WAVELET SDF-REPS: SOLID MODELING WITH VOLUMETRIC SCANS

Duane Storti Mark A. Ganter William R. Ledoux Randal P. Ching
Department of Mechanical Engineering
University of Washington
Seattle, WA 98195

Yangqiu Patrick Hu David Haynor
Department of Radiology
University of Washington
Seattle, WA 98195

ABSTRACT

This paper describes a new formulation of solid modeling that addresses the issue of including parts whose geometry is determined from volumetric scans (CT, MRI, PET, etc.) along with parts whose geometry is designed by traditional computer-aided design (CAD) operations. Such issues arise frequently in the design of medical devices or prostheses where fit and/or interference between man-made artifacts and existing anatomy are essential considerations, but the modeling formulation presented is not limited to medical applications and can be applied to any parts whose volume can be actually or virtually scanned. Scanner data typically comprises a grid of intensity values and segmentation must be performed to determine the extent of the part. In current practice, the segmented scanner data is run through a polygonizer to obtain an approximate tessellation of the object's surface. Even in the best case scenario where the triangles obtained form a closed surface that accurately approximates the surface of the scanned object, such triangulated models can be problematic due to excessive size.

We present an alternative approach based on recent advances in segmentation with level set methods. The output of the level set computation is a grid of approximate values for the signed distance from each grid point to the nearest point on the surface of the scanned object. We propose interpolating the grid of signed distance values to obtain an implicit or function-based representation (f-rep) for the object, and we introduce appropriate wavelets to effectively perform the interpolation while also providing a number of other useful properties including data compression, inherently multi-scale modeling, and capabilities for skeletal-based modeling operations.

INTRODUCTION

Recent technological advances have led to adoption of three-dimensional (3D) scanning as standard practice in fields ranging from medical diagnostics to security and quality assurance. While the scanning technologies, including magnetic resonance (MR), computed tomography (CT), and positron emission tomography (PET), have made major advances, the modeling technology seems to have lagged behind. It is not yet common to see the output of 3D scanners employed directly in the design process even though it would be highly desirable for applications such as design of medical devices and

prostheses. Some medical device design applications do make routine use of surface scan data (e.g. surface scans of the ear canal are used to produce custom hearing aids), but 3D scanners seem to be used rarely to create solid models. Even with 3D scan data, the common practice is to perform segmentation (to identify interior and exterior voxels) and then to use a polygonizer to derive an approximate tessellation of the surface dividing interior from exterior. The result is either a surface mesh or, if the mesh is properly closed and appropriate topological information is provided to establish connectivity of the surface triangulation, a traditional boundary representation (b-rep). In real applications, such triangulated models can become so large (triangle counts in the millions are not unheard of) that typical solid modeling operations become impractical with commonly available CAD software.

While modeling advances have been limited, some greater progress has been made in visualization due at least in part to the willingness to consider a wider variety of representations and algorithms [1] including both surface visualization and volume visualization. Surface visualization methods model the scattering of light primarily from the boundaries of objects in the scene, while volume visualization methods operate directly on intensity values for each voxel. 2D images are produced applying models of opacity, attenuation, scattering and emission to the voxel values (and gradients) of intensity.

The success of volume visualization in establishing a major presence in computer graphics and movie special effects should provide cause for optimism and expectations of progress. However, volume visualization methods do not provide a direct approach to solid modeling which requires an effective means of point membership classification. While volume visualization produces attractive images, the intensity values do not suffice to exactly specify boundaries between the interior and exterior of a particular object in the scene.

Traditionally (and in the current commercial market), solid modeling employs b-reps comprised of a collection of surface patches with connectivity information that together specify how the collection of patches wraps the surface of the object being modeled. Even if we simplify by specifying that the patches are triangles, the problem of obtaining a b-rep of a scanned object is non-trivial. The scanner data is typically quite noisy and simply choosing a threshold intensity value does not suffice to separate interior from exterior. Recent algorithmic developments, however, have produced significant advances in terms of segmentation (determining the set of voxels belonging to a particular object in the scan region). Most relevant to this paper is the recent work of Hu et al. [2] that combines graph cut methods [3] with level set methods [4, 5] to efficiently achieve segmentation for a variety of imaging modalities and applications. Using the graph cuts approach, weighted links are established between voxels (with the weights depending on intensity values and proximity) and an initial segmentation is achieved by determining how to cut the resulting graph at minimum cost. Level set methods are then employed to improve the accuracy of the segmentation while accounting for penalties associated with local surface characteristics such as curvature. The final output of the level set computation is a grid of values that approximate the signed distance function (SDF) at each voxel.

The traditional approach to go from SDF data to solid model involves polygonization to produce a b-rep model, and there are a variety of established algorithms for doing so.

Bloomenthal [6] provides a review of methods for polygonization of isosurfaces including exhaustive enumeration schemes such as marching cubes [7] or dividing cubes [8], continuation methods including predictor-corrector methods [9] and simplicial continuation methods [10,11] reviewed by Velho et al. [12]. The result of polygonization is, as the name implies, a collection of polygons comprising a polygonal tessellation of a surface approximating that of the scanned object. If topological information is included specifying the connectivity between the edges and vertices (e.g., using formats such as OFF [13] or PLY [14]), the connected tessellation can function as a solid model and is, at least theoretically, suitable for import to a traditional CAD environment. Note, however, that most polygonizers do not produce file formats that are readily imported to CAD environments. When import capabilities do exist, the imported objects may have restricted functionality, and a specialty market has arisen for translation of scan data to CAD format.

MOTIVATIONAL PROBLEMS

To motivate consideration of alternative approaches, we present two real problems that we have encountered that illustrate shortcomings of the traditional approach.

The first problem arose in the course of a project aimed at rapid prototyping the bones of a specific human foot in order to evaluate use of prototypes in surgical planning. Since the foot contains 20 bones, it was necessary to insert pins into the virtual model so that the physical model would maintain proper relative orientation. Interacting with a collection of 20 bones each containing tens of thousands of triangles turned out to be painfully laborious. It took hours to properly position the pins, and the actual Boolean union operation proved intractable.

A more sophisticated motivational problem was proposed to us by a medical device designer. His development team had produced a catheter device for treating a particular type of cardiac defect and demonstrated the effectiveness of the device on porcine test specimens. While such progress represented a significant accomplishment, the device was intended to treat human patients. Refining the design to ensure that it would be safe and effective for use with particular human anatomy (or at least with a specific segment of the human population) was expected to take an additional year which would delay the delivery of the product to market and cost the company millions of dollars in revenue.

To solve each of these problems, it is essential to have a modeling environment that can effectively handle both human-designed artifacts (i.e. the output of CAD systems) and anatomical structures (derived from volumetric scans).

POSSIBLE ALTERNATIVE FORMULATIONS

Based on the difficulties encountered in our motivational problems, we survey the range of possible solid modeling approaches in search of a promising alternative approach.

The first taxonomy of solid modeling representation schemes was compiled by Voelker and Requicha and, as nicely described in Hoffmann [15], included b-reps along with spatial partitioning and constructive solid geometry (CSG). Spatial partitioning methods such as octrees [16] operate by dividing space into sub-regions that are full, empty or partially occupied, and partially occupied regions may be further subdivided and classified to refine the model. CSG involves describing primitive shapes as volume

sets and creating more complicated geometries via Boolean operations on the volume sets. By the mid-1990s, the taxonomy included a wider range of representations: template generation, spatial enumeration, cell decomposition, sweeps, medial axis transformations, ray representations, parametric sample sets, parametric function representations and real function representations.

As mentioned previously, the commercial marketplace remains dominated by b-rep modelers. Octrees still find significant use in visualization applications and some of the other representations may still find specialized application. However, the alternative representation that merits special interest for modeling objects derived from volumetric scans involves real function representations or f-reps [17] (also referred to as implicit models [6]). Before getting into the details of f-reps, why are they of special interest here? Signed distance functions (SDFs) are a subset of f-reps that have very desirable properties (Velho et al. [12] go so far as to identify them as optimal f-reps), and a grid of sampled SDF values is exactly what is produced as the output of modern segmentation algorithms based on level set methods. All that is needed to convert the grid of SDF values to a useful f-rep is an appropriate interpolant. On that basis, we proceed to discuss necessary background on f-reps before returning to demonstrate that wavelets very nicely fill our interpolation needs.

