Multiresolution Wavelet Representations for Arbitrary Meshes

Apostol Natsev
Department of Computer Science
Duke University

July 3, 1997

Abstract

Wavelets and multiresolution analysis are instrumental for developing efficient methods for representing, storing and manipulating functions at various levels of detail. Although alternative methods such as hierarchical quadtrees or pyramidal models have been used to that effect as well, wavelets have picked up increasing popularity in recent years due to their energy compactness, efficiency, and speed. Wavelet representations have achieved a great success in a wide variety of applications, including graphics, data compression, signal processing, physical simulation, hierarchical optimization, and numerical analysis, among others.

This paper gives an overview of wavelets and their construction, as well as some applications to graphics and 3-D mesh processing at multiple levels of detail. The emphasis is on meshes of arbitrary topology and their multiresolution analysis by means of subdivision wavelets and their generalizations. In particular, both the traditional (first generation) wavelets and the more recent second generation wavelets are considered, together with the lifting scheme for their construction. Examples of such wavelets and their use for representation of arbitrary meshes are given, along with a survey of applications of wavelet-based multiresolution analysis to graphics problems.

1 Introduction

Wavelets have enjoyed a considerable attention in the past few years among the computer graphics community because of their success in solving various problems, such as global illumination and radiosity computation [GSCH93, CSSD96], volume rendering [Mur92], curve and surface modeling [GC95], animation [LGC94], and morphing [HWK94]. In addition, wavelets have proved themselves useful for image processing [Mal89b], image enhancement, noise removal, and feature detection [Mal92]. For an excellent survey of wavelet applications in computer graphics, see [SDS96]. In computer graphics and geometric modeling, objects are typically represented by triangular meshes of various shapes. Applications like special effects and animation often warrant the design and editing of complex geometric models of arbitrary topology. When using laser range scanners for automatic digitizing of graphics models, meshes of extreme complexity and very high resolution are becoming increasingly common. Such fine meshes are expensive to store, manipulate, render, animate, or transmit. Aside from that, designers typically want to be able to change models both globally (e.g., the overall shape) and locally (e.g., minor details), which requires multiresolution editing capabilities. In light of that, wavelet-based multiresolution analysis offers a simple, efficient, and theoretically sound framework for dealing with the above problems. However, wavelet approaches, albeit useful for graphics applications, also need to overcome certain challenges, such as definition and construction on general domains. Traditionally, wavelets have been restricted to infinite domains, such as the real line or plane, while in practice most applications require finite domains, and in general domains of arbitrary topology may be needed. Two-dimensional wavelets are central to a variety of problems, such as image compression, and are typically constructed as tensor products of one-dimensional wavelets [Dau92]. Unfortunately, the tensor-product formulation requires the target functions to be defined on $\mathbb{R}^2$ or a periodic version of it, such as the cylinder or torus. Some alternative constructions [Dau92, JM91] of wavelets on $\mathbb{R}^2$ do not use tensor products but are still not applicable to functions defined on general surfaces (e.g., the sphere). Wavelets defined on arbitrary meshes were pioneered by Lounsbery *et al.* [LDW94, Lou94], and were extended to represent functions defined on arbitrary surfaces (rather than representing the surfaces themselves) by Sweldens *et al.* [Swe, DS96, SS95b, SS95a].

The main idea of multiresolution analysis is to decompose the given function into a simpler low-resolution part and a high-resolution detail part so that the original function can be recovered from the combination of the two parts, while trying to decorrelate the data by compacting most of the energy in the low-resolution part and keeping the data perturbations, called wavelet coefficients, in the second part. This process of filtering the function into subbands continues recursively on the low-resolution part until the most basic (crude) representation is reached. The motivation behind the low-pass and high-pass filtering is that for most practical functions, the data will be correlated and predictable so that most of the wavelet coefficients (or the prediction error) are going to be negligible, and therefore, good approximations of the function could be obtained with only a few large wavelet coefficients. This is the basis for wavelet compression methods, where the wavelet coefficients are subjected to a thresholding scheme that preserves only the most significant ones, given a certain threshold value. By manipulating the threshold (also called quantization parameter), one can achieve multiresolution representations and varying level of detail. This scheme was used in [DJL92], where the authors report excellent lossy compression rates.

