Medial Descriptors for 3D Shape Segmentation, Reconstruction, and Analysis

Kustra, Jacek

1 INTRODUCTION

> I don't pretend we have all the answers. But questions are certainly worth thinking about.
>
> Arthur C. Clarke

1.1 SHAPES

We live in a world which, through our senses, we perceive as three dimensional. At this level of perception, a basic property of everything that surrounds us is its *shape*. Shape determines the way we identify objects and interact with our surroundings. It is an intrinsic dynamic property of virtually everything that we perceive, therefore its understanding is of extreme importance. Furthermore, as our visual and tactile perception of an object is often limited to its boundary, *i.e.* the outer interface between the object and the space surrounding it, our visual interpretation of an object is driven by the way we perceive its boundary.

Given the above, it is natural that many computer applications have been developed in the last decade to represent, manipulate, and analyze 3D shapes. Such applications range from classical computer graphics and virtual reality (which use shapes to depict realistically-looking 3D environments), augmented reality (which add computer-generated shapes to images and video capturing a real-world environment), computer-aided medicine (which use shapes extracted from 3D scans to depict and analyze various parts of the human body), to data visualization (which use 2D and 3D synthetic shapes to depict phenomena captured by large amounts of spatial or non-spatial data).

To be able to deal with 3D shapes, a computer needs ways to represent these shapes and to measure and determine its properties. Briefly put, any software application that is concerned with such shapes needs two main components: a shape *representation* part, which deals with the actual storage of the shape and its properties, and a shape *processing* part, which deals with the operations that the application wishes to execute on the shape representation. These two components, which parallel the fundamental data structures and algorithms in any computer program, are outlined next.

1.2 SHAPE REPRESENTATION

The first step towards reasoning about a shape is having a convenient way of representing it. The representation choice is dependent on several factors such as the type of sensor used to acquire the shape and the type of analysis one wants to perform on the representation. In practice, such representations can be classified as *continuous* or *discrete*. Continuous representations are able to faithfully capture all details of a shape, and allow in principle any type of analysis to be performed thereon without loss of precision. However, they are not practical or often not even feasible, as we cannot (easily) construct them for all but simple shapes. Discrete representations have the key advantage of allowing one to capture practically any type of shape in existence by a number of relatively simple data structures. While they typically cannot represent all details of any shape, they allow a controlled trade-off between representation size and complexity (on one hand) and representation costs and accuracy (on the other hand). As such, discrete shape representations are predominant in most computer-based shape processing applications.

Besides the above, shape representations can be classified into *surface* or *volumetric* ones. Surface, or boundary, representations capture only the form of the shape's surface or interface that separates the inner part of the shape from its surroundings. They are useful in cases when we are interested to reason only about the shape boundary or in cases when we only have information on this boundary, and not on the shape's interior. Discrete boundary representations can be readily acquired by *e.g.* laser scanners or constructed by modeling tools, and consist typically in a point-sampling of the object surface, with optional additional connectivity. These two representations are also known under the names of unstructured point clouds and polygonal meshes, respectively. Volumetric representations capture both the shape's surface and its interior. They can be acquired from 3D volumetric scanners, such as CT and MRI scanners, or constructed synthetically, *e.g.* by using field-based methods. Discrete volumetric representations usually consist of an uniform sampling of the shape's interior on a regular grid, *i.e.*, of a collection of voxels whose values indicate the distinction between shape interior and exterior and, optionally, the value(s) of one or more measured properties inside the shape. In contrast to boundary representations, volumetric representations require more memory and computing power to handle.

1.3 SHAPE PROCESSING

Both boundary and volumetric representations outlined above are essentially anchored in the $\mathbb{R}^3$ (three-dimensional) space in which the shape is embedded. However, this is not the only space in which shapes can be represented. For instance, shapes can be represented in the Fourier domain. This allows certain shape processing operations, such as *e.g.*

smoothing, matching, or filtering, to be implemented in simpler and/or more efficient ways than in the spatial domain. Conversely, the spatial domain supports better other types of shape processing, such as analyses pertaining to metrology or topology.

Shape processing operations encompass a huge spectrum of manipulations, such as matching, compression, simplification, denoising and smoothing, segmentation, and rendering. As outlined above, different shape *representations* support different types of *processing* operations better (or worse) in terms of simplicity, accuracy, robustness, and computational scalability. As such, the quest of the best match between specific processing operations and existing (or new) shape representations is an important evolving field of research.

1.4 MEDIAL REPRESENTATIONS

As outlined in Section 1.2, the boundary and volumetric representations are not the only spatial representations of 3D shape known. Although these are arguably the most convenient representations for shape acquisition and display, other representations can offer important advantages for specific shape processing operations.