FUNCTION-BASED REPRESENTATIONS (F-REPS)

The most basic ideas of f-reps are typically encountered in introductory calculus and analytic geometry. A real function is specified and geometric regions are identified by the sign of the function; thus point membership classification is performed by evaluating the sign of the function. (Here we employ the convention that negative values occur in the interior.) Thus the function $f(x,y,z) = x^2 + y^2 + z^2 - r^2$ provides a real function that represents a sphere of radius r centered at the origin. While this example is nearly trivial, f-reps became useful for modeling when methods were developed for constructing representations of more complex geometries

The earliest known f-rep models are due to Rvachev and appear in the Russian literature starting in the 1950's. Significant outcomes of that work, including a basic set of primitives and a set of CSG operators, became available in English via the translations and ensuing independent work of Shapiro [18]. Some f-rep foundations were arrived at independently [19,20], and the work of both Rvachev and Ricci have inspired follow on work involving f-rep formulations of sweeps [21, 22, 23], reconstruction of f-reps from surface point data [24, 25], and use of f-reps for efficient implementation of haptic feedback [26]. Sourin and colleagues have maintained an active f-reps presence on the World Wide Web [27] and integrated f-reps into virtual world description languages including VRML and its successor, X3D [28].

The f-rep methods employed in this paper flow most directly from Ensz et al. [29] that presents function-based formulations for the range of operators needed to produce a functional CAD system and identifies desirable properties of functions used to describe geometry. While an f-rep uniquely specifies a particular geometric shape, the representation is not unique and different f-reps are available for a particular object (just as the object's surface can be divided in different ways to create an infinite number of b-reps). Among the various choices of f-reps, a class of defining functions called "linear manifold" functions was identified as particularly desirable because of their utility in

blending operations. Linear manifold functions have the mathematical property that the gradient, wherever it exists, has unit magnitude. Ensz et al. [29] also identified an important subclass of linear manifold functions called "global offset manifold" functions whose level sets are the offset surfaces of the zero level set that defines the object boundary. In other words, the global offset manifold is the graph of the SDF that specifies the distance from any point to the boundary of the object, with interior points indicated with a negative sign. Velho et al. [12] specify two important measures of effectiveness for an implicit or f-rep model: (1) the information conveyed by the value of the function and (2) the extent of the domain for which the function gives valid information; and they conclude that "the optimal implicit function is the Euclidean distance from the surface". For the simple example of a sphere, the signed distance f-rep is readily obtained: $f_{SDF}(x,y,z) = \sqrt{x^2 + y^2 + z^2} - r$.

However, the question remains whether SDFs, with their desirable properties, can be readily produced for objects with more complex geometries. Ensz [30] gives SDFs for a variety of simple primitives and points out [29] that Boolean operations only partially preserve the SDF property (in the exterior/interior for union/intersection respectively).

Thus "manual" construction of SDFs is non-trivial and our approach, as described in the following section, will involve interpolating a grid of sampled values, and the literature includes several significant related works. Raviv and Elber [31] created a freeform sculpting environment using sampled values of a scalar field, but not specifically a distance field, to control a trivariate spline. Schmitt et al. [32] used trivariate B-splines for sculpting heterogeneous F-rep objects and applied wavelet techniques to their splines to support multi-resolution modeling. Their splines are constructed to provide a "distance property" in order to work properly in F-rep modeling operations, but do not directly model Euclidean distance. Freytag et al. [22] also employ multivariate B-splines for creating F-rep models, and their work is closest to ours in that they are more concerned about accuracy in modeling the distance field. They also combine the distance field model with a second field so that the two field models together provide analysis capabilities by giving approximate solutions of partial differential equations subject to conditions specified on the boundary of the object. Given their context, it is natural to focus on accuracy of the distance (and its derivatives) near the boundary while allocating degrees of freedom to the second field to enhance solution accuracy in the interior. Their approach also applies to the case when the distance values are not sampled on a structured grid. Our approach is distinct in seeking to accurately model the entire distance field and to do so directly through the use of wavelets without relying on an intermediate spline construction. We focus on SDF data sampled on a regular grid, so we need to first establish that such data can be obtained for objects from the desired range of model sources.

We know that grids of SDF data are available for objects derived from volumetric scans because the output of the level set segmentation algorithm is exactly that: a grid of SDF values. Can we also obtain grids of SDF values for CAD artifacts? The answer is very direct and positive if the artifact is designed with an f-rep modeler. If the f-rep is an SDF, all that is required is to perform a "virtual volumetric scan" by evaluating the SDF on a grid of points. For more general f-reps, a level set method is applied (just as it was during segmentation) to the virtual scan data to convert it to a grid of SDF data.

For models arising from conventional modelers, there is now a significant literature on computation of SDFs. Jones et al. [1] provide a review of computation of SDF values by a variety of methods including not only level set methods, but also chamfer distance transforms, vector distance transforms, and brute force methods. Representations of distance functions discussed in the review include regular voxel grids, adaptive distance fields, complete distance fields, enhanced distance fields, and dynamic tubular grids. More recently, Houston et al. [34] have introduced hierarchical RLE level sets as a compact data structure for representation of SDF data in the neighborhood of a deformable surface.

We have now arrived at the point where methods have been specified for obtaining a grid of SDF data for any object whether it is a designed artifact or derived from a volumetric scan. Starting from this common base of SDF data, we proceed directly to construction of SDFs via interpolation.

ONE-DIMENSIONAL WAVELET SDF INTERPOLATION

To extend a grid of SDF data to a complete SDF function that will serve as an f-rep solid model, we need an interpolant that allows evaluation to a specified accuracy at any point in the workspace and is sufficiently well-behaved to maintain the unit magnitude property of the gradient of an SDF. In 1D, "solids" consist of a collection of intervals and the unit magnitude gradient property implies that SDFs must be piecewise linear with slope ±1. More specifically, a 1D SDF consists of linear segments with slope +1 terminating at a positive value alternating with segments with slope -1 terminating at a negative value. The terminal points of the linear segments occur at the midpoints of the interior and exterior intervals and comprise the interior and exterior skeleton or medial axis [35, 36]. For concreteness, consider a 1D example with data shown in Fig. 1a. (See the Appendix for numerical details.) This particular data set corresponds to a regular sampling of the SDF for the 1D "solid" corresponding to the union of the pair of intervals $8 \le x \le 24$ and $48 \le x \le 60$. So while this data may seem extremely non-generic, it is typical of 1D SDFs and points us toward the desired interpolation properties: (1) the interpolant should exactly describe low degree polynomials with a finite number of coefficients, and (2) the interpolant should have compact support so that edge effects are localized to the neighborhood of segment boundaries. The first property is also referred to as "vanishing moments" and Daubechies wavelets [37] were designed to provide exactly such properties: compact support and vanishing moments. Daubechies wavelets also provide guaranteed smoothness (depending on the genus of the wavelet), simplicity of operation, and an inherently multi-scale description. Daubechies wavelets have found numerous applications [38] including volume visualization [39], and computations associated with Daubechies wavelets can be implemented on a graphics processor [40].

A brief general discussion of wavelet methods is appropriate before proceeding with the details of Daubechies wavelets. Analysis, in general involves describing functions as linear combinations of basis functions that are preferred because of their simplicity or known properties. The most common embodiment, Fourier analysis, expresses a fairly general class of periodic functions as linear combinations of sinusoidal functions whose frequencies satisfy a scaling relationship. Specifically, a whole number of periods of each basis function must coincide with the period of the function being analyzed. Wavelet analysis is similar but employs scaling relationships on both frequency and amplitude

[38]. Operationally, wavelet analysis involves list correlations [41] of the data with a scaling vector, to identify the low frequency component, and with a wavelet vector, to localize the high frequency components. (Much of the wavelet literature deals with time series analysis, so frequency is the common term. Here we are dealing with functions over a spatial domain, so it is convenient to think of resolution instead of frequency. Correlation with the scaling vector produces a lower resolution model, and correlation with the wavelet vector identifies the size and location of finer details.)

