem to finding a path on each patch with two or more boundary components which joins them; in this way, we convert it to a new surface with one boundary. The produced cuts are well shaped and stable with respect to the quality of the input patch in terms of tiny triangles and irregular sampling. After this step, the geometric properties that have been used for the patch identification (i.e., curvature, pre-defined shape) can be taken into account by a specific parameterization method of disc-like surfaces. A straightforward solution could be subdividing each patch into triangular regions; however, in this manner we lose the identification of patches with significant shape features, which limits in practice the use of the parameterization to simplification-based approaches.

Applications. The low distortion provided by the feature-based segmentation corresponds to a more regular sampling of $\mathcal{M}$ and remeshed surfaces have extraordinary vertices localized on smooth boundary components of adjacent patches. In Chapter 5, the proposed method is used for extracting a skeleton of the input surface *aligned with its parameterization*, and the skeletal line of generalized cones and cylinders is used for local deformations which are free of artifacts. Furthermore, the approach enables to resample and apply arbitrary textures to the patch while it evolves from the initial to the final state.

Global parameterization. The set of cut graphs for a given triangle mesh $\mathcal{M}$ with an arbitrary genus depends on the underlining topology and the selection of a specific one should be guided by the surface geometry and targeted applications. The influence of the mesh connectivity on the cut graph search affects previous work due to the use of algorithms based on mesh traversal techniques for the evaluation of the geodesic metric. Our solution is to search a cut graph made of the iso-contours of a fair function $f : \mathcal{M} \rightarrow \mathbb{R}$ and to work in a planar domain where geodesic curves are defined by line segments whose counterparts on $\mathcal{M}$, with respect to an appropriate diffeomorphism $\phi : \mathcal{M} \rightarrow \mathbb{R}^2$, give smooth approximations of geodesic paths. The emphasis of the proposed approach is on the definition of a simple method for finding a family of cut graphs of $\mathcal{M}$ and guided by several criteria which spread from the global parameterization (e.g., minimal length, minimization of the parameterization distortion, or interpolation of points as required by remeshing and texture mapping) to the calculation of polygonal schemes for surface classification. The proposed approach finds a cut graph of a closed or bordered triangle mesh $\mathcal{M}$ with n vertices in $O(n)$ time if $\mathcal{M}$ has 0-genus, and $O(n\log(n))$ time if $g \geq 1$; finally, the associated polygonal schema is reduced if $g = 0$ and with a constant number of redundant edges otherwise.

The chapter is organized as follows: in Section 4.2 and 4.3 we review basic concepts on surfaces and intrinsic geometry, Sections 4.4 and 4.5 introduce the parameterization problem. In Section 4.6, we describe the segmentation of surfaces with an arbitrary genus based on different methods, define its coding as a shape-graph, and discuss the properties of the graph. In Section 4.7, the parameterization of the graph nodes is presented, and its application to deformation and remeshing are detailed in Chapter 5. The global parameterization problem is presented in Section 4.8 and conclusions are discussed in the last section.

4.2 Surfaces and fundamental forms

Definition 14. *A* regular parametric representation of class C^m*,* $m \geq 1$*, of a set of points* $\mathcal{S}$ *in* $\mathbb{R}^3$ *is a mapping* $\varphi = \varphi(u, v)$ *of an open set* U *in the uv plane onto* $\mathcal{S}$ *such that:*

- φ *is of class* C^m *in* U*;*
- *if* (e_1, e_2, e_3) *is a basis in* $\mathbb{R}^3$ *and* $\varphi(u, v) := (\varphi_1(u, v), \varphi_2(u, v), \varphi_3(u, v))$*, then for all* $(u, v) \in U$
$$rank \begin{pmatrix} \partial_u f_1 & \partial_v f_1 \\ \partial_u f_2 & \partial_v f_2 \\ \partial_u f_3 & \partial_v f_3 \end{pmatrix} = 2.$$

From the previous definition it follows that a mapping φ of an open set U onto $\mathcal{S}$ is a regular parametric representation of class C^m of $\mathcal{S}$ if and only if (i) φ belongs to class C^m $(m \geq 1)$ on U, (ii) $n := (\varphi_u \wedge \varphi_v) \neq 0$ in U; the condition (ii) guarantees that the tangent plane at $\mathcal{S}$ in $\varphi(u_0, v_0)$, $(u_0, v_0) \in U$, is well defined and its equation is $< n(u_0, v_0), (x, y, z) > -\varphi(u_0, v_0) = 0$.

Definition 15. *A* coordinate patch of class C^m, $m \geq 1$, in $\mathcal{S}$ *is a mapping* $\varphi = \varphi(u, v)$ *of an open set* U *into* $\mathcal{S}$ *such that*

- φ *is of class* C^m *on* U*;*
- $\varphi_u \wedge \varphi_v \neq 0$ *for all* $(u, v) \in U$*;*
- φ *is injective and bi-continuous on* U*.*

Let $\varphi = \varphi(u, v)$ be a coordinate patch on a surface of class C^m, $m \geq 1$; then, the *differential* of φ at (u, v) is a 1-1 linear mapping $d\varphi = \varphi_u du + \varphi_v dv$ of the vector (du, dv) in the uv plane onto the vectors $\varphi_u du + \varphi_v dv$ parallel to the tangent plane at $\varphi(u, v)$. We now consider the *first fundamental form* defined as

$$I := \|d\varphi\|_2^2 = Edu^2 + 2Fdudv + Gdv^2 \tag{4.1}$$

where we set

$$E := \|\varphi_u\|_2^2, \quad F :=< \varphi_u, \varphi_v >, \quad G := \|\varphi_v\|_2^2.$$

The function $I(du, dv)$ is homogeneous of degree two in du and dv; arranging the coefficients E, F, G in a symmetric matrix

$$T := \begin{pmatrix} E & F \\ F & G \end{pmatrix} \tag{4.2}$$

we have

$$I = (du\, dv)\, T \begin{pmatrix} du \\ dv \end{pmatrix}.$$

We recall that I depends only on the surface $\mathcal{S}$ and not on the particular parametric representation φ, and that under the assumptions of regularity for φ the matrix T is positive definite (or equivalently, I is a positive definite quadratic form). The first fundamental coefficients E, F, G play a basic role in calculating arc length, angles, and surface area; in fact, the following relations hold.

- Let $\gamma(t) := \varphi(u(t), v(t))$, $a \leq t \leq b$, be a regular arc on a patch $\varphi = \varphi(u, v)$; then, its length is the integral

$$s = \int_a^b \Big|\frac{d\gamma}{dt}\Big| dt = \int_a^b \left[E\left(\frac{du}{dt}\right)^2 + 2F\frac{du}{dt}\frac{dv}{dt} + G\left(\frac{dv}{dt}\right)^2 \right]^{1/2} dt; \tag{4.3}$$

- the area of a region R on $\mathcal{S}$ parameterized by $\varphi = \varphi(u, v)$ is defined as

$$A(R) := \iint_U \sqrt{EG - F^2} dudv \tag{4.4}$$

  where U is the set of points in the parameter plane which maps onto R. We note that in (4.4) $g := det(T) = (EG - F^2) > 0$ in U;

- let $\gamma(t) := \varphi(u(t), v(t))$, $a \leq t \leq b$, and $\beta(\tau) := \varphi(\eta(\tau), \xi(\tau))$, $c \leq \tau \leq d$, be two curves on $\mathcal{S}$ intersecting at p with tangents $\gamma^{'} = \varphi_u u^{'} + \varphi_v v^{'}$ and $\beta^{'} = \varphi_u \eta^{'} + \varphi_v \xi^{'}$. If $\theta = \angle(\beta, \gamma)$ is the angle between $\gamma^{'}$ and $\beta^{'}$ at p, then

$$\cos(\theta) = \frac{Eu^{'}\eta^{'} + F(u^{'}\xi^{'} + v^{'}\eta^{'}) + Gv^{'}\xi^{'}}{\left[E(u^{'})^2 + 2Fu^{'}v^{'} + G(v^{'})^2\right]^{1/2} \left[E(\eta^{'})^2 + 2F\eta^{'}\xi^{'} + G(\xi^{'})^2\right]^{1/2}}. \tag{4.5}$$

4.3 Intrinsic geometry

Let $\mathcal{S}$ be a surface of class C^m, $m \geq 1$, and $f : \mathcal{S} \rightarrow \mathcal{S}^\star$ be a function (see Figure 4.1). If for every coordinate patch $\varphi = \varphi(u, v)$ on $\mathcal{S}$ with domain U, the composite mapping $\varphi^\star = \varphi^\star(u, v) := f(\varphi(u, v))$ of U into $\mathcal{S}^\star$ is a regular parametric representation of class C^r (see Definition 14), $r \leq \min\{m, n\}$, then f is called a *regular differentiable mapping of $\mathcal{S}$ into $\mathcal{S}^\star$ of class C^r*. Using the fact that a regular parametric representation is locally 1-1 and bicontinuous, it is possible to prove the following theorem.

Theorem 9. *Let f be a regular differentiable mapping of class C^r, $r \geq 1$, from a surface $\mathcal{S}$ into a surface $\mathcal{S}^\star$.*

- *For every point $p \in \mathcal{S}$ there exists a coordinate patch $\varphi = \varphi(u, v)$ on $\mathcal{S}$ containing p such that $\varphi^\star = f(\varphi(u, v))$ is a coordinate patch on $\mathcal{S}^\star$.*

- *If $\gamma = \gamma(t)$, $a \leq t \leq b$, is a regular curve of class C^r on $\mathcal{S}$, then $\varphi^\star = \varphi^\star(t) := \varphi(\gamma(t))$ is a regular curve of class C^r on $\mathcal{S}^\star$.*

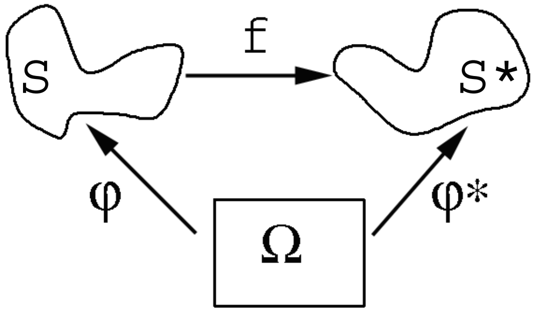


Figure 4.1: Regular differentiable mapping f between two surfaces $\mathcal{S}$ and $\mathcal{S}^\star$.

A 1-1 mapping f of a surface $\mathcal{S}$ onto $\mathcal{S}^\star$ is called an *isometry* if the length of an arbitrary regular arc $\gamma = \gamma(t)$ on $\mathcal{S}$ is equal to the length of its image $\gamma^\star = \gamma^\star(t) := f(\gamma(t))$ on $\mathcal{S}^\star$ (see equation (4.3)). If there exists an isometry from $\mathcal{S}$ onto $\mathcal{S}^\star$, then $\mathcal{S}$ and $\mathcal{S}^\star$ are said to be *isometric*; finally, if f is an isometry from $\mathcal{S}$ onto $\mathcal{S}^\star$ then f^{-1} is an isometry from $\mathcal{S}^\star$ into $\mathcal{S}$.

A regular differentiable mapping f of a surface $\mathcal{S}$ into a surface $\mathcal{S}^\star$ is said to be *equiareal* if every region of $\mathcal{S}$ is mapped onto a region of $\mathcal{S}^\star$ with the same area (see equation (4.4)). The mapping f is defined as *conformal*, or *angle-preserving*, if the angle between two oriented curves $\gamma = \gamma(t)$ and $\beta = \beta(\tau)$ on $\mathcal{S}$ intersecting at p is the same as that of the corresponding images on $\mathcal{S}^\star$ through f at the corresponding point $f(p)$ (see equation (4.5)).

Using the expression of the arc-length (4.3), area (4.4), and angle between two curves (4.5) as function of the coefficients E, F, G of the first fundamental form I in (4.1), it is possible to prove that isometric, angle-preserving, and equiareal mappings between $\mathcal{S}$ and $\mathcal{S}^\star$ are fully characterized by I as stated by Proposition 9.

Proposition 9. *Let $f : \mathcal{S} \rightarrow \mathcal{S}^\star$ be a 1-1 mapping, $I = Edu^2 + 2Fdudv + Gdv^2$ and $I^\star = E^\star du^2 + 2F^\star dudv + G^\star dv^2$ be the first fundamental form of $(\mathcal{S}, \varphi)$ and $(\mathcal{S}^\star, \varphi^\star := f(\varphi))$ respectively. Then,*

- *f is an isometry if and only if $I = I^\star$, that is,*

$$E = E^\star, \quad F = F^\star, \quad G = G^\star;$$

- *f is equiareal if and only if the matrices T and $T^\star$ associated to I and $I^\star$ have the same determinant, that is,*

$$det(T) = det(T^\star) \quad \Longleftrightarrow \quad EG - F^2 = E^\star G^\star - {F^\star}^2;$$

- *f is conformal if and only if there exists a function $\eta = \eta(u, v) \neq 0$ in U such that $I = \eta I^\star$, that is,*

$$E = \eta E^\star, \quad F = \eta F^\star, \quad G = \eta G^\star;$$

- *f is an isometry if and only if f is equiareal and conformal.*

These concepts will be used in Section 4.5 for characterizing the distortion of the parameterization of triangle meshes.

Conformal and harmonic mappings. We now want to investigate the link between conformal and harmonic mappings proving that a function φ which is conformal is also harmonic. Let φ be any mapping from a surface $\mathcal{S}$ to the plane and be (u^1,u^2) the related coordinates of $\mathcal{S}$. By Proposition 9, if φ is conformal there exists a scalar function $\eta=\eta(u^1,u^2)$ such that

$$ds^2=\eta(u^1,u^2)\left((du^1)^2+(du^2)^2\right). \tag{4.6}$$

If $\mathcal{S}^\star$ has coordinates $(\tilde{u}^1,\tilde{u}^2)$ and $E^\star$, $F^\star$, $G^\star$ are the coefficients of its first fundamental form, it is possible to show that (4.6) implies the equations [DoC76]:

$$\frac{\partial u^1}{\partial \tilde{u}^1}=\frac{E^\star}{\sqrt{\tilde{g}}}\frac{\partial u^2}{\partial \tilde{u}^2}-\frac{F^\star}{\sqrt{\tilde{g}}}\frac{\partial u^2}{\partial \tilde{u}^1},\quad \frac{\partial u^1}{\partial \tilde{u}^2}=-\frac{G^\star}{\sqrt{\tilde{g}}}\frac{\partial u^2}{\partial \tilde{u}^1}+\frac{F^\star}{\sqrt{\tilde{g}}}\frac{\partial u^2}{\partial \tilde{u}^2}, \tag{4.7}$$

where $\tilde{g}:=E^\star G^\star-(F^\star)^2$.

If we differentiate one equation in (4.7) with respect to $\tilde{u}^1$ and the other one with respect to $\tilde{u}^2$ (and viceversa), by summing them we obtain the generalization of the Laplace equation

$$\Delta_S u^1=0,\quad \Delta_S u^2=0,$$

where Δ_S is the Laplace-Beltrami operator

$$\Delta_S:=\frac{1}{\sqrt{\tilde{g}}}\left(\frac{\partial}{\partial\tilde{u}^1}\left(\frac{G^\star}{\sqrt{\tilde{g}}}\frac{\partial}{\partial\tilde{u}^1}-\frac{F^\star}{\sqrt{\tilde{g}}}\frac{\partial}{\partial\tilde{u}^2}\right)+\frac{\partial}{\partial\tilde{u}^2}\left(\frac{E^\star}{\sqrt{\tilde{g}}}\frac{\partial}{\partial\tilde{u}^2}-\frac{F^\star}{\sqrt{\tilde{g}}}\frac{\partial}{\partial\tilde{u}^1}\right)\right).$$

Finally, we note that a harmonic function is not necessarily conformal; an example is given by $f(x,y):=(x(1+y),y)$.

4.4 Graphs and triangulation

In this section, we introduce the definition of parameterization and we review some concepts on surface triangulations (see Section 1.1) from the perspective of graph theory [Die00]. A *simple graph* $\mathcal{G}=\mathcal{G}(V,E)$ is a set of nodes $V:=\{i:i=1,\dots,n\}$ and a set of edges $E:=\{(i,j)\in V\times V: i\neq j\}$ subset of all unordered pairs of distinct nodes. A graph $\mathcal{G}$ is *planar* if it can be embedded in the plane such that:

- each node $i\in V$ is mapped to a point $u_i\in\mathbb{R}^2$;
- each edge $(i,j)\in E$ is mapped to a curve whose end points are u_i, u_j;
- the only intersections between two curves corresponding to two distinct edges are at common endpoints.

Such a planar embedding of $\mathcal{G}$ is referred to as a *plane* graph. A plane graph $\mathcal{G}$ partitions the plane into connected regions called *faces* which can be bounded or unbounded; in this last case, each face is called an *outer* face. Let $\partial\mathcal{G}$ be the set of nodes and edges incident to the outer faces; if $\partial\mathcal{G}$ is a simple planar curve, $\mathcal{G}$ is called a *simply connected* graph with boundary $\partial\mathcal{G}$. Finally, a plane graph is *triangulated* if all its bounded faces are triangular, that is, they have three edges.

Definition 16. *A* planar triangulation $\mathcal{G}$ *is a simply-connected triangulated plane graph whose edges are straight.*

Let F be the set of bounded triangular faces of $\mathcal{G}$, then we write $\mathcal{G} = \mathcal{G}(V, E, F)$.

Definition 17. *A* surface triangulation $\mathcal{S} = \mathcal{S}(\mathcal{G}, X)$ *is an embedding in* $\mathbb{R}^3$ *of a simply-connected triangulated plane graph* $\mathcal{G} = \mathcal{G}(V, E, F)$*,* $V := \{i : i = 1, \ldots, n\}$*, with node set* $X := \{x_i \in \mathbb{R}^3 : i = 1, \ldots, n\}$*, straight lines for edges, and triangular facets for faces.*

Since the boundary $\partial\mathcal{G}$ of a simply-connected plane graph $\mathcal{G}$ is a simple closed planar curve it has a well-defined orientation, which we take as the anti-clockwise direction.

Definition 18. *Two simply connected triangulated plane graphs* $\mathcal{G}_1$ *and* $\mathcal{G}_2$ *are* isomorphic *if there is an injective correspondence between their nodes, edges, and faces in a such way that corresponding edges join corresponding nodes, corresponding faces join corresponding nodes and edges, two anti-clockwise sequences of nodes in* $\partial\mathcal{G}_1$ *and* $\partial\mathcal{G}_2$ *correspond.*

The above definitions enable to define the parameterization as follows.

Definition 19. *A* parameterization *of a surface triangulation* $\mathcal{S} = \mathcal{S}(\mathcal{G}, X)$ *is any planar triangulation* $\mathcal{P}$ *which is isomorphic to* $\mathcal{G}$*.*

Definition 19 does not depend on the set X which defines the geometry of the surface triangulation $\mathcal{S}$; however, a “good” parameterization (see Section 4.5.3) will be defined in a such way that corresponding neighborhoods on $\mathcal{S}$ and $\mathcal{P}$ will have a local geometry with a low “distortion”.

Example 1. *Given* n *distinct points* $\Omega := \{u_i := (a_i, b_i), i = 1, \ldots, n\}$ *in* $\mathbb{R}^2$*, let* $\mathcal{P}$ *be any triangulation (e.g., the Delaunay triangulation [BE92]) of* Ω *and* $X := \{x_i := (a_i, b_i, c_i), i = 1, \ldots, n\}$ *a point cloud in* $\mathbb{R}^3$ *with* $\{c_i\}_{i=1,\ldots,n} \subseteq \mathbb{R}$ *arbitrary values. Then,* $\mathcal{S}(\mathcal{P}, X)$ *is a surface triangulation with parameterization* $\mathcal{P}$*. An example is given in Figure 4.2.*

4.5 The parameterization problem

In this section, we make explicit the concept of surface parameterization and how it is approached. Let $\mathcal{S} = \mathcal{S}(\mathcal{G}, X)$ be a surface triangulation and $S_k(\mathcal{S})$, $k \in \mathbb{N}$, be the linear space of all continuous functions $\varphi : \mathcal{S} \to \mathbb{R}^k$ which are linear on each triangle of $\mathcal{S}$, that is,

$$S_k(\mathcal{S}) := \left\{\varphi : \mathcal{S} \to \mathbb{R}^k : \ \varphi \in C^0, \ \varphi|_\Delta \in \mathbb{P}^1 \ \forall \Delta \in \mathcal{S}\right\}$$

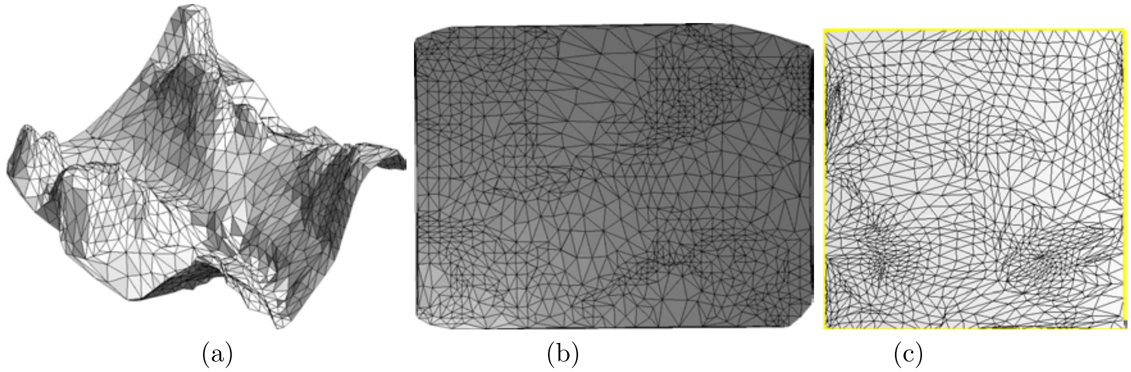


Figure 4.2: (a) Input surface triangulation, (b) planar parameterization achieved as described in the example 1, (c) shape-preserving parameterization on the unit square (see Section 4.5.1 and 4.5.2).

with $\mathbb{P}^1$ the linear space of linear functions from $\mathbb{R}^3$ to $\mathbb{R}^k$. From now on, we consider the case $k := 2$ (i.e., we search an embedding of $\mathcal{S}$ on the plane), while more general considerations are given in Section 4.8 and 4.8.1. If we focus our attention on a single triangle $\Delta := [x_{i_1}, x_{i_2}, x_{i_3}] \in \mathcal{S}$, the linearity of φ on Δ enables to write

$$\varphi(x) = \lambda_1\varphi(x_{i_1}) + \lambda_2\varphi(x_{i_2}) + \lambda_3\varphi(x_{i_3}), \quad x \in \Delta \tag{4.8}$$

where

$$\lambda_1 := \frac{\text{area}([x, x_{i_2}, x_{i_3}])}{\text{area}(\Delta)}, \quad \lambda_2 := \frac{\text{area}([x_{i_1}, x, x_{i_3}])}{\text{area}(\Delta)}, \quad \lambda_3 := \frac{\text{area}([x_{i_1}, x_{i_2}, x])}{\text{area}(\Delta)}, \tag{4.9}$$

$\lambda_i \geq 0, i = 1, 2, 3,\ \lambda_1 + \lambda_2 + \lambda_3 = 1$, are the barycentric coordinates of x with respect to the vertices of the triangle Δ (i.e., $x = \lambda_1 x_{i_1} + \lambda_2 x_{i_2} + \lambda_3 x_{i_3}$). Therefore, the parameterization of $\mathcal{S}$ is an isomorphism $\varphi : \mathcal{S} \rightarrow \mathcal{P} \subseteq \mathbb{R}^2$, with $\mathcal{P} = \mathcal{P}(\mathcal{G}, \Omega)$ parameterization domain, such that

$$\begin{aligned} \varphi|_X : \quad & X \rightarrow \Omega \subseteq \mathbb{R}^2 \\ & x_i \mapsto \varphi(x_i) := u_i \end{aligned} \tag{4.10}$$

and $\mathcal{P}$ is a valid triangulation (i.e., without foldovers). The map φ is extended from the set of vertices X to the surface $\mathcal{S}$ by using barycentric coordinates as in (4.8).

4.5.1 A family of parameterizations based on convex combinations

The method that we are going to present [Flo97] defines a parameterization of a surface triangulation $\mathcal{S}$ (i.e., 0-genus and one boundary component) through the solution of a linear system which requires that each internal vertex of $\mathcal{P}$ is a convex combination of its neighborhoods, while the boundary vertices are constrained to belong to a convex polygon. The choice of the coefficients of the convex combination provides several embeddings of $\mathcal{S}$ onto $\mathcal{P}$ (i.e., a family of parameterizations) whose effectiveness for applications such as surface approximation depends on the way $\mathcal{P}$ locally preserves the shape of $\mathcal{S}$ (see Section 4.5.3).

Let $\mathcal{S} = \mathcal{S}(\mathcal{G}, X)$, $\mathcal{G} = \mathcal{G}(V, E, F)$, be a planar surface triangulation with vertices $X := \{x_i : i = 1, \ldots, n\}$; without loss of generality, we assume that $X_I := \{x_i : i = 1, \ldots, \overline{n}\}$ are the internal vertices and $X_B := \{x_i : i = \overline{n}+1, \ldots, n\}$ are the $k := n-\overline{n}$ boundary vertices in any anti-clockwise order. The construction of the embedding φ in (4.10) is achieved as follows:

- choose $\{u_{\overline{n}+1}, \ldots, u_n\}$ to be the vertices of any k-sided convex polygon $\Omega \subseteq \mathbb{R}^2$ in an anti-clockwise sequence;
- for each internal node $i = 1, \ldots, \overline{n}$, choose any set of real numbers $\lambda_{i,j}$, $j = 1, \ldots, n$, such that
$$\lambda_{i,j} = 0, \quad (i,j) \notin E, \qquad \lambda_{i,j} > 0, \quad (i,j) \in E, \qquad \sum_{j=1}^{n} \lambda_{i,j} = 1; \tag{4.11}$$
- define $\{u_i : i = 1, \ldots, \overline{n}\}$ to be the solution of the linear system of equations
$$u_i = \sum_{j=1}^{n} \lambda_{i,j} u_j, \qquad i = 1, \ldots, \overline{n}; \tag{4.12}$$
- let $\mathcal{P} = \mathcal{P}(\mathcal{G}, \Omega_B, \Lambda)$, with $\Omega_B := \{u_{\overline{n}+1}, \ldots, u_n\}$ and $\Lambda := (\lambda_{i,j}) \in M_{\overline{n},n}(\mathbb{R})$, be the embedding of $\mathcal{G}$ in $\mathbb{R}^2$ with nodes $\{u_i : i = 1, \ldots, n\}$, straight lines for edges, and triangles for faces.

In the following, we present the main properties of the method proving that $\mathcal{P}$ is a parameterization of $\mathcal{S}$. First of all, we note that the relation (4.12) can be written as
$$u_i - \sum_{j \in N(i) \cap I} \lambda_{i,j} u_j = \sum_{j \in N(i) \cap B} \lambda_{i,j} u_j, \quad i = 1, \ldots, \overline{n}$$
with $N(i) := \{j \in V : (i,j) \in E\}$ 1-star of i in $\mathcal{G}$, $I := \{i : i = 1, \ldots, \overline{n}\}$, and $B := \{i : i = \overline{n} + 1, \ldots, n\}$, or as a matrix equation
$$Au = b \tag{4.13}$$
where $A := (a_{ij})_{i,j} \in M_{\overline{n},\overline{n}}(\mathbb{R})$ is defined by the relations
$$a_{ii} = 1, \quad a_{ij} = -\lambda_{i,j}, \quad i, j \in I, \quad i \neq j,$$
$u := (u_i)_{i=1,\ldots,\overline{n}} \in M_{\overline{n},2}(\mathbb{R})$, $b := \left(\sum_{j \in N(i) \cap B} \lambda_{i,j} u_j\right)_{i=1,\ldots,\overline{n}} \in M_{\overline{n},2}(\mathbb{R})$. As stated by Theorem 10, the existence and uniqueness of a solution to (4.12) is guaranteed by the non-singularity of the matrix A in (4.13), while the meaningfulness of the solution as parameterization of $\mathcal{S}$ depends on the geometric properties of the convex combinations [1].

Theorem 10. *For the parameterization problem (4.12) the following properties hold:*

- *the matrix A in (4.13) is non-singular;*

[1] If a point $p \in \mathbb{R}^2$ is a convex combination of s points $\{p_i : i = 1, \ldots, s\}$ and l is any line passing through p, then either all the points p_i belong to l or there is at least one point p_j on the other side of l.

- *every internal point u_i, $i = 1, \ldots, \overline{n}$, of $\mathcal{P}$ belongs to the convex polygon Ω;*
- *let $P(\mathcal{S})$ be the class of all parameterizations of the surface triangulation $\mathcal{S}$. Define $T(\mathcal{S}) \subseteq P(\mathcal{S})$ to be those parameterizations of the form (4.12) and let $C(\mathcal{S}) \subseteq P(\mathcal{S})$ be those parameterizations whose boundary nodes are the vertices of a convex polygon in an anti-clockwise sequence. Then, $T(\mathcal{S}) = C(\mathcal{S})$.*

Therefore, the convexity of the polygon Ω is of basic importance because it guarantees that φ is an isomorphism between the surface triangulation $\mathcal{S}$ and its parameterization $\mathcal{P}$, and it fully characterizes the set of mappings which embed $\mathcal{S}$ onto convex polygons of the plane. In [LKL02], the drawback of mapping the boundary of the 3D mesh onto a convex planar curve has been solved by using a multi-layered virtual boundary. Finally, the method of barycentric-coordinates has been extended in [GGS03] to the case of closed manifold triangle meshes with 0-genus, defining all its possible *spherical parameterizations*.