The rest of this paper is organized as follows. In Section 2 we present the necessary background, including the first and the second generation wavelets, as well as the so called lifting scheme for constructing them. Section 3 provides an overview of multiresolution analysis of arbitrary meshes, while the following section surveys some of its applications. In particular, the problems of compression and simplification, continuous level-of-detail animation, multiresolution editing of surfaces, global illumination and radiosity computation, surface optimization, and morphing are considered. Finally, Section 5 gives a summary and conclusions.

2 Wavelet Background

As mentioned in the introduction, the basic idea of multiresolution analysis is the decomposition of a function into a low-resolution part (or subband) and a "detail" part. The low-resolution part typically consists of a subset of the data points with weighted averages of neighbors at each of the selected data points. These averages basically implement a *low-pass filter* L that eliminates fine-level details. On the other hand, the detail part usually consists of abstract *wavelet coefficients*, computed as weighted differences that implement a *high-pass filter* H. This decomposition process, also called *analysis*, is typically used to recursively decompose the low-resolution part further into two new subbands, and so on. The whole cascading process is referred to as a *filter bank* and terminates when reaching the coarsest level representation, which, along with the wavelet coefficients from each level, forms the final wavelet representation of the decomposed signal.

The analysis filters L and H are built in such a way that the original function can be reconstructed from the decomposed subbands. The recovery process, called *synthesis*, starts from the lowest (coarsest) level and typically uses some kind of subdivision scheme together with the wavelet coefficients to successively refine the model until the finest (or desired) resolution is achieved. The refining and perturbing steps are implemented through two other filters, collectively called synthesis filters. The challenge in developing good wavelet models (as stated in [Lou94]) is to design the four analysis and synthesis filters so that:

1. the low-resolution versions are good approximations of the original object (in a least-squares sense);

2. the magnitude of a wavelet coefficient reflects the coefficient's importance by measuring the error introduced when the coefficient is set to zero;

3. analysis and synthesis filter banks have time complexity that is linear in the number of data points.

2.1 Definition and basic properties

The basic theory of wavelets and wavelet transforms was developed in the eighties by Yves Meyer, Ingrid Daubechies, and Stephane Mallat [Mey90, Dau92, Mal89a]. A vague definition of wavelets, borrowed from [Swe95], states:

"Wavelets are *building blocks* that can *quickly decorrelate* data."

The emphasized words point out three of the main characteristics of wavelets: mathematical basis, data decorrelation, and speed of implementation. The first feature means that wavelets form a mathematical basis and can serve as building blocks for representing arbitrary functions as a linear combination of wavelet bases. The second means that wavelets possess energy compactness and data decorrelation properties, or in other words, the wavelet representation is more compact than the original one (i.e., has smaller entropy). This is the basis for the use of wavelets for data compression purposes, where only a small fraction of wavelet coefficients is used to approximate the original signal. The decorrelation power of wavelets comes from the fact that they are localized both in space (or time) and frequency. This is accomplished by designing wavelets with compact support (space localization), which are smooth (decay towards high frequency), and which have vanishing moments (decay towards low frequencies). The third main characteristic of wavelets is their speed of implementation. Switching between original and wavelet representation can usually be done in time linear in the size of the data set.