Daubechies wavelets are designed to be compact, so the scaling and wavelet vectors have finite length. (The length of the scaling and wavelet vectors depends on the genus of the wavelet. Increasing the genus provides smoother interpolants at the cost of longer vectors and decreased localization of edge effects.) To be concrete, we consider the Daubechies wavelets of rank 2 and genus 3 for which the scaling vector is $\vec{a} = \frac{1}{2}\{a_0,\ a_1,\ a_2,\ a_3,\ a_4,\ a_5\}$. (Please see the Appendix for numerical values.) The low resolution component is obtained by performing a list correlation of scaling data with the data array followed by 2:1 decimation. The high resolution component is obtained by a similar list correlation and decimation process using the wavelet vector $\vec{b} = \frac{1}{2}\{a_5,\ -a_4,\ a_3,\ -a_2,\ a_1,\ -a_0\}$.

Since the lists of high resolution and low resolution coefficients are each half as long as the original list of data, they are often joined to be stored in the memory originally used to store the data (referred to as the "in-place wavelet transform" [42]). Plots of the low resolution and high resolution coefficients (See Appendix for numerical values.) are shown in Figs. 1b and 1c. The low resolution coefficients provide a lower resolution sampling of the original SDF while the high reolution coefficients are almost all essentially zero as a direct result of the vanishing moment property of Daubechies wavelets. Segments corresponding to samples from low degree polynomials do not contribute to the high resolution component. (In particular, with Daubechies wavelets of genus 3, polynomials of degree $d \leq 2$ produce high resolution coefficients that essentially vanish. For more details on computing wavelet coefficients, see [43].) Exceptional non-negligible high resolution entries occur only near the slope discontinuities. A full wavelet analysis recursively analyzes the low and high resolution components. Numerical results for a full wavelet analysis of the sample data is given in the Appendix, but we note here that 30 of the 64 wavelet coefficients are negligibly small (more than 10 orders of magnitude smaller than the significant coefficients).

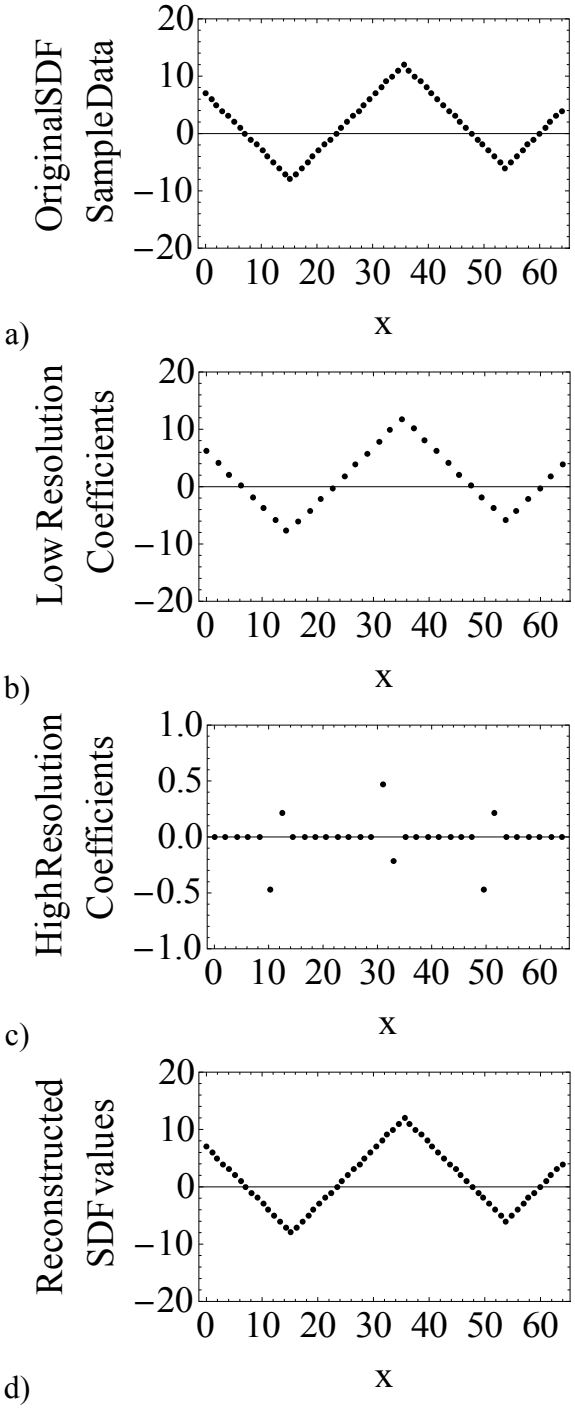


Fig. 1 – Analysis of 1D SDF data. a) Original sampled data. b) Coarser sampling provided by low resolution coefficients. c) High resolution coefficients. Note that the vertical scale differs from the previous plots by a factor of 20. High resolution coefficients vanish except near the slope discontinuities. d) SDF sample data reconstructed after ~40% data compression.

The other basic operations, synthesis (or reconstruction) and differentiation, with Daubechies wavelets work almost exactly the same as analysis but with different vectors used in the list correlation. Synthesis reproduces refined models from low and high resolution wavelet coefficients at the previous level of refinement. First, the low and high resolution coefficients are placed in alternating positions to produce a data vector with doubled length. List correlations are performed with the low resolution synthesis vector $\{a_4, a_1, a_2, a_3, a_0, a_5\}$ at odd-numbered positions and with the high resolution synthesis vector $\{a_5, -a_0, a_3, -a_2, a_1, -a_4\}$ at the even positions to produce data corresponding to an additional level of resolution. Synthesis is constructed as the inverse of analysis, and repeated synthesis reproduces the original sample data (to within round-off error).

Since synthesis involves list correlations of the wavelet coefficients with fixed vectors, negligibly small wavelet coefficients cannot contribute significantly to the synthesis result, and this provides a direct mechanism for data compression. By choosing a threshold criterion for treating wavelet coefficients as zero, we can store only the non-trivial coefficients to attain a compressed representation that, via wavelet synthesis, provides a very good approximation of the original data. In the 1D example, we see that 30 of the 64 wavelet coefficients have magnitudes smaller than 10^{-15}. Treating those coefficients as zeros and synthesizing, we obtain the data shown in Fig. 1d which is visually indistinguishable from the original sample data. In fact, the largest error in the reconstructed values is less than 2×10^{-14}.

At any level of resolution, there is also a wavelet estimate of the derivative that is produced, once again, by a list correlation of the data with a fixed vector. In this case, the vector is comprised of the fundamental connection coefficients [38, Ch. 10]. (See Appendix for numerical values.). The wavelet derivative of the sample data is shown in Fig. 2. Note that the wavelet derivative very nicely maintains the slope at ±1 in the interior of the linear intervals. The only points where the magnitude of the slope differs significantly from unity reside near the slope discontinuities where the gradient of the SDF fails to exist. Thus, non-unit magnitude values of the gradient identify the neighborhood of the skeleton. While 1D skeletons are fairly trivial sets of interval midpoints, we will see that they become more interesting in higher dimensions.

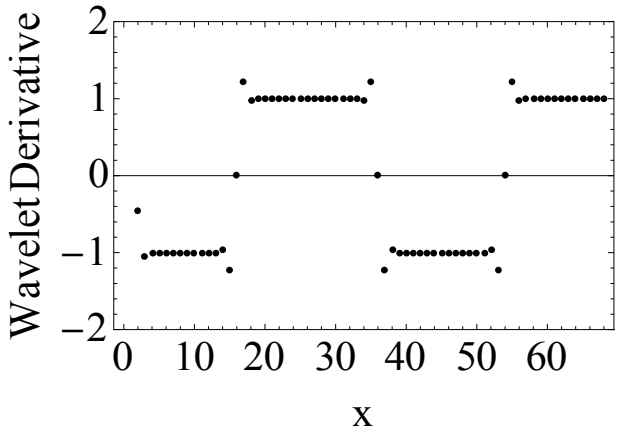


Fig. 2 – Wavelet derivative of sample 1D SDF data.