Since $\sum_{j=1}^{n} \lambda_{i,j} = 1$, $i = 1, \ldots, \overline{n}$, the coefficient matrix in (4.13) is sparse and diagonally dominant; therefore, the associated linear system can be solved in $O(\overline{n}^3)$ time with direct methods such as the QR and the LU factorization, or iteratively in $O(\overline{n})$ time by using the Bi-CGSTAB conjugate gradient [GL96]. Usually, iterative solvers are preferable to direct methods due to the growth of the dimension of the matrix A with the number of vertices of the input surface triangulation.

4.5.2 Weights choice and related approaches

By allowing each internal vertex $u_i \in \Omega$, $i = 1, \ldots, \overline{n}$, to be any convex combination of its neighboring vertices, the procedure provides all possible valid planar embeddings of the graph $\mathcal{G}$ associated to $\mathcal{S}$ onto a convex and planar domain. The choice of the weights $\lambda_{i,j}$ affects the distortion of the parameterization (see Section 4.5.3); therefore, they have to be chosen in order to minimize the triangle deformation with respect to the geometry of the 3D mesh. We now describe several methods for selecting the weights $\lambda_{i,j}$ and whose choice depends on the targeted applications; Figure 4.3 shows the different embeddings of a given surface $\mathcal{S}$.

Constant weights [Tut63]. If we consider the parameterization problem as the search of a planar embedding of a plane graph $\mathcal{G}$, it is natural to choose the coefficients of Λ as $\lambda_{i,j} := 1/d_i$, where d_i is the number of nodes incident in i, that is, $d_i = \sharp N(i)$. This choice is independent of the geometry X of $\mathcal{S}$ and it is only related to the connectivity of the graph $\mathcal{G}$.

Weighted least-square coefficients [Flo97]. Suppose the boundary vertices $u_{\overline{n}+1}, \ldots, u_n$ have been chosen and are fixed; then, we define the internal points $\{u_i : i = 1, \ldots, \overline{n}\}$ as the minimum of the convex functional $F : \mathbb{R}^{\overline{n} \times 2} \rightarrow \mathbb{R}$:

$$F(u_1, \ldots, u_{\overline{n}}) := \sum_{(i,j) \in E} w_{i,j} \|u_i - u_j\|_2^2, \tag{4.14}$$

where the weights $w_{i,j} = w_{j,i} > 0$ are symmetric for all $(i,j) = (j,i) \in E$. If $u_i := (u_{i,x}, u_{i,y}) \in \mathbb{R}^2$, the global minimum of (4.14) is attained when $(\partial_{u_{i,x}} F, \partial_{u_{i,y}} F) = 0$ for all $i = 1, \ldots, \overline{n}$, that is, at the points

$$u_i = \sum_{j:(i,j)\in E} \frac{w_{i,j}}{\sum_{j:(i,j)\in E} w_{i,j}} u_j, \quad i = 1, \ldots, \overline{n}. \tag{4.15}$$

Therefore, we have that the previous equation is equivalent to solving (4.12) with

$$\lambda_{i,j} := \frac{w_{i,j}}{\sum_{j:(i,j)\in E} w_{i,j}},$$

and the weighted least-squares coefficients are defined proportional to the edge lengths of $\mathcal{S}$ as

$$w_{i,j} := \|x_i - x_j\|_2^{-q}, \quad q \in \mathbb{N}.$$

Even thought these weights take into account the geometry of the input surface triangulation $\mathcal{S}$, they present some oscillations probably due to the constraints of having symmetric weights $w_{i,j} = w_{j,i}$, $(i,j) \in E$, [Flo97].

Shape-preserving weights [Flo97]. The search of the coefficients $\lambda_{i,j}$ in (4.11) is also related to the problem of finding the barycentric coordinates as defined below.

Definition 20. *Let $\mathcal{N} := (v_1, \ldots, v_k)$ be k points in the plane arranged in an anti-clockwise ordering around v_0. The coefficients $(\lambda_1, \ldots, \lambda_k)$ are* barycentric coordinates *of v_0 with respect to $\mathcal{N}$ if*

$$\begin{cases} \lambda_1 > 0, \ldots, \lambda_k > 0, \\ \sum_{i=1}^k \lambda_i = 1, \\ \sum_{i=1}^k \lambda_i v_i = v_0. \end{cases} \tag{4.16}$$

In the simplest case $k = 3$ (see also (4.8) and (4.9)), the weights $\lambda_1, \lambda_2, \lambda_3$ are the barycentric coordinates of v_0 with respect to the triangle $\Delta := [v_1, v_2, v_3]$ and they are uniquely determined by the previous conditions (4.16). If $k > 3$, the problem (4.16) has not a unique solution; in fact, since v_0 belongs to the convex hull of $\mathcal{N}$ there must exist at least one triangle $\Delta := [v_{i_1}, v_{i_2}, v_{i_3}]$ such that $v_0 \in \Delta$ and therefore we can choose as coordinates

$$\lambda_i := \begin{cases} \lambda_{i_l} & i = i_l,\ l = 1,2,3, \\ 0 & \text{else.} \end{cases}$$

However, these coordinates depend randomly on the choice of the triangle Δ which contains v_0. In order to supply to this drawback, the shape-preserving coefficients related to the node i are calculated by averaging the barycentric coordinates of u_i over certain covering triangles belonging to the local conformal parameterization of the 1-neighborhood of x_i. These coefficients depend continuously on $\{x_j : j \in \{i\} \cup N(i)\}$ and, as stated by Proposition 10, their main feature is the *reproduction property* of the parameterization with respect to affine transformations.

Proposition 10 (Reproduction property)**.** *Suppose that $\mathcal{S}$ is a planar surface triangulation and that its boundary vertices $x_{\overline{n}+1}, \ldots, x_n$ form a convex polygon Ω containing $\mathcal{S}$. Let $\mathcal{P}(\mathcal{G}, \Omega_B, \Lambda)$ be the shape-preserving parameterization and such that the boundary vertices $x_{\overline{n}+1}, \ldots, x_n$ are mapped affinely into $u_{\overline{n}+1}, \ldots, u_n$. Then, the parameterization $\mathcal{P}$ is an affine mapping of $\mathcal{S}$.*

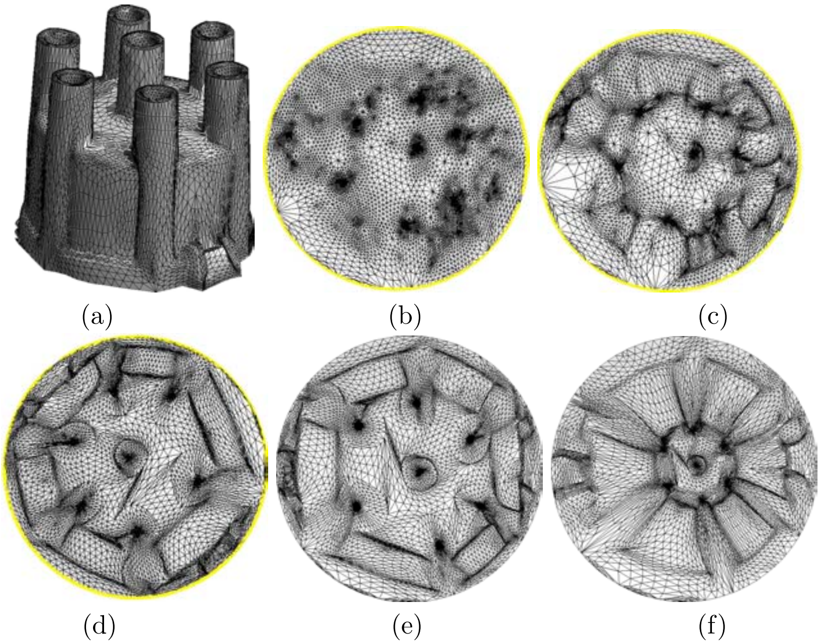


Figure 4.3: (a) Input surface triangulation, parameterization with (b) constant, (c) least-squares, (d) shape-preserving, (e) conformal, and (f) harmonic weights.

Mean-value weights [Flo03]. The Mean-Value Theorem 11 [Rud70] enables to derive new (barycentric) coordinates of v_0 with respect to $\mathcal{N}$ which depend infinitely smoothly on the data points $v_0, v_1, \ldots, v_k$ as summarized by Proposition 11.

Theorem 11 (Mean-value Theorem)**.** *Let $f : \Omega \to \mathbb{R}$ be a harmonic function on an open and bounded set Ω in $\mathbb{R}^2$ (i.e., $\Delta f = 0$ in Ω). Then, for a disc $B(v_0, r) \subseteq \Omega$ with boundary Γ*

$$f(v_0) = \frac{1}{2\pi r} \int_\Gamma f(v) ds.$$

Proposition 11 (Mean-value coordinates)**.** *Let α_i, $0 < \alpha_i < \pi$, be the angle at v_0 in the triangle $\Delta := [v_0, v_i, v_{i+1}]$ defined cyclically (see Figure 4.4(a)). Then,*

- *the weights*

$$\lambda_i := \frac{w_i}{\sum_{j=1}^k w_j}, \quad w_i := \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{\|v_i - v_0\|_2}, \tag{4.17}$$

are barycentric coordinates for v_0 with respect to $v_1, \ldots, v_k$;

- *each weight λ_i is bounded*

$$\frac{1}{Ck} < \lambda_i < \frac{C}{k},$$

where

$$C := \frac{L^\star \tan(\alpha^\star/2)}{L_\star \tan(\alpha_\star/2)} \geq 1,$$

with $L^\star := \max_i\{\|v_i - v_0\|_2\}$, $L_\star := \min_i\{\|v_i - v_0\|_2\}$, $\alpha^\star := \max_i\{\alpha_i\}$, and $\alpha_\star := \min_i\{\alpha_i\}$.

- *Let $\mathcal{S} = \mathcal{S}(\mathcal{G}, X)$ be a planar surface triangulation of a domain Ω. If $f_{\mathcal{G}} : \Omega \rightarrow \mathbb{R}$ is a piecewise linear function which satisfies the mean value theorem for each interior vertex v_0, and $r > 0$ is a suitable radius such that Γ belongs to the convex hull of $\mathcal{N}$, then $f_{\mathcal{G}}(v_0)$ is given by the convex combination*

$$f_{\mathcal{G}}(v_0) = \sum_{i=1}^{k} \lambda_i f_{\mathcal{G}}(v_i)$$

with the weights λ_i in (4.17).

Given a planar surface triangulation $\mathcal{S}$ and an internal node $i \in V$, we can consider its 1-star $N(i) := \{j_1, \ldots, j_k\}$; if $\mathcal{S}$ is a 2D surface (i.e., $X \subseteq \mathbb{R}^2$), from the previous discussion it follows that the problem (4.12) is equivalent to searching the coordinates $(\lambda_{j_l})_{l=1}^{k}$ of x_i with respect to $\{x_{j_l}\}_{l=1}^{k}$. If $\mathcal{S}$ is a 3D surface, the use of the mean-value weights locally mimics the planar case.

Harmonic weights [EDD$^+$95, PP93]. We now describe a parameterization method which is based on the discretization of harmonic maps using the Finite Element Methods [QA94]; then, we analyze its relations with the convex combination maps. Given a surface $\mathcal{S}$ with regular parametric representation $\varphi : U \rightarrow \mathcal{S}$, the method consists of the following steps:

- fix the boundary condition, i.e. $\varphi|_{\partial\mathcal{S}} = \varphi_0$, by mapping the polygon boundary $\partial\mathcal{S}$ homeomorphically to some polygon in the plane;
- find the function $\varphi \in S_2(\mathcal{S})$ which minimizes the Dirichlet energy

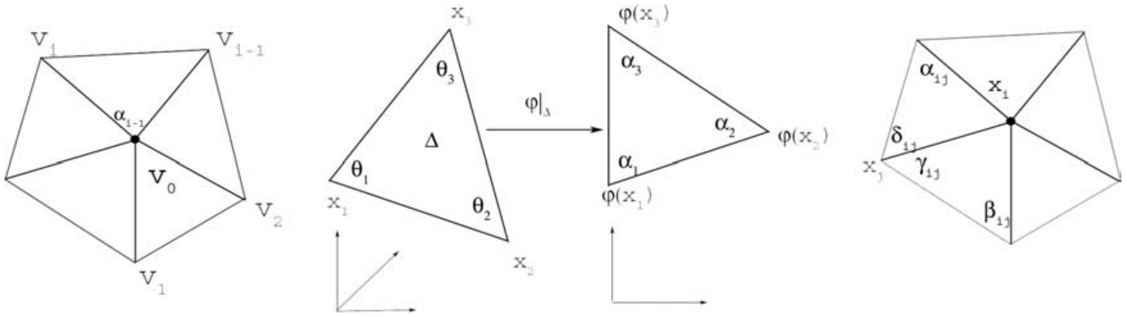
$$E_D(\varphi) := \frac{1}{2} \iint_U \|\nabla\varphi\|_2^2 dudv$$

with boundary condition $\varphi|_{\partial\mathcal{S}} = \varphi_0$.

From equation (4.4), it follows that

$$\begin{aligned} A(\mathcal{S}) = \iint_U \sqrt{EG - F^2} dudv = \iint_U \sqrt{\|\varphi_u\|_2^2 \|\varphi_v\|_2^2 - | < \varphi_u, \varphi_v > |^2} dudv = \\ \iint_U \|\varphi_u \wedge \varphi_v\|_2 dudv \leq^{(a)} \iint_U \|\varphi_u\|_2 \|\varphi_v\|_2 dudv \leq^{(b)} \\ \frac{1}{2} \iint_U \left(\|\varphi_u\|_2^2 + \|\varphi_v\|_2^2\right) dudv = \frac{1}{2} \iint_U \|\nabla\varphi\|_2^2 dudv, \end{aligned}$$

where the last integral is the Dirichlet energy of $\mathcal{S}$ achieved by using its parametric representation φ over U. The inequality (a) becomes an equality if and only if $< \varphi_u, \varphi_v >= 0$, while (b) does iff $\|\varphi_u\|_2 = \|\varphi_v\|_2$. Furthermore, the minimum of the Dirichlet energy is the area of $\mathcal{S}$ and it is attained only for conformal mappings, i.e. when the two conditions on φ previously introduced hold.



(a) (b) (c)

Figure 4.4: (a) 1-star of v_0, (b) corresponding triangles on the input triangle mesh and the parameterization domain, (c) angles related to the triangles belonging to the 1-neighborhood of x_i.

Let us now consider the discrete case where $\mathcal{S}$ is a surface triangulation and φ is a piecewise linear function. If $\Delta := [x_1, x_2, x_3]$ is a triangle in $\mathcal{S}$, it is possible to prove that [DMA01, EDD$^+$95]

$$\int_\Delta \|\nabla\varphi\|_2^2 dx = \frac{1}{2}\left[\cot(\theta_3)\|\varphi(x_1)-\varphi(x_2)\|_2^2 + \cot(\theta_2)\|\varphi(x_1)-\varphi(x_3)\|_2^2 + \cot(\theta_1)\|\varphi(x_2)-\varphi(x_3)\|_2^2\right], \tag{4.18}$$

where the angles θ_i, $i = 1, 2, 3$, are shown in Figure 4.4(b). Therefore, using the additivity of the integral, the symmetry of the expression $\|\varphi(x_i) - \varphi(x_j)\|_2$ with respect to $(i, j) \in E$, and differentiating $E_D(\varphi)$ with respect to the unknowns $\varphi(x_i)$, $i = 1, \ldots, \overline{n}$, the minimum of the convex functional E_D is the solution of the linear system of equations:

$$\sum_{j\in N(i)} w_{i,j}(\varphi(x_i) - \varphi(x_j)) = 0, \quad i, \ldots, \overline{n}, \tag{4.19}$$

where (see Figure 4.4(c))

$$w_{i,j} := \cot(\alpha_{ij}) + \cot(\beta_{ij}) = \frac{\sin(\alpha_{ij} + \beta_{ij})}{\sin(\alpha_{ij})\sin(\beta_{ij})}. \tag{4.20}$$

Since the coefficient matrix (4.20) associated to the equations (4.19) is symmetric and positive definite, the system admits a unique solution $(\varphi(x_i))_{i=1,\ldots,\overline{n}}$. Expression (4.19) is equivalent to (4.15) but the weights (4.20) might be negative; in fact,

$$w_{i,j} > 0 \quad \Longleftrightarrow \quad (\alpha_{ij} + \beta_{ij}) < \pi.$$

This condition is fulfilled when all angles of $\mathcal{S}$ are less than $\pi/2$; if it is not satisfied, the discrete harmonic map may not be injective and some triangles may have the wrong orientation. Proposition 12 gives the link between the injectivity of piecewise linear harmonic functions and convex combination maps for the parameterization problem.

Proposition 12. *A discrete harmonic map $\varphi : \mathcal{S} \to \mathcal{P}$ solution of (4.19) is injective if it maps $\partial\mathcal{S}$ homeomorphically into a convex polygon $\partial\mathcal{P}$ and if the sum of every pair of opposite angles in $\mathcal{S}$ is less than π. In this case, φ is a convex combination map.*

We explicitly underline the analogy between the above discussion and the definition of *harmonic scalar fields* over arbitrary surface triangulations given in Section 3.2.2. In fact, equation (3.2) at page 75 is equivalent to (4.19) with the difference that there we deal with an arbitrary manifold $\mathcal{M}$, and the constraints on f are given on the set $\{x_i : i \in C\}$ which not necessarily coincides with the boundary of $\mathcal{M}$ (if any).

4.5.3 Parameterization distortion

From Proposition 9, it follows that it is possible to measure the "quality" of a given parameterization $\varphi : \mathcal{S} \rightarrow \mathcal{P} \subseteq \mathbb{R}^2$ by analyzing if φ preserves intrinsic properties such as the arc length, angles, and area. This aim is achieved by studying the properties of the matrix T in (4.2) associated to the first fundamental form $I = \|d\varphi\|_2^2$ of φ (see Section 4.3); the further T deviates from the identity matrix, the more the intrinsic properties of $\mathcal{S}$ are distorted when it is mapped onto $\mathcal{P}$. We specialize the discussion given in Section 4.2 to the case of a triangle mesh; then, new embeddings of $\mathcal{S}$ onto a planar domain are defined by different techniques which aim at producing least-distorted parameterizations, and vary by the distortion functionals they deal with.

Let $\mathcal{S} = \mathcal{S}(\mathcal{G}, X)$ [2] be a planar surface triangulation and $\varphi : \mathcal{S} \rightarrow \mathcal{P} \subseteq \mathbb{R}^2$, $\varphi \in S_2(\mathcal{S})$, $\mathcal{P} = \mathcal{P}(\mathcal{G}, \Omega)$, be a parameterization of $\mathcal{S}$. The distortion of φ over $\mathcal{S}$ is defined by averaging the stretch between each triangle $\Delta \in \mathcal{S}$ and its image $\varphi(\Delta) \in \mathcal{P}$ [FH02]. Since φ is a piecewise linear mapping, by introducing an orthonormal coordinate system over Δ with the third axis perpendicular to Δ, equation (4.8) can be written as $\varphi|_\Delta = f|_\Delta$ where

$$f(x) = Fx + c, \quad x \in \mathbb{R}^2 \tag{4.21}$$

with $F \in \mathbb{R}^{2\times 2}$ and $c \in \mathbb{R}^2$. Using (4.21), we want to express the shape deformation of Δ in terms of the matrices (F, c); to this end, we set

$$\Pi_1 := \{f : \mathbb{R}^2 \rightarrow \mathbb{R}^2 : \quad f(x) = Lx + m,\, L \in \mathbb{R}^{2\times 2},\, m \in \mathbb{R}^2\}$$

and $\pi \in \Pi_1$. Then, for all $f \in \Pi_1$ we have that $(\pi \circ f) \in \Pi_1$. A functional $E : \Pi_1 \rightarrow \mathbb{R}$ is defined *invariant with respect to* π if $E(\pi \circ f) = E(f)$, $\forall f \in \Pi_1$, and E is a *deformation functional* if it is invariant with respect to affine transformations (i.e., $\pi(x) := Qx + r$ with Q orthogonal matrix). The deformation functionals are characterized by Proposition 13.

Proposition 13. *Let $E : \Pi_1 \rightarrow \mathbb{R}$ be a deformation functional and f be as in (4.21). Then, $E(f)$ only depends on the ratio $\rho := \sigma_1/\sigma_2$ of the singular values $\sigma_1 \geq \sigma_2 \geq 0$ of F. In particular, E can be expressed as a function from $\{\rho \in \mathbb{R} : \rho \geq 1\}$ to $\mathbb{R}$.*

We note that if

- $\rho = 1$, the singular values of F are proportional and φ does not distort the shape of the triangle Δ;

[2] The definition of parameterization distortion is valid for any triangulated manifold without assumptions on its genus and number of boundary components. For more details, we refer the reader to Section 4.8.

- $\rho \rightarrow +\infty$, the shape of Δ is increasingly distorted and Δ is collapsed to lines when $\sigma_2 = 0$.

Then, if E is a deformation functional we can define the *global deformation functional* as the average of the values of E on $\varphi|_\Delta$ for each triangle Δ in $\mathcal{S}$; that is,

$$D(\varphi) := \sum_{\Delta \in \mathcal{S}} E(\varphi|_\Delta). \tag{4.22}$$

The functional D in (4.22) satisfies the following properties:

1. *rotation and translation invariance*: since these affine transformations do not change the surface geometry, the functional D is not affected by rotations and translations;
2. *continuity*: the deformation functional E used to define D converges to a continuous measure when we provide finer triangulations of $\mathcal{S}$; then,

   $$D(\varphi_{\mathcal{S}_k}) \rightarrow D(\varphi_{\mathcal{S}})$$

   where $\varphi_{\mathcal{S}_k}$ is the parameterization of $\mathcal{S}_k$ and $\lim_{k \rightarrow +\infty} \mathcal{S}_k = \mathcal{S}$ in $\mathbb{R}^3$;
3. *additivity*: the functional D does not depend on the sampling of the input surface, that is,

   $$D(\varphi_{\mathcal{S}_1 \cup \mathcal{S}_2}) + D(\varphi_{\mathcal{S}_1 \cap \mathcal{S}_2}) = D(\varphi_{\mathcal{S}_1}) + D(\varphi_{\mathcal{S}_2})$$

   for all $\mathcal{S}_1$ and $\mathcal{S}_2$ triangle meshes.

We explicitly underline that the previous properties can be defined on $\mathcal{S}$ instead of its parameterization φ; furthermore, the area, the Euler characteristic, and the perimeter of the boundary of $\mathcal{S}$ fulfill the previous conditions.

Definition 21. *A* proper deformation functional *is a deformation functional E which is monotonically increasing as a function of ρ.*

As suggested in [FH02] and stated by Proposition 14, several proper deformation functionals can be defined by considering the condition number $\kappa(F) := \|F\|\|F^{-1}\|$ of $F \in Gl_2(\mathbb{R})$ with respect to a matrix norm $\| \,\|$ [3] [GL96].

Proposition 14. *The condition numbers κ_2 and $\kappa_{Frob.}$ of the matrix F in (4.21) and of its first fundamental form $F^T F$ are proper shape deformation functionals. Furthermore, the following relations hold:*

- $E_1(f) := \kappa_2(F) = \rho$;

[3] A norm $\| \,\|$ in $\mathbb{R}^n$ induces a norm in $M_{n,n}(\mathbb{R})$ defined as

$$\|A\| := \sup_{x \neq 0} \left\{ \frac{\|Ax\|}{\|x\|} \right\}, \qquad A := (a_{ij}) \in M_{n,n}(\mathbb{R}).$$

The *Frobenius norm* of A is defined as $\|A\|_{Frob.} := \sqrt{\sum_{i,j} |a_{ij}|^2}$.

- $E_2(f) := \kappa_{Frob.}(F) = \rho + \frac{1}{\rho}$;
- $E_3(I) := \kappa_2(F^T F) = \rho^2$;
- $E_4(I) := \kappa_{Frob.}(F^T F) = \rho^2 + \frac{1}{\rho^2}$.

An alternative estimation of the surface stretch [SSGH01] which will be used for comparing different cuts on a given surface $\mathcal{S}$ (see Section 4.7.1) is defined by the functionals

$$L^2(\varphi) := \sqrt{\frac{\sum_{\Delta\in\mathcal{S}} A(\Delta)\left(L^2(\Delta)\right)^2}{\sum_{\Delta\in\mathcal{S}} A(\Delta)}}, \quad L^\infty(\varphi) := \max_{\Delta\in\mathcal{S}}\{L^\infty(\Delta)\}, \tag{4.23}$$

with $L^2(\Delta) := \sqrt{(\sigma_1+\sigma_2)/2}$, $L^\infty(\Delta) := \sigma_1$, and $A(\Delta)$ area of the triangle Δ.

Given a surface $\mathcal{S}$, it is possible to define an isometry φ on $\mathcal{S}$ if it is *developable* [4]; if $\mathcal{S}$ is a piecewise linear surface, it is developable if the sum of the angles incident to each interior vertex of $\mathcal{S}$ is 2π. Usually, 3D surface triangulations do not satisfy this last condition and the common approach to the search of a parameterization of $\mathcal{S}$ consists of finding φ which minimizes the parameterization distortion and which is as close as possible to an isometry, or to a conformal or area-preserving mapping. In the following paragraphs, we describe linear and quadratic methods that accomplish this strategy and that have been defined during the last years.

Discrete isometric parameterization. If we consider one of the previous deformation functionals E_i, $i = 1,\dots,4$, the corresponding functional D in (4.22) is not quadratic as it happens in (4.14) or for the discretization of harmonic maps (4.18). This means that if we want to select as φ the minimum of D we have to solve a non-linear problem; for example [HG99], the proper shape deformation functional E_2 is given by

$$E_2(\varphi|_\Delta) = \frac{\cot(\theta_1)\|\varphi(x_3)-\varphi(x_2)\|_2^2 + \cot(\theta_2)\|\varphi(x_3)-\varphi(x_1)\|_2^2 + \cot(\theta_3)\|\varphi(x_2)-\varphi(x_1)\|_2^2}{<\varphi(x_3)-\varphi(x_1), \varphi(x_2)-\varphi(x_1)>}$$

[4] We introduce developable surfaces as a special case of ruled surfaces. A *ruled surface* $\mathcal{R}$ carries a one parameter family of straight lines $\mathcal{L}$; these lines are called *generators*. The general parameterization of a ruled surface $\mathcal{R}$ is

$$x(u,v) := c(u) + ve(u)$$

where $c(u)$ is called *directrix curve* and $e(u)$ is a vector field along $c(u)$. For fixed values u, this parameterization represents the straight lines $\mathcal{L}(u)$ on $\mathcal{R}$. The normal vector $n(u,v)$ of the ruled surface $x(u,v)$ is computed as cross product of the partial derivative vectors x_u and x_v, and we have that

$$n(u,v) = c^{'}(u) \wedge e(u) + ve^{'}(u) \wedge e(u).$$

For fixed $u = u_0$, the normal vectors $n(u_0,v)$ along $\mathcal{L}(u_0)$ are linear combinations of the vectors $c^{'}(u_0)\wedge e(u_0)$ and $e^{'}(u_0)\wedge e(u_0)$. The parameterization $x(u,v)$ represents a *developable surface* $\mathcal{D}$ if for each generator $\mathcal{L}$ all points $x \in \mathcal{L}$ have the same tangent plane (with the exception of the singular points on $\mathcal{L}$). Any regular generator $\mathcal{L}(u)$ of a developable surface $\mathcal{D}$ carries a unique singular point $s(u)$, and $s(u) = x(u,v_s)$ is determined by the parameter value

$$v_s := -\frac{<c^{'}\wedge e, e^{'}\wedge e>}{\|e^{'}\wedge e\|_2}.$$

If e and $e^{'}$ are linear independent, the singular point is at infinity; otherwise, it is a proper point. In $\mathbb{R}^3$, there exist three basic classes of developable surfaces: cylinders, cones, and surfaces consisting of the tangent lines of a regular space curve $s(u)$ which is the singular curve of the surface. For more details on the analysis of developable surfaces, we refer the reader to [Pet04].

where the angles $\{\theta_i\}_{i=1,2,3}$ of the triangle $\Delta := [x_1, x_2, x_3]$ are shown in Figure 4.4(b). In this case, it is not necessary to fix the boundary of the parameterization domain which evolves according to the optimization process. The global deformation functional associated to E_2 has twice the number of triangles in $\mathcal{S}$ as upper bound and it is achieved by choosing $E_2(\varphi|_\Delta) = 2$, that is, when φ is conformal. Since a conformal and piecewise linear function φ is an isometry, minimizing this functional gives a parameterization which is as isometric as possible.

Discrete conformal parameterization. From (4.21), it follows that we can consider intrinsic properties for piecewise linear mappings $\varphi|_\Delta \in \mathbb{P}^1$, $\Delta \in \mathcal{S}$. To this end, let $g : \mathbb{R}^2 \to \mathbb{R}^2$, $g(x, y) := (u(x, y), v(x, y))$ be an arbitrary function; therefore, we have that the Jacobian matrix of g is given by

$$J := \begin{pmatrix} u_x & u_y \\ v_x & v_y \end{pmatrix},$$

and the matrix T in (4.2) associated to the first fundamental form I is

$$T = J^T J = \begin{pmatrix} u_x^2 + v_x^2 & u_x v_y - u_y v_x \\ u_x v_y - u_y v_x & u_y^2 + v_y^2 \end{pmatrix}; \tag{4.24}$$

if λ_i, $i = 1, 2$, $\lambda_1 \geq \lambda_2$, are the eigenvalues of T, and σ_i, $i = 1, 2$, $\sigma_1 \geq \sigma_2$, the singular values of J, we have that $\lambda_i = \sigma_i^2$, $i = 1, 2$.