This thesis focuses on one such alternative spatial representation for 3D shapes: the *medial* representation. This representation, first proposed by Blum in 1967 [15], encodes a (2D or 3D) shape into the collection of the medial loci of the maximally inscribed (2D or 3D) balls inside the shape's boundary. For 2D shapes, medial axes (also called 2D skeletons), consist of a collection of 1D curves. For 3D shapes, several medial representations are known, as follows: Medial surfaces, also called surface skeletons, consist of a (complex) collection of 3D manifolds with boundaries (Figure 1.1). They provide a full representation of the entire shape boundary, but are relatively complex to compute. Curve skeletons, in contrast, consist of a (relatively simple) collection of 3D curves. They can be computed relatively easily, but only capture the topological characteristics and main geometry characteristics of the shape.

Medial representations, also called medial descriptors, are very effective in supporting shape processing applications which focus on analyzing, or manipulating, the topological and symmetry-related properties of a shape. Intuitively put, medial descriptors capture the essential 'branching structure' or part-in-whole structure of a shape, as well as its symmetry characteristics. As such, they are often used in applications where such properties are important, such as shape matching, simplification, part-in-whole segmentation, and recognition.

In this thesis, we explore the usage of 3D medial descriptors for supporting a range of shape analysis and processing operations. Hence, we next overview the different types of medial representations (Section 1.4.1), applications that such representations can support (Section 1.4.2), and outline the key challenges that medial representations face when supporting such application (Section 1.4.3).

1.4.1 *Types of medial representations*

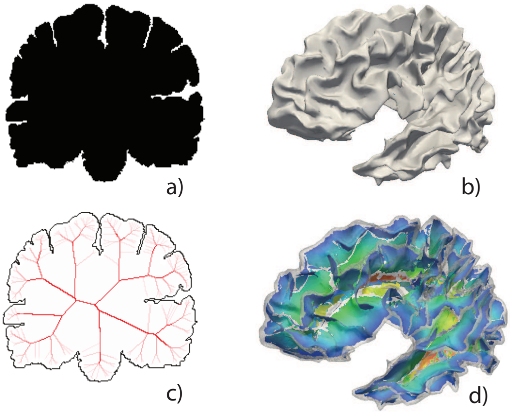

Figure 1.1: 2D and 3D shape (brain) boundary representation (a, b) and its corresponding medial representation (c, d)

As briefly outlined above, several types of medial representations have been proposed for several types of shapes and shape-processing applications. For 2D shapes, the widest-used medial representation directly follows the maximally inscribed ball definition outlined earlier. This representation has also been the most explored (and used) in practice, due to its simple and efficient computation. Currently, many algorithms exist for computing 2D skeletons, *e.g.* [45, 118, 122, 155, 183]. Such algorithms are able to robustly deliver accurate 2D skeletons of large and complex shapes, represented either as pixel images or polygonal contours, at interactive-frame rates. As such, 2D skeletons are frequently used in many applications on 2D shape matching, simplification, retrieval, recognition, and analysis (Figure 1.1).

For 3D shapes, curve skeletons are the widest-used medial representation, as these are relatively simple and efficient to compute and analyze. Similar to the 2D skeleton case, many curve-skeleton extraction algorithms have been proposed for both polygonal and volumetric shape representations, *e.g.* [7, 16, 64, 172, 174, 181, 192]. Such algorithms can efficiently extract accurate curve skeletons of large and complex 3D shapes, and have been used in applications such as 3D shape recognition and matching, path planning, and virtual navigation.

However, as already outlined, curve skeletons cannot fully capture the entire geometry and topology of a 3D shape. As such, they are most effective in describing shapes which have a locally tubular structure, such as blood vessel networks or plants. In contrast, medial surfaces do capture all topological and geometric information present on the boundary of a 3D shape. As such, they enable more complex shape processing applications, such as shape reconstruction and simplification, denoising, and smoothing. Several methods exist for extracting 3D medial surfaces from both volumetric and polygonal shape representations, *e.g.* [43, 50, 59, 101, 119, 126, 135]. However, most if not all such methods are considerably more complex, slower, and less scalable than 2D skeleton or 3D curve-skeleton extraction methods. As such, 3D surface skeletons have been used considerably less often in shape processing applications than their other two counterparts.

1.4.2 *Applications of medial representations*

Medial representations have been used for a wide range of analysis and processing operations involving both 2D and 3D shapes. Below we overview a number of important applications where such descriptors have been used, without the claim of being exhaustive.