HIGHER DIMENSIONAL WAVELET SDF-REPS

Now we address the question of how the concept of wavelet interpolation of SDF data extends to higher dimension. The essential difference occurs at the extension from one to two dimensions, so we now consider a 2D example and, to provide direct comparison with established methods based on polygonization, we choose for our example an SDF sampling of a polygon. While Ensz [30] provides SDF formulations for a variety of polygons, for simplicity of presentation we employ his formulation of a rectangle

$$f(x,y)=\sqrt{\max\left\{\begin{matrix}0,\\ |x|-w\end{matrix}\right\}^2+\max\left\{\begin{matrix}0,\\ |y|-h\end{matrix}\right\}^2}-\max\left\{\begin{matrix}0,\\ \min\left\{\begin{matrix}w-|x|\\ h-|y|\end{matrix}\right\}\end{matrix}\right\}$$

where $2w$ and $2h$ are the width and height. A rotation was applied to illustrate that alignment with the coordinate axes is not required. We performed a virtual scan by evaluating this function on a regular grid of points in the xy-plane to produce a grid of SDF values illustrated in Fig. 3. (Note that this figure utilizes the entire 2D domain to illustrate the nature of the SDF values, but that SDF values are as yet only defined on the grid points.) To enhance interpretation of the figure, we provide a geometric description of the graph of the SDF. In the interior, the SDF graph consists of portions of 4 planes: two of the segments are isosceles trapezoids whose shorter parallel sides meet midway between the long sides of the rectangle. The gaps between the trapezoids are filled by two isosceles triangles. Together these four polygons form the shape of an inverted simple hipped roof. Each of the planar segments extends into the exterior region within the perpendiculars to the edges through the vertices (corresponding to the SDF graph in Voronoi regions generated by each edge). These planar regions are connected by sectors of 45° cones with vertices located at the corners of the rectangle (corresponding to the SDF graph in Voronoi regions generated by each vertex). Along any coordinate direction, the SDF data corresponds to regular samples of the distance function along a parametric line through the 2D modeling space. Geometrically, we can think of this SDF graph as the intersection of the plane through the parametric line and perpendicular to the xy-plane with the SDF surface described in the previous paragraph.

Along any parametric line, the SDF graph consists piecewise of intersections of two planes and intersections of a vertical plane with a vertical cone. The plane-plane intersections give rise to linear segments while the plane-cone intersections give rise to hyperbolic segments. Just as in the 1D case, the linear segments are completely described by the low resolution components and the high resolution components vanish. To discover what happens in the hyperbolic regions, consider wavelet analysis of the array of data corresponding to $y=50$. The original SDF data is illustrated in Fig. 4a, and the low resolution and high resolution wavelet coefficients are illustrated in Figs. 4b and 4c respectively. The low resolution coefficients again provide a coarser sampling of the SDF function, and the high resolution coefficients essentially vanish in the linear regions. In the hyperbolic regions, the graph corresponds to a function of the form $f(t)=\sqrt{c_0+c_1t+c_2t^2}$, i.e. the square-root of a quadratic function of the parameter describing the line, as a direct consequence of the Pythagorean theorem. Without the square-root, the graph would be a parabola, the function would be a degree 2 polynomial,

and the high resolution components would again vanish (because the genus 3 Daubechies wavelets have 3 vanishing moments and "kill off" second degree polynomials). What happens with the square-root that produces hyperbolic segments? If we think of expanding the square-root in a Taylor series, the genus 3 wavelet annihilates the high resolution coefficients associated with the first 3 terms of the Taylor series. So in the hyperbolic regions, the high resolution coefficients do not vanish, but they are still relatively small (typically 5 orders of magnitude smaller than the significant coefficients).

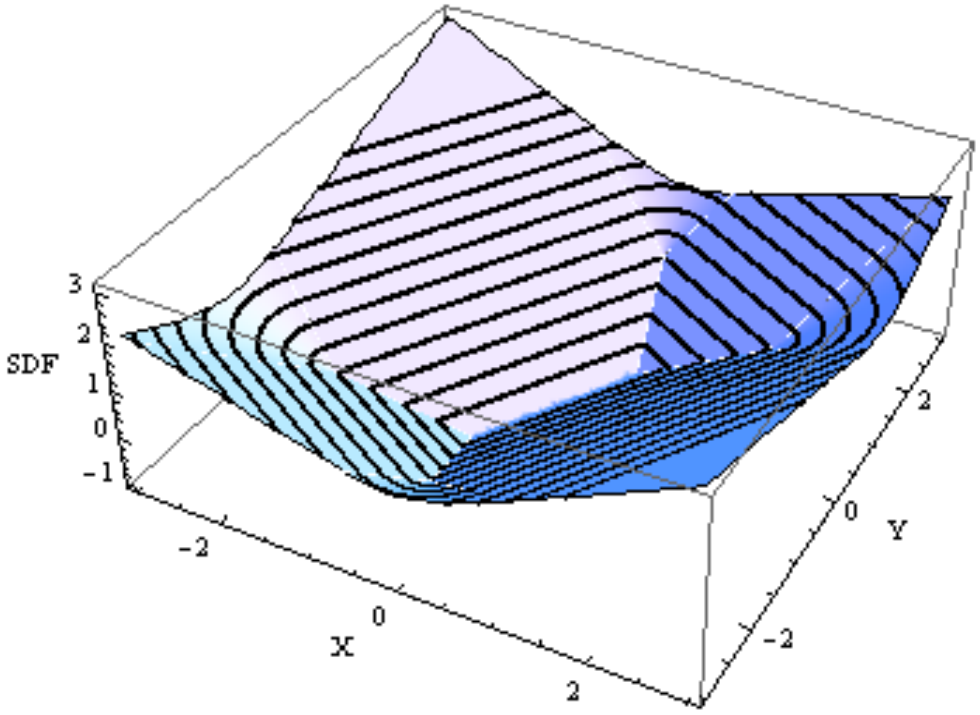
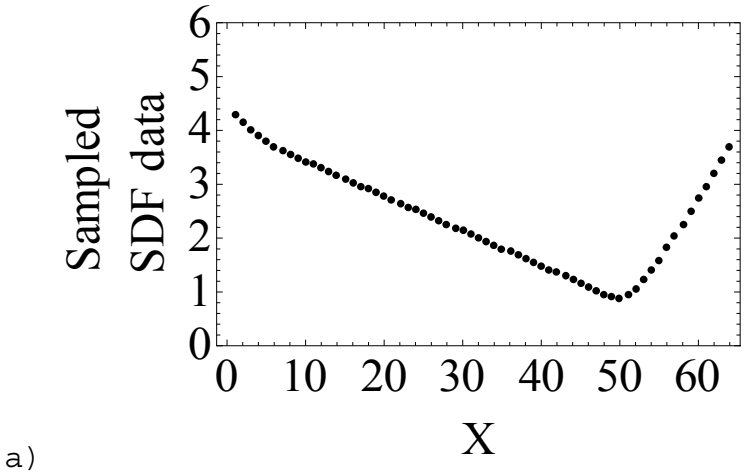


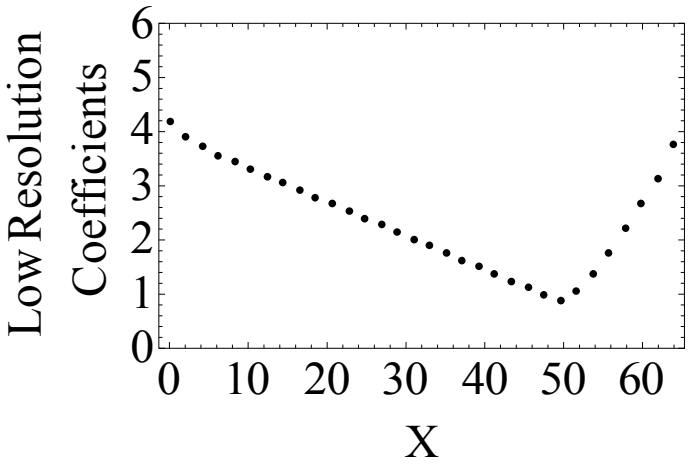
Fig. 3 – SDF for a rectangle in 2D: Contour plot of the level sets superposed on carpet plot of SDF values.

Thus, we can now state the impact of the first major change in extending to higher dimensions. Along an axis of the grid, the SDF sample data contains regions corresponding to roots of quadratic functions. Wavelet analysis is still suitable for describing such functions and, while high resolution coefficients are not completely annihilated, they are small enough to maintain significant promise for data compression.