Using (4.24), we can write

$$\text{trace}(T) = \text{trace}(J^T J) = \begin{cases} u_x^2 + u_y^2 + v_x^2 + v_y^2, \\ \lambda_1 + \lambda_2 = \sigma_1^2 + \sigma_2^2, \end{cases}$$

$$det(J) = \begin{cases} u_x v_y - u_y v_x, \\ \sigma_1 \sigma_2, \end{cases}$$

and therefore

$$E_D(g) = \frac{1}{2} \iint_U \|\nabla g\|_2^2 dxdy = \frac{1}{2} \iint_U \left(\|\nabla u\|_2^2 + \|\nabla v\|_2^2\right) dxdy = \frac{1}{2}(\sigma_1^2 + \sigma_2^2) A(U), \tag{4.25}$$

and

$$\sigma_1 \sigma_2 = \frac{A(g(U))}{A(U)} \tag{4.26}$$

with $A(U)$ the area of U. Following [DMA01], we introduce the quadratic area-energy related to $u_i \in \mathcal{P}$ as

$$A(u_i) := \sum_{j \in N(i)} \mu_{ij} \|u_i - u_j\|_2^2, \quad \mu_{ij} := \frac{\cot(\gamma_{ij}) + \cot(\delta_{ij})}{\|x_i - x_j\|_2^2}$$

where the angles γ_{ij} and δ_{ij} are depicted in Figure 4.4(c). Since a conformal mapping $g : \mathbb{R}^2 \to \mathbb{R}^2$ satisfies the Cauchy-Reimann conditions $u_x = v_y$, $u_y = -v_x$, these last relations can be used for defining the *conformal energy* of g as

$$E_C(g) := \frac{1}{2} \left[(u_x - v_y)^2 + (u_y + v_x)^2 \right].$$

Then, an almost conformal parameterization $\varphi : \mathcal{S} \to \mathcal{P}$ is achieved by minimizing the global conformal functional

$$E_C(\varphi) := \sum_{\Delta \in \mathcal{S}} A(\Delta) E_C(\varphi|_\Delta)$$

where the restriction $\varphi|_\Delta$ of φ to Δ has to be intended as in (4.21). From (4.25) and (4.26), it follows that $\forall \Delta \in \mathcal{S}$

$$E_C(\varphi|_\Delta) = \frac{1}{2}(\sigma_1 - \sigma_2)^2$$

and

$$E_C(\varphi) = E_D(\varphi) - A(\varphi)$$

which shows that the minimization problem is quadratic and the solution is achieved by solving the linear system

$$\sum_{j \in N(i)} (w_{i,j} - \mu_{ij}) u_j = u_i, \quad i = 1, \dots, \overline{n}. \tag{4.27}$$

In [DMA01], E_C is generalized by considering a linear combination of E_D and A, that is,

$$\tilde{E}_C(\varphi) := \alpha E_D(\varphi) + \beta A(\varphi)$$

where α and β are two arbitrary real constants. In both cases, the *intrinsic parameterization* which solves (4.27) is invariant to rotation and translation, and it minimizes the distortion defined as the linear combination of the Dirichlet [PP93] and area-energy (also called χ-energy).

An alternative approach to the search of a conformal parameterization consists of defining an angle-based flattening which minimizes the distortion of the planar angles with respect to their 3D counterparts [SdS00]. To each node $i \in V$ we assign the sum of the angles around its corresponding vertex $x_i \in \mathcal{S}$ and $u_i \in \mathcal{P}$; that is,

$$\theta^{(i)} := \sum_{j \in N(i)} \theta_j, \quad \alpha^{(i)} := \sum_{j \in N(i)} \alpha_j = 2\pi.$$

Then, we define the set of weights on the vertices as

$$s(i) := \begin{cases} \frac{2\pi}{\theta^{(i)}} & i \in I, \\ 1 & i \in B, \end{cases}$$

and the angles of the triangles in $\mathcal{P}$ are defined as the minimum of the quadratic functional

$$E(\alpha_1, \dots, \alpha_n) := \sum_{i \in V} \left[\frac{\alpha_i}{\beta_i} - 1 \right]^2, \quad \beta_i := \theta_i s(i). \tag{4.28}$$

The minimum is attained if and only if $\mathcal{S}$ is developable with $\theta^{(i)} = 2\pi$, $i \in I$, and in this case the embedding of $\mathcal{S}$ onto $\mathcal{P}$ is conformal with $\alpha_i = \beta_i = \theta_i$, $i \in I$. Therefore, the solution of (4.28) is used for constructing the corresponding piecewise linear mapping φ which is the most conformal parameterization of $\mathcal{S}$. However, for achieving a good parameterization the proposed approach requires to insert a set of additional constraints on the solution which converts the problem to a non-linear one.

Discrete area-preserving parameterization. As stated by Proposition 9, the condition that φ has to satisfy in order to be a conformal mapping is weak and this implies that is possible to build several area-preserving mappings on a given surface $\mathcal{S}$. As a consequence, minimizing area-based functionals such as

$$E_A(\varphi) := \sum_{\Delta \in \mathcal{S}} \left[A(\varphi(\Delta)) - A(\Delta)\right]^2 \tag{4.29}$$

gives an ill-posed problem due to the non-uniqueness of the solution of (4.29) which is no longer quadratic in its unknowns.

In the following sections, we consider the local and global parameterization of a triangle mesh $\mathcal{M}$ with an arbitrary genus; therefore, we will use and generalize the previous discussion where we assumed that $\mathcal{S}$ was a planar surface triangulation (i.e., 0-genus and one boundary component).

4.6 Segmentation methods for local parameterization

Local parameterization methods build a segmentation of a surface triangle mesh into patches homeomorphic to a disc, and then map each of them into the plane by a local convex-combination map. The segmentation of a surface $\mathcal{M}$ into a family of disc-like regions uses algorithms for mesh decimation [KCS98], e.g. vertex-removal or half-edge-collapse, where the simplified triangular mesh $\mathcal{M}_S \subseteq \mathcal{M}$ approximates the shape of $\mathcal{M}$, sharing the same genus, and provides a correspondence among the points of $\mathcal{M}$ and $\mathcal{M}_S$ (see Figure 4.5). The number of patches is equal to the number of triangles in $\mathcal{M}_S$, and the simplification is guided by a set of criteria and runs until the coarse mesh is as simple as possible without violating quality measures, which aim to reduce the distortion of the embedding. In general, an over-segmentation can create distortions along the boundaries of adjacent regions which affect the texture mapping and remeshing results in spite of local adjustments (e.g., smoothing, re-sampling, constrained boundaries). The decomposition of the input manifold into a set of arbitrarily-shaped patches enables to remesh each chart with a regular subdivision, which cannot be extended to the entire surface due to the induced irregular covering of $\mathcal{M}$; therefore, they achieve only a *semi-regular* remeshing. The properties of the segmentation usually depend on the application of the parameterization: for example, for remeshing or morphing purposes, triangle-shaped or quadrilateral-shaped patches are often used to build the base mesh, as they well support subdivision schemes and spline-based representations [FHR02, LSS$^+$98, IBMJ04]. The main problems to be solved are the smoothness of the mapping across the patch boundaries, the control over the shape, and the number of patches which, for complex shapes, may produce a large number of small charts with simple borders.

We approach the search of an extension of simplification-based methods for the local parameterization by discussing surface decomposition from a general perspective; to this end, we consider the class of methods $\mathcal{J}$ which subdivide $\mathcal{M}$ into disjoint regions of 0-genus, i.e. a family of patches $\{R_i\}_{i=1,\ldots,m}$ such that:

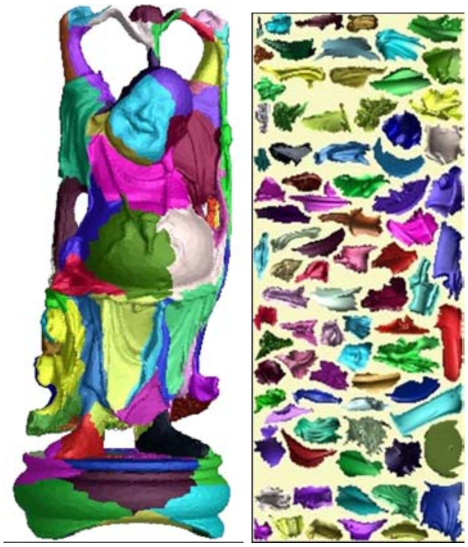


Figure 4.5: Input model of genus six and decomposition into disc-like charts as shown in [SWG$^+$03].

1. $\bigcup_{i=1}^{m} R_i = \mathcal{M}$;
2. R_i is a connected region, $i = 1, \ldots, m$;
3. $R_i^\circ \cap R_j^\circ = \emptyset, i \neq j$, with X° internal part of X;
4. $R_i \neq \emptyset$ has 0-genus and k_i boundary components $\{\gamma_j\}_{j=1}^{k_i}$.

Each primitive $\{R_i\}_i$ has 0-genus but a different number of boundary components; more precisely, a *cone* has one boundary, a *cylinder* has two boundary components, and finally a *body* has k_i loops, with $k_i \geq 3$. We now review different methods which belong to $\mathcal{J}$ and we show how the use of these primitives enables to segment $\mathcal{M}$ in a natural way, both in terms of the geometry of the primitives and their adjacency relations.

4.6.1 Topological segmentation based on the Extended Reeb equivalence

Our approach to the topology-based segmentation builds on the shape decomposition method discussed in Chapter 3 and it uses the concept of quotient space introduced by Reeb [Ree46, Mil63] to classify manifolds under the action of a real smooth function f. The function f induces a segmentation of the manifold into cells, or primitives, related to its critical points. Each cell has 0-genus and only three types of patches are possible, according to the number of their boundary components: *conical primitives* have one boundary, *cylinders* have two, and *bodies* have three or more boundary components. More precisely according to the description given in Section 3.2.1, we define the following patches:

- a *cone* primitive is defined for each terminal node of the *Extended Reeb Graph* (ERG), by merging the regions associated to the terminal node and all the normal regions associated to the arc connecting the terminal node to its adjacent one. Therefore, a cone primitive has only one boundary component by definition, it identifies a critical region containing either a maximum or minimum, and it covers the portion of the shape until it meets the next critical region;
- a *cylinder* primitive is related to each arc connecting two internal nodes n_1 and n_2 of the ERG, by merging all the regions associated to the arc connecting n_1, n_2. This primitive has two boundary components, and it either corresponds to a degenerate maximum or minimum (e.g., a volcano rim) or it simply corresponds to the part of the mesh that connects two critical regions;
- a *body* primitive is defined for each critical region associated to an inner node, containing a saddle point of the ERG. Therefore, a body primitive may have k boundary loops, with $k \geq 3$, and it corresponds to a critical region containing a saddle point.

The most important point is that each primitive $\{R_i\}_{i=1,\dots,m}$ has 0-genus, but a different number of boundary components. The correctness of the chosen primitives, i.e. their topological equivalence, is based on the fact that the topological type of a patch R_i depends only on the number of its boundary components and on the genus of the surface $R_i^\star$ obtained by gluing a disc to each loop $\gamma_j, j = 1, \dots, k_i$. The general case is detailed by the following theorem [Mas67] (see also Theorem 5 at page 14).

Theorem 12 (Classification of compact surfaces)**.** *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be two compact bordered surfaces; assume that their boundaries have the same number of components. Then, $\mathcal{M}_1$ and $\mathcal{M}_2$ are homeomorphic if and only if the surfaces $\mathcal{M}_1^\star$ and $\mathcal{M}_2^\star$, obtained by gluing a disc to each boundary component, are homeomorphic.*

While the properties (1-4) at page 109 are completely independent of the chosen mapping function f, different f functions generate different decompositions and this allows to adapt the decomposition to the characteristics of the specific shape (see Section 3.2.1). The decomposition method as well as the parameterization of the primitives (see Section 4.7) is fully automatic. The user may, however, control the process in three different stages: the choice of f, the granularity of the decomposition, and the optimization criteria for the parameterization of each primitive. The use of the Reeb quotient space ensures the minimal number of charts with respect to the number of critical points of $(\mathcal{M}, f)$; however, the user could further segment the primitives of $\mathcal{M}$ to produce a finer segmentation. Finally, the user could select any other criteria for optimizing the parameterization, even a different one for each cell, as the zippering of the parameterization is highly simplified by the connectivity of the patches. The generality of the approach, indeed, enables to adapt the segmentation to the parameterization distortion of each patch (see Section 4.6.2), and to choose any other embedding of the charts. Moreover, the connectivity of the segmentation is very simple because the cells can be adjacent only by a whole boundary component, therefore preventing the creation of corner vertices.

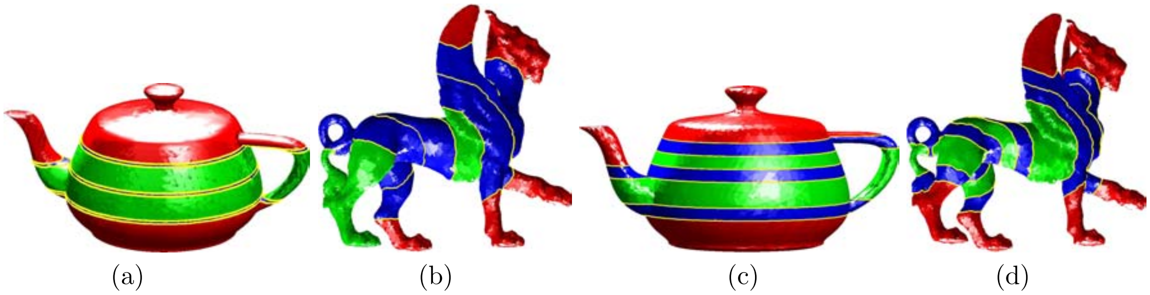


Figure 4.6: (a-b) Minimal decomposition with respect to the height function for the teapot and the geodesic distance from curvature extrema for the feline. (c-d) Adaptive segmentation induced by the parameterization distortion.

4.6.2 Adaptive approach to the topological decomposition

We deal with complex 3D shapes in a more flexible way through an adaptive implementation of the previous method which slices the surface in correspondence of topological changes and where high parameterization distortions or curvature extrema [MPS$^+$04a, CSM03] occur. To this end, after the identification of the regions $\{R_i\}_{i=1,\dots,m}$ we proceed as follows. For each patch R_i,

1. we locate its subregions S with high L^2, or L^∞ (see Equation (4.23)) stretch (resp., multiresolutive curvature [MPS$^+$04a]);
2. we use the averaged value $\overline{\alpha}$ of f on its boundary components to define the corresponding iso-contour $f^{-1}(\overline{\alpha})$ on R_i;
3. we subdivide R_i in correspondence of the new contours (if any).

The type of each new sub-patch can be different with respect to that of R_i, but it always falls in the previous classification of the building primitives. In this way, the refined segmentation of $\mathcal{M}$ enables to reduce the distortion of the parameterization; alternative criteria can be introduced with the unique constraint of defining 0-genus patches. In Figure 4.6, the minimal and adaptive segmentation are shown; Table 4.1 gives the number and type of primitives of the segmented surfaces related to the examples shown in the chapter.

Table 4.1: Number of patches of the segmentation.

Surface	Figure	♯Patches	♯Cones	♯Cylinders	♯Bodies
Teapot	4.14(a)	9	3	3	3
Teapot	4.14(b)	9	3	3	3
Feline	4.6(a)	12	6	2	4
Feline	4.6(b)	17	7	4	6

4.6.3 Morphological and geometric segmentation

We now discuss segmentation methods based on geometric and morphological properties of the input surface which provide a decomposition belonging to $\mathcal{J}$. As discussed in Section 2.7 and 2.8, *Plumber* is a specialized shape classification method for detecting tubular features (i.e., generalized cones and cylinders) of 3D surfaces. The algorithm segments a surface into connected components that are either body parts or elongated features, that is, handle-like and protrusion-like features, together with their concave counterparts, i.e., narrow tunnels and wells. The multi-resolutive segmentation codes each primitive with the level of detail at which it has been identified. In [KT03], the segmentation method is defined as a fuzzy clustering of vertices where the probability that a face belongs to a patch depends on its distance from the other faces. The advantage of the method is the avoidance of over-segmentation and that boundaries among adjacent regions are not jagged.

The segmentation provided by these methods is meaningful in our context if each patch R_i has 0-genus. Therefore, this last constraint has to be added as additional stop criterion and it is simply tested by evaluating the Euler formula on R_i. In special situations, such as presence of topological noise, it may happen that the previous methods are not able to proceed in the segmentation until the patch has 0-genus; in this case, the region is partitioned by using the topological segmentation induced by the Reeb graph. Even though topological noise can be removed during a pre-processing step, an adaptive segmentation seems to be a possible solution for handling these situations without modifying the input surface.

The use of the primitives provided by these different segmentation methods enables to segment $\mathcal{M}$ in a more natural way with respect to [FHR02, LSS$^+$98], while maintaining their parameterization and distortion control at a simple level. The factors which mainly differentiate this decomposition from simplification-based segmentation are the geometry of the allowed patches and their connectivity relations. Since we deal with a simplified situation on the adjacency between patches, we guarantee a regular remeshing on the whole surface with the exception of localized vertices (see Section 5.2).

4.7 Patch parameterization

In this section, the method of barycentric coordinates discussed in Section 4.5 is extended to the case of a surface triangulation $\mathcal{M}$ with 0-genus and k simply-connected boundary components $\{\gamma_i\}_{i=1}^{k}$; this result has an importance which is independent of the local parameterization and which will be used in Section 4.8 for the global parameterization of surfaces with an arbitrary genus. Then, we describe how each patch R_i of the segmentation $\mathcal{J}$ is parameterized.

Proposition 15 (Extended convex combination maps)**.** *Given a triangulated manifold $\mathcal{M}$ of 0-genus and k boundary components $\{\gamma_i\}_{i=1}^{k}$, there exists an embedding of $\mathcal{M}$ onto a planar domain with k convex boundary components.*

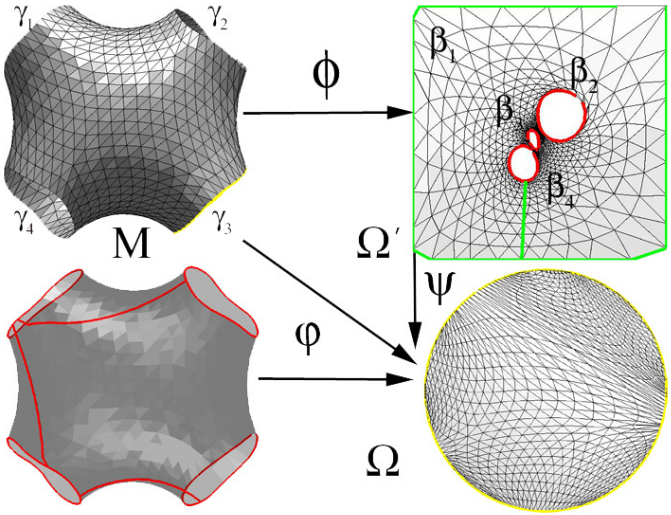


Figure 4.7: General framework for the parameterization of a surface patch with $k = 4$ boundary components.

Proof. Chosen one boundary component γ_1 (for example, the one with the greater number of vertices or length), we parameterize $\mathcal{M}$ with respect to γ_1, that is, we consider the vertices of the remaining boundary components as unknowns in the linear system (4.13). In this way, we get a parameterization $\phi : \mathcal{M} \to \mathcal{D} := (\Omega^{'}, T)$ where $\Omega^{'} \subseteq \mathbb{R}^2$ is a planar domain with k boundary components $\{\beta_i := \phi(\gamma_i)\}_{i=1}^{k}$. Clearly, different initial choices of γ_i produce different ϕ. We now prove that each curve β_i, $i = 1, \ldots, k$ is convex. Let us suppose that the internal boundary component β is not convex; then, there exist two non-consecutive vertices u_p and u_q on β given by

$$u_p = \sum_{j \in N(p)} w_{p,j} u_j, \qquad u_q = \sum_{j \in N(q)} w_{q,j} u_j,$$

a point $a := \lambda u_p + (1 - \lambda) u_q$, $\lambda \in [0, 1]$, on the line segment which joins u_p to u_q, and a triangle $\Delta := [u_{i_1}, u_{i_2}, u_{i_3}] \in \mathcal{D}$ such that $a \in \Delta$. Therefore, we express a as a linear combination of the vertices of Δ

$$a = t_1 u_{i_1} + t_2 u_{i_2} + t_3 u_{i_3}, \qquad t_1, t_2, t_3 \geq 0, \qquad t_1 + t_2 + t_3 = 1$$

and from the identities

$$u_{i_k} = \sum_{j \in N(i_k)} w_{i_k,j} u_j, \qquad k = 1, 2, 3$$

we have that

$$\sum_{j \in N(p)} w_{p,j}(\lambda u_j) + \sum_{j \in N(q)} w_{q,j}((1-\lambda) u_j) + \sum_{j \in N(i_1)} w_{i_1,j}(-u_j) + \\ + \sum_{j \in N(i_2)} w_{i_2,j}(-u_j) + \sum_{j \in N(i_3)} w_{i_3,j}(-u_j) = 0.$$

The previous identity implies that A in (4.13) is singular; from Theorem 10, it follows that it is a contradiction. □

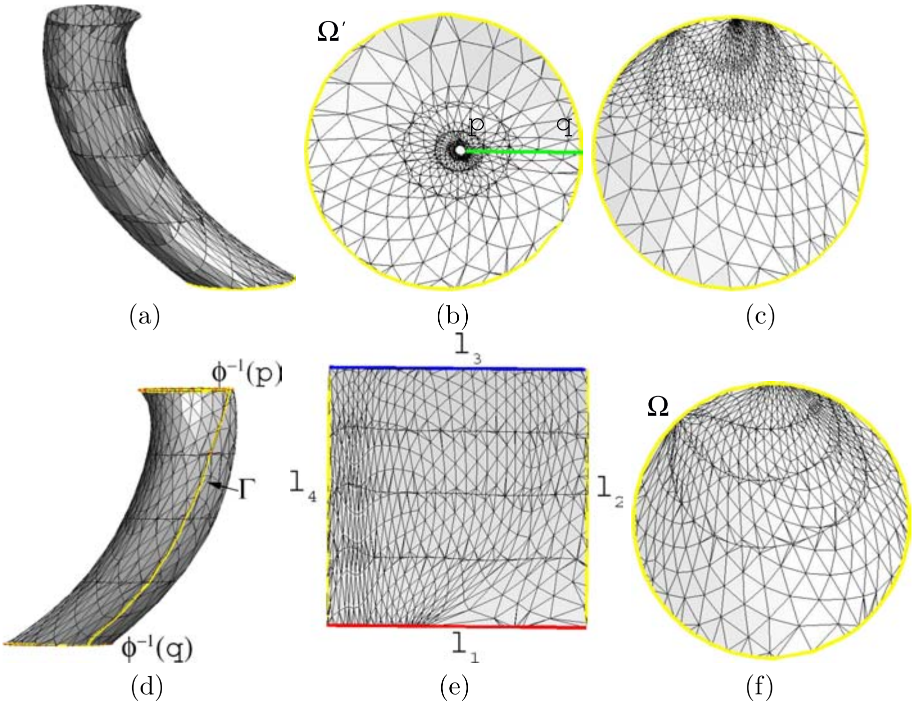


Figure 4.8: (a) Input cylindrical surface R; (b) least-squares parameterization of R with respect to the boundary depicted in yellow in (a), and minimal cut γ (green line); (c) shape-preserving unfolding of $\Omega^{'}$ with respect to the depicted cut in (b); (d) 3D cut Γ; shape-preserving parameterization of R with respect to Γ on the (e) unit square, and (f) on the unit circle.

Another choice, which resembles the statement of Theorem 12 at page 111, consists of closing $(k-1)$ boundary components of $\mathcal{M}$, thus defining a new surface triangulation $\mathcal{M}^{\star}$ with just one boundary, mapping $\mathcal{M}^{\star}$ to $\Omega^{'}$, and then removing the closed loops; we discard this method because the internal loops of the planar parameterization are generally non-convex. Moreover, even if $\Omega^{'}$ is homeomorphic to the input mesh, this approach is not useful for texture mapping, especially when the texture is non-uniform. In this case indeed, the entire image should be mapped onto the surface and not a part of it only. For remeshing applications or uniform textures, the presence of holes is simply solved by avoiding operations on them.

First of all, conical primitives, i.e. for $k=1$, fall into the standard case and they are parameterized by solving the equation (4.12) with shape-preserving weights. We now analyze the case $k \geq 2$ (see Figure 4.7) distinguishing the case $k=2$ (i.e., parameterization of cylindrical primitives) and $k \geq 3$ (i.e., parameterization of bodies). Using Proposition 15, we build a parameterization $\phi : \mathcal{M} \rightarrow \Omega^{'}$ where $\Omega^{'}$ is a planar domain with k convex loops $\{\beta_i := \phi(\gamma_i)\}_{i=1}^{k}$. As detailed by Proposition 16, for the removal of the internal closed curves $\{\beta_i\}_{i=2}^{k}$ we define a new planar domain Ω with only one boundary and a homeomorphism $\psi : \Omega^{'} \rightarrow \Omega \subseteq \mathbb{R}^2$ with the following properties:

- the boundary $\partial\Omega$ of Ω is convex;

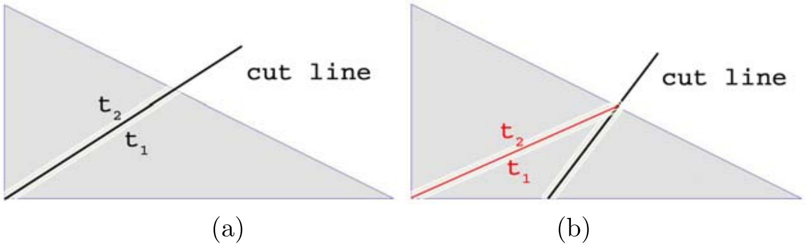


Figure 4.9: (a) Cut of the triangle through a vertex, (b) re-triangulation.

- each internal boundary component β_i of $\Omega^{'}$ is mapped to a curve $\psi(\beta_i)$ of $\partial\Omega$, $i = 2, \ldots, k$.

Proposition 16. *Given a triangulated manifold $\mathcal{M}$ of 0-genus and with k boundary components,*

1. *it is always possible to convert $\mathcal{M}$ to a surface of 0-genus with only one boundary component γ in $O(n_V)$ time, with n_V number of vertices of $\mathcal{M}$;*
2. *the curve γ joins the boundary components of $\mathcal{M}$ without self-intersections and it is not affected by the mesh connectivity.*

Proof. Parameterization of cylindrical surfaces. If $k = 2$, the surface triangulation is topologically equivalent to a cylinder and the parameterization $\phi : \mathcal{M} \rightarrow \Omega^{'}$ embeds $\mathcal{M}$ onto a domain $\Omega^{'}$ with two boundary components $\beta_1 := \phi(\gamma_1)$ and $\beta_2 := \phi(\gamma_2)$. Let p and q be the closest points, with respect to the Euclidean distance, on β_1 and β_2; that is,

$$\|p - q\|_2 \equiv \min_{r \in \beta_1, s \in \beta_2} \{\|r - s\|_2\},$$

and $[p, q]$ the line segment from p to q (see Figure 4.8(a-b)).

The cut $[p, q]$ is inserted in the input triangulation before unfolding the parameter domain onto Ω. To this end, if $[p, q]$ intersects a triangle t passing through one of its vertices t is split into two new faces t_1, t_2 which share a part of the cut (see Figure 4.9(a)). Otherwise, $[p, q]$ splits t into one triangle and a quadrilateral q; then, q is re-triangulated by subdividing it along its shortest diagonal (see Figure 4.9(b)). This update is also performed on the 3D triangulation in order to ensure that $\mathcal{M}$ and $\Omega^{'}$ share the same connectivity. Successively, we cut $\Omega^{'}$ along $[p, q]$, thus unfolding $\Omega^{'}$ to Ω through ψ. The unfolding map ψ is achieved by duplicating the cut $[p, q]$ to $[q, p]$, converting β_1, β_2 and the cuts into one connected loop $\{\beta_1, [p, q], \beta_2^{-1}, [q, p]\}$, and applying the Floater's parameterization (see Figure 4.8(c)). Each closed curve γ has an anticlockwise orientation; γ^{-1} means that we have reversed the ordering of its vertices (i.e., clockwise orientation). Finally, the cut operation is coded into a relation $\mathcal{R}$ between $\psi([p, q])$ and $\psi([q, p])$. Therefore, the map $\varphi := \psi \circ \phi$ embeds $\mathcal{M}$ onto Ω.