2.2 Second generation wavelets

Classic wavelet constructions historically define the wavelet functions as translates and dilates of a single *mother wavelet*. As defined in [Swe, Swe95], we refer to these traditional wavelets as *first generation wavelets*. They have been extensively used for representing, manipulating, or compressing data on the real line (e.g. audio) or plane (e.g., images) but they are not sufficient when it comes to multiresolution analysis of meshes with arbitrary topology that are most frequently found in graphics applications. Indeed translation and dilation make sense in a Euclidean space but not on complex manifolds, and therefore, in such cases we resort to the more general *second generation wavelets*. These generalized wavelets, as defined in [Swe, Swe95], need not be necessarily translates and dilates of one function but they still possess the three main characteristics of wavelets mentioned above: basis properties, decorrelation power, and fast algorithms.

The main philosophy behind second generation wavelets is to achieve all desirable properties for wavelets adapted to more general settings than the real line or plane. The key realization here is the fact that translation and dilation are not required in order to obtain wavelets with the desired properties. The only invariant that needs to be maintained is that a basis function can be expressed as a finite linear combination of wavelet basis functions in a deeper, finer level. The main distinction from classical wavelets is that the filter coefficients of a second generation wavelet need not be the same throughout, but can change locally in order to adapt to the local changes in the manifold. Typical examples of second generation wavelets are:

- *Wavelets on bounded domains:* The construction of wavelets on domains in a Euclidean space is used in applications such as image segmentation and numerical solution of PDEs. A special case is the construction of wavelets on an interval, where care must be taken near the endpoints in order to avoid introducing artifacts at the boundaries due to zero padding and mirror reflection techniques.

- *Wavelets on curves and surfaces:* To analyze, store, manipulate, or compress data defined on curves or surfaces, one needs wavelets intrinsically defined on these manifolds, independent of parametrization.

- *Weighted wavelets:* Diagonalization of differential operators and weighted approximation require a basis adapted to weighted measures. Wavelets biorthogonal with respect to a weighted inner product are needed.

- *Wavelets and irregular sampling:* Many real life problems require basis functions and transforms adapted to irregularly sampled data.

2.3 The lifting scheme

First generation wavelets are typically constructed with Forier analysis because translation and dilation become algebraic operations under the Forier transform. In the case of second generation wavelets, though, the Forier transform can no longer be used as a construction tool. An alternative construction method, called *the lifting scheme* was introduced by Sweldens [Swe, Swe95]. The method was inspired by the work of Donoho [Don92] and Lounsbery *et al.* [LDW94, Lou94], and generalizes the latter. Donoho [Don92] shows how to build wavelets from interpolating scaling functions by subdivision rather than by the Forier transform, while Lounsbery *et al.* [LDW94, Lou94] build a multiresolution analysis of surfaces using the so called *subdivision wavelets*, wavelets defined on arbitrary meshes with *subdivision connectivity* (i.e. meshes that can be derived from a base mesh with a finite number of subdivision steps). The lifting scheme doesn't use Forier analysis and so can be used for both first and second generation wavelet construction. Some of the lifting scheme advantages, compared to the classic Forier construction or alternative schemes, are:

- *Generality:* can be used to construct second generation wavelets;
- *In-place computation:* no extra memory is required;
- *Immediately invertible:* trivial inverse transform of the same complexity as the forward transform;
- *Efficient implementation:* could be used for classical wavelet construction just because it is faster than the Forier construction.

The lifting scheme consists of three fundamental steps: *split, predict,* and *update*. Assume we start with an abstract data set that we refer to as λ_0. In the first stage we *split* the data into two subsets λ_{-1} and γ_{-1} and refer to γ_{-1} as the wavelet subset. Choices for splitting include a simple scissors-like cut of the data set or an interlaced partitioning into even and odd data sets (the *Lazy wavelet*). In the second stage, we try to use the λ_{-1} subset to *predict* the γ_{-1} subset based on the correlation present in the original data. To that purpose, we construct a prediction operator P, so that $\gamma_{-1} \approx P(\lambda_{-1})$. We then replace γ_{-1} with the difference between itself and the prediction: $\gamma_{-1} = \gamma_{-1} - P(\lambda_{-1})$. If the prediction is reasonable, this difference will contain much less information then the original γ_{-1} set. The wavelet subset now encodes how much of the data deviates from the model on which P was build.