Path planning and navigation: Given a (2D or 3D) shape, the skeleton thereof represents the locus of points which are locally centered with respect to the shape boundary. This implies that such points are, also, locally at maximal distance from the shape's boundary. As such, the skeleton can be used to construct a navigation path for a vehicle that moves inside the respective shape and avoid colliding with the shape's boundary. Following this idea, skeletons have been used for path planning applications in both 2D map-like contexts [24, 54, 55, 109, 166] and 3D contexts, such as navigating a virtual camera inside tubular organs such as arteries, bronchies, or the colon [79, 97, 192]. For 3D applications, curve skeletons have been mainly used, given that the desired path planning involved tube-like structures.

Segmentation: Given a (2D or 3D) shape, the branches of its skeleton correspond to protrusions (convex bumps) of the shape's boundary. As such, the natural idea arose to segment these protrusions from the main shape body or rump, by cutting the shape around points corresponding to the branch junction points or around points located on the skeletal ligature branches. This leads to so-called *part-based* segmentations, which work well for natural objects. Examples of part-based segmentation using skeletons include [147] for 2D shapes and [129] for 3D. In both examples, skeletons consisting of 1D structures (*i.e.*, 2D skeletons and 3D curve skeletons) are used.

Surface skeletons can be also used to segment shapes, leading to a different class of so-called *patch-based* segmentations, where segments represent quasi-flat shape surface areas separated by sharp creases. Examples of patch-based segmentations using surface skeletons are shown in [130].

Both part-based and patch-based segmentation approaches are further explored in Chapters 4 - 6.

Shape matching and retrieval: Skeletons represent the geometry and topology of a shape in a compact manner, *i.e.*, in a space which is typically of lower dimension than the given shape. As such, they can be used in shape retrieval and matching contexts. The overall idea is simple: The skeleton is reduced to an attributed graph, where nodes represent skeleton branches and edges represent skeleton junctions where several branches meet. Using this compact descriptor, shapes can be searched or matched following a top-down comparison process based on graph distances. Examples of such applications include [9, 60, 169, 196]. Similar to the segmentation case, in most current applications 2D skeletons and 3D curve skeletons are used, as surface skeletons are deemed too complex and/or computationally expensive for this context.

Shape analysis: Shape analysis concerns itself with finding specific features on the surface of 3D shapes, such as edges, corners, or concave or convex regions. Such regions can be often found easier by analyzing the shape skeleton, and using known connections between the skeleton and shape to locate them on the shape surface. For example, convex curvature maxima (ridges and corners) of 3D shapes can be found by locating the surface skeleton, respectively curve skeleton, boundaries [134]. Similar approaches can be used for 2D shapes, leading to enhanced descriptors for shape classification [8]. Compared to classical curvature-based shape analysis methods, skeletal methods have the advantage of enhanced robustness in the presence of small-scale surface noise.

Shape simplification: Shapes can be simplified by using their skeletons. Applications include removing small-scale surface noise and/or accentuating the important sharp edges thereof. The main idea here follows the fact that the boundary of a shape can be reconstructed from its skeleton. By applying suitable skeleton simplification techniques, the shape reconstructed from the simplified skeleton can thus emphasize certain important details and remove other irrelevant ones. Examples of such techniques include [41, 62, 175, 177]. Among other applications, this process leads to the production of a multiscale representation of shapes, based on progressively simplified skeletons thereof. However, as for the applications mentioned above, 3D curve skeletons and 2D skeletons are the main tools used here, the 3D surface skeletons being rarely used. Besides simplification of shapes represented by their boundaries, skeletons also have been used to simplify other types of spatial datasets, such as 2D grayscale and color images [199].

Mesh construction: Since the medial axis can be seen as the 'symmetry locus' of a boundary, it offers a useful instrument to construct a partitioning of the shape that links 'opposite' boundary parts. This property is instrumental in producing quad-dominant meshes from both 2D and 3D domains [5, 48, 137, 143]. Creating such quad meshes is of great importance for highly-accurate and efficient numerical simulations of complex phenomena, such as air flows around road vehicles and planes. Such meshing methods based on medial descriptors have also found their way into state-of-the-art commercial products [187].

Information visualization: As mentioned above, medial descriptors compactly capture the symmetry locus, or symmetry set, of a complex boundary. As such, they can encode the 'essence' of complex boundaries using more spatially compact shapes than the boundaries themselves. This property is useful, among others, in information visualization applications, where one design goal is to squeeze as much information as possible in a limited screen space – the so-called 'space filling' design approach [163]. In this context, medial axes have been used for the visualization of large graphs, by simplifying the drawing of such graphs using an edge-bundling approach [44, 179].