General case. We solve the case of k boundaries, with $k \geq 3$, by using an iterative procedure. Supposed that we have removed $(i - 1)$ boundary components, at the i^{th} step we cut the current parameter domain $\Omega_i^{'}$ along the line segment of minimal length which joins its external loop with

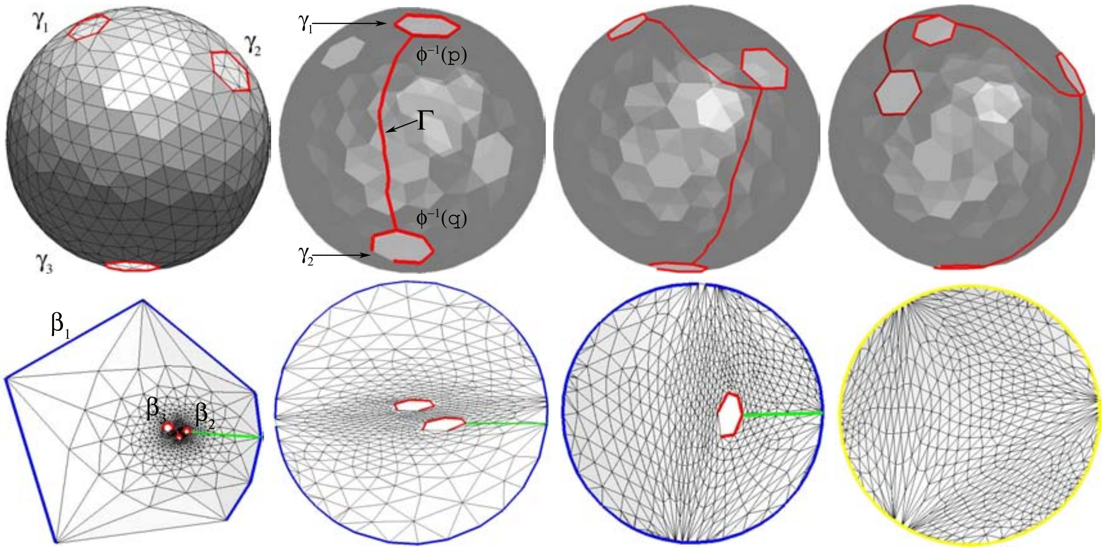


Figure 4.10: Cut evaluation and parameterization with shape-preserving weights of a surface of 0-genus with four boundary components. The first row shows the different steps on the 3D surface and the second row their 2D counterparts on the parameterization domain.

the closest internal boundary component. If $\Omega_1^{'} := \Omega^{'}$, let $\psi_i : \Omega_i^{'} \rightarrow \Omega_{i+1}^{'}$ be the i^{th} embedding; at the end of this process, the internal loops of Ω' are mapped onto $\partial\Omega_k^{'}$ through the function $\psi := \psi_{k-1} \circ \ldots \psi_2 \circ \psi_1$ (see Figure 4.10). Each step of the procedure used for joining the boundary components of $\mathcal{M}$ requires to update and introduce the only entries of the matrix A in (4.13) related to those points in $\Omega^{'}$ that have been affected by the cut. The computational cost for updating W (resp., for solving (4.13) with an iterative solver [GL96]) is linear in the number of vertices of Γ (resp., $\overline{\mathcal{M}}$).

The independence of γ of the mesh connectivity is a consequence of the way we have built the path γ; self-intersections are avoided because at each step we map the joined boundary components on the external boundary component of the current parameterization domain. □

The linear path $[p, q]$ exploited for mapping $\Omega^{'}$ to Ω defines a curve $\Gamma := \phi^{-1}([p, q])$ on $\mathcal{M}$, called *link path*; the more ϕ maintains the geometry of the input mesh the better the line connecting $\phi^{-1}(p)$ to $\phi^{-1}(q)$ approximates the corresponding geodesic (see Figure 4.8(d)). Alternatively to the previous approach, we can parameterize a primitive directly using the 3D cut Γ without applying two consecutive unfoldings with the aim of avoiding to accumulate the distortion in the proximity of the boundary of Ω (see Figure 4.8(e-f)). We use this new embedding $\varphi : \mathcal{M} \rightarrow \Omega$ in all the following examples. Removing one boundary component at each step is the only constraint of the proposed approach, while different criteria can drive the cut selection. In the following, we detail the main options which can also be combined to define a *hybrid method*; then, the properties of γ and a comparison with previous work are discussed in Section 4.7.1.

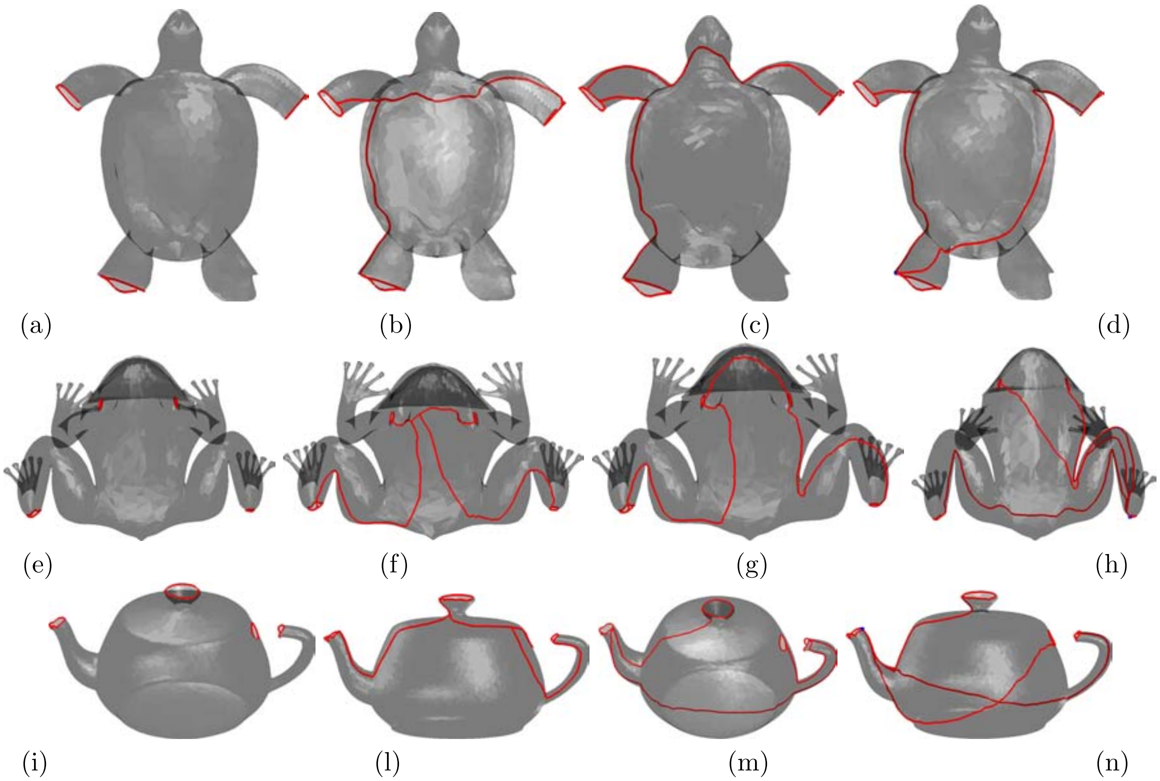


Figure 4.11: (a, e, i) Input dataset with three (resp., four) boundary components, (b, f, l) the criterion used for the cut identification is its minimal length in the parameterization domain and the search includes link paths; therefore, bifurcations of γ are admitted. (c, g, m) The cut is constrained to join the boundary components without bifurcations on the surface, (d, h, n) all the link paths interpolate a given source (blue) point. The cut length and the L^2-stretch of the corresponding parameterization are given in Table 4.2.

Joining boundary components with or without bifurcations. If $\mathcal{M}$ has three or more boundary components, at a given iteration $k \geq 2$, the external boundary $\partial\Omega^{'}$ of the current parameter domain $\Omega^{'}$ includes the boundary components removed at the previous steps and the link paths which join them. Therefore, the line segment $[p, q]$ of minimal length which links $\partial\Omega^{'}$ to its closest internal boundary component may have its origin p on an arc which corresponds to a link path Γ on $\mathcal{M}$; in this case, the point $\phi^{-1}(p) \in \Gamma$ will be a bifurcation of the cut (see Figure 4.11(b, f, l)). When we evaluate the cut $[p, q]$, we can restrict its search to the part of the external boundary which corresponds to the boundary components that were removed at the previous steps and neglecting the link paths which join them. In this way, we constrain the cut to join the boundary components without admitting bifurcations on the input surface (see Figure 4.11(c, g, m)).

Joining boundary components with a given source point. In some applications such as the search of the canonical schema of $\mathcal{M}$ (see Section 4.9.2), it is useful that each link path interpolates

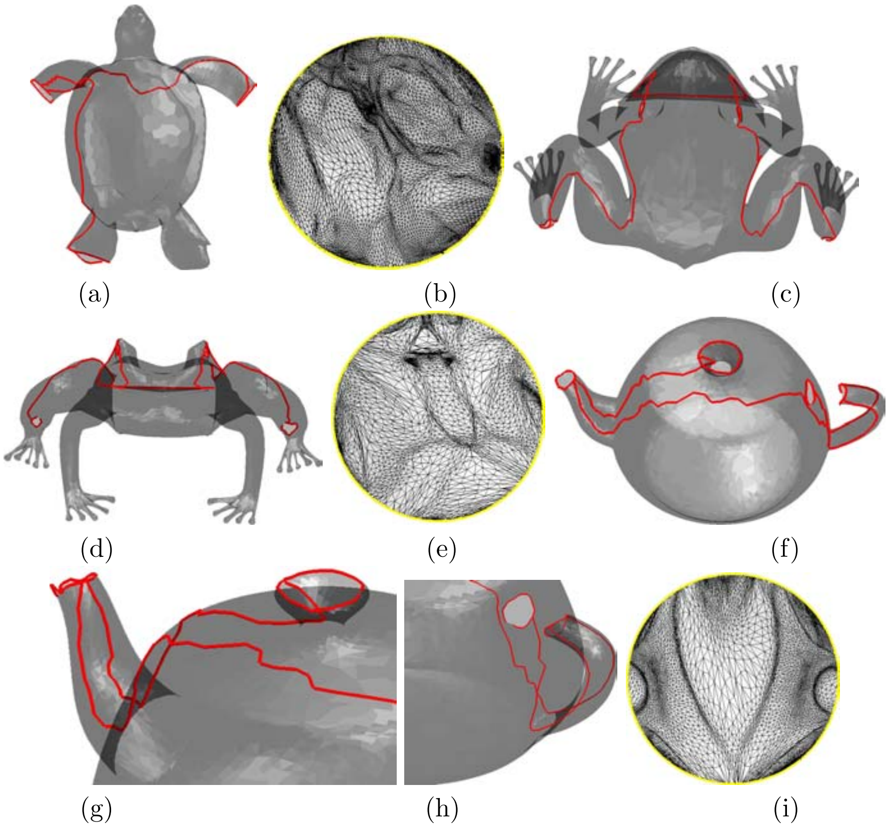


Figure 4.12: (a, c, d, f) Geodesic cut constrained to join the boundary components without bifurcations on the link paths and achieved by running the Dijkstra algorithm; (g-h) zoom-in on the spout and the handle of the teapot; (b, e, i) embedding of the input surface with respect to the cut shown in Figure 4.11 (c, g, m). Note the difference of the cuts in terms of smoothness and length which are given in Table 4.2.

a given point $s \in \mathcal{M}$. Let us suppose that p is the counterpart of s on the parameterization domain. At each step, the search of the minimum length cut is constrained to join p to its closest boundary component with respect to the Euclidean distance. However, the use of a base point can accumulate the parameterization distortion around s and therefore it is not optimal for applications such as remeshing and texture mapping (see Figure 4.11(d, h, n)).

Minimizing the cut length and the related parameterization distortion. The iterative approach to the unfolding avoids to check if self-intersections of the cut disconnect the surface and it can also be applied to the 3D setting. In this case, at the first step we select the longest boundary component which is then joined to its closest boundary component by the shortest path [Dij59]. These two boundaries are then merged into a unique loop γ and we set the weights of the edges in γ equal to $+\infty$. This update avoids that the next link paths will cross the cut found at the previous steps; then, the following iterations proceed as previously discussed. A comparison

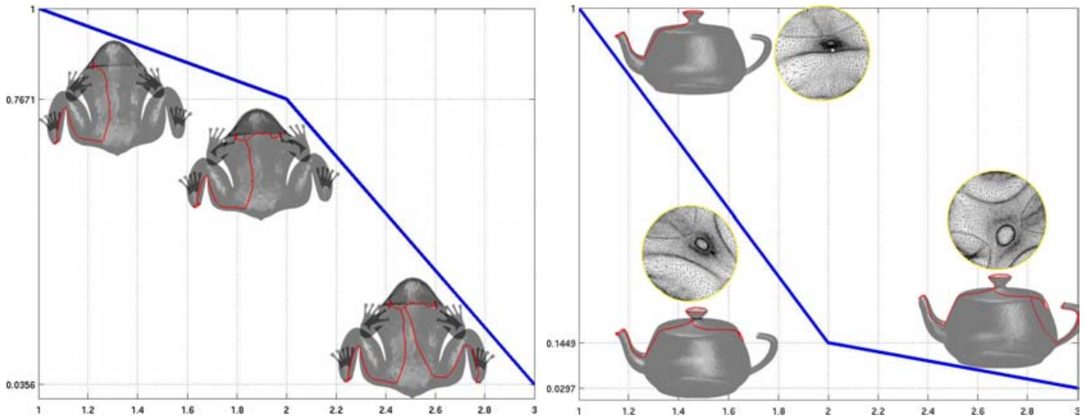


Figure 4.13: Cut, parameterization with shape-preserving weights, and normalized L^2-stretch values (y-axis) achieved by removing one boundary component at each step (x-axis).

between the cuts achieved by using the exact and the approximated approach are given in Figure 4.12.

A more general strategy consists of defining γ as the minimum of the functional which is a convex combination between the cut length and the related parameterization stretch; this choice avoids to have long and visible seams on the remeshed/textured dataset while guaranteeing a low distortion. In this case, each link path is achieved by running the Dijkstra algorithm on $\Omega^{'}$ with weights

$$\varrho_{ij} = \lambda \varrho_{ij}^{(g)} + (1-\lambda)\varrho_{ij}^{(s)}$$

where (i,j) is an edge of T, $\lambda \in [0,1]$, $\varrho_{ij}^{(g)} := \|x_i - x_j\|_2$ is the edge length on $\mathcal{M}$, and $\varrho_{ij}^{(s)} := (s_{ij}^{(1)} + s_{ij}^{(2)})^{-1}$ is the average of the L^2-stretch of the two triangles which share the edge (i,j) with respect to the current ϕ (see also Figure 4.26). We explicitly underline that if $\lambda = 1$ we are using the approach previously discussed.

4.7.1 Discussion

Searching and optimizing a cut on a disc-like surface $\mathcal{M}$ by taking into account the correspondence between $\mathcal{M}$ and its parameterization domain was firstly proposed in [GGH02] where an iterative procedure augments an initial coarse cut with a path which joins it to a sub-region with high L^2-stretch until the distortion is below a given threshold. The computational cost after k iterations is $O(kn\log(n))$ with n number of input vertices. In a similar way, we connect the boundary components on $\mathcal{M}$ but for solving a different problem and providing a grater flexibility for the cut selection and a lower computational cost. In fact, working on the parameterization domain is motivated by its simplicity, a greater degree of flexibility of adding constraints on the final cut (e.g., interpolation of a source point $p \in \gamma$, the possibility of joining the boundary components with respect to a given order, the control of bifurcations on γ), a greater smoothness of γ which is not affected by the mesh connectivity, and a linear computational cost for its evaluation. Since each

Table 4.2: Length of the cut graphs achieved as approximation of the geodesic ones; the table also shows the L^2-stretch.

Cut type	Figure	Length $\|\gamma\|$	$L^2(\mathcal{M})$
		Turtle	
Bifurcations	4.12(b)	55.88	3.66×10
No bifurcations	4.12(c)	59.32	1.38×10
Source	4.12(d)	78.81	1.16×10
		Frog	
Bifurcations	4.12(f)	12.92	9.84
No bifurcations	4.12(g)	13.54	1.10×10
Source	4.12(h)	18.23	2.19×10^2
		Teapot	
Bifurcations	4.12(l)	4.23	3.66×10
No bifurcations	4.12(m)	5.08	1.43×10
Source	4.12(n)	6.74	4.23×10

parameterization $\varphi : \mathcal{M} \rightarrow \Omega$ with respect to γ maps the cut to the smooth boundary of a planar polygon, an important property of γ is its regularity. A link path achieved by using the Dijkstra algorithm is C^0, in fact it is a polyline made of edges of $\mathcal{M}$, while a path $\Gamma(\lambda) := \phi^{-1}(\lambda p+(1-\lambda)q)$, $\lambda \in [0,1]$, is C^s if $\phi^{-1} \in C^s$. If we consider the parameterization ϕ with respect to harmonic weights, ϕ is the solution of the Laplacian equation $\Delta\phi = 0$ and $s \geq 2$. It follows that each link path of the global cut γ achieved as approximated geodesic curve is at least C^2 everywhere with the exception of the points where it intersects the other boundary components or where bifurcations occur; in these points a C^0-regularity holds (see Figure 4.12). The cut-graph achieved as best compromise between length and induced parameterization distortion requires $O(n\log(n))$ time and it is C^0 everywhere.

The parameterization of $\mathcal{M}$ onto a domain without holes is necessary for generating a geometry image [GGH02] or performing a regular remeshing; otherwise, black pixels or extraordinary vertices will appear in a neighborhood of the boundary vertices. Furthermore, the cut generally reduces the parameterization distortion with respect to $\Omega^{'}$ because boundary vertices are not constrained to be internal to the parameterization domain (see Figure 4.13 and Table 4.2).

Joining the boundary components of $\mathcal{M}$ by using a minimum spanning tree weighted on the edge length or geodesic curves is computationally expensive (i.e., $O(n\log(n))$), sensible to local noise, and it depends on the input simplicial complex on which the Djikstra algorithm runs; therefore, these approaches usually require a post-processing step which makes γ smoother. Furthermore, the Dijkstra algorithm can compute paths which are quite different from true geodesic curves; this is because the algorithm takes shortcuts passing through edges instead of vertices. Searching the cut on the parameter domain enables to forget the input triangulation because geodesic curves are represented by line segments; therefore, the proposed approach is stable with respect to the

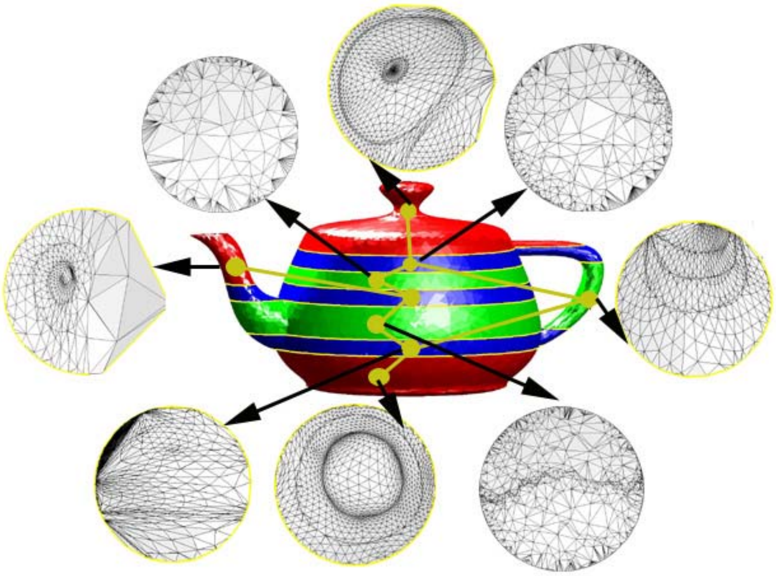


Figure 4.14: Shape-graph and parameterization of the building primitives related to the minimal segmentation induced by the Reeb graph with respect to the height function.

presence of tiny triangles, irregular sampling and connectivity.

4.7.2 Shape graph and coding of the parameterization

In the previous sections, using a segmentation $\mathcal{J}$ of the input surface we have identified and parameterized cones, cylinders, and bodies. Starting from the partition $\{R_i\}_{i=1,\dots,m}$ of $\mathcal{M}$ parameterized by $\varphi_i : R_i \rightarrow \Omega_i$, $i = 1, \dots, m$, the *graph-based parameterization* of $\mathcal{M}$ is the collection $\mathcal{G} := \{(R_i, \varphi_i, \Omega_i)\}_{i=1,\dots,m}$ enriched with the adjacency relations among primitives coded by the graph arcs. Equivalently, we substitute each node of the shape-graph with the parameterization of the corresponding primitive calculated by using standard (e.g., cones) or specialized methods (e.g., cylinders and bodies). $\mathcal{G}$ gives a high-level geometric and topological knowledge on where and how regions are located and glue together; also, $\mathcal{G}$ specifies how we can generate the input surface with the building primitives (or equivalently, with their planar embeddings). Moreover, $\mathcal{G}$ stores information on the size and generators of cylinders and cones. Figures 4.14 and 4.15 give a complete view of the parameterization graph. As described in Chapter 5, this high-level description of 3D shapes can be enriched with additional geometric structures which support applications such as deformation or editing and stored as graph attributes.

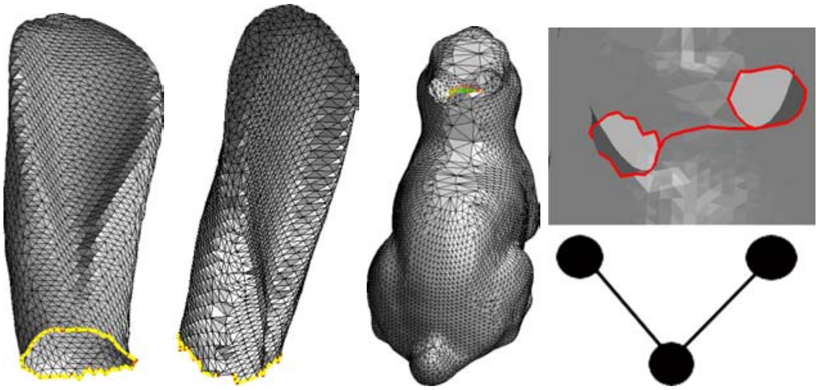


Figure 4.15: Shape-graph of the rabbit where the topological segmentation into three patches is induced by the Reeb graph with respect to the distance from curvature extrema located on the ears.

4.8 Global paramterization of bordered triangle meshes with arbitrary genus

Let $\mathcal{M}$ be a compact, connected triangulated surface with n vertices, e edges, f faces, and b boundary components; then, $\chi(\mathcal{M}) := n - e + f$ is called the *Euler characteristic* of $\mathcal{M}$ and it is related to the genus g of $\mathcal{M}$ by the relation $g = \frac{1}{2}(2-\chi(\mathcal{M})-b)$. According to the discussion given in Section 4.5, the *global parameterization* of $\mathcal{M}$ is an isomorphism $\varphi : \mathcal{M} \rightarrow \mathcal{P} \subseteq \mathbb{R}^k$, $k = 2, 3$, with $\mathcal{P}$ parameterization domain (see Figure 4.16); therefore, the problem consists of finding an injective map

$$\begin{aligned} \varphi|_M : \quad & M \rightarrow \Omega \subseteq \mathbb{R}^k \\ & x_i \mapsto \varphi(x_i) := u_i \end{aligned} \tag{4.30}$$

with $\Omega := \{u_i : i = 1, \ldots, n\}$ and such that $\mathcal{P} := (\Omega, T)$ is isomorphic to $\mathcal{M}$. The map φ is extended from the set of vertices M to the surface $\mathcal{M}$ by using barycentric coordinates: if $p \in \mathcal{M}$ belongs to the triangle of vertices a, b, c, then $p = \alpha a + \beta b + \gamma c$ with $\alpha \geq 0$, $\beta \geq 0$, $\gamma \geq 0$, $\alpha + \beta + \gamma = 1$, and $\varphi(p)$ is set as $\varphi(p) := \alpha\varphi(a) + \beta\varphi(b) + \gamma\varphi(c)$.

During the last years, the global parameterization of an arbitrary 2-manifold triangle mesh has gained an increasing attention due to the impact on several research fields: its theoretical aspects are related to surface classification and evaluation of polygonal schema, the computation of a global embedding enables to apply standard techniques of approximation (i.e., remeshing and polynomial approximation), texture mapping and compression are further applications. Even though these problems are different they share the idea of switching each of them from the 3D surface with arbitrary topology to one of its 0-genus representation whose choice depends on the problem characteristics we are dealing with.

The set of cut graphs for a given triangle mesh $\mathcal{M}$ with an arbitrary genus depends on the underlining topology and the selection of a specific one should be guided by the surface geometry and targeted applications. The influence of the mesh connectivity on the cut graph search affects

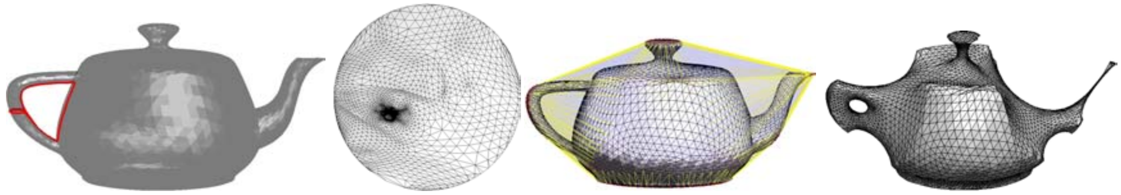


Figure 4.16: Two examples of global parameterization of the same surface. First row: $\mathcal{M}$ is embedded onto a planar domain (i.e., $k = 2$) by using the red cut which converts it to a disc-like surface. Second row: the parameterization domain is a 3D surface (i.e., $k = 3$) achieved by running the shape-preserving parameterization with the (red) vertices of the convex hull of $\mathcal{M}$ (blue faces) as boundary constraints.

previous work due to the use of algorithms based on mesh traversal techniques for the evaluation of the geodesic metric. Our solution [Pat04, PSF04] is to search a cut graph made of the iso-contours of a fair function $f : \mathcal{M} \rightarrow \mathbb{R}$ and to work in a planar domain where geodesic curves are defined by line segments and whose counterparts on $\mathcal{M}$, with respect to an appropriate diffeomorphism $\phi : \mathcal{M} \rightarrow \mathbb{R}^2$, give smooth approximations of geodesic paths. The emphasis of our approach is on the definition of a simple method for finding cut graphs of $\mathcal{M}$ and guided by several criteria which spread from the global parameterization (e.g., minimal length, minimization of the parameterization distortion, or interpolation of points as required by remeshing and texture mapping) to the calculation of polygonal schemes for surface classification. We propose an algorithm which finds a reduced polygonal schema in $O(n)$ time if $\mathcal{M}$ has 0-genus, and with a constant number of redundant edges in $O(n \log(n))$ time in the general case. The proposed approach treats closed and bordered surfaces with a unique approach; furthermore, using standard reduction operations [VY90] it is possible to convert the cut graph of $\mathcal{M}$ to a canonical one while the corresponding polygonal schema is converted to its canonical form.

The *classification theorem* of compact manifolds [Mas67] guarantees the existence of a representative $\mathcal{P}$ for each class of homeomorphic surfaces, that is, a bordered sphere or disc for the family of 0-genus surfaces and a bordered g-holed torus for the set of manifolds with genus g. In this way, applications such as remeshing and texture mapping can be performed on $\mathcal{P}$ and then the result is mapped back to $\mathcal{M}$ by φ^{-1}. This approach has been applied when $\mathcal{M}$ is a closed 0-genus surface (i.e., *spherical parameterization* [GGS03]) or with one boundary component (i.e., *convex combination maps* [Flo97] discussed in Section 4.5.1); the general case has not yet been solved due to the difficulty of finding both $\mathcal{P}$ and φ. In Section 4.8.1, we give a first investigation on this problem focusing our attention on the search of a "simpler" isomorphic representative of an arbitrary triangulated surface, as done in [IGG01, SCO04], but also taking into account the underlining parameterization problem. Then, in the following sections we discuss the global embedding achieved by cutting the surface along appropriated cuts which convert it to a disc-like surface (i.e., with 0-genus and one boundary component). An example of the results achieved by using these two approaches is given in Figure 4.16.

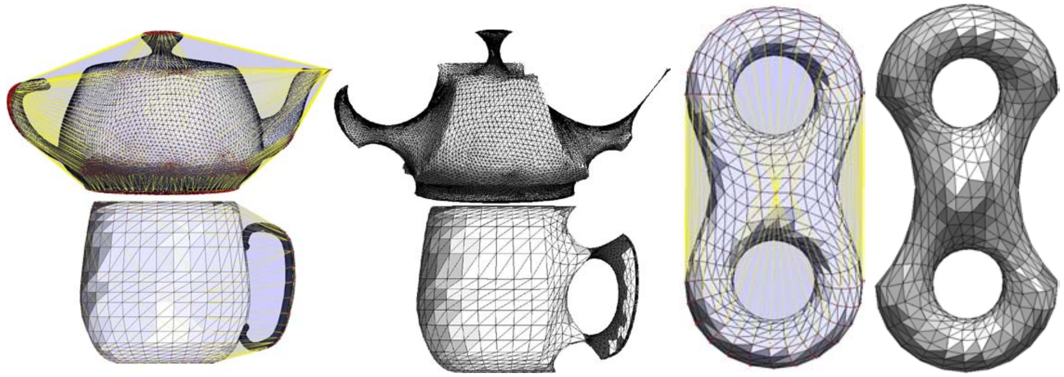


Figure 4.17: Intrinsic shape-preserving parameterization of surfaces with different genus and geometry; blue faces represent the convex hull of the input dataset with vertices depicted in red.

4.8.1 3D parameterization

Given an arbitrary triangulated surface $\mathcal{M}$, we want to define an isomorphic 3D domain $\mathcal{P}$; the parameterization methods previously discussed are special cases of this problem when $\mathcal{P}$ is a planar convex polygon or a sphere. We show that the convex combination maps [Flo97] can be applied to map $\mathcal{M}$ onto a 3D domain; the main idea consists of locating a set of *virtual boundary vertices* of $\mathcal{M}$ which are fixed through φ to a 3D convex (virtual) boundary and determine the remaining "internal" vertices.

Proposition 17. *Any arbitrary surface $\mathcal{M} \subseteq \mathbb{R}^3$ can be embedded onto a 3D domain $\mathcal{P}$ which belongs to the convex hull of $\mathcal{M}$.*

Proof. Let

$$\Delta(M) := \left\{ \sum_{i=1}^{n} \alpha_i x_i : \alpha_i \geq 0, i = 1, \ldots, n, \sum_{i=1}^{n} \alpha_i = 1 \right\}$$

be the convex hull of M [GP74] and $B := \{x_i : i \in H\}$ the set of its vertices (see Section 1.1.1). We can apply the shape-preserving parameterization with $V_B := H$ where the 2D boundary of the planar convex polygon has been replaced by the 3D convex surface $\partial\Delta(M)$. Then, the solution of the linear system (4.12) defines the *intrinsic parameterization* of $\mathcal{M}$, $\varphi|_M : M \rightarrow \Omega \subseteq \mathbb{R}^3$, as done in the case of disc-like surfaces.