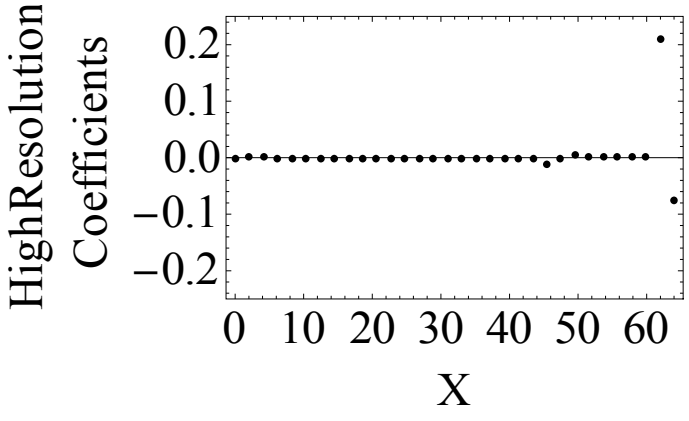
The second major impact is the need for higher dimensional wavelets. While inherently higher dimensional wavelet formulations exist [38], simple tensor product wavelets (in which 1D wavelet analysis is applied sequentially in each coordinate direction) suffice for our purposes. The significant change is that while the initial 1D wavelet analysis is applied directly to the SDF sample data, subsequent analysis applies to the wavelet coefficients produced by the initial analysis. Our initial treatment of the SDF data relied on special properties of the SDF (i.e., that SDFs are piecewise smooth and well-approximated by low degree polynomials), and we need to ensure those properties are inherited by the wavelet coefficients.



a)



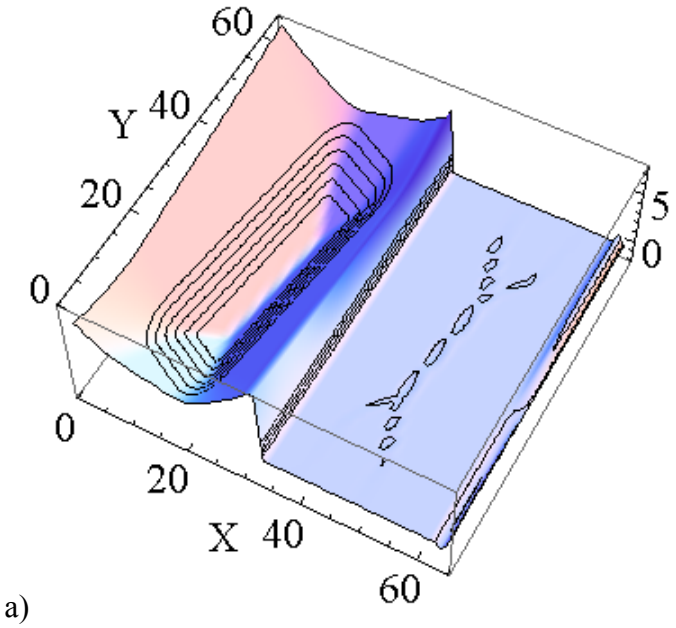
b)



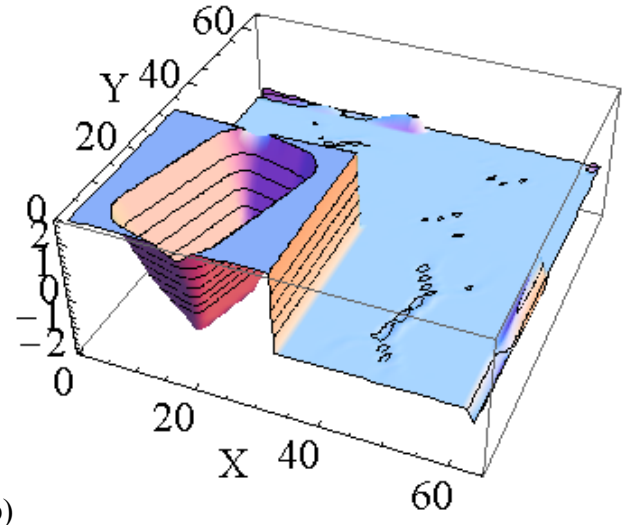
c)

Fig. 4 – Wavelet analysis of one slice of a 2D SDF. a) SDF data points corresponding to Y=50 in Fig. 3. b) Low resolution coefficients provide coarser sampling. c) High resolution coefficients have, with few exceptions, small magnitudes. Note that vertical scale differs by a factor of 24 from a) and b).

As discussed above, the 1D wavelet analysis provides low resolution coefficients that provide a coarser sampling of the SDF function and high resolution coefficients that are, with rare exception, negligibly small. Thus when 1D wavelet analysis is applied to the wavelet coefficients along the second coordinate direction, the first half of the data (arising from low resolution coefficients) corresponds to a scaled coarse sampling of the SDF (which is known to be amenable to wavelet analysis) and the second half of the data consists primarily of small coefficients that approximate the zero function (which is also amenable to wavelet analysis). These ideas are illustrated in Fig. 5.



a)



b)

Fig. 5 – Tensor product wavelet analysis of 2D SDF data.
a) Result of 1D analysis in x-direction.
b) Result of 1D analysis in y-direction of wavelet coefficients plotted in a).

For the raw rectangle SDF data shown in Fig. 3, the result of wavelet analysis along the x-direction is shown in Fig. 5a. The low resolution component (a scaled, coarser sampling of the SDF) appears in the left half of the plot, while the right half is almost everywhere nearly zero. (The largest of the high resolution coefficients occur near the skeleton but are still small compared to typical low resolution coefficients.) Applying wavelet analysis in the y-direction to these wavelet coefficients produces the wavelet coefficients shown in Fig. 5b. Here we see that one quadrant of the plot corresponds to a uniformly scaled coarser sampling of the SDF while the other 3 quadrants contain mostly negligibly small coefficients indicating that the wavelet analysis is suitable for application not only to the raw SDF data, but also to the intermediate wavelet coefficients.

3D WAVELET SDF-REPS

Do any further complications arise when we extend to modeling in 3D? We begin by considering the case of polyhedral models to ensure that the traditional models are included in the modeling domain. The boundary of a polyhedron consists of faces, edges, and vertices, and the global SDF can be thought of as an assembly of pieces of the SDFs for each boundary entity within its Voronoi domain. The distance function in the Voronoi domain generated by a face is linear, and the vanishing moment property ensures very desirable behavior (annihilation of high resolution coefficients) in such regions. For Voronoi domains generated by either an edge or a vertex, the distance function along any parametric line is described by the square root of a quadratic function of the parameter. The situation is exactly like the 2D case in that high resolution coefficients, while not completely annihilated, are still relatively small. The only difference is that an additional 1D wavelet analysis must be applied in the z-direction.

On this basis, wavelet SDF-reps are expected to be effective for modeling polyhedra, but what can we expect from other types of solids? In particular, if this approach is to be of use for biomedical applications, anatomical objects with smooth “organic” shapes are of particular interest. We will illustrate the effectiveness of wavelet SDF-reps for organic solids, but first we discuss why positive results are expected based on the consideration of polyhedra. While ideal behavior (annihilation of high resolution components) occurs in Voronoi cells generated by faces, the “worst case” behavior is associated with the Voronoi cells generated by edges and vertices. For objects with smooth surfaces, the optimal case is not realized, but the worst case scenario is also avoided since there are no actual edges or vertices. With that expectation in mind, we look at some examples of 3D wavelet SDF-reps and their applications.

Figure 6a shows a ray-cast image of a wavelet SDF-rep of a human talus (a bone in the hindfoot) based on a 128 x 128 x 128 grid of SDF data generated from CAT scan data using the Multirigid software package [2]. No polygons are involved in the creation of this image; it is produced directly by finding surface points as roots of the SDF along rays from the view direction and computing shading based on the wavelet derivative estimate of the gradient at the surface point. Note that while root-finding of generic functions can be problematic, the unit magnitude property of the SDF supports simple and efficient root-finding. (For example, a simple stepping method with step size equal to the function value works nicely.)