1.4.3 *Medial representation challenges*

For a medial descriptor to be useful and usable in practice, it should meet several conditions. Globally put, these conditions regard two activities: the *computation* of the medial descriptor from an input shape; the *interpretation* of the medial descriptor in terms of supporting relevant shape processing operations. We discuss these two aspects below.

Medial computation: The extraction of medial representations, specially for 3D surface skeletons, is a challenging proposal. For a medial extraction method (also called skeletonization method) to be usable, it has to meet several desirable properties, as follows:

- *accurate:* The computed medial descriptors have to match the skeleton definition (locus of maximally-inscribed balls) as well as possible, subject to the inherent limitations of the sampling resolution used to capture both the input shape and the output descriptor. Even small inaccuracies can lead to significant errors in the ensuing shape processing applications, such as wrong classifications or incorrect smoothing.

- *scalable:* Medial descriptors should be computable for large and complex 3D shapes, such as the ones produced by modern 3D scanners or modeling applications. These can easily have millions of polygons (for boundary representations) or billions of voxels (for volumetric representations). Additionally, the speed of computing such medial descriptors has to be high – seconds or even less for such complex shapes, in case we want to support interactive applications.

- *robust:* Skeletonization methods should be able to produce 'clean' medial descriptors from shapes having different sampling densities and/or variable amounts of sampling noise. By this, we mean that the resulting descriptors should not be strongly affected by the input sampling or noise characteristics.

- *generic:* Skeletonization methods should be able to handle shapes of a wide variety in terms of geometry and topology, *e.g.* closed or open boundaries, shapes of genus 0 or higher, and shapes representing smooth (natural) objects or shapes having a faceted structure (*e.g.* man-made objects). This way, such methods can be used for the largest possible class of applications.

While many 3D skeletonization methods exist, very few, if any, fully comply with *all* above requirements. For instance, thinning methods working on volumetric representations are simple and scalable, but are not very accurate [43, 119, 126]. Curve skeletonization methods are relatively scalable and robust, but are not generic [31, 32, 37, 64, 172]. Surface skeletonization methods are generic and accurate but relatively slow and less robust [108]. Without complying with all above requirements, the applicability of 3D skeletonization methods in real-world shape processing applications will be inherently limited.

Medial interpretation: Producing a 'raw' skeleton from a 3D shape is only the first step to actually using the skeleton for shape processing. Indeed, one further needs to extract information from the skeleton which is relevant to the specific processing operations we aim to support. This process is also known as computing higher-level medial *features*. Examples of such features are the feature points (contact points of each skeletal point with the input shape's boundary), local thickness (distance from each skeletal point to the closest input shape point), skeleton importance (a metric that encodes the relevance of each skeletal point to the description of the input shape), skeletal components (the distinct manifolds or branches that form the skeleton), junctions (the intersection points or curves where the skeletal manifolds meet), skeleton topology (the graph formed by the manifolds and their intersections), skeleton boundary (the skeletal points which appear on the border of the skeleton itself). Such features can be, in turn, refined to compute higher-level features, which support more advanced analysis and processing operations on the input shape. For example, the skeleton boundary and junctions can be further classified to detect convex and concave parts of the input shape; the skeleton importance can be analyzed to remove small-scale noise details of the input shape; skeletal boundaries can be used to detect (sharp) edges on the shape; and feature points can be analyzed to detect so-called ligature branches that correspond to protrusions, or convex parts, of the input shape.

As for skeletonization itself, several techniques have been proposed to compute higher-level features from 3D curve and surface skeletons. However, many such methods suffer from the same accuracy, scalability, robustness, and genericity problems outlined above for the skeleton extraction. As computing refined skeletal features is, thus, hard, using such features to support shape processing applications is inherently limited. On the other hand, the difficulty of computing refined skeletal features has made it hard to *explore* new ways by which such features could be used to support existing shape processing applications.

Research question: Based on the above points, we can state that computing 3D curve skeletons and their respective features is a relatively well developed field, including many applications. In contrast, we find a major challenge in the practical computation of 3D surface skeletons and their relevant higher-level features, and in the usage of such features to support shape processing applications. As such, the actual usability and usefulness of 3D surface skeletons in practice is still an open question that needs to be answered. Separately, we see that the costs of handling volumetric representations of 3D shapes (and their skeletons) appear to be intrinsically higher than the comparative costs involved by boundary representations.

Given this context, we can now formulate our **first research question**:

How can we compute 3D surface skeletons and related refined features accurately, scalably, robustly, and generically for a wide set of 3D shapes encoded by a discrete boundary representation?