We now show that the vertices of Ω belongs to $\Delta(M)$ by adapting the results in [Flo97] to the 3D case. Let us suppose that there exists at least one vertex which does not belong to $\Delta(M)$ and among all of them we select the point p which has maximum distance from $\partial\Delta(M)$ [5]. Let q be the orthogonal projection of p on $\partial\Delta(M)$, and π the plane passing through p with normal vector $(p - q)$. The way we selected p guarantees that all the points of Ω belong to the half-space $\mathcal{H}$ which contains q. Since p is a strict convex combination of its neighbors which do not belong to

[5] The distance of a point p from a closed convex set $\mathcal{C}$ is defined as $\|p-q\|_2 = \min_{x \in \mathcal{C}}\{\|p-x\|_2\}$, and the unique solution $q := pr_{\mathcal{C}}(p)$ is the orthogonal projection of p on $\mathcal{C}$.

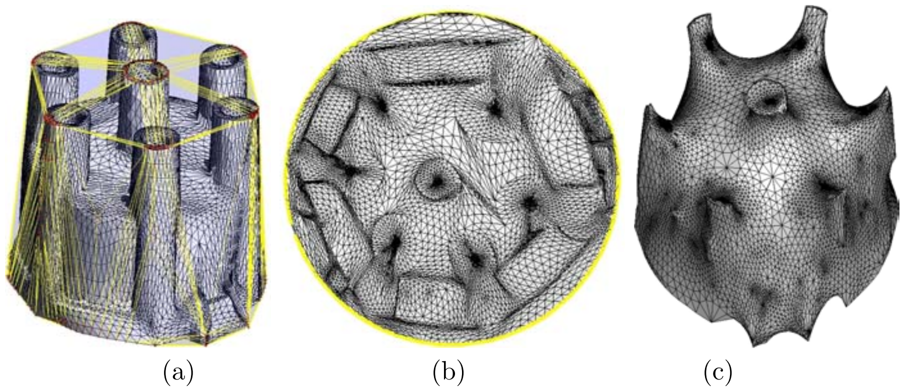


Figure 4.18: (a) Disc-like surface with its convex hull (blue triangulation), (b) shape-preserving, and (c) intrinsic parameterization.

$\mathcal{H}^C$ (i.e., the complementary set of $\mathcal{H}$), they belong to π. Growing from this 1-star to the adjacent neighborhoods, we have that a point of $\partial\Delta(M)$ belongs to π and this is a contradiction. □

We explicitly underline that:

$$\varphi(\mathcal{M}) = \mathcal{M} \leftrightarrow \mathcal{M} \text{ is convex.}$$

In fact, $\varphi(\mathcal{M}) = \mathcal{M} \leftrightarrow A = I$ in (4.13) $\leftrightarrow \Delta(M) = \mathcal{M} \leftrightarrow \mathcal{M}$ is convex. From this relation, it follows that the convexity of $\mathcal{M}$ represents a degree of regularity for the embedding of $\mathcal{M}$. Furthermore, the previous result can be considered as an *intrinsic* 3D property of the parameterization which is independent on the choice of the weights but only depends on the geometry of the input surface. In Figure 4.17 and 4.18, the intrinsic parameterization of several surfaces is shown.

4.8.2 Basic concepts and related work on global parameterization

From now on, we consider the case $k = 2$; therefore, the global parameterization of $\mathcal{M}$ consists of cutting it along a path γ, called *cut graph*, which enables its unfolding onto a planar domain (see also Section 1.1.4). Cutting $\mathcal{M}$ along γ defines the disc-like surface $\overline{\mathcal{M}} := (\mathcal{M} - \gamma)$ which is homeomorphic to a simple polygon Ω and by identifying the edges of $\partial\Omega$ we obtain the input surface $\mathcal{M}$ (see Figure 4.19). The polygon Ω with the correspondence of its boundary edges is called *polygonal schema* of $\mathcal{M}$ and among all of them the following ones have a theoretical importance for surface classification and Computer Graphics applications:

- *polygonal schema of minimal length*: γ has minimal length;
- *reduced polygonal schema*: γ is a *fundamental system of loops with base point* $s \in \mathcal{M}$, that is, a set of closed, simple, and pair-wise disjoint paths which share s;
- *canonical polygonal schema*: $\partial\Omega$ is a $(4g + 3b)$-sided polygon of the form

$$\gamma_1\beta_1\gamma_1^{-1}\beta_1^{-1}\dots\gamma_g\beta_g\gamma_g^{-1}\beta_g^{-1}\Gamma_1\alpha_1\Gamma_1^{-1}\dots\Gamma_b\alpha_b\Gamma_b^{-1} \tag{4.31}$$

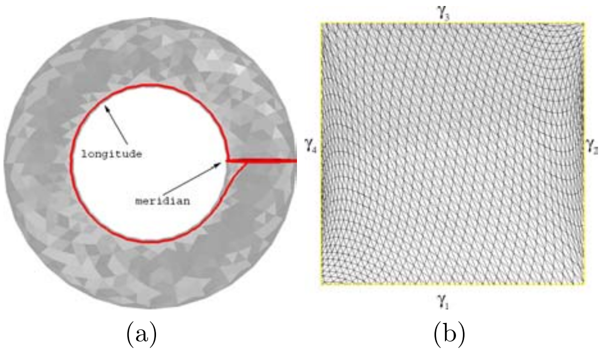


Figure 4.19: (a) Cut graph (red line) on a torus $\mathcal{M}$ with its meridian and longitude. (b) Embedding of $\mathcal{M}$ into the plane and polygonal schema; γ_1 and γ_3 correspond to the meridian while γ_2 and γ_4 give the longitude identification.

with $\gamma_1\gamma_1^{-1}$ in the case of the sphere. We explicitly underline that the canonical schema of a manifold without boundary is reduced.

Related work. Arbitrary cut graphs can be found in $O(gn)$ time and paths interpolating a common base point can be optimized in polynomial time [dVL02]. Finding the cut of minimal length is NP-hard [EHP02], but any given cut can be converted to a $O(\log^2(g))$-approximation of the minimum cut graph, which is neither reduced nor canonical, in $O(g^2 n \log(n))$ time. In [NGH04], the cut graph of a closed surface interpolates $2g+2$ points arbitrarily chosen on the input surface and which are the critical points of a Morse function f on $\mathcal{M}$ (see Section 3.2.2 and 4.9.1).

The method discussed in [SF02] consists of locating a longitude for each topological handle by using the EdgeBreaker algorithm [LTR$^+$02] which moves a contour σ on $\mathcal{M}$ maintaining it homotopic to a circle. While traversing the mesh, if σ touches itself it is split into two new contours and if two contours intersect a third one they are merged and a longitude is generated. Then, the cut graph γ is built as the minimum spanning tree [Kru56] of the $2g$ boundaries generated by duplicating the g longitudes previously found. The identification and cut of the topological handles as well as γ depend on the triangle mesh connectivity; therefore, a post-processing based on the Dijkstra algorithm [Dij59] refines these generators. For bordered surfaces, longitudes cannot be extracted because boundary components prevent to sweep σ on $\mathcal{M}$ in a homotopic way. A specialization of this method for texture mapping on closed surfaces of genus one has been proposed in [SF04].

In [GGH02], the cut graph γ is built by using an iterative procedure. At first, a seed triangle is removed from $\mathcal{M}$ and its boundary is marked as active; then, at each step one triangle adjacent to the current boundary is removed while maintaining the set of removed faces homeomorphic to a disc. The iteration stops when all the triangles have been visited and the boundary of the region gives the cut graph. The visiting process is performed by considering at first those faces which are closest to the seed one; furthermore, γ is made smoother and shorter by replacing each of its arcs

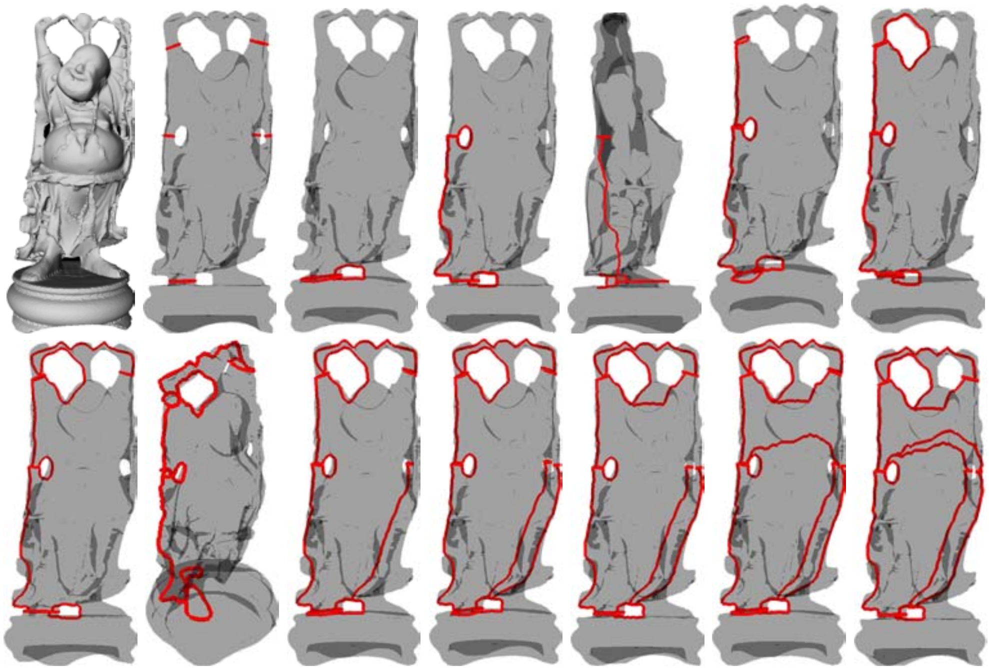


Figure 4.20: Overview: input model of genus six and steps used for converting it to a disc-like surface.

with a constrained shortest path that connects the adjacent cut nodes within a neighborhood of γ.

The degree of freedom in the cut selection is strongly smaller if we search the reduced polygonal schema or a homology base of $\mathcal{M}$ [Mas67] for surface classification and global *conformal* parameterization (i.e., φ preserves the angles). The evaluation of the canonical polygonal schema is a complex problem because the cut loops of the topological handles and the boundary components of $\mathcal{M}$ have to be linked with a minimal number of paths in spite of their shape and location on $\mathcal{M}$. Due to this difficulty, the common approach to the problem determines a polygonal schema which is successively converted to the canonical form with a set of reduction operations whose computational cost for a closed surface is $O(t\log(t))$ with t total number of vertices, edges, and faces [VY90].

In [GY03], the global conformal parameterization of an arbitrary surface $\mathcal{M}$ is achieved by solving a sparse linear system and the method is based on the properties (i.e., closedness, harmonity, conjugacy, duality, and symmetry) of gradient fields of conformal maps on $\mathcal{M}$. The parameterization is independent of the mesh connectivity and it is only related to the topology of the input surface and to the homology base used for the construction of the linear system.

Overview and contributions. All the previous methods give important theoretical and practical information for the global parameterization problem; however, they share the dependence

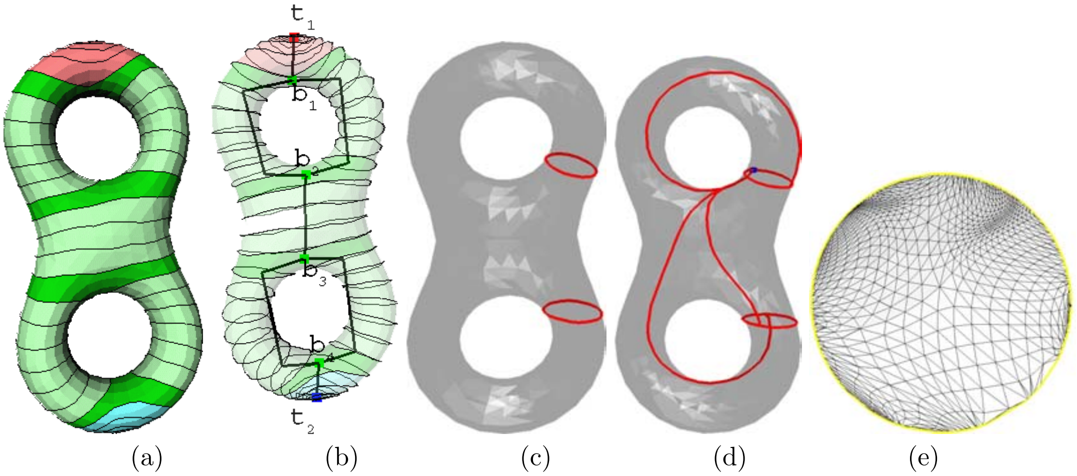


Figure 4.21: (a) Iso-contours of a harmonic function with a unique minimum (blue point) and maximum (red point) (see Section 3.2.2 and [NGH04]), (b) Reeb graph whose nodes are the barycenters of the corresponding iso-contours, (c) cuts along the meridian loops of the topological handles that convert the input surface $\mathcal{M}$ to a new one with 0-genus and four boundary components, (d) cut graph with a source (blue) point, and (e) shape-preserving parameterization on the unit circle. An alternative cut used for the polynomial interpolation of $\mathcal{M}$ is given in Figure 4.23 and other possible choices are shown in Figure 4.24.

of the cut graph on the mesh connectivity, it is not possible to constrain its shape, and with the exception of [GGH02] they assume to deal with closed surfaces. These elements become drawbacks if the embedding is finalized at remeshing and texture mapping where the irregular sampling and connectivity of the input surface is what has to be optimized without assumptions on its regularity.

Since a cut graph γ which converts $\mathcal{M}$ to a disc-like surface enables to parameterize $\overline{\mathcal{M}}$ with any method which works on surfaces of 0-genus with one boundary component (see Section 4.5), the search of the cut that we propose is mainly focused on its characteristics such as minimal length, smoothness, and induced parameterization distortion. The main features of the proposed approach are (see Figure 4.20):

- *flexibility*: we provide a method for building a family of cut graphs with several options for the selection of the loops that reduce $\mathcal{M}$ to a 0-genus surface with $2g + b$ boundary components;
- *generality* and *openess*: closed and bordered surfaces are treated with a unique approach. Furthermore, the method proposed for joining the cut loops of the topological handles of $\mathcal{M}$, as described in Section 4.7, does not rely neither on the fact that they are meridians or longitudes, nor on the specific methods that are used for their identification. The only assumption is that the cut surface has 0-genus; therefore, it is also useful for parameterizing bordered 0-genus surfaces which are the output of algorithms that cut the topological handles of $\mathcal{M}$ on the base of a different approach;

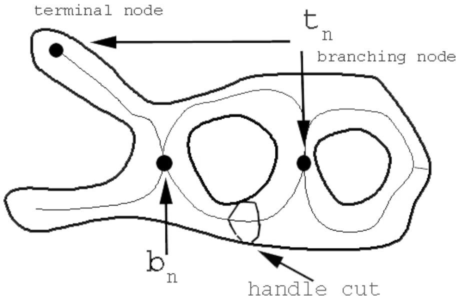


Figure 4.22: Identification and cut of the topological handles.

- *smoothness* and *"independence" of the mesh connectivity*: the cut graph is not affected by the mesh connectivity. If harmonic weights [PP93] are chosen, the cut graph is C^s, $s \geq 2$, everywhere with the exception of few points where a C^0-regularity holds. On the contrary, previous work guarantees a C^0-smoothness because they are based on mesh traversal techniques such as the Dijkstra algorithm and the Minimum Spanning Tree;
- *computational cost*: the cut graph is achieved in $O(n)$ time if $g = 0$ and $O(n \log(n))$ time if the input surface has genus $g \geq 1$. Finally, the cut graph defines a reduced polygonal schema or with a constant number of redundant edges which depends only on g and b.

The rest of the chapter is organized as follows: in the following section we detail the core of the proposed framework, a discussion of its properties with respect to the state of the art in the field is given in Section 4.9.3, and future work concludes the chapter.

4.9 Cutting surfaces of arbitrary genus

We approach the search of a cut graph of a given triangle mesh $\mathcal{M}$ by reducing it to the following sub-problems (see Figure 4.21):

1. locate the g topological handles of $\mathcal{M}$ and cut each of them along one of its meridian loops which is then duplicated into two identical loops (see Section 4.9.1). Therefore, we achieve a 0-genus surface $\mathcal{M}^\star$ with $2g + b$ boundary components;
2. join the $2g + b$ boundary components in order to convert $\mathcal{M}^\star$ to a new surface $\overline{\mathcal{M}}$ with 0-genus and a unique boundary (see Section 4.7). Then, the polygonal schema of $\overline{\mathcal{M}}$ is calculated (see Section 4.9.2) and the surface is parameterized by using the shape-preserving parameterization. However, any other parameterization method for disc-like surfaces is meaningful [FH04].

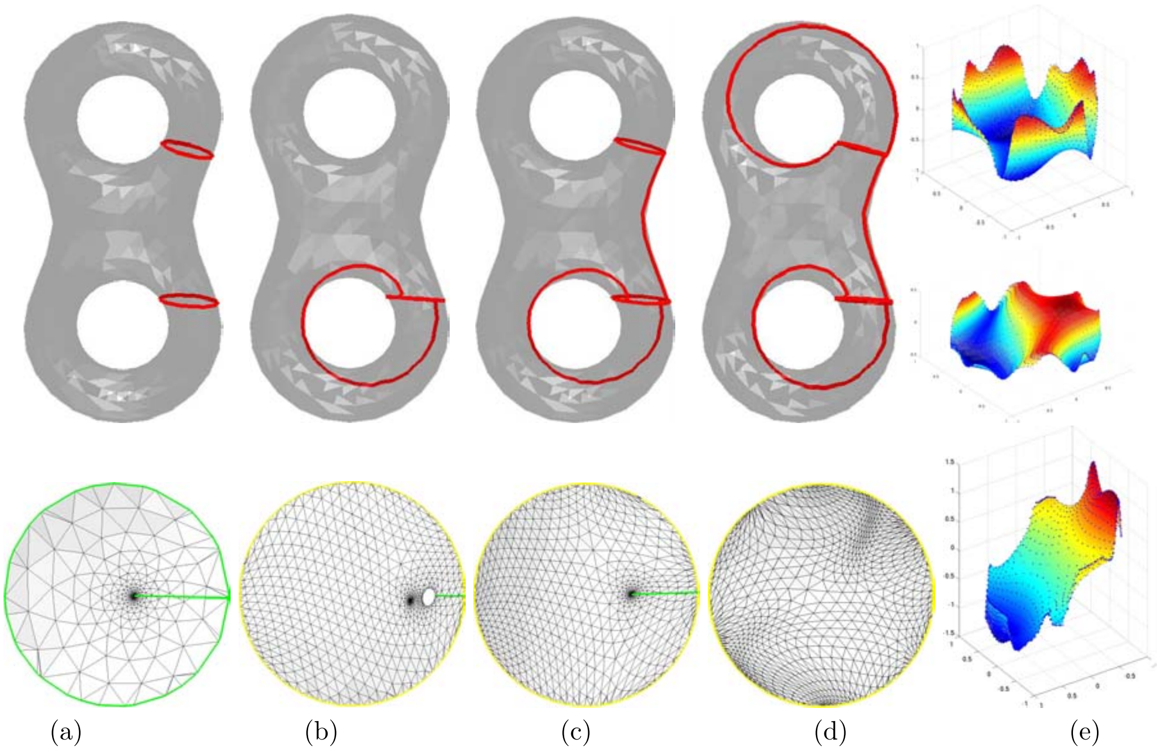


Figure 4.23: (a-d) Cut evaluation and shape-preserving parameterization of a surface of 0-genus with four boundary components. The first row shows the different steps on the 3D surface and the second row their 2D counterparts on the parameterization domain. (e) Tri-linear interpolation of the components x, y, and z of the vertices using the parameterization domain in (d).

4.9.1 Locating and cutting topological handles

The genus of $\mathcal{M}$, which is evaluated in linear time through the Euler formula, gives useful information for finding the cut graph. In fact, it is the maximum number of disjoint and non-separating *loops* $\gamma_1, \dots, \gamma_g$ on $\mathcal{M}$; that is, the set of closed curves such that $\gamma_i \cap \gamma_j = \emptyset$, $i \neq j$, and $\mathcal{M} - (\gamma_1 \cup \ldots \cup \gamma_g)$ is a connected 0-genus surface. It follows that we have to perform g cuts on $\mathcal{M}$ in order to reduce it to a 0-genus surface; however, the Euler formula does not give information for locating and cutting topological handles.

Our approach is based on the Reeb graph of $\mathcal{M}$ with respect to a mapping function $f : \mathcal{M} \rightarrow \mathbb{R}$ (see Chapter 3) which codes $\mathcal{M}$ as a graph whose cycles are used to identify the topological handles of $\mathcal{M}$, and the associated iso-contours correspond to the meridian curves along which the surface has to be cut for reducing its genus. From Proposition 6 at page 67, it follows that if $\mathcal{M}$ is a closed surface the choice of f does not affect the number of loops in the Reeb graph but only the shape of the iso-contours, that is, the meridians used for constructing the cut graph. If we consider a bordered surface some cycles of R_G may not be related to the topological handles of $\mathcal{M}$, but they are *spurious cycles* due to the boundary components; therefore, they have to be recognized and

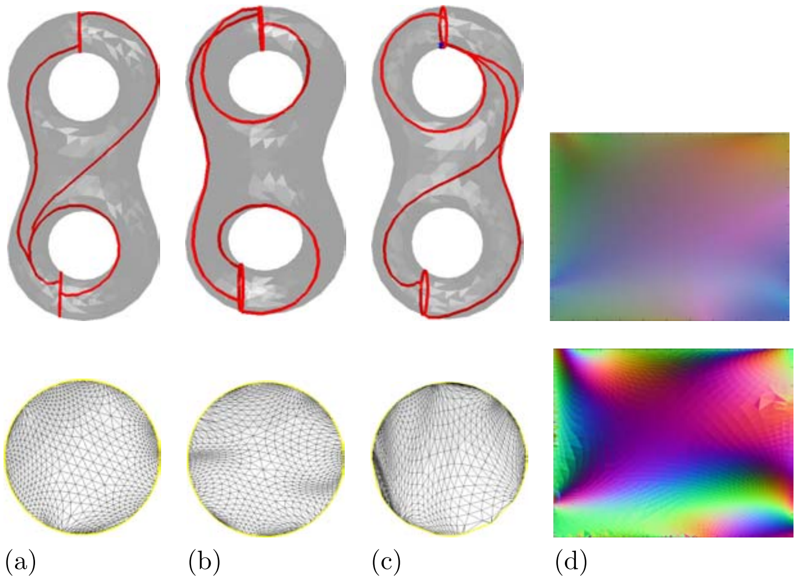


Figure 4.24: Cut graph of the bitorus; each column shows the cut and the corresponding embedding. (a) Bifurcations are admitted ($|\gamma| = 19.7$, $L^2(\mathcal{M}) = 13$), (b) bifurcations are discarded ($|\gamma| = 19.06$, $L^2(\mathcal{M}) = 20.8$), (c) a given source point is interpolated ($|\gamma| = 23.74$, $L^2(\mathcal{M}) = 59.9$). (d) Geometry and normal image with respect to the cut in (a).

discarded. As discussed in Section 3.2.1, each node i of R_G corresponds to a connected component of the iso-contours $f^{-1}(\alpha_i)$, $\alpha_i \in \mathbb{R}$, and its degree d_i is defined as the number of edges incident to i in R_G. A node i is classified as *terminal* if $d_i = 1$, as *internal* if $d_i = 2$, and as *branching* if $d_i \geq 3$. In all the examples, each node is represented by the barycenter of the connected component of the corresponding iso-contour (see Figure 4.21(a-b)).

We now describe how the topological handles of $\mathcal{M}$ are identified and cut without disconnecting the input surface while reducing its genus to zero (see Figure 4.22). Starting from a branching node b_n of R_G we walk on one of its branches along a path $\mathcal{R}$ of internal nodes until a terminal or a new branching node occurs. Let $\mathcal{M}_{\mathcal{R}}$ be the region of $\mathcal{M}$ around $\mathcal{R}$, that is, the sub-mesh delimited by the iso-contours associated to b_n and t_n, with t_n last node of $\mathcal{R}$. If t_n is a terminal node, $\mathcal{M}_{\mathcal{R}}$ has 0-genus [Ree46]; therefore, cuts will not be performed. In the case that t_n is a branching node, we distinguish two situations. If $\mathcal{M}$ is a closed surface, Proposition 6 guarantees that $\mathcal{M}_{\mathcal{R}}$ is part of a topological handle which can be cut by considering the iso-contour associated to any internal node of $\mathcal{R}$; for instance, the one of shorter length or associated to the node which is in the middle of $\mathcal{R}$ are possible choices. If $\mathcal{M}$ is a bordered surface, we have to decide if $\mathcal{M}_{\mathcal{R}}$ is a topological handle. In both cases, this decision is taken by considering the genus $\overline{g}$ of $\mathcal{M}$ after the cut; if $\overline{g} = (g-1)$ and $\mathcal{M}$ has not been disconnected, $\mathcal{M}_{\mathcal{R}}$ is sliced otherwise the handle cut is discarded. Then, the nodes of $\mathcal{R}$ are marked as visited and the next iterations consider all the other paths which outcome from b_n and not yet analyzed. The algorithm proceeds on the non-visited arcs of R_G until g cuts have been performed; therefore, the stop can happen before

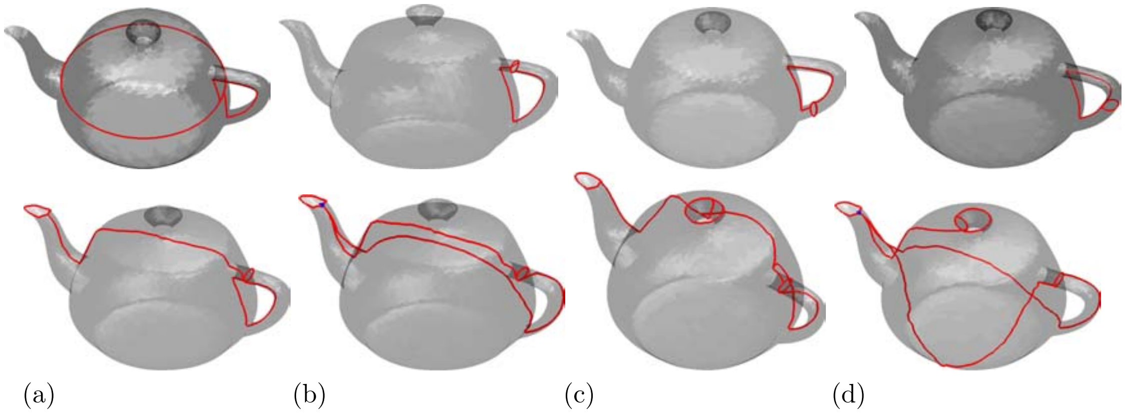


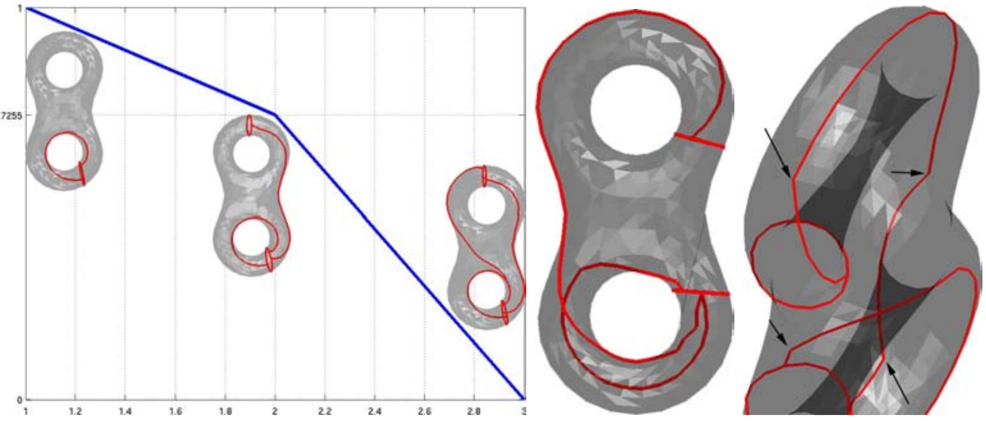
Figure 4.25: First row: family of cut graphs achieved by selecting different loops on a closed surface. Second row: cut graph (a, c) without bifurcations and (b, d) with a source (blue) point of a surface with one and two boundary components respectively.

visiting the whole graph and the computational cost of the loop search is sub-linear in the number of nodes of R_G.

With reference to Figure 4.21, starting from the branching node b_1 in (b) we walk on the arc of the Reeb graph until the terminal node t_1 is reached; the surface patch does not identify a topological handle of the bitorus and cuts will not be done. Then, we move from b_1 to b_2 along the arc on the right side and we perform a cut using one of the shown iso-contours thus reducing the genus to one and adding two boundary components. Let us now suppose that the algorithm visits the graph arc from b_2 to b_1 on the left; in this case, each iso-contour disconnects the surface and therefore will no be considered as a meaningful cut. The same discussion applies to the surface patch identified by the arc from b_2 to b_3, and finally a cut is performed along the arc b_3b_4. Since $g = 2$, the algorithm stops without continuing in the visiting process and its output is a surface of 0-genus with four boundary components (see (c)). By using the method discussed in Section 4.7, these handle cuts are converted into a connected boundary component (see Figure 4.23 and (d)) which is then used for the surface embedding on the parameterization plane (see (e)). Other possible cut graphs are shown in Figure 4.24.