Often, we might want to maintain some global properties of the original data set. For example, in the case of an image, we would like the smaller images λ_{-j} to have the same overall brightness, i.e., the same average pixel value. This is achieved in the third stage, where we *update* λ_{-1} so that a certain scalar quantity $Q()$ (e.g., the mean) is preserved: $Q(\lambda_{-1}) == Q(\lambda_0)$. To that effect, we use the already precomputed wavelet set γ_{-1}so that the latter preserves $Q()$. In other words, we construct an operator U and update $\lambda_{-1} = \lambda{-1} + U(\gamma_{-1})$. At this moment we can recursively replace the original data set with the smaller set λ_{-1} and the wavelet set γ_{-1}. With a good prediction, the two subsets $\{\lambda_{-1}, \gamma_{-1}\}$ yield a more compact representation than the original set λ_0. We then iterate the scheme, and eventually replace the original data with the wavelet representation $\{\lambda_{-n}, \gamma_{-n}, \ldots, \gamma_{-1}\}$, hoping that this would be a more compact representation because of the intrinsic correlation in the original data. To derive the inverse transform all we need to do is reverse the order of the above steps and replace + signs with −, and vice versa. This illustrates one of the nice properties of the lift scheme—no complicated Forier analysis is needed in order to obtain the inverse of the transform.

In [DS96] Daubechies and Sweldens show that every wavelet or subband transform with finite filters can be obtained with a finite number of lifting steps. By manipulating the three steps in the scheme—splitting (or subdivision), prediction, and update—one can use the lifting scheme to construct both first and second generation wavelets. The Lazy wavelet and the Haar wavelet are typically used as a basis for the lifting scheme, and by means of the update step they can be lifted to various other well-known wavelets. Other examples of subdivision schemes that combine the splitting and the prediction steps include piecewise linear subdivision (which produces polyhedra); the schemes of Catmull & Clark [CC78, HKD93], Loop [Loo87], or the butterfly scheme of Dyn *et al.* [DLG90] (which produce tangent-plane-smooth G^1 subdivision surfaces); and the modifications by Hoppe *et al.* [Hop94, HDH$^+$94] that produce piecewise smooth surfaces with selected discontinuities.

2.4 An example: spherical wavelets

A typical example of the use of lifting to construct second generation wavelets is on the sphere, which is a 2-manifold (i.e., a surface). The construction starts with the tessellation of the sphere into cells (i.e., spherical triangles, their edges, and their vertices). This is done starting from a base complex that is combinatorically isomorphic to a pyramid, octahedron, or icosahedron. The base complex is then recursively subdivided (by adding edge midpoints and subdividing each triangle into four subtriangles) until the desired resolution is reached. Other subdivision schemes are certainly possible. A data (or function) value is associated with each vertex (as opposed to each edge or triangle, which are alternative choices), which leads to a natural splitting of the vertices into an "even" and an "odd" subset, as needed in the first step of the lifting scheme. The even vertices are the ones from the coarser level, while the odd ones are the new vertices introduced in the finer level. This by itself defines a Lazy wavelet transform, which does nothing but subsampling (no prediction or update).

Examples of the most popular prediction operators are the linear, quadratic, and butterfly schemes, among others, but higher order polynomial and B-spline predictors are also possible. Examples of criteria for the update lifting phase include preserving the average energy or making sure that the resulting wavelet has at least one vanishing moment (i.e., it will model polynomials of degree at most 1 losslessly). So defined, these spherical wavelets can be used to process scalar functions defined on a sphere. Sample applications are spherical texture [SS95a], environmental maps and earth elevation data (topography) [SS95b, KL97], and illumination [SS95b].