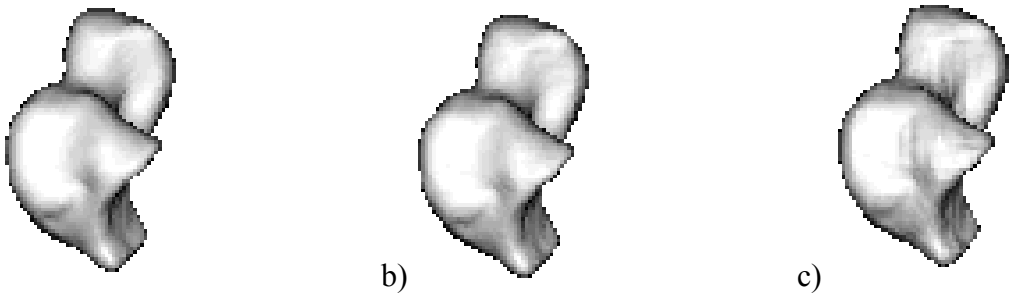


Fig. 6 – 3D wavelet SDF-rep for a human talus. a) Raycast image of the SDF-rep based on 128^3 data grid. b) Raycast image of coarser model based on 64^3 grid of low resolution wavelet coefficients. c) Wavelet data compression of 3D SDF-rep. For this raycast image, 99% of the original data is suppressed.

Figure 6b illustrates the inherent multi-resolution nature of the wavelet SDF-rep. After one level of wavelet analysis, the low resolution coefficients provide a coarser (64 x 64 x 64) sampling of the SDF, and the ray-cast image is shown in Fig. 6b. Note that no anti-aliasing or other image enhancement techniques are applied since the intent is to convey the reduced resolution of the model.

Figure 6c illustrates the potential for data compression. After performing a full wavelet analysis, only the largest 1% of the wavelet coefficients are retained while the remaining 99% were treated as zeros. The compressed data was reconstructed (via wavelet synthesis), and the compressed model nicely preserves the salient features of the original talus model.

To demonstrate the feasibility of traditional CAD operations, we directly take on one of the motivational problems: inserting pins between bone models. Figure 7 shows a raycast image of a pin connecting the talus to the calcaneus (heel bone). The calcaneus is represented as an SDF-rep obtained from CAT scan data, and the pin is represented by an SDF-rep derived from a virtual scan of the SDF-rep of the pin [30]. The Boolean union operation is performed directly on the grids of data (by taking the minimum of the SDF values at the grid point), and the image is obtained by raycasting. Once again, no polygons were involved in the production of this image, and a raster fabrication scheme such as Three-Dimensional Printing [44] could be employed to manufacture the part without polygonization.

To demonstrate non-traditional CAD operations, we consider skeleton-based modeling operations. In earlier work, we surveyed the potential applications of skeleton-based operations [45] and produced some sample skeletal operations on polyhedral models [46]. While the reliability and practicality of polyhedral skeletonization algorithms limited the effectiveness of the earlier skeletal modeler (the result of skeletal modeling operations typically could not be re-skeletonized), the wavelet SDF-rep provides a more promising alternative approach.

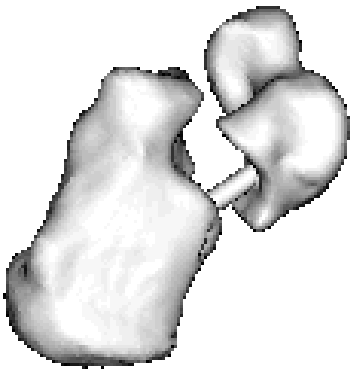


Fig. 7 – Talus and calcaneus SDF-reps derived from 3D scan data unioned with SDF-rep obtained by virtual scan of cylindrical pin.

As in lower dimensional settings, the skeleton corresponds to the set of internal points where the gradient of the SDF is not defined. At points away from the skeleton, the magnitude of the gradient is unity. Translating to our formulation based on a grid of data, grid points in the neighborhood of the skeleton are identified by evaluating the gradient (by applying the wavelet derivative along each coordinate direction) and testing whether the magnitude differs from unity by more than a threshold value. Figure 8a illustrates the skeletal grid points for the talus. Each point on the skeleton has an associated maximal sphere, and the maximal sphere radius is specified directly by the magnitude of the SDF at the skeletal grid point (i.e., the radius of the maximal sphere is the distance to the nearest surface point, and the distance to the nearest surface point is given by the magnitude of the SDF). The skeleton together with the maximal sphere radii comprise the skeletal data [46] that provides an alternate specification of the geometry of the solid. (Note that the finite length of the connection coefficient vector may produce an apparent thickening of the skeleton. Thickening can be managed by employing lower genus versions of the wavelet derivative to improve localization, but it is not really problematic from a modeling perspective. Spheres associated with non-skeletal points are not maximal; such spheres are contained within maximal spheres associated with skeletal points and do not affect the geometric description. In the discrete case of spheres centered on a grid, non-skeletal spheres can provide some additional detail at the cost of a small increase in storage requirements.) Figure 8b illustrates the skeletal-based model of the talus produced as a ray-traced image of the spheres identified as violating the unit magnitude gradient criterion.

Figures 8c, 8d and 9 illustrate skeletal-based solid modeling operations. Fig. 8c shows a solid whose boundary is an offset surface of the talus boundary. The “thickened” talus model consists of the talus skeletal data with the sphere radii increased by a constant value, and the image is produced by conventional raytracing [47] of spheres. Figure 8d shows a transparent thickened talus superposed on the original talus for comparison.

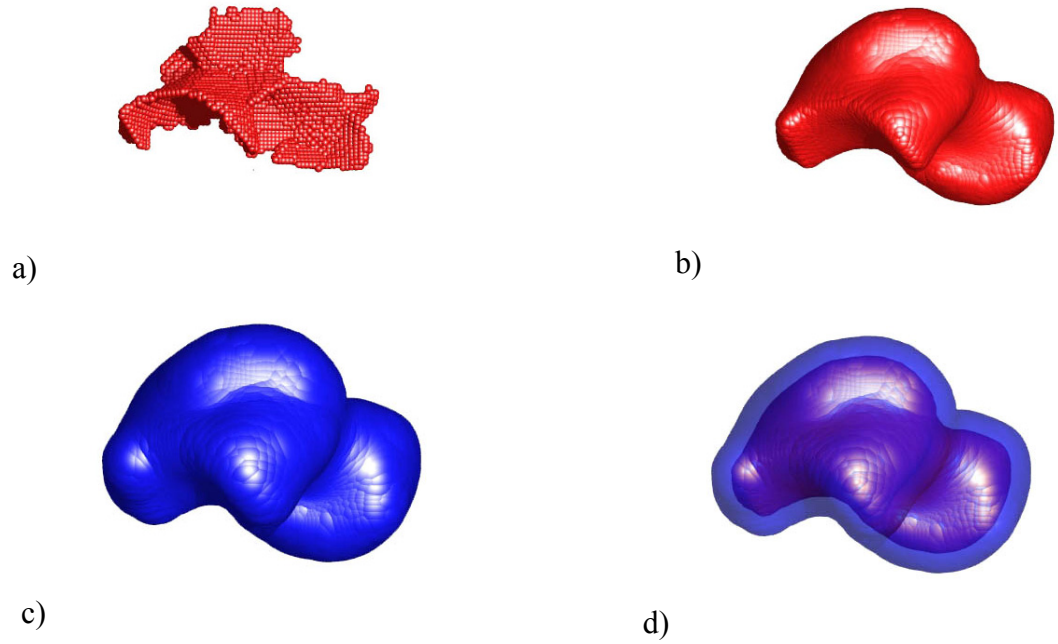
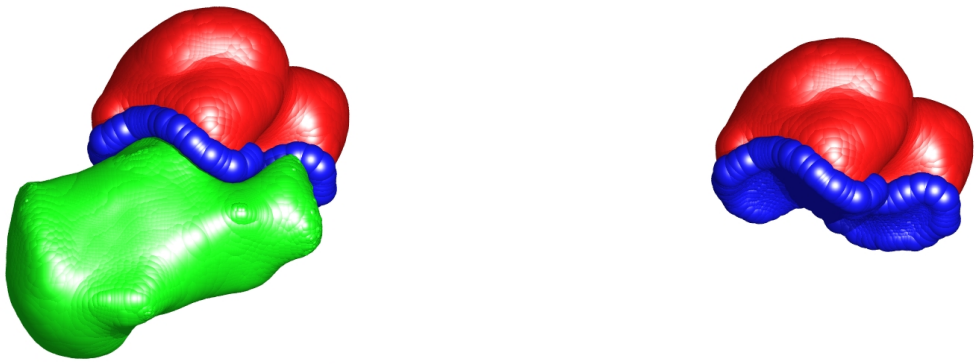


Fig. 8 – Skeletal model derived from wavelet SDF-rep. a) Skeletal grid points of the talus identified using wavelet derivatives to evaluate the gradient. b) Skeletal data (skeleton points with radii specified by SDF) as alternative geometric representation. c) Offset talus obtained by uniformly increasing radii. d) Semi-transparent rendering of offset talus and original talus.