To answer this question, we explore the development of novel skeletonization methods for surface skeletons, and validate them in practice by using them to compute such skeletons from a wide range of 3D shapes and several shape processing applications. This leads us to our **second research question**:

How can we use refined skeletal features extracted from 3D surface skeleton to efficiently support shape processing applications?

To answer this question, we will study how the refined skeletal features computed by the methods developed under the scope of our first question can be used to capture relevant aspects of 3D shapes, such as curvature, parts, and edges. Next, we will use the identified feature-to-shape-characteristic mappings to implement several shape processing operations, such as shape classification and shape segmentation. Finally, we will compare the results of our shape processing with results of established methods in the same class. Thereby, we will validate the usefulness and effectiveness of 3D surface skeletons (and their extracted features) for these applications.

Content-wise, with respect to the above two research questions, this thesis has two contributions:

- **Theory:** The usage of new or existing surface-skeletal *features* to support various shape processing applications highlights connections between shapes and their surface skeletons which have not been explored so far (partly due to the practical inability of computing such skeletons and features). We expect this will strengthen the interest of researchers in exploring additional skeletal features and their use in similar or different applications.

- **Practice:** The creation of efficient and effective 3D surface-skeletonization *algorithms* will support the practical applicability of surface skeletons in real-world applications. Slightly simplifying the discourse, one of the aims of this work is to show that surface skeletons can be made to be as easy and efficient to use in practice as the better-known curve skeletons.

1.5 STRUCTURE OF THIS THESIS

In line with the two main research questions stated above, this thesis has the following structure:

CHAPTER 2 provides an overview of 3D shape representations and introduces several skeleton-related concepts and definitions which will be used on all subsequent chapters. Additionally, it provides an overview of existing skeletonization methods and methods for computing related skeletal features, with a focus on 3D surface skeletons. Related work which is, by its nature, more specific to individual chapters is addressed next in the context of the respective chapters.

CHAPTER 3 presents a method for the fast, robust, and scalable extraction of 3D surface and curve skeletons from large and complex 3D shapes represented as point clouds and polygonal meshes. Additionally, we present how the proposed method can compute related skeleton features such as feature-points and skeleton importance values, used further for skeleton regularization. We also present how the resulting surface skeletons can be used to efficiently reconstruct the input shape, and how to reconstruct compact (meshed) representations of the surface skeletons from a skeleton point cloud. Given the scalability and accuracy of the presented method, this method will form the backbone for our subsequent work on extracting refined skeletal features and using such features in shape processing applications in the following chapters. Separately, we present here a different method that extracts curve skeletons from 2D views of 3D shapes. Similar to our surface-skeletonization proposal, this method is accurate, scalable, robust, and generic, and can handle complex and large shapes represented by point clouds or polygons. Globally, the two skeletonization methods presented in this chapter address the issues of scalability, robustness, and ease of computation of 3D surface and curve skeletons raised by our research questions.

CHAPTER 4 dives deeper into the challenges presented by the analysis of surface skeletons represented by point clouds, such as produced by our method proposed in Chapter 3. Two challenges are addressed here: (1) the extraction of smooth surfaces from noisy point clouds, which enables the use of our surface skeletonization methods directly on such point clouds; and (2) the extraction of the separate manifolds that compose a surface skeleton, which is an important type of refined skeletal feature. To underscore the added-value of our method, we also show its application for the denoising, segmentation, and extraction of meshed surfaces from general point clouds apart from skeletal ones. As such, this chapter partially answers the question of extracting refined features from 3D surface skeletons – in this context, these are skeletal manifolds.

CHAPTER 5 investigates the density properties of a 3D surface-skeleton point cloud. By exploring and exploiting these properties, unique to this type of skeleton representation (in contrast to *e.g.* voxel based representations), we show next how we can support shape segmentation in contexts where known segmentation methods fail to produce good results. We demonstrate our proposal by a practical application for the segmentation of orthodontic dental casts. This chapter thus targets our second research question by showing how the skeletal point-cloud density is instrumental in supporting segmentation applications.

CHAPTER 6 extends our quest for the computation of refined skeletal abstractions. We show how we can compute features such as edges, medial sheets, sheet-intersection curves, and skeleton point classifiers from the surface-skeleton point clouds delivered by our method proposed in Chapter 3. Next, we show how such features can be effectively and efficiently used to support applications in shape classification and segmentation, and compare our results with traditional techniques in these areas. As such, this chapter addresses both the first research question (extracting higher-level skeletal features) and the second research question (using the extracted features to support shape processing applications).

CHAPTER 7 concludes this thesis by discussing our answers to the two main research questions stated in Section 1.4.3 and outlines potential directions for future work in the area of using surface skeletons for additional shape processing applications.