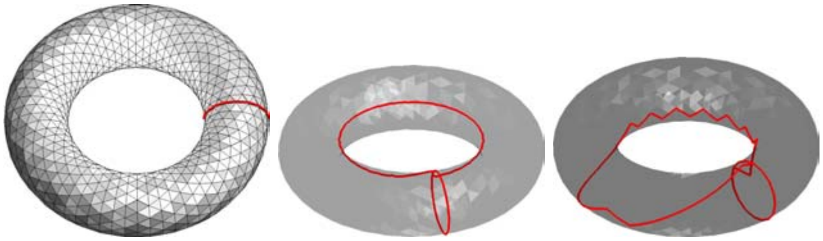
4.9.2 Global parameterization and polygonal schema

In this section, we provide the polygonal schema of a surface $\mathcal{M}$ with respect to the cuts described in the previous section proving that they guarantee a polygonal schema which is reduced or with a constant number of redundant edges which depends only on g and b. We remind the reader that if $g \geq 1$, $\mathcal{M}$ is converted to a 0-genus surface with $2g + b$ boundary components; therefore, we evaluate the polygonal schema of $\mathcal{M}$ through $\mathcal{M}^\star$ and $\overline{\mathcal{M}}$.



(a) (b) (c)

Figure 4.26: Cut, parameterization with shape-preserving weights, and normalized L^2-stretch values (y-axis) achieved by removing one boundary component at each step (x-axis). (b) Cut without bifurcations achieved as best compromise between cut length and induced parameterization distortion, (c) zoon-in where the arrows indicate sharp vertices ($|\gamma| = 12.26$, $L^2(\mathcal{M}) = 9.80$).



(a) (b) (c)

Figure 4.27: (a) Input surface with a smooth meridian loop, (b) cut graph achieved by using the proposed approach, and (c) by running the Dijkstra algorithm. The shape-preserving parameterization with respect to the cut graph in (b) is shown in Figure 4.19(b).

Cut-graph without bifurcations. Let us assume that the cut graph has been calculated by joining the boundary components of $\mathcal{M}$ without admitting bifurcations on the link paths. For calculating the polygonal schema, we can suppose without loss of generality that the boundary component γ_i has been linked to γ_{i+1} by the link path Γ_i, with $i = 1, \ldots, b-1$. Since each cut Γ_i has an opposite counterpart Γ_i^{-1}, we conclude that the polygonal schema has $2(b-1)$ identified edges corresponding to the pairs $(\Gamma_i, \Gamma_i^{-1})$ and γ_i has been subdivided by Γ_i into two sub-parts $\gamma_i^{'}$, $\gamma_i^{''}$ such that $\gamma_i^{'} \cup \gamma_i^{''} = \gamma_i$. Therefore, the polygonal schema of $\mathcal{M}$ is

$$\gamma_1\Gamma_1\gamma_2^{'}\Gamma_2\gamma_3^{'}\ldots\gamma_{b-1}^{'}\Gamma_b\gamma_b^{'}\Gamma_b^{-1}\gamma_{b-1}^{''}\ldots\gamma_2^{''}\Gamma_1^{-1}$$

which corresponds to a $(4b-4)$-sided polygon. From (4.31), it follows that if $g = 0$ the polygonal schema has $b-2$ redundant edges, and if $g \geq 1$ this schema is not canonical but it has a constant number (i.e., $4g + b - 4$) of redundant edges.

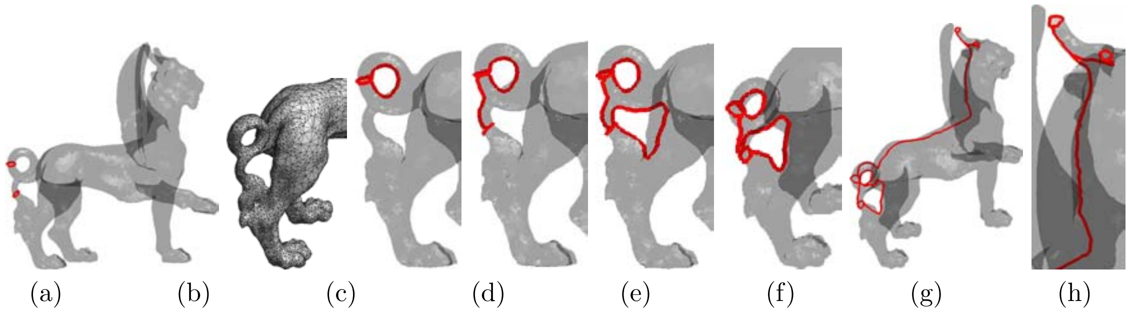


Figure 4.28: (a) Cut of the topological handles of the feline ($g = 2$, $b = 0$), (b) local irregular connectivity, (c-e) the three steps which convert the surface (a) of genus zero and with four boundary components to (e) a disc-like surface, (f) zoom-in. (g-h) Cut graph on the feline model ($g = 2$, $b = 2$) with two boundary components located on its korns.

Cut-gaph interpolating a source point. In this case, all the link paths share a common point $p \in \gamma_1$; therefore, the polygonal schema is

$$\gamma_1\Gamma_1\gamma_2\Gamma_1^{-1}\Gamma_2\gamma_3\Gamma_2^{-1}\dots\Gamma_{b-1}\gamma_b\Gamma_{b-1}^{-1}$$

which has $3b$ edges. The number of redundant edges is $2g$ and the schema is canonical if $g = 0$.

4.9.3 Discussion

Calculating the meridians of the topological handles as iso-contours associated to the loops of the Reeb graph guarantees that they are smooth curves because pre-images of a fair function f. This set of possible cuts for the topological handles of $\mathcal{M}$ provides several degrees of freedom for reducing $\mathcal{M}$ to a 0-genus surface $\mathcal{M}^\star$ with $2g + b$ boundary components. Furthermore, as previously discussed, these boundaries can be joined in different ways to define a *family of cut graphs* for $\mathcal{M}$ (see Figure 4.25). Figure 4.26 shows the reduction of the L^2-stretch of $\mathcal{M}$ with respect to the current shape-preserving parameterization while the cut loops of the topological handles are joined for reducing $\mathcal{M}^\star$ to a disc-like surface $\overline{\mathcal{M}}$; we refer the reader to Section 4.7.1 for the discussion on the procedure which joins the $(2g + b)$ boundary components of $\mathcal{M}^\star$. As shown in Figure 4.27, dealing with smooth cut loops and a surface with regular connectivity does not ensure that the Dijkstra algorithm builds smooth cut graphs. Additional examples of cut graphs of closed triangle meshes are given in Figure 4.28(a-f), 4.29, 4.30, 4.31, 4.32, while Figure 4.28(g-h) takes into account a bordered surface.

4.10 Conclusions

The first part of the chapter has discussed an extension of the previous work on local parameterization where the constraint of having disc-like patches has been relaxed to deal with 0-genus

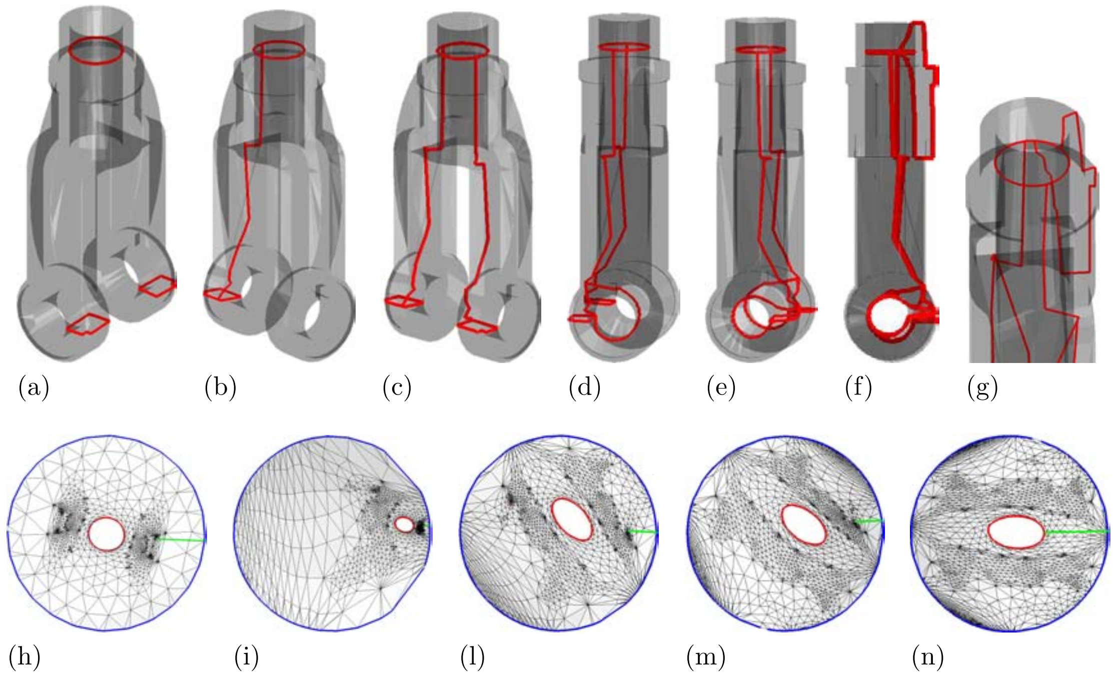


Figure 4.29: (a) Cut loops around the topological handles of a surface with genus three. After the cut and duplications of each curves, the six boundary components are joined step by step as described by the couples of images: (b,h), (c,i), (d,l), (e,m), and (f,n). (g) Zoom-in on the upper part of the cut graph.

patches. The proposed framework defines a user-independent model for graph-based parameterization of 3D shapes into a minimal number of maximal charts, and it is associated to a centerline skeleton (aligned with the parameterization), which stores geometric information as node labels. The patch identification locates regions, which are global features of the input shape; in fact, cones and cylinders represent protrusions, while bodies are junctions among them. As discussed in Chapter 5, the possibility of working with different segmentations of the same surface reduces the number of patches with respect to the atlas generation based on the mesh simplification, and thus improves the regularity and smoothing of the remeshing through the boundaries of adjacent regions, which usually require a specific identification and treatment. The adaptive surface segmentation enables to reduce the distortion in the parameterization of each region R_i with benefits on the remeshing; finally, the localization of extraordinary vertices is an improvement with respect to previous methods which do not control their location on the surface.

The proposed feature-based parameterization enables a greater degree of flexibility on the alignment of patches along shape features and it is based on a generalization of the parameterization proposed in [Flo97] for embedding 0-genus charts with an arbitrary number of boundary components. We provided several segmentation methods based on topological and geometric properties which give a partition of surfaces with arbitrary genus meaningful for the proposed approach. The

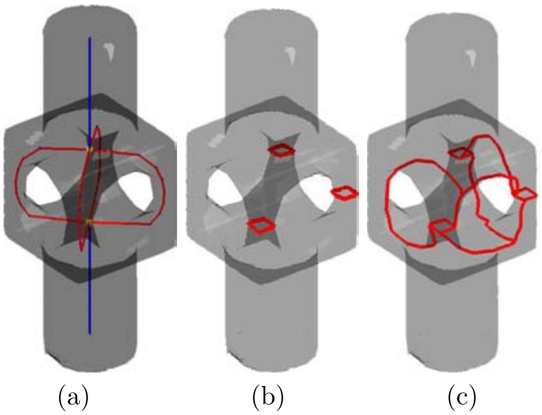


Figure 4.30: (a) Reeb graph with respect to the height function of a surface of genus three, (b) cuts of the topological handles, and (c) cut graph without bifurcations of the link paths.

choice of using the shape-preserving parameterization as the main element of the framework and in all the examples was intended to focus the attention on the generality of our approach; however, any other parameterization method of disc-like surfaces is meaningful with possible specializations and optimizations of its single steps. The procedure used for joining the boundary components enables to add constraints on the final cut; for instance, cuts interpolating feature points and lines are important for remeshing while cut graphs which share a common source point are used for extracting *reduced polygonal schemes*.

In the second part of the chapter, we discussed a computational method for calculating cut graphs for the global parameterization of bordered surfaces with an arbitrary genus proving that it extracts reduced cut graphs in linear time if $g = 0$, and with a constant number or redundant edges in $O(n \log(n))$ time if $g \geq 1$. In this last case, applying standard reduction operations [VY90] it is possible to convert redundant cut graphs to a canonical form. The proposed framework builds on the Reeb graph R_G which codes in a compact form the topology of a given surface $\mathcal{M}$ and stores additional information such as the critical points, the genus of $\mathcal{M}$ as number of cycles, and the iso-contours of $(\mathcal{M}, f)$. In the same way, a cut graph γ is an alternative high-level representation of $\mathcal{M}$ whose genus and boundary components correspond to a set of loops and link paths together with geometric properties such as minimal length, interpolation of a given source point. We note that the reduced polygonal schema can also be considered as the counterpart of the Reeb graph induced by a smooth function with a minimal number of critical points as proposed in [NGH04]. If we suppose that $\mathcal{M}$ is a closed surface, R_G has g loops while the cut graph contains $2g$ loops related to the g iso-contours (i.e., meridians) of $(\mathcal{M}, f)$ that have been cut and duplicated for converting a bordered $\mathcal{M}$ to a 0-genus surface. The way these loops are joined can be different and the resulting link paths not always are along each topological handle (i.e., longitude). Therefore, the most interesting extension consists of evaluating a homology base of an arbitrary surface $\mathcal{M}$ and it is based on the possibility of mapping the surface $\mathcal{M}^\star$ to a planar domain whose internal boundary components are constrained to have a pre-defined position on the parameterization domain.

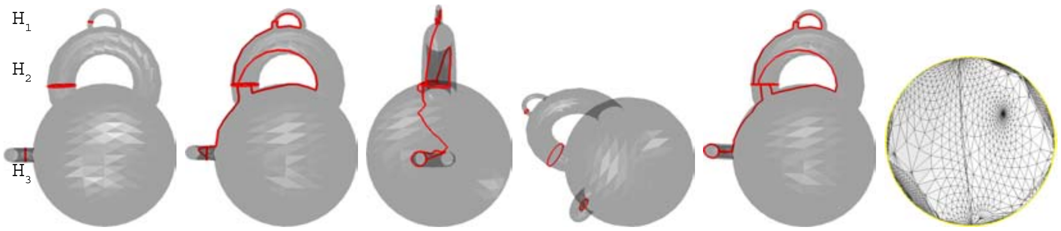
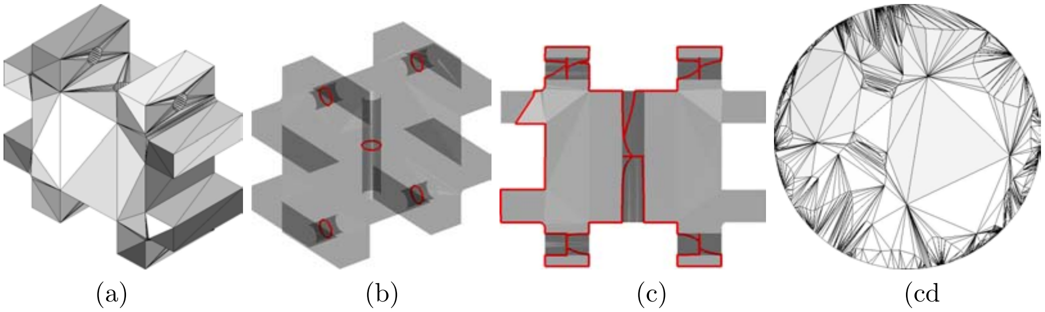


Figure 4.31: Two different cut graphs on a surface of genus three achieved by cutting H_3 around and along the topological handle.



(a) (b) (c) (cd

Figure 4.32: (a) Input dataset of genus five, (b) cuts of its topological handles, (c) cut graph, (d) planar embedding.

Related pubblications

1. G. Patanè, M. Spagnuolo, B. Falcidieno. "Para-Graph: graph-based parameterization of triagnle meshes with arbitrary genus". In *Computer Graphics Forum, 2004*, vol.23 n.4, pp.783-797.

2. G. Patanè. *Parameterization of 3D triangle meshes with arbitrary genus.* Rapporto Tecnico N. 7/2004, Istituto di Matematica Applicata e Tecnologie Informatiche, Consiglio Nazionale delle Ricerche.

Chapter 5

Applications and conclusions

This chapter presents possible applications related to the local and global parameterization; more precisely, in Section 5.1 we present a simple method for extracting a local skeleton of a given surface starting from its local parameterization and used for performing shape deformation. Section 5.2 discusses the remeshing of triangle meshes, and Section 5.3 presents conclusions and future work of the thesis.

5.1 Local surface deformation

We describe the extraction of a skeleton of a surface $\mathcal{M}$ with arbitrary genus, which is locally induced by the local parameterization $\mathcal{G} := \{(R_i, \varphi_i, \Omega_i)\}_{i=1,\dots,m}$ (see Section 4.7.2) and used for local shape deformations. More precisely, each patch R, which has to be deformed, is *injectively* mapped onto a skeletal line Σ *aligned with its parameterization*

$$\begin{array}{rccl} \varphi : & R & \rightarrow & \Omega \\ & p_i & \mapsto & u_i := (u_{i,s}, u_{i,t}). \end{array} \tag{5.1}$$

Furthermore, Σ has the same complexity as R (i.e., the number of vertices of Σ is equal to that of R) and R is simply coded as a displacement field with respect to Σ (see Section 5.1.2).

More in details, for extracting the skeleton Σ of a conical primitive (see Figure 5.1, first row), we consider as Ω the unit circle with a set of circles $\{S(0, r_i)\}_i$ centered in the origin and with radii $0 \leq r_i \leq r_{i+1} \leq 1$; then, Σ is built by joining the barycenters of the curves $\varphi^{-1}(S(0, r_i))$ on $\mathcal{M}$. We can optimize the previous step by choosing $\varphi(p)$ as center of the circles, where p is a critical point of $(\mathcal{M}, f)$ (see Figure 5.1, second row).

For extracting a skeleton Σ of a cylindrical primitive (i.e., $k = 2$), we choose as Ω the unit square $[0, 1] \times [0, 1]$ where the cuts Γ and Γ^{-1} are mapped onto the line segments l_2, l_4, while the two boundary components γ_1, γ_2 are identified with l_1, l_3 respectively (see Figure 5.2(a-b)).

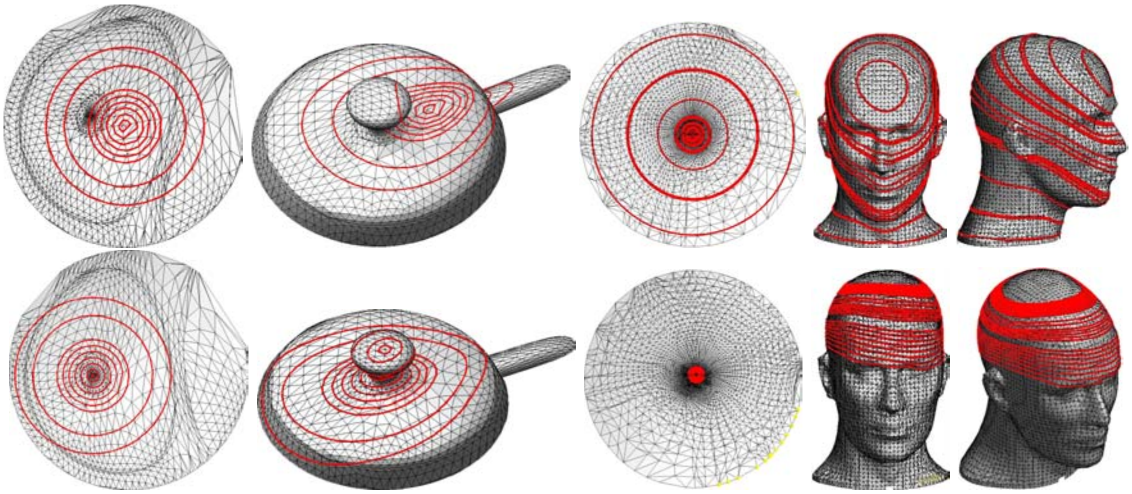


Figure 5.1: First row: set of concentric circles on the parametric domain and their corresponding curves on two disc-like surfaces. Second row: sections with respect to the critical point (f is the height function).

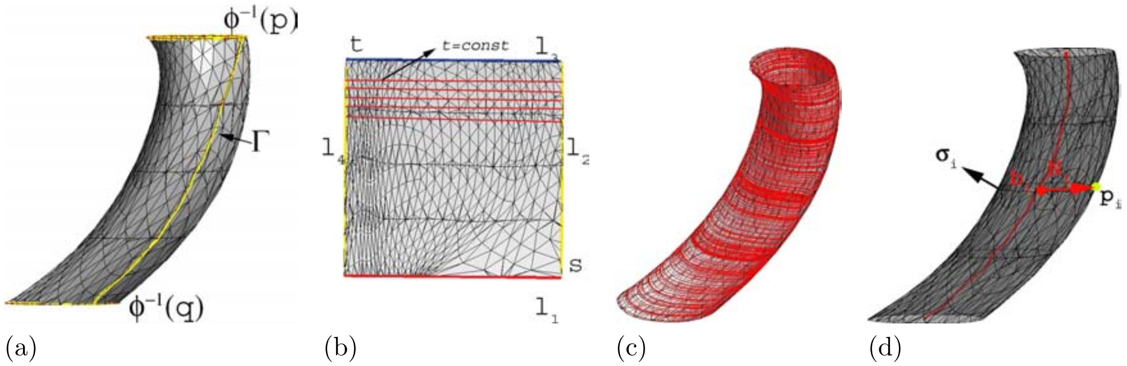
First of all, we associate to the primitive a *coordinate network* of lines; each loop is achieved as $\varphi^{-1}(\{t = constant\})$ and each parallel as $\varphi^{-1}(\{s = constant\})$. Then, each loop is collapsed to its barycenter thus defining a skeletal line whose number of nodes can be refined by increasing the number of loops in spite of the mesh density and connectivity (see Figure 5.2(c-d)). The coordinate network is useful for visualizing the presence and location of distortions induced by the parameterization and due to the selection of a specific set of weights in (4.12) or to a particular cut (e.g., along feature lines) for the unfolding. Finally, Γ and a section $\varphi^{-1}(\{t = constant\})$ represent the topological generators of the cylindrical patch.

We explicitly underline that chosen a conical (resp., cylindrical) primitive R embedded by (5.1), its skeletal line Σ is aligned with the parameterization in the sense that Σ is achieved by joining the barycenters of the iso-curves with respect to $f(x) := \|\varphi(x)\|_2, \forall x \in R$ (resp., $f(x) := pr_2(\varphi(x))$, with $pr_2(s,t) = t, \forall (s,t) \in \mathbb{R}^2$). This property enriches existing methods based on the Reeb graph [FK97] with the information and applications of parameterization; we thus provide a tool which uses topology for the patch decomposition and geometry for the parameterization. Another example is given in Figure 5.3.

A body primitive (i.e., $k \geq 3$) is segmented by clustering those vertices which are closest to the same boundary component of $\Omega^{'}$, that is, a vertex p is associated to the boundary β_s such that

$$\|p - pr_{\beta_s}(p)\|_2 = \min_{j=1,\dots,k}\{\|p - pr_{\beta_j}(p)\|_2\}$$

where pr_{β_j} is the orthogonal projection onto the convex curve β_j. Each achieved sub-patch $\tilde{R}_j$ is a *Voronoi*-like region (for the definition, see Section 1.2) which includes β_j, $j = 1,\dots,k$, (see Figure 5.4); if $\tilde{R}_j$ has two boundary components, we extract the skeletal line by using the method previously discussed, otherwise it is a body primitive and we perform a different step depending



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

Figure 5.2: (a) Cut Γ on a cylindrical primitive R and (b) its embedding onto the unit square where the edges l_2, l_4 are identified with Γ. The parallel circles on R are mapped onto parallel lines in Ω which are orthogonal to the edge l_2. The parameterization domain is used for building the iso-contours σ_i (red curves) in (c), whose barycenters b_i correspond to the nodes of the skeletal line in (d). We note that each iso-contour interpolates a given vertex p_i of the primitive. The example shows the stability of the cut identification and parameterization in spite of the high-curvature and irregular connectivity of R.

on its area $a(\tilde{R}_j)$. If $a(\tilde{R}_j)/a(R) < \epsilon$ with ϵ given threshold, we build a skeleton by connecting its center of mass with the barycenters of the boundary components; otherwise, a new segmentation step is performed on the current sub-patch. This procedure stops when all the sub-regions are cylinders or the area of bodies is less than $\epsilon a(R)$.

In Section 5.1.1, we perform shape transformations which affect the object geometry and supervised by a check of the topological meaningfulness of the editing operations; for example, a cylindrical primitive can be modified by altering its sections but cannot be replaced by shapes of different genus. From now on, the term feature corresponds to a region on the input surface identified during the segmentation or selected by the user. In the last case, if $k = 1$ we consider $\varphi(p)$, with p point selected by the user, as center of the circles in Ω; for simplifying the discussion and without loss of generality, we suppose that circles are centered in $0 \equiv (0, 0)$.

5.1.1 Surface deformation

A standard approach for deforming a surface $\mathcal{M}$ or finding a correspondence between two polyhedra is to define a common embedding on a high-level description such as a multi-resolution representation [KCVS98, ZSS97], the Medial Axis [Blu67], the Reeb graph [FK97], or the parameterization domain [FG99].

In *multi-resolution modeling*, the intrinsic representation of $\mathcal{M}$ is the base mesh $\mathcal{M}_S$ with respect to a given level of detail specified by the user and the geometric details are encoded with respect to $\mathcal{M}_S$. Using mesh decimation [KCS98] instead of nested spaces related to subdivision schemes [ZSS97] ensures a greater degree of flexibility without assumptions on the regularity of the mesh connectivity. In both cases, modifications are applied to $\mathcal{M}_S$ which has to include all the structural

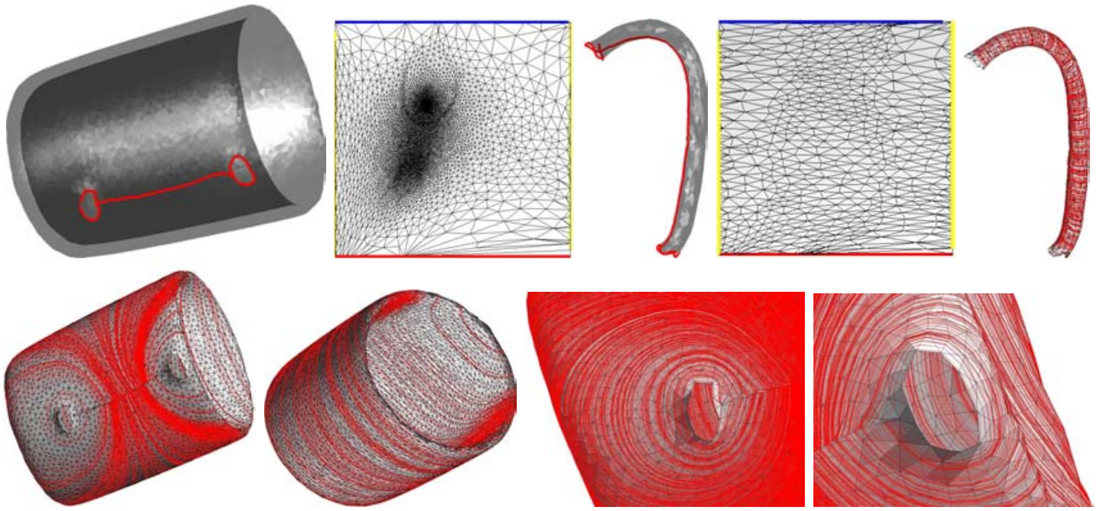


Figure 5.3: First row: segmentation induced by *Plumber*, cut, and parameterization of the two cylindrical patches which compose the input surface. Second row: iso-contours induced by the parameterization and zoom-in on the iso-contours around the two boundary components.

details needed for the deformation; then, local details are added thus avoiding to create a bottle neck in real time operations.

Drawbacks of using descriptors such as the Medial Axis [YBS03] are that tiny perturbations of the surface can produce a new element in the representation without distinction between local and global details. Furthermore, the presence of both lines and surfaces in the Medial Axis gives a correspondence between the surface and its abstraction which can be unnatural for the user, and the pruning process required to get a stable representation can create loss of details and artifacts during the deformation.

In implicit modeling [BW97], skeletons, considered as a collection of elements with associated implicit primitives, provide a compact representation that is useful for defining both motion and deformation. In fact, during animation the skeleton attributes may change, varying surface details, or blend. Moreover, depending on the kind of manipulation (e.g., animation, metamorphosis, growth, etc.) skeletal elements may rotate, stretch, appear, or disappear [Blo02, CGC^{+}02, LCF00]. Linear skeletons, such as the Reeb graph, overcome some of the problems related to the Medial Axis but they do not provide an injective correspondence between the surface vertices and the graph nodes [FK97].

Deforming or morphing 3D surfaces by using their planar embeddings requires to deal with patches of 0-genus with the same connectivity [FG99]. If a global approach is used, the identification of features on the parameterization domain is difficult and complicated by the cut that can distort and/or disconnect features to be deformed or morphed. The local parameterization, as approached by previous work, identifies patches which are not meaningful for deformation because they do not represent features of the input surface.

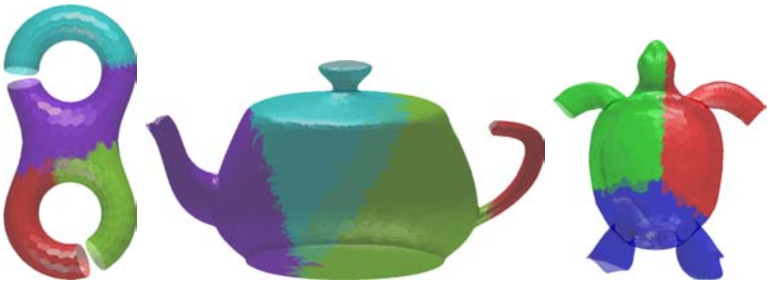


Figure 5.4: Voronoi-like regions of the body primitives depicted in Figure 4.12.