Figure 9 simulates a more realistic problem. The goal is to design a pad to fit between the talus and calcaneus in the scanned configuration. By conventional methods, this would require cutting two large b-reps out of the volume of the entire modeling space and then manually trimming to isolate the pad. (Specifying the trimming object and then cutting out the bones is also possible, but it still requires Boolean operations with multiple large surface triangulations.) Here we illustrate the alternative approach enabled by wavelet-based skeletal methods. We determine the complement of the union of the bones by operating directly on the grids of SDF data. Skeletal points of the complement are identified using the wavelet derivative estimate of the gradient. Skeletal points relevant for the pad are identified by eliminating those that do not lie within a threshold distance of both bones. The skeletal data for the retained grid points comprises a model for a pad that fits between the talus and calcaneus. Fig. 9a shows the pad in position between the two bones. In Fig. 9b, the calcaneus is removed to provide a direct view of the pad. Again the image is produced by ray-tracing spheres, but the corresponding SDF-rep can be produced by specifying the sphere radii on the skeletal grid points as initial data for a level set computation that yields the SDF data on the remaining grid points. The apparent texture of the surface is an artifact of visualizing the spheres. The surfaces of the waveletSDF-rep objects are smooth and tangent to each of the spheres.



a) b)

Fig. 9 – Skeletal modeling examples: Skeletal design of a talar/calcaneal pad. a) The pad (shown in blue) fits between the talus (red) and calcaneus (green). b) The calcaneus is removed to provide a better view of the pad.

CONCLUSION

In this paper, we have presented a new formulation for solid modeling based on wavelet interpolation of grids of signed distance function values (wavelet SDF-reps). The wavelet SDF-rep formulation is motivated by, and well suited for, modeling of organic shapes derived from volumetric scan data. However, we have also indicated that wavelet SDF-reps are capable of modeling artifacts designed by conventional CAD systems. The nature of SDFs, well behaved functions that are locally very well approximated by low degree polynomials and have unit magnitude gradient away from exceptional points, makes them very amenable to wavelet analysis. Basic wavelet operations of analysis, synthesis, and differentiation provide immediate support for useful modeling capabilities including data compression, multi-resolution representation, and skeletal-based solid editing. Using the wavelet SDF-rep formulation, modeling operations including visualization, Boolean operations, skeletal editing, and computation of inertial and surface properties can all be performed directly on the grid of SDF data without ever producing surface triangulations.

To finish on a philosophical note, problems such as biomedical device design typically involve simultaneous modeling of anatomy (derived from scan data) and artifacts (produced by CAD software). Of these two kinds of objects, the anatomical objects typically have relatively more complex geometry, so traditional modeling methods end up having to translate the more complex geometry into the native CAD domain of the simpler artifacts. The wavelet SDF-rep formulation enables the modeling and design process to take place in an environment much closer to the native domain of the objects with the more complicated geometries and thereby offers the potential for reducing the effort required for model translation. Based on this potential savings, the straightforward nature of implementation, and the inherent data compression and multi-resolution modeling capabilities, we consider wavelet SDF-reps to hold significant promise, and we continue to pursue their development in our ongoing research. Note that

the methods presented in this paper are inspired by the treatment of volumetric imaging data that is typically regularly sampled. Readers interested in handling irregularly or randomly sampled distance functions should see the alternative treatment of modeling and analysis based on sampled distance functions by Freytag, et al. [33].

NOMENCLATURE

b-rep: boundary representation

CAD: computer-aided design

CSG: constructive solid geometry

CT: computed tomography

f-rep: function-based representation

MRI: magnetic resonance imaging

RLE: run length encoding

SDF: signed distance function

ACKNOWLEDGMENTS

The research effort described in this paper continues under the support of Grant #CMMI-0733831 from the National Science Foundation. The authors wish to express their appreciation to NSF for that support and to Dr. Mark Ensz of Sandia National Laboratory whose creative, thorough, and well-documented Ph.D. research continues to become more useful as time progresses. We also wish to thank the reviewers for numerous comments that served to enhance the clarity and completeness of the paper.

APPENDIX – NUMERICAL DETAILS OF WAVELET EXAMPLE

The 1D test data is:

$$\vec{v} = \begin{matrix}\{7, 6, 5, 4, 3, 2, 1, 0, -1, -2, -3, -4, -5, -6, -7, -8, -7, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8,\\ 9, 10, 11, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0, -1, -2, -3, -4, -5, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4\}\end{matrix}$$

Note that integer values were chosen for clarity of presentation, not out of necessity.

The numerical values of the coefficients in the scaling/wavelet vectors for Daubechies wavelets of rank 2 and genus 3 are:

$a_0 = 0.4670,\ a_1 = 1.1411,\ a_2 = 0.6504,\ a_3 = -0.1909,\ a_4 = -0.1208,$ and $a_5 = 0.04982.$

One stage of wavelet analysis applied to the sample vector produces the following vector, where the first half of the list are low resolution coefficients and the second half are high resolution coefficients:

$$\{6.2,4.2,2.2,0.18,-1.8,-3.8,-5.8,-7.7,$$
$$-6.2,-4.2,-2.2,-0.18,1.8,3.8,5.8,7.8,$$
$$9.8,12.,10.,8.2,6.2,4.2,2.2,0.18,$$
$$-1.8,-3.8,-5.7,-4.2,-2.2,-0.18,1.8,3.8,$$
$$2.2\times10^{-16},5.6\times10^{-17},-1.1\times10^{-16},-2.8\times10^{-16},-4.4\times10^{-16},-0.47,0.22,-3.9\times10^{-16},$$
$$-2.2\times10^{-16},-5.6\times10^{-17},1.1\times10^{-16},2.8\times10^{-16},4.4\times10^{-16},6.1\times10^{-16},7.8\times10^{-16},0.47,$$
$$-0.22,7.2\times10^{-16},5.6\times10^{-16},3.9\times10^{-16},2.2\times10^{-16},5.6\times10^{-17},-1.1\times10^{-16},-2.8\times10^{-16},$$
$$-0.47,0.22,-2.2\times10^{-16},-5.6\times10^{-17},1.1\times10^{-16},2.8\times10^{-16},4.4\times10^{-16},6.1\times10^{-16}\}$$

Full wavelet analysis of the sample data produces the following set of coefficients:

$$\{1.28,2.1,-5.15,-1.99,2.29,-4.07,1.64,-0.538,0.375,0.29,1.5,$$
$$-0.87,-1.06,0.482,0.728,-0.653,-4.76\times10^{-17},-0.0369,0.246,0.137,$$
$$4.76\times10^{-17},5.12\times10^{-16},0.0369,-0.246,-0.137,3.31\times10^{-16},$$
$$0.00499,-0.788,0.426,-0.215,0.857,-0.435,3.05\times10^{-16},1.39\times10^{-16},$$
$$-2.78\times10^{-17},-1.94\times10^{-16},-3.61\times10^{-16},-5.27\times10^{-16},0.2,0.0498,$$
$$-3.05\times10^{-16},-1.39\times10^{-16},2.78\times10^{-17},1.94\times10^{-16},3.61\times10^{-16},5.27\times10^{-16},$$
$$6.94\times10^{-16},8.6\times10^{-16},-0.2,-0.0498,6.38\times10^{-16},4.72\times10^{-16},$$
$$3.05\times10^{-16},1.39\times10^{-16},-2.78\times10^{-17},-1.94\times10^{-16},-3.61\times10^{-16},0.2,$$
$$0.0498,-1.39\times10^{-16},2.78\times10^{-17},1.94\times10^{-16},1.14,-0.391\}$$

The connection coefficient vector for Daubechies wavelets of rank 2 and genus 3 is:

$$\frac{1}{4}\{0.000342,\ 0.0146,\ -1.42,\ 0.745,\ 0,\ -0.745,\ 1.42,\ -0.0146,\ -0.000342\}.$$