3 Multiresolution Analysis of Arbitrary Meshes

3.1 Overview

As mentioned before, a multiresolution representation of a mesh, as developed by Lounsbery *et al.* [LDW94, Lou94], consists of a simple base mesh together with a sequence of local correction terms, called wavelet coefficients, capturing the detail present in the object at various resolutions. The tools for constructing such multiresolution representations are based on wavelets that are defined over meshes of arbitrary topological type, and were pioneered by Lounsbery *et al.* [LDW94, Lou94] through his subdivision wavelets, later generalized to second generation wavelets by Sweldens and others [Swe, Swe95, DS96, SS95b, SS95a], who also provided the lifting scheme for their construction. Both methods, though, have a prerequisite that the input mesh have *subdivision connectivity*, or in other words, that the mesh can be obtained from a simple base mesh by recursive subdivision (typically, 4-to-1 triangle splitting). Even though both methods deal with meshes of arbitrary topology, the subdivision connectivity restriction limits the domain of applications severely because few meshes have that condition in practice. The complex meshes obtained from automatic laser range scanning and digitizing, for example, certainly do not have the subdivision connectivity property, which is a serious limitation. Recently Eck *et al.* [EDD+95] addressed this problem by use of harmonic mapping techniques that interpolate input meshes to genus-equivalent triangular meshes that possess the desired property. The basic idea of their remeshing method is to construct a parametrization of the input mesh M over a suitably defined domain mesh K, and then to resample it in order to produce a new mesh M^* that has subdivision connectivity and is of the same topological type as M. The remeshing process consists of three parts: *partitioning* of M into a small number of triangular regions by means of harmonic maps (i.e., maps that preserve as much of the metric structure, such as lengths, angles, etc., of M as possible); *parametrization* of each region over the corresponding face (made to fit continuously so that collectively they define a globally continuous parametrization); and *resampling* through recursive 4-to-1 splits on each of the faces. The method is guaranteed to produce a mesh that differs from the original by no more than a specified remeshing tolerance. Combined with either Lounsbery's or Sweldens' method, the remeshing procedure of Eck *et al.* [EDD+95] serves as a preprocessing step and forms a complete multiresolution analysis system for meshes of arbitrary topology. The range of possible applications, related to this wavelet-based representation, is discussed in the next section.

4 Applications

Second generation wavelets extend the benefits of multiresolution analysis to a wide variety of new applications. A survey of these, based mainly on [Lou94], include, but is not necessarily limited to, the following:

Compression. Multiresolution analysis is often used for data compression applications. Mallat [Mal89a] and De Vore [DJL92], among others, use wavelet techniques for efficient image compression. Finkelstein and Salesin [FS94] develop a wavelet-based method for approximating B-spline curves within an L^∞ tolerance. Other examples of wavelet-based compression include [CS92, Luc92, MH92, BBS94]. The general scheme of wavelet compression is typically implemented in three steps: wavelet decomposition, quantization (or thresholding) of wavelet coefficients, and reconstruction. Lounsbery *et al.* [LDW94, Lou94] extend the traditional wavelet-based compression approaches from the plane to surfaces of general topology. They provide examples of polyhedral compression and compression of functions defined on surfaces. The same problem is also tackled in [KL97, SS95b, SS95a], where spherical wavelets (cf. Section 2.4) are used to compress elevation topographic data, for example, defined over the Earth globe. One thing to note in compression applications is the underlying data structure that stores the wavelet coefficients. In order to achieve good compression ratios, one needs to be very careful in its design, as exemplified in [KL97, LDW94, Lou94]. Important criteria to be considered in polyhedral compression schemes include curvature match (the graphics equivalent of an approximation quality measure), closeness measure (e.g., based on L^2 or L^∞ norm), topology preserving properties, capabilities for smooth blending of different levels of detail, specifiable compression rate requirements, and speed, among others. The multiresolution analysis compression schemes usually offer very good quality and adaptation to local curvature at reduced computation cost. Such schemes also provide continuous transmission and smooth blending of various levels of detail. For comparison, non-wavelet polyhedral compression methods are surveyed by Heckbert and Garland [HG94], and also by Lounsbery [Lou94], with the most notable solutions to the problem being the following:

- Turk [Tur91, Tur92] develops an algorithm that places a user-specified number of vertices over the original polyhedral surface, and then uses repelling forces to spread them out. The repelling forces vary according to curvature information (which is sometimes estimated unsuccessfully if not given), and the algorithm results in a mesh with vertices on the original surface, but there are no guarantees as to how far the approximation is from the original data at points other than the sampled ones.

- Rossignac and Borrel [RB93] describe a compression method that divides the ambient space into subregions so that vertices in the same region are coalesced into a single vertex using a weighted approximation, such as their centroid. The new vertices are then reconnected to form a final collection of faces. This method is simple and highly efficient, and thus has one of the best running times among polyhedral compression methods. In addition, the method does not preserve the topological type of the original shape, which can be a significant advantage. The paper also presents an additional method for blending between levels of detail. However, is does not adapt well to curvature changes and tends to blur finer detail.

- Schroeder *et al.* [SZL92] provide an iterative mechanism, which they call decimation, for reducing mesh complexity by repeatedly passing over the mesh looking for areas that can be simplified. Being an iterative algorithm, this technique does not have any guarantees about how far the end result may deviate from the original, but empirical results indicate that the method adapts well to changes in curvature. This technique doesn't treat the problem of blending between approximations.

- Hoppe *et al.* [HDMS93, Hop94, HDH$^+$94] develop methods for compressing both piecewise linear polyhedral surfaces as well as meshes that represent subdivision surfaces by balancing both the simplicity of the representation and an L^2 distance measure from the original surface. This method appears to have excellent quality that adapts very well to local curvature but it comes at the price of very long running time. No mechanism for smooth level-of-detail blending is provided.

Interactive viewing. A high-quality animation usually involves very large and complex models containing thousands or millions of polygons. Rendering of such fine detail at all times is usually unnecessary—for example, detail may be required in close-ups but not needed in farther views. The cost of showing all detail is particularly prohibitive in animation applications, where up to 30 frames per second may be required. In that respect, multiresolution analysis offers solutions to several problems in that area. Level of detail has been used in flight simulators for many years [NN84]. Funkhouser and Sequin [FS93] use it for real-time walkthroughs through architectural models. They use simplified models for distant objects and detailed models for close-up views. Their approach, however, does not generate the various levels from a single representation, and does not address the issue of varying the level of detail smoothly. Lounsbery [Lou94] shows how multiresolution analysis can be used for animation previewing by providing low-cost rough model approximations, as well as for level-of-detail control. His multiresolution scheme not only provides natural multiple levels of detail, represented and stored efficiently in a single model, but also addresses the problem of varying the level-of-detail continuously so that transition between different levels is smooth and gradual.

Multiresolution editing. Hierarchical B-splines, as introduced by Forsey and Bartels [FB88], provide a powerful mechanism for editing shapes at various levels of detail but can be applied to a restricted class of surface topologies. In contrast, the wavelet-based multiresolution analysis provides an alternative capable of handling meshes of arbitrary topological type. One approach was developed by Finkelstein and Salesin [FS94] who demonstrate the potential of wavelets for editing B-spline curves and tensor product B-spline surfaces. Their techniques were further modified for surfaces of arbitrary topological type by Hoppe *et al.* [Hop94, HDH$^+$94] and adapted for editing polyhedra, smooth subdivision surfaces, surfaces with boundaries, and smooth subdivision surfaces with corners and "darts." Zorin, Schröder and Sweldens [ZSS97] describe another approach for multiresolution mesh editing. They develop a scalable interactive multiresolution editing system for meshes of arbitrary topology