5.1.2 Skeleton-driven deformation

A simple scheme for surface modeling is given by the *handle metaphor* [WW92], where the user selects a *region of interest* on the surface $\mathcal{M}$ containing a second region R, i.e. the *handle*, which can be moved, rotated, and scaled. The boundary of the region of interest is built as the geodesic line connecting the surface vertices picked during the patch selection. The user is free to manipulate the handle and the mesh in the region of interest is updated using a constrained energy [KCVS98] or a Laplacian-based functional minimization [SLCO$^+$04].

The core of our approach (see Figure 5.1 and 5.2) is based on the encoding of each vertex p_i of R relative to the barycenter of the iso-contour σ_i induced by $\varphi : R \rightarrow \Omega$ and interpolating p_i. In this way, we derive an intrinsic and affine-invariant coordinate system represented by a linear skeleton Σ whose vertices are the barycenters b_i of the curves σ_i. Each skeletal line Σ of a conical (resp., cylindrical) patch aligned with its parameterization (5.1) has a *multi-resolutive structure* which is achieved by refining the number of circles (resp., parallel lines) in Ω. For deforming R, it is sufficient to code each of its vertices into one node of Σ; to this end (see Figure 5.1 and 5.2(d)), we consider the map

$$\Theta : R \rightarrow \mathbb{R}^3, \quad \Theta(p_i) = b_i$$

where b_i is the barycenter of the 3D curve $\sigma_i := \varphi^{-1}(S(0, \|u_i\|_2))$ (resp., $\sigma_i := \varphi^{-1}(\{t = u_{i,t}\})$) on R. We note that given two vertices $p_i, p_j \in \mathcal{M}, i \neq j$, the curves σ_i, σ_j do not have multiple self-intersections even though they can coincide when $\|u_i\|_2 = \|u_j\|_2$ if $k = 1$, or $u_{i,t} = u_{j,t}$ if $k = 2$; this property guarantees that Θ is injective "almost" everywhere. This process brings to the identity

$$R = \Sigma + DN \tag{5.2}$$

where $D := (d_i)_{i=1,\dots,n_V}$, $d_i := \|p_i - b_i\|_2$ is the set of distances of the vertices of R from Σ and $N := (N_i)_{i=1,\dots,n_V}$, $N_i := \frac{p_i - b_i}{d_i}$ is the unit displacement vector of R from Σ. Clearly, Σ can be approximated by a B-spline [GL96] in order to work with a continuous representation instead of a polygonal line; this step is useful for producing deformations on the skeleton guided by physically-based animations. Through the relation (5.2), deforming R is equivalent to modify the skeletal line Σ and the vector D; the displacement field N is fixed and its role is to map changes on (Σ, D) to R. Examples are given in Figure 5.5 and 5.6; Table 5.1 gives the computational cost of the

Table 5.1: Computational cost of the main steps of the proposed framework; n_V (resp., n) is the number of vertices of the input mesh (resp., cut). For the fuzzy clustering, I is the number of iterations.

Method	Comptationa cost	Reference
Shape decomposition		
Reeb graph	$O(n_V \log(n_V))$	[BMMP03]
Plumber	$O(n_V^2)$	[MPS+04b]
Fuzzy clustering	$O(n_V^2 \log(n_V) + In_V^2)$	[KT03]
Cut & unfolding		
Approx. geodesic	$O(n)$	
Geodesic	$O(n_V \log(n_V))$	[Dij59]
Unfolding	$O(n_V)$	[Flo97]
Deformation		
Reeb Graph	$O(n_V \log(n_V))$	[BMMP03]
Medial Axis	$O(n_V^2 \log(n_V))$	[YBS03]
Shape-graph	$O(n_V)$	

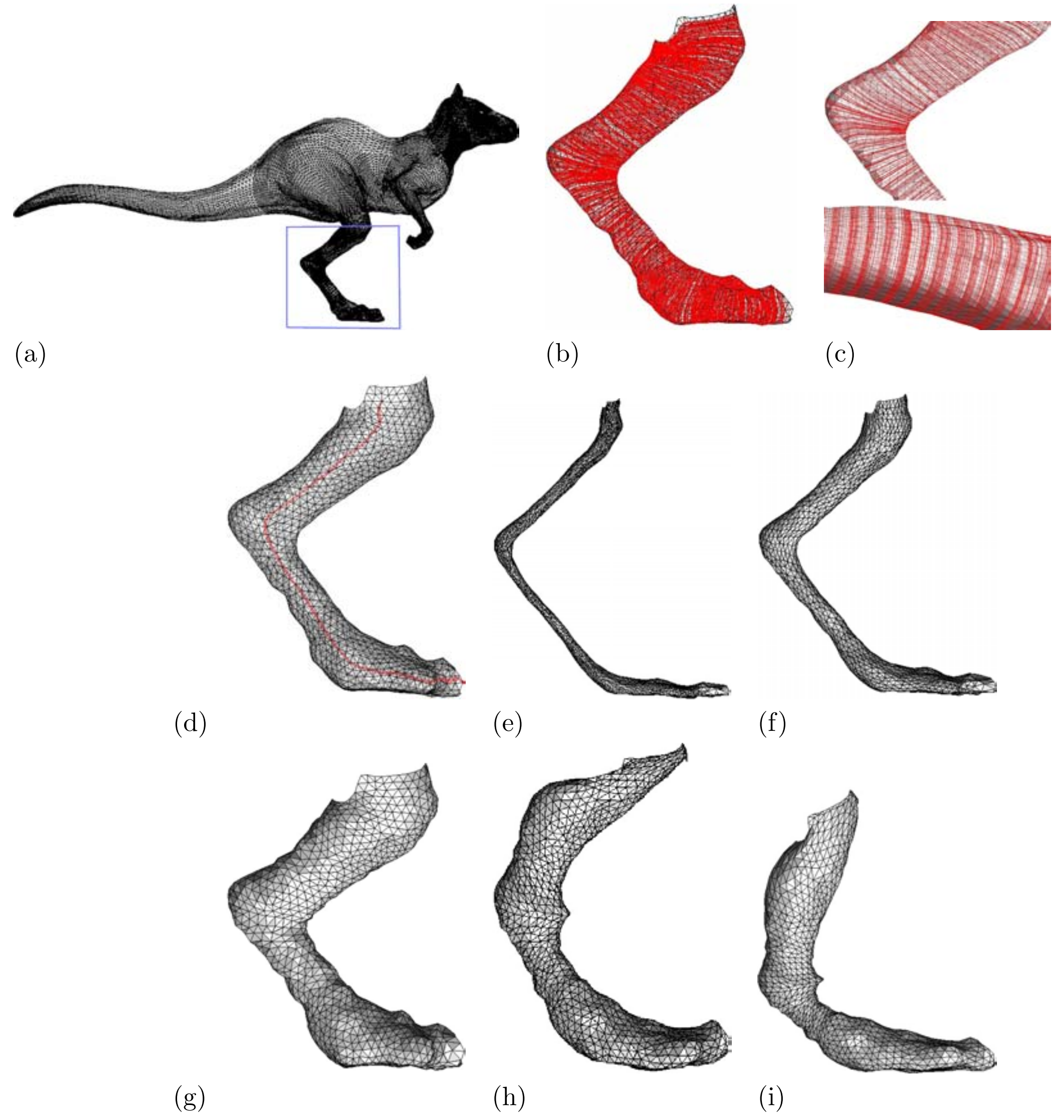
main steps of the proposed framework.

Strong deformations can create intersections between parts of the input surface; therefore, a simple method for recognizing these degenerate situations is necessary. We give a simple solution to the problem by considering a *maximum* and a *minimum envelope* of the patch and built as follows. Since each vertex of R is coded in Σ, we approximate the corresponding iso-contour σ_i with the maximum (resp., minimum) circle of center b_i and radius $\max_{p\in\sigma_i}\{\|p-b_i\|_2\}$ (resp., $\min_{p\in\sigma_i}\{\|p-b_i\|_2\}$). Testing self-intersections is thus reduced to check the Euclidean distance among points of Σ.

5.2 Remeshing

The advantages of using the proposed method for remeshing are discussed in this section. To this end, we review the remeshing techniques commonly used to approximate a given surface $\mathcal{M}$ with a new triangulation $\overline{\mathcal{M}}$ which has a specific connectivity (i.e., regular, semi-regular) and geometry (i.e., anisotropic, isotropic vertex sampling).

Given a disc-like surface $\mathcal{M}$ parameterized by $\varphi : \mathcal{M} \to \Omega$, a *regular remeshing* [EDD+95, HLG01] defines a base mesh S^0 with a minimal number of triangles which are subdivided to obtain a sequence $S^0, S^1, \ldots, S^r$ of triangulations with subdivision connectivity and contained in Ω. The iteration proceeds until S^r (resp., $\varphi^{-1}(S^r)$) is a good approximation of Ω (resp., $\mathcal{M}$) with respect to some geometric error (e.g., L^2 norm). Therefore, this method uses the same number of samples on the parametric domain in spite of the presence in Ω of triangles with a different area. As a result,



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 5.5: (a) Conical (blue box) primitive, (b-c) iso-contours induced by its parameterization, (d) skeletal line. Deformations are achieved by altering (e-g) the vector D, and (h-i) the skeletal line of the leg. From (5.2), it follows that the primitive converges to its skeleton Σ when D tends to zero; clearly, the previous modifications can be combined to give a more flexibility to the user.

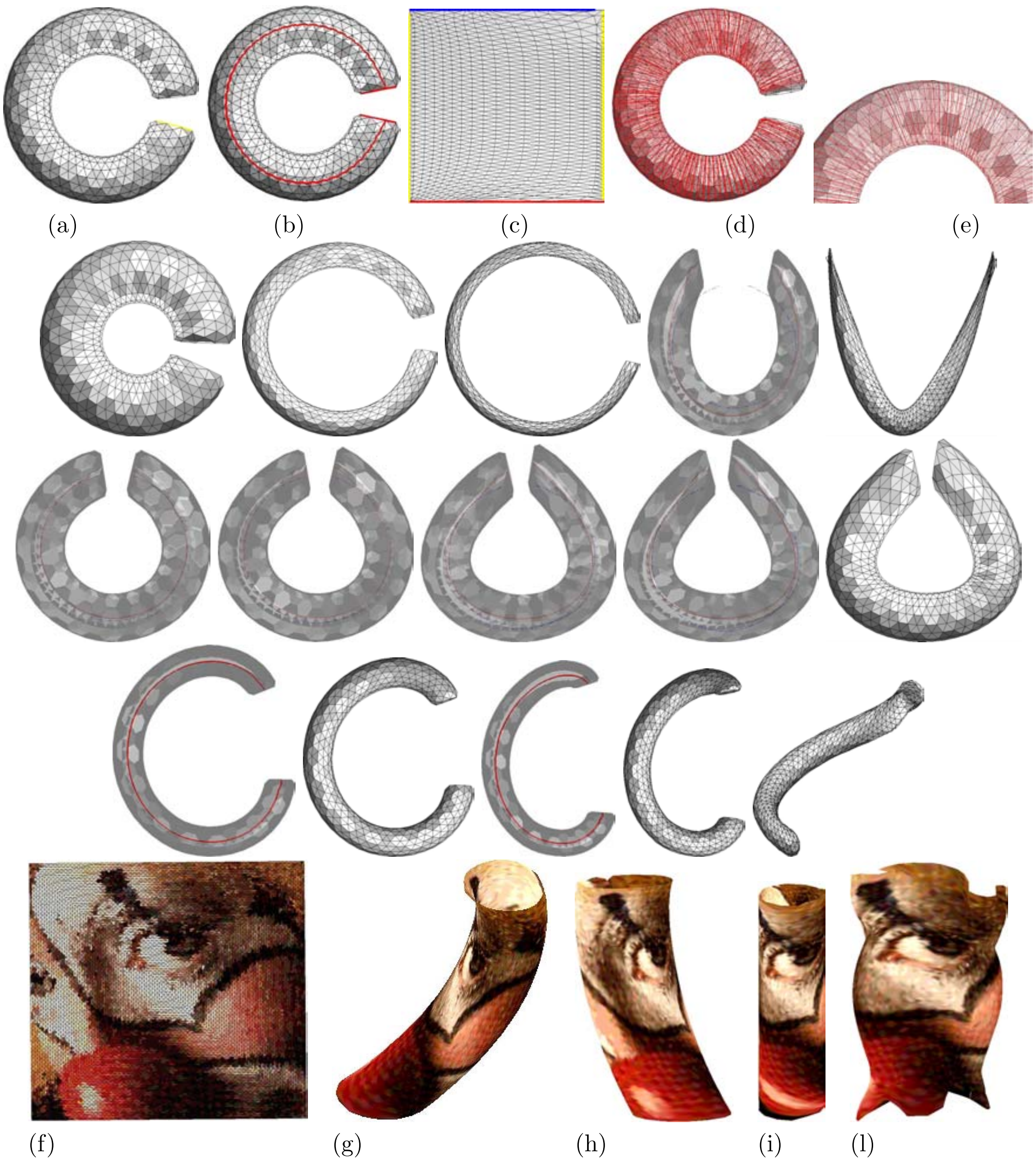


Figure 5.6: First row: (a) input cylindrical primitive, (b) cut, (c) shape-preserving parameterization on the unit square, (d) ring network, (e) zoom-in. From the second to the fourth row: deformations achieved by affecting the skeletal line and the radius of the primitive: the marked (resp., dotted) skeletal line represents the target (resp., input) skeleton. Fifth row: non-uniform texture mapping and deformation of the cylindrical primitive R in Figure 5.2. (f) Discretization of a non-uniform texture on a regular grid, (g-h) different views of its mapping on R which is successively deformed (i) to a cylinder with boundary components of the same length. In (l), deformation of (g) into a cylinder with irregular shape.

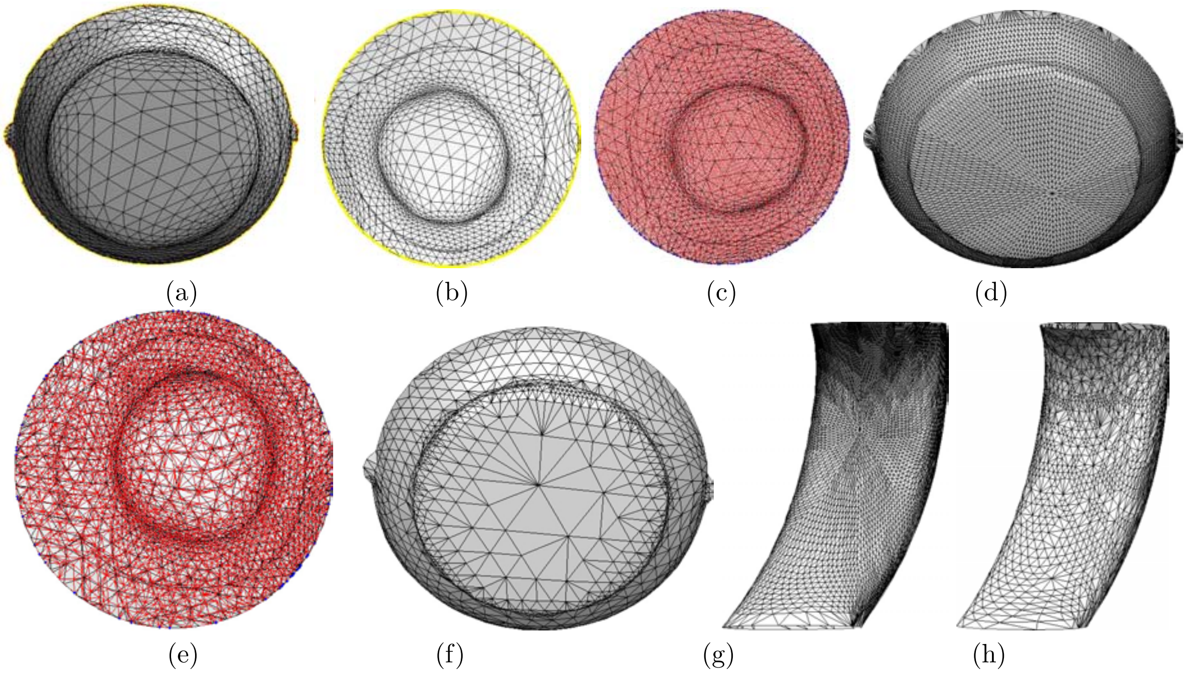


Figure 5.7: (a) Input conical primitive, (b) shape-preserving parameterization, (c) regular grid on the parameterization domain, and (d) it counterpart on $\mathcal{M}$. (e-f) Adaptive remesh of (a). The examples (e) and (f) show the uniform and adaptive remesh of the cylinder in Figure 4.8.

we have an oversampling of those triangles with a greater area and an undersampling in regions where small local features are located. To overcome this drawback, two strategies are available. The first one consists of re-parameterizing Ω to a new domain $\overline{\Omega}$ with the aim of equalizing the area of each triangle; internal vertices are updated using an area-based smoothing [HLG01] or a relocation strategy [SG03]. This optimization step can produce inverted triangles and in the case of small features it does not avoid a large number of subdivision steps.

To overcome the exponential growth of the number of vertices, the second choice is an *adaptive remeshing* $(S^k)_k$ of Ω which localizes on the current remesh S^k those triangles where the approximation error between $\mathcal{M}$ and $\varphi^{-1}(S^k)$ exceeds a given error ϵ and which have to be split [LSS+98]. A triangle $t \in S^k \subseteq \Omega$ is subdivided if the following condition

$$E(t) := \max_{p_i \in t, p_i \in \Omega} \{dist(\varphi^{-1}(p_i), \pi)\} > \epsilon$$

holds, where π is the plane defined by the triangle $\varphi^{-1}(t)$. After this phase, the mesh S^{k+1} is achieved from S^k by using the red-green triangulation [BSW83]. The iteration terminates when the local error $E(t)$ does not exceed ϵ for all $t \in S^r$; finally, $\varphi^{-1}(S^r)$ is evaluated combining the point plane location and barycentric coordinates.

Local remeshing [EDD+95, GVSS00, LSS+98] is based on partitioning 3D meshes into disc-like patches, each one is parameterized and then remeshed. Main problems of this approach are the lack

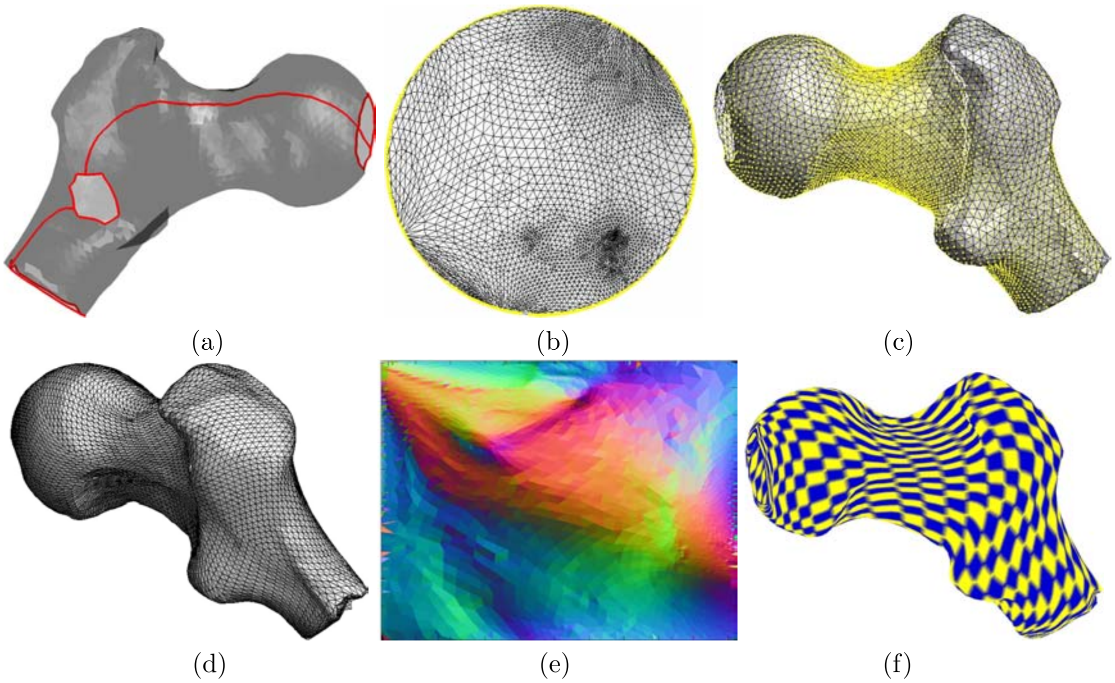


Figure 5.8: (a) Cut on a surface of 0-genus with three boundary components, (b) unfolding onto the unit circle (shape-preserving weights), (c) regular sampling, (d) regular remeshing, (e) normal-map image (1024×1024) [GGH02], (f) texture mapping.

of symmetry in the patch decomposition and the dependence of its structure from the simplified mesh $\mathcal{M}_S$ which defines the atlas identification. The regular connectivity of each patch cannot be extended to the entire mesh and extraordinary vertices are “randomly” located on the remeshed surface $\overline{\mathcal{M}}$.

Global remeshing [AMD02, GGH02, HLG01] re-samples the global parameterization of $\mathcal{M}$ (see Section 4.8) with a regular or an adaptive grid and thus treats the original mesh as a whole while guaranteeing the regularity of the connectivity. The problems which affect the definition of a “good” global parameterization, i.e. the choice of the cuts for the unfolding of surfaces with an arbitrary genus and the parameterization distortion of regions with a high-curvature, are reflected on the final result.

An alternative to both approaches is to work directly on the surface with an iterative procedure which modifies local regions on $\mathcal{M}$ [Hop96, SG03], usually the 1-star of each internal vertex, and controls their smoothness and sampling rate. This *mesh adaptation process* reduces the computational cost and avoids the drawbacks of decomposing and cutting $\mathcal{M}$; however, the locality of the approach does not guarantee to avoid error accumulation through feature lines and to effectively optimize the global sampling of the original surface.

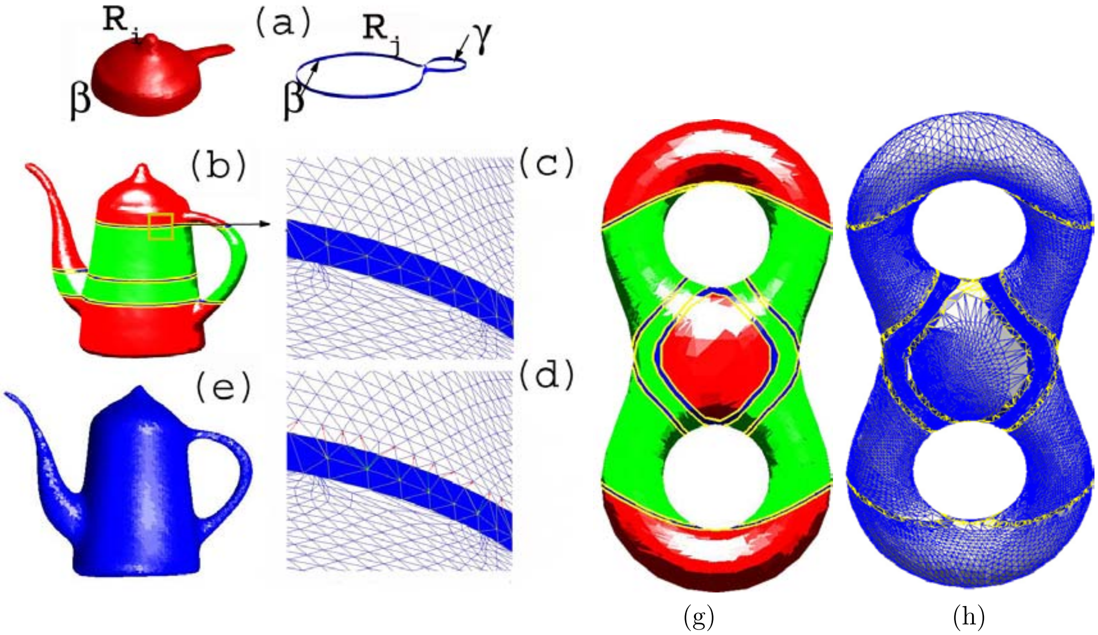


Figure 5.9: (a) Adjacent patches (R_i is a conical primitive and R_j is a body with four boundary components), (b) topological segmentation, remeshed dataset (c) before and (d) after the update of the common boundary, (e) final result. (g) Shape segmentation with f Euclidean distance from the barycenter. (h) Uniform local remeshing of a bitorus; yellow curves locate boundary components between adjacent regions.

5.2.1 Feature-based local remeshing

We apply the graph-based parameterization for the remeshing of a 3D shape with an arbitrary genus and density. As refinement operator for uniform remeshing we consider the 1-to-4 operator that recursively subdivides each triangular face into four sub-triangles by introducing three new vertices on the edges. In this way, each remeshed patch has an implicitly-defined connectivity which is exploited in several tasks such as compression, progressive transmission, and multi-resolution editing. The adaptive remeshing is based on the method presented in [LSS$^+$98]. We first discuss how each primitive is remeshed by taking into account their different parameterizations and then combined to give a coherent and global remesh of the input surface.

Remesh of conical primitives. Since $S^r \subseteq \Omega$, we ensure the remeshing of $\partial\Omega$ by projecting ∂S^r onto $\partial\Omega$; to this end, we map each vertex $p \in \partial S^r$ to the point of intersection between the line $\{\lambda p :\ \lambda > 0\}$ and $\partial\Omega$. Once computed S^r, the remeshed dataset $\overline{\mathcal{M}}$ is achieved as $\varphi^{-1}(S^r)$; each vertex $p \in S^r$ is written using its barycentric-coordinates with respect to the vertices of a triangle $t := (i,j,k) \in T$ which contains it, i.e.

$$p = \alpha v_i + \beta v_j + \gamma v_k, \qquad \alpha \geq 0, \beta \geq 0, \gamma \geq 0,\ \alpha + \beta + \gamma = 1,$$

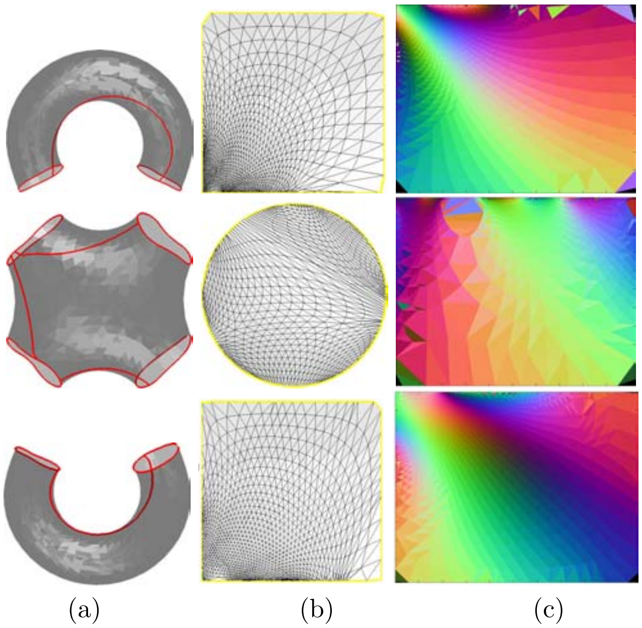


Figure 5.10: (a) Local segmentation of the bitorus into a body and two cylinders induced by *Plumber*, (b) parameterization, and (c) normal-map image [GGH02]. On each patch, the red line represents the cut used for its unfolding.

whose 3D counterpart is $\varphi^{-1}(p) = \alpha\varphi^{-1}(v_i) + \beta\varphi^{-1}(v_j) + \gamma\varphi^{-1}(v_k)$. The remeshing is reduced to a point-location problem in an irregular triangulation whose computational cost is $O(\sqrt{n})$ with n number of vertices (see Figure 5.7(a-f)).

Remesh of cylindrical primitives. For a cylindrical primitive $\mathcal{M}$, we can proceed in two different ways. As first choice, we apply the previous approach with its parameterization $\varphi : \mathcal{M} \rightarrow \Omega$ onto the unit square; the edges l_2, l_4 have to be coherently re-sampled (i.e., $(1, \alpha) \in l_2$ if and only if $(0, \alpha) \in l_4$, $0 \leq \alpha \leq 1$), in order to ensure that $\varphi^{-1}(l_2)$ and $\varphi^{-1}(l_4)$ coherently join when mapped back to $\Gamma \subseteq \mathcal{M}$. In the case of uniform remeshing, the vertices of $\partial\Omega$ have 3-connectivity and each vertex on $\varphi^{-1}(l_2)$ has 6-connectivity. Therefore, the remeshed patch has 6(resp., 3)-connectivity for internal (resp., boundary) vertices and the number of vertices on the two remeshed boundary components is the same (see Figure 5.7(g)). An example of adaptive remeshing is given in Figure 5.7(h).

The second possibility is to use the embedding $\phi : \mathcal{M} \rightarrow \Omega^{'}$ onto the unit circle with the boundary component β_2 in its interior. In this case, we take care of introducing in $\Omega^{'}$ the intersection between β_2 and S^r. This procedure is preferred to the previous one when the parameterization of the cylindrical patch onto the unit square is strongly distorted by an irregular sampling of its geometry. We explicitly note that we cannot know in advance how many vertices of β_2 belong to S^r. The same considerations apply to bodies (see Figure 5.8).