REFERENCES

1. Jones, M., J. Baerentzen and M. Sramek, “3D distance fields: A survey of techniques and applications,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 581-599, 2006.
2. Hu, Y., Haynor, D. , Fassbind, M., Rohr, E., and Ledoux, W., "Image Segmentation and Registration for the Analysis of Joint Motion from 3D MRI,” Proc. SPIE Vol. 6141, p. 133-142, *Medical Imaging 2006: Visualization, Image-Guided Procedures, and Display*, 2006.
3. Boykov, Y. and Veksler, O., “Graph cuts in vision and graphics: Theories and applications,” In: N. Paragios, Y. Chen, and O. Faugeras, (Eds.), Handbook of Mathematical Models in Computer Vision, Springer-Verlag, pp. 79–96, 2006.
4. Sethian, J. A., Level Set Methods and Fast Marching Methods, 2nd Edition, Cambridge University Press, New York, 1999.
5. Osher, S. and Fedkiw, R., Level Set Methods and Dynamic Implicit Surfaces, Springer Verlag, New York, 2002.
6. Bloomenthal, J., Introduction to Implicit Surfaces, Morgan Kaufmann, San Francisco, 1997.
7. Lorensen, W. and Cline, H., “Marching Cubes: A High Resolution 3D Surface Construction Algorithm,” *Computer Graphics*, vol. 21, no. 4, pp. 163-169, 1987.
8. Cline, H., Lorenson, W., Ludke, S., Crawford, C., and Teeter, B., “Two Algorithms for the Three-Dimensional Reconstruction of Tomograms,” *Medical Physics*, vol. 15, no. 3, pp. 320-327, 1988.

9. Rheinbolt, W., “On the computation of multi-dimensional solution manifolds of parametrized equations,” *Numerische Mathematik*, vol. 53, pp. 165-182, 1988.
10. Allgower, E. and Georg, K., Numerical Continuation Methods, An Introduction, (Springer Series in Computational Mathematics, vol. 13), Springer-Verlag, New York, 1990.
11. Allgower, E. and Gnutzmann, S., “Simplicial pivoting for mesh generation of implicity defined surfaces,” *Computer-Aided Geometric Design*, vol. 8, no. 4, pp. 305-325, 1991.
12. Velho, L., Gomes, J., and de Figueiredo, L. H., Implicit Objects in Computer Graphics, Springer-Verlag, New York, 2002.
13. www.geomview.org
14. www-static.cc.gatech.edu/projects/large_models/ply.html
15. Hoffmann, C., Geometric and Solid Modeling: An Introduction, Morgan Kaufmann, San Francisco, 1989.
16. Glassner, A., “Space Subdivision for Fast Ray Tracing,” *IEEE Computer Graphics and Applications*, vol. 4, no. 10, pp. 15-22, 1984.
17. Pasko, A., Adzhiev, V., Sourin, A., and Savchenko, V., “Function Representation in Geometric Modeling: Concept, Implementation, and Applications”, *Visual Computer*, Vol. 11, pp. 429-446, 1995.
18. Shapiro, V., “Real functions for representation of rigid solids,” *Computer-Aided Geometric Design*, vol. 11, pp. 153-175, 1994.
19. Ricci, A., “A Constructive Geometry for Computer Graphics,” *The Computer Journal*, vol. 16, no. 2, pp. 157-160, 1973.
20. Sabin, M., “The Use of Potential Surfaces for Numerical Geometry”, Technical Report VTO/MS/153, British Aircraft Corporation, Weybridge, UK, 1968.
21. Ganter, M. and Storti, D., “Object Extent Determination for Algebraic Solid Models,” ASME Journal on Mechanical Design, vol. 117, pp.20-26, 1995.
22. Ensz, M., “Implicit swept solids,” M. S. Thesis, University of Washington, Seattle, 1994.
23. Sourin, A. and Pasko, A., “Functional representations for sweeping by a moving solid,” *Proceedings 3rd Symposium on Solid Modeling and Applications*, pp. 383-391, ACM Press, NY, 1995.
24. Savchenko, V., Pasko, A., Okunev, O., Kunii, T., "Function representation of solids reconstructed from scattered surface points and contours", *Computer Graphics Forum*, vol.14, no.4, pp.181-188, 1995.
25. Lim, C.-T., Ensz, M., Ganter, M., and Storti, D., “Object reconstruction from layered data using implicit solid modeling,” *International Journal of Manufacturing Systems special focus issue on Layered Manufacturing*, pp. 260-272, 1997.
26. Maneewarn, T., Hannaford, B., Storti, D., and Ganter, M., “Haptic rendering for internal content of an implicit object,” *Proceedings of ASME IMECE*, Nashville, TN, 1999.
27. http://www.ntu.edu.sg/home/assourin/FVRML.htm
28. www.web3d.org
29. Ensz, M., Storti, D., and Ganter, M., “Implicit methods for geometry creation,” *International Journal of Computational Geometry & Applications*, vol. 8, nos. 5 & 6, pp. 509-536, 1998.
30. Ensz, M., “Implicit solid modeling through manifold modification,” PhD Thesis, University of Washington, Seattle, 1997.
31. Raviv, A. and Elber, G., “Three Dimensional Freeform Sculpting via Zero Sets of Trivariate Functions”, Computer-Aided Design, Vol. 32, pp. 513-526, 2000.

32. Schmitt, B. Pasko, A., and Schlick, C., “Constructive Sculpting of Heterogeneous Volumetric Objects Using Trivariate B-Splines”, Visual Computer, Vol. 20, pp. 130-148, 2004.
33. Freytag, M., Shapiro, V., and Tsukanov, I., “Field Modeling with Sampled Distances,” *Computer Aided Design,* vol. 38, no. 2, pp. 87-100, 2006.
34. Houston, B., Nielsen, M., Batty, C., Nillson, O., and Museth, K., “Hierarchical RLE level set: a compact and versatile deformable surface representation,” *ACM Transactions on Graphics*, vol. 25, no. 1, pp. 151-175, 2006.
35. Blum, H., “Biological shape and visual science,” *Journal of Theoretical Biology*, vol. 38, no. 1, 1973.
36. Wolter, F., “Cut locus and medial axis in global shape interrogation and representation,” Design Laboratory Memorandum 92-2, MIT, Department of Ocean Engineering, Cambridge, MA, 1993.
37. Daubechies, I., Wavelets, CBMS-NS Series in Applied Mathematics, SIAM Publications, Philadelphia, 1992.
38. Resnikoff, H. and Wells, R., Wavelet Analysis: The Scalable Structure of Information, Springer-Verlag, New York, 1998.
39. Norton, A. and Rockwood, A., “Enabling view-dependent progressive volume visualization on the grid”, *IEEE Computer Graphics and Applications*, Vol. 23, pp. 22-31, 2003.
40. Garcia, A. and Shen, H.-W., “GPU-based 3D wavelet reconstruction with tileboarding”, *Visual Computer*, Vol. 21, pp. 755-763, 2005.
41. http://reference.wolfram.com/mathematica/tutorial/ConvolutionsAndCorrelations.html
42. Nievergelt, Y., Wavelets Made Easy, Birkhauser, Boston, 1999.
43. Storti, D., Ganter, M., Ledoux, W., Ching, R., Hu, Y. P., Haynor, D., Artifact vs. Anatomy: Dealing with Conflict of Geometric Modeling Descriptions, Proceedings of the SAE Human Modeling Conference, Paper #2007-01-2450, Seattle, WA, 2007.
44. Sachs, E., Williams, P., Brancazio, M., Cima, M., and Kremmin, K., “Three-dimensional printing: Rapid tooling and prototypes directly from a cad model”, *Proceedings of the Manufacturing International*, ASME, Vol. 4, pp.131-136, 1990.
45. Storti, D., Turkiyyah, G., Ganter, M., Lim, C.-T., and Stal, D., “Skeleton-based modeling operations on Solids,” *Proceedings of the Third ACM Solid Modeling Conference*, pp. 141-154, 1997.
46. Blanding, R., Brooking, C., Ganter, M., and Storti, D., “A skeletal-based solid editor,” Proceedings of ACM *Fifth Symposium on Solid Modeling*, Eds. W. Broosvoort and D. Anderson, ACM Press, NY, pp. 141-150, 1999.
47. www.povray.org