We now consider what happens on the boundary components of adjacent patches in $\mathcal{G}$. If the same boundary component γ is used to parameterize two adjacent regions R_i, R_j (therefore, $\gamma \subseteq R_i \cap R_j$), we consider the same orientation and starting point on it in order to ensure that it has the same parameterization for both regions. This choice guarantees that $\partial\Omega_i \equiv \partial\Omega_j$ and that the corresponding remeshed regions (with respect to the same S^r) coherently join on the common (remeshed) boundary whose vertices have valence six.

Let us now consider the common boundary component β of two adjacent patches R_i and R_j, and suppose that one of them has been parameterized with respect to a loop $\gamma \neq \beta$ (see Figure 5.9(a)). In this case, their remeshed boundaries Θ_i, Θ_j in $\mathbb{R}^3$ corresponding to β do not coherently join even though they are two different descriptions of $\varphi^{-1}(\beta)$. Supposed that Θ_i has a greater number of vertices with respect to Θ_j, we substitute Θ_j with Θ_i (see Figure 5.9(b-c)) and we insert it into the remeshed patch corresponding to R_j (see Figure 5.9(d-e)). This choice is intended to minimize the number of extraordinary vertices, and we use this strategy if we consider a different r in S^r for each patch or the adaptive remeshing (see Figure 5.9). An alternative zippering of adjacent patches is discussed in [SWG$^+$03]. An example of feature-based parameterization with respect to a different decomposition (see Figure 5.9) of the bitorus is given in Figure 5.10.

5.3 Conclusions and future work

Topological and geometric methods for the local and global analysis, approximation, and abstraction of arbitrary triangle meshes have been proposed, discussed, and compared with respect to the current state of the art. Future work is mainly related to specializing the feature-based segmentation detailed in Chapter 4 for applications such as shape recognition, retrival [BMM$^+$03], local parameterization and deformation as discussed in this chapter. The local and global parameterization give the most interesting perspectives for possible specializations from a theoretical and applicative point of view. More precisely, starting from the characterization of convex-combination maps given in Section 4.7 (i.e., Proposition 15 and 16 at page 113 and 116) it would be possible to improve it in such a way of constraining the boundary components $\{\gamma_i\}_{i=1,\dots,k}$ of a 0-genus triangle mesh $\mathcal{M}$ to be mapped onto k convex planar curves with a predefined shape and position on the parameterization domain. The possibility of building such an embedding of $\mathcal{M}$ would give a deeper understanding of the link between the connectivity and geometry of $\mathcal{M}$ with generalizations to surfaces with an arbitrary genus as discussed in Section 4.8.1, and benefits on constrained texture mapping and remeshing. Furthermore, solving the previous problem is of basic support to the search of a homology base of $\mathcal{M}$, the evaluation of reduced polygonal schemes, and the construction of conformal parameterizations.

The global parameterization detailed in Section 4.8 gives also a starting point for visualizing, modeling, and analyzing scalar and vector fields defined on an arbitrary triangle mesh $\mathcal{M}$. More precisely, let us suppose that $\mathcal{M}$ has been parameterized with respect to a cut graph γ by $\varphi : \mathcal{M} \to \Omega \subseteq \mathbb{R}^2$ and that a scalar field $f : \mathcal{M} \to \mathbb{R}$ is assigned. As detailed in Chapter 3 and 4, any

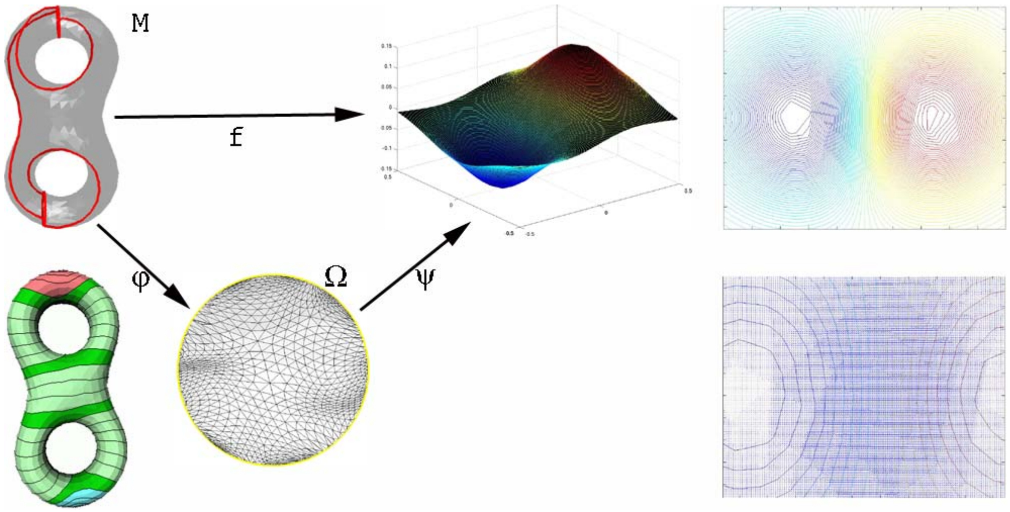


Figure 5.11: Proposed framework for analyzing, modeling, and visualizing a scalar field $f : \mathcal{M} \rightarrow \mathbb{R}$ defined on an arbitrary triangle mesh $\mathcal{M}$ and based on the global parameterization. The two figures on the right show the variation of the gradient field associated to f.

scalar field (e.g., harmonic scalar field, distance functions) $(\mathcal{M}, h)$ can be used for defining γ; in some cases, we have $h = f$ but we do not make this assumption in the following of the discussion. Instead of characterizing f by studying the evolution of its iso-contours $C_\alpha(\mathcal{M}) := \{x \in \mathcal{M} : f(x) = \alpha\}$, $\alpha \in \mathbb{R}$, we reduce the "dimension" of the problem by exploiting the parameterization φ of $\mathcal{M}$ (see Figure 5.11). Since φ is a simplicial isomorphism, we consider the composite mapping $\psi := f \circ \varphi^{-1} : \Omega \rightarrow \mathbb{R}$ whose graph $\Sigma := \{(u, \psi(u)) \in \mathbb{R}^3 : \; u \in \Omega\}$ is a 2.5D surface; therefore, the analysis of $(\mathcal{M}, f)$ is equivalent to that of Σ and the values $\{f(x) : x \in \mathcal{M}\}$ correspond to the z-values of Σ. The choice of moving from $(\mathcal{M}, f)$ to Σ is motivated by a simpler approach to the following tasks: scalar field analysis, approximation, and modeling.

Scalar field analysis: the critical points of f have a straightforward interpretation as maxima, minima, and saddles of Σ. Furthermore, the analysis of f and of the associated iso-contours is simplified by the visualization of Σ as an elevation model (i.e, a digital terrain model); for any $\alpha \in \mathbb{R}$, the relation among the iso-contours on $\mathcal{M}$ and Σ is given by $\psi^{-1}(\alpha) = \varphi(C_\alpha(\mathcal{M}))$. On the contrary, on $\mathcal{M}$ the analysis of f is indirect because related to the visualization of the contour levels $C_\alpha(\mathcal{M})$, $\alpha \in \mathbb{R}$.

Scalar field interpolation and approximation: if $x \in \mathcal{M}$, $f(x)$ is usually evaluated with linear precision by interpolating the values of f on the vertices of the triangle $\Delta := [x_{i_1}, x_{i_2}, x_{i_3}]$ which contains the point x. If we consider Σ, we can interpolate or approximate the values of f (or equivalently, the z-values of Σ) by using an arbitrary functional space $\mathcal{F} \subseteq C^k(\Omega, \mathbb{R})$ of real functions defined on Ω; common choices are the space of polynomials or generated by an appropriate set of radial basis functions [GL96]. Finding a function $\psi^\star \in \mathcal{F}$ such that $\psi^\star(u_i) \approx f(x_i)$, $i = 1, \dots, n_V$, is a standard approximation problem on the parameterization domain Ω of 0-genus (and not of arbitrary genus as it happens for $\mathcal{M}$) and it enables to approximate the value $f(x)$ with a greater precision with respect to a linear interpolation, and to include in the choice of $\mathcal{F}$ a-priori information on the scalar field f such as its degree of smoothness $\overline{k} \leq k$.

Scalar field modeling: strictly related to the above-mentioned problem is the modeling of scalar fields on arbitrary triangle meshes. Defining a scalar field $f : \mathcal{M} \to \mathbb{R}$ which interpolates a given set of samples $\{(x_i, f(x_i)) : i \in C\}$, $C \subseteq \{i = 1, \ldots, n_V\}$, is of basic importance when f has to be recovered from a partial knowledge of its values on $\mathcal{M}$. In this case, two options are available depending on the meaning of the input data; if we want to characterize f focusing its modeling on the set of critical points, the solution is to define a harmonic scalar field f thus constraining the location on $\mathcal{M}$ and the values of maxima and minima of f (see Section 3.2.1). If the emphasis of the modeling phase of f is on its values, the solution is to find an approximation of f in $\mathcal{F}$ as previously discussed thus guaranteeing a greater accuracy of the sampling of f on the whole surface $\mathcal{M}$; in this case, the critical points of f cannot be included during the construction of the scalar field.

The previous discussion has a simple extension to the case of vector and time-depending scalar fields. The analysis of a vector field $F := (f_1, \ldots, f_d) : \mathcal{M} \to \mathbb{R}^d$, $d \geq 2$, is reduced to the analysis of its d scalar components $f_i : \mathcal{M} \to \mathbb{R}$, $i = 1, \ldots, d$. A time-depending scalar field $f_t : \mathcal{M} \to \mathbb{R}$, $t \in I := [0, 1]$, can be analyzed by taking into account its 2.5D counterpart $\psi_t : \mathcal{M} \to \mathbb{R}$, $\psi_t := f_t \circ \varphi^{-1}$, whose graph $\Sigma_t := \{(u, \psi_t(u)) : u \in \Omega\}$ gives an interpretation of the variation of its critical points with respect to t that is simpler than the analysis of the iso-contours $C_\alpha^t := \{x \in \mathcal{M} : f_t(x) = \alpha\}$, $\alpha \in \mathbb{R}$, $t \in I$. Finally, we underline that the global parameterization φ of $\mathcal{M}$ is calculated and then used to evaluate the whole set of 2.5D surfaces Σ_t, $t \in I$, through the function ψ_t.

Bibliography

[ABS03] M. Attene, S. Biasotti, and M. Spagnuolo. Shape understanding by contour-driven retiling. *The Visual Computer*, 19(2-3):127–138, 2003.

[AIM] AIM@SHAPE. Network of Excellence (Advanced and Innovative Models and Tools for the development of Semantic-based systems for Handling, Acquiring, and Processing knowledge Embedded in multidimensional digital objects) - FP6 IST NoE 506766 URL: http://www.aimatshape.net.

[AMD02] P. Alliez, M. Meyer, and M. Desbrun. Interactive geometry remeshing. *Acm Transactions on Graphics*, 21(3):347–354, 2002.

[Aur91] F. Aurenhammer. Voronoi diagrams: a survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23(3):345–405, 1991.

[Ban67] T. Banchoff. Critical points and curvature for embedded polyhedra. *Journal of Differential Geometry*, 1:245–256, 1967.

[BE92] M. Bern and D. Eppstein. Mesh generation and optimal triangulation. In *Computing in Euclidean Geometry, Edited by Ding-Zhu Du and Frank Hwang, Lecture Notes Series on Computing – Vol. 1.* World Scientific, 1992.

[Bia04] S. Biasotti. *Computational Topology Methods for Shape Modelling Applications.* PhD thesis, Universitá degli Studi di Genova, May 2004.

[Blo02] J. Bloomenthal. Medial-based vertex deformation. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer animation*, pages 147–151. ACM Press, 2002.

[Blu67] H. Blum. A transformation for extracting new descriptors of shape. In *Models for the Perception of Speech and Visual Forms*, pages 362–380. MIT Press, 1967.

[BMM$^+$03] S. Biasotti, S. Marini, M. Mortara, G. Patanè, M. Spagnuolo, and B. Falcidieno. 3D shape matching through topological structures. *Lecture Notes in Computer Science*, 2886:194–203, 2003.

[BMMP03] S. Biasotti, S. Marini, M. Mortara, and G. Patanè. An overview on properties and efficacy of topological skeletons in shape modelling. In M.S. Kim, editor, *SMI '03: Proceedings of Shape Modeling International 2003*, pages 245–254, Los Alamitos, May 2003. IEEE Computer Society.

[BSW83] R.E. Bank, A.H. Sherman, and A. Weiser. Re-finement algorithms and data structures for regular local mesh refinement. *Scientific Computing*, pages 3–17, 1983.

[BW97] J. Bloomenthal and B. Wyvill. *Introduction to Implicit Surfaces.* Morgan Kaufmann Publishers Inc., 1997.

[CGC$^+$02] S. Capell, S. Green, B. Curless, T. Duchamp, and Z. Popovich. Interactive skeleton-driven dynamic deformations. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 586–593. ACM Press, 2002.

[CMEH$^+$03] K. Cole-McLaughlin, H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Loops in reeb graphs of 2-manifolds. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pages 344–350. ACM Press, 2003.

[CRS98] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: Measuring error on simplified surfaces. *Computer Graphics Forum*, 17(2):167–174, June 1998.

[CSM03] D. Cohen-Steiner and J.-M. Morvan. Restricted delaunay triangulations and normal cycle. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pages 312–321. ACM Press, 2003.

[dBKOS97] M. de Berg, M. Van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry Algorithms and Applications.* Springer-Verlag, Berlin Heidelberg, 1997.

[DGG03] Tamal K. Dey, J. Giesen, and S. Goswami. Shape segmentation and matching with flow discretization. *Proceedings of the 8th International Workshop onAlgorithms and Data Structures*, LNCS 2748:25–36, 2003.

[Die00] R. Diestel. *Graph Theory.* Springer-Verlag, 2000.

[Dij59] E.W. Dijkstra. A note on two problems in connection with graphs. *Numerical Mathematics*, 24(1):269–271, 1959.

[DLN87] A. R. Dill, M. D. Levine, and P. B. Noble. Multiple resolution skeletons. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(4):495–504, 1987.

[DM02] L. DeFloriani and P. Magillo. Multiresolution mesh representation: Models and data structures. In *Tutorials on Multiresolution in Geometric Modeling*, pages 363–418. Springer-Verlag, 2002.

[DMA01] M. Desbrun, M. Meyer, and P. Alliez. Intrinsic parameterizations of surface meshes. In *Computer Graphics Forum*, volume 21, pages 209–218, 2001.

[DMSB99] M. Desbrun, M. Meyer, P. Schroeder, and A. H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 317–324, 1999.

[DMSB02] M. Desbrun, M. Meyer, P. Schroeder, and A.H. Barr. Discrete differential geometry operators for triangulated 2- manifolds. In *Visualization and Mathematics III*, 2002.

[DoC76] M. P. DoCarmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1976.

[dVL02] E. Colin de Verdiere and F. Lazarus. Optimal system of loops on an orientable surface. In *Proceedings of the 43rd Symposium on Foundations of Computer Science*, pages 627–636. IEEE Computer Society, 2002.

[EDD$^+$95] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 173–182. ACM Press, 1995.

[Ede87] H. Edelsbrunner. *Algorithms in combinatorial geometry.* Springer-Verlag New York, Inc., 1987.

[EHNP03] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Morse-smale complexes for piecewise linear 3-manifolds. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pages 361–370. ACM Press, 2003.

[EHP02] J. Erickson and S. Har-Peled. Optimally cutting a surface into a disk. In *Proceedings of the eighteenth annual symposium on Computational geometry*, pages 244–253. ACM Press, 2002.

[ES92] R. Engelking and K. Svekhicki. *Topology: A Geometric Approach.* Oxford University Press, 1992.

[FG99] M. S. Floater and C. Gotsman. How to morph tilings injectively. *J. Comput. Appl. Math.*, 101(1-2):117–129, 1999.

[FH02] M. S. Floater and K. Hormann. Parameterization of triangulations and unorganized points. In A. Iske, E. Quak, and M. S. Floater, editors, *Tutorials on Multiresolution in Geometric Modelling*, Mathematics and Visualization, pages 287–316. Springer, 2002.

[FH04] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In N. A. Dodgson, M. S. Floater, and M. A. Sabin, editors, *Multiresolution in Geometric Modelling.* Springer, 2004.

[FHR02] M. S. Floater, K. Hormann, and M. Reimers. Parameterization of manifold triangulations. In *Approximation Theory X: Abstract and Classical Analysis*, pages 197–209. Vanderbilt University Press, 2002.

[FK97] A.T. Fomenko and T.L. Kunii. *Topological Modeling for Visualization.* Springer, 1997.

[Flo97] M. S. Floater. Parametrization and smooth approximation of surface triangulations. *Comput. Aided Geom. Des.*, 14(3):231–250, 1997.

[Flo03] M. S. Floater. One-to-one piecewise linear mappings over triangulations. *Mathematics of Computation*, 72(242):685–696, 2003.

[FS93] B. Falcidieno and M. Spagnuolo. Geometric reasoning for the extraction of surface shape properties. In *Communicating with the Virtual World*, pages 10–15. IEEE Computer Society, 1993.

[GGH02] X. Gu, S. J. Gortler, and H. Hoppe. Geometry images. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 355–361. ACM Press, 2002.

[GGS03] C. Gotsman, X. Gu, and A. Sheffer. Fundamentals of spherical parameterization for 3d meshes. *ACM Trans. Graph.*, 22(3):358–363, 2003.

[GL96] G. H. Golub and C. F. Van Loan. *Matrix computations (3rd ed.).* Johns Hopkins University Press, 1996.

[GP74] V. Guillemin and A. Pollack. *Differential Topology.* Englewood Cliffs, NJ: Prentice-Hall, 1974.

[GS85] L. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of voronoi. *ACM Trans. Graph.*, 4(2):74–123, 1985.

[GVSS00] I. Guskov, K. Vidime, W. Sweldens, and P. Schroeder. Normal meshes. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 95–102, 2000.

[GW92] R.C. Gonzales and R.E. Woods. *Digital Image Processing.* Reading, MASS.: Addison-Weshley, 1992.

[GW01] I. Guskov and Z. J. Wood. Topological noise removal. In *Graphics Interface 2001*, pages 19–26. Canadian Information Processing Society, 2001.

[GY03] X. Gu and S.-T. Yau. Global conformal surface parameterization. In *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 127–137. Eurographics Association, 2003.

[Ham93] B. Hamann. Curvature approximation for triangulated surfaces. *Geometric modelling*, pages 139–153, 1993.

[HDD$^+$92] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 71–78. ACM Press, 1992.

[HG99] K. Hormann and G. Greiner. MIPS: An efficient global parametrization method. In *Curve and Surface Design: Saint-Malo 1999*, pages 153–162. Vanderbilt University Press, 1999.

[HLG01] K. Hormann, U. Labsik, and G. Greiner. Remeshing triangulated surfaces with optimal parameterizations. *Computer-Aided Design*, 33(11):779–788, 2001.

[Hop96] H. Hoppe. Progressive meshes. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 99–108. ACM Press, 1996.

[HSKK01] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 203–212. ACM Press, 2001.

[IBMJ04] H. Rushmeier I. Boier-Martin and J. Jin. Parametrization of triangle meshes over quadriangular domains. In *Eurographics Symposium on Geometry Processing (SGP2004)*, pages 197–207. Eurographics Association, 2004.

[IGG01] M. Isenburg, S. Gumhold, and C. Gotsman. Connectivity shapes. In *Visualization'01 Conference Proceedings*, pages 135–142, 2001.

[Jol86] I. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.

[Kar99] E. Kartasheva. The algorithm for automatic cutting of three-dimensional polyhedrons of h-genus. In *Proceedings of the International Conference on Shape Modeling and Applications*, pages 26–35. IEEE Computer Society, 1999.

[KCS98] L. Kobbelt, S. Campagna, and H.-P. Seidel. A general framework for mesh decimation. In *Graphics Interface*, pages 43–50, 1998.

[KCVS98] L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel. Interactive multi-resolution modeling on arbitrary meshes. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 105–114. ACM Press, 1998.

[Kru56] J. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. American Mathematical Society*, 7:48–50, 1956.

[KT03] S. Katz and A. Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Transactions on Graphics (TOG)*, 22(3):954–961, 2003.

[LCF00] J. P. Lewis, M. Cordner, and N. Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 165–172. ACM Press, 2000.

[Lip69] M. M. Lipschutz. *Theory and Problems of Differential Geometry*. Schaum's Outline Ser, 1969.

[LK98] J. Li and C. Kuo. A dual graph approach to 3D triangular mesh compression. In *IEEE International Conference on Image Processing, Chicago, Oct. 4-7*, pages 891–894, 1998.

[LKL02] Y. Lee, H. S. Kim, and S. Lee. Mesh parameterization with a virtual boundary. *Computers and Graphics*, 26(5):677–686, 2002.

[Lon98] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.

[LPVV01] F. Lazarus, M. Pocchiola, G. Vegter, and A. Verroust. Computing a canonical polygonal schema of an orientable triangulated surface. In *Proceedings of the seventeenth annual symposium on Computational geometry*, pages 80–89. ACM Press, 2001.

[LSS$^+$98] A. W. F. Lee, W. Sweldens, P. Schroeder, L. Cowsar, and D. Dobkin. MAPS: Multiresolution adaptive parameterization of surfaces. *Computer Graphics*, 32(Annual Conference Series):95–104, 1998.

[LTH01] X. Li, T. W. Toon, and Z. Huang. Decomposing polygon meshes for interactive applications. In *Symposium on Interactive 3D graphics*, pages 35–42, 2001.

[LTR$^+$02] H. Lopes, G. Tavares, J. Rossignac, A. Szymczak, and A. Safanova. Edgebreaker: a simple compression for surfaces with handles. In *Proceedings of the seventh ACM symposium on Solid modeling and applications*, pages 289–296. ACM Press, 2002.

[LV99] F. Lazarus and A. Verroust. Level set diagrams of polyhedral objects. In *Proceedings of the fifth ACM symposium on Solid modeling and applications*, pages 130–140. ACM Press, 1999.

[Man88] M. Mantyla. *Introduction to Solid Modeling.* W. H. Freeman & Co., 1988.

[Mas67] W. S. Massey. *Algebraic Topology: An Introduction.* Harbrace College Mathematics Series, 1967.

[Mil63] J. Milnor. *Morse Theory.* Princeton University Press, 1963.

[Moh91] B. Mohar. The laplacian spectrum of graphs. *Graph Theory, Combinatorics and Applications.*, pages 871–898, 1991.

[Mor85] M. E. Morten. *Geometric Modeling.* John Wiley, 1985.

[MP02a] M. Mortara and G. Patanè. Affine-invariant skeleton of 3d shapes. In *Proceedings of the Shape Modeling International 2002 (SMI'02)*, pages 245–252. IEEE Computer Society, 2002.

[MP02b] M. Mortara and G. Patanè. Shape covering for skeleton extraction. *International Journal of Shape Modelling*, 8(2):139–158, 2002.

[MPS$^+$04a] M. Mortara, G. Patanè, M. Spagnuolo, B. Falcidieno, and J. Rossignac. Blowing bubbles for the multi-scale analysis and decomposition of triangle meshes. *Algorithmica (Special Issue on Shape Algorithms)*, 38(1):227–248, 2004.

[MPS$^+$04b] M. Mortara, G. Patanè , M. Spagnuolo, B. Falcidieno, and J. Rossignac. Plumber: a multiscale decompostion of 3D shapes into tubular primitives and bodies. In *ACM Symposium on Solid Modelling*, pages 339–344, 2004.

[NGH04] X. Ni, M. Garland, and J. C. Hart. Fair morse functions for extracting the topological structure of a surface mesh. *ACM Trans. Graph.*, 23(3):613–622, 2004.

[Ogn94] R. L. Ogniewicz. Skeleton-space: A multiscale shape description combining region and boundary information. In *Proc. Computer Vision and Pattern Recognition*, pages 746–751, 1994.

[Pat04] G. Patanè. *Parameterization of 3D triangle meshes with arbitrary genus.* Istituto di Matematica Applicata e Tecnologie Informatiche,Consiglio Nazionale delle Ricerche, Technical Report N. 7/2004, *Submitted for pubblication*, 2004.

[PCM02] V. Pascucci and K. Cole-McLaughlin. Efficient computation of the topology of level sets. In *VIS '02: Proceedings of the conference on Visualization '02*, pages 187–194. IEEE Computer Society, 2002.

[Pet04] M. Peternell. Developable surface fitting to point clouds. *Comput. Aided Geom. Des.*, 21(8):785–803, 2004.

[POB87] S. M. Pizer, W. R. Oliver, and S. H. Bloomberg. Hierarchical shape description via the multiresolution symmetric axis transforms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(4):505–511, 1987.

[PP93] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, pages 15–36, 1993.

[PS85] F. P. Preparata and M. I. Shamos. *Computational geometry: an introduction.* Springer-Verlag New York, Inc., 1985.

[PS99] K. Polthier and M. Schmies. Straightest geodesics on polyhedral surfaces. In *Mathematical Visualization*, pages 135–150, 1999.

[PSF04] G. Patanè, M. Spagnuolo, and B. Falcidieno. Para-graph: Graph-based parameterization of triangle meshes with arbitrary genus. *Computer Graphics Forum*, 23(4):783–797, 2004.

[QA94] A. Quarteroni and A.Valli. *Numerical Approximation of Partial Differential Equations.* Springer-Verlag, Heidelberg, 1994.

[Ree46] G. Reeb. Sur les points singuliers d'une forme de pfaff completement integrable ou d'une fonction numerique. In *Comptes Rendu Acad. Sciences*, pages 847–849. Sciences Park, 1946.

[Rud70] W. Rudin. *Real and complex analysis.* Mac Graw-Hill, 1970.

[SCO04] O. Sorkine and D. Cohen-Or. Least-squares meshes. In *Proceedings of Shape Modeling International*, pages 191–199. IEEE Computer Society Press, 2004.

[SdS00] A. Sheffer and E. de Sturler. Parameterization of faceted surfaces for meshing using angle based flattening. *Engineering with Computers*, 17(3):326–337, 2000.

[SF02] D. Steiner and A. Fischer. Growing surface methods for cutting 3d freeform objects with genus-n. In *Bi-National UK-Israel Conference on Geometric Modeling Methods*, 2002.

[SF04] D. Steiner and A. Fischer. Planar parameterization for closed manifolds genus-1 meshes. In *Proceedings of the ninth ACM symposium on Solid modeling and applications*, pages 83–92. Blackwell Plubisher, 2004.

[SG03] V. Surazhsky and C. Gotsman. Explicit surface remeshing. In *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 20–30. Eurographics Association, 2003.

[SKK91] Y. Shinagawa, T. L. Kunii, and Y. L. Kergosien. Surface coding based on morse theory. *IEEE Comput. Graph. Appl.*, 11(5):66–78, 1991.

[SLCO$^+$04] O. Sorkine, Y. Lipman, D. Cohen-Or, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian surface editing. In *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 179–188. Eurographics Association, 2004.

[SPB96] E. C. Sherbrooke, N. M. Patrikalakis, and E. Brisson. An algorithm for the medial axis transform of 3d polyhedral solids. *IEEE Transactions on Visualization and Computer Graphics*, 2(1):44–61, 1996.

[SSGH01] P. V. Sander, J. Snyder, S. J. Gortler, and H. Hoppe. Texture mapping progressive meshes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 409–416. ACM Press, 2001.

[SWG$^+$03] P. V. Sander, Z. J. Wood, S. J. Gortler, J. Snyder, and H. Hoppe. Multi-chart geometry images. In *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 146–155. Eurographics Association, 2003.

[Tau95a] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *Proceedings of the Fifth International Conference on Computer Vision*, pages 902–907. IEEE Computer Society, 1995.

[Tau95b] G. Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 351–358. ACM Press, 1995.

[Tau00] G. Taubin. Geometric signal processing on polygonal meshes. In *Eurographics'2000, State of the Art Report*, August 2000.

[Tau02] G. Taubin. Dual mesh resampling. *Graph. Models*, 64(2):94–113, 2002.

[Tut63] W. T. Tutte. How to draw a graph. *Proc. London Mathematical Society*, 13:743–768, 1963.

[TZG96] G. Taubin, T. Zhang, and G. H. Golub. Optimal surface smoothing as filter design. In *Proceedings of the 4th European Conference on Computer Vision-Volume I*, pages 283–292. Springer-Verlag, 1996.

[VF02] A. Verroust and M. Finiasz. A control of smooth deformations with topological change on polyhedral mesh based on curves and loops. In *Proceedings of the Shape Modeling International 2002 (SMI'02)*, pages 10–15. IEEE Computer Society, 2002.

[VL99] A. Verroust and F. Lazarus. Extracting skeletal curves from 3d scattered data. In *Proceedings of the International Conference on Shape Modeling and Applications*, pages 194–201. IEEE Computer Society, 1999.

[VY90] G. Vegter and C. K. Yap. Computational complexity of combinatorial surfaces. In *Proceedings of the sixth annual symposium on Computational geometry*, pages 102–111. ACM Press, 1990.

[WDSB00] Z. Wood, M. Desbrun, P. Schroeder, and D. Breen. Semi-regular mesh extraction from volumes. In *Proceedings of the 11th IEEE Visualization 2000 Conference (VIS 2000)*, pages 275–282. IEEE Computer Society, 2000.

[WW92] W. Welch and A. Witkin. Variational surface modeling. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 157–166. ACM Press, 1992.

[YBS03] S. Yoshizawa, A. G. Belyaev, and H.-P. Seidel. Free-form skeleton-driven mesh deformations. In *Proceedings of the eighth ACM symposium on Solid modeling and applications*, pages 247–253. ACM Press, 2003.

[ZS01] Denis Zorin and Peter Schröder. A unified framework for primal/dual quadrilateral subdivision schemes. *Computer Aided Geometric Design*, 18(5):429–454, 2001.

[ZSS97] D. Zorin, P. Schroeder, and W. Sweldens. Interactive multiresolution mesh editing. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 259–268. ACM Press, 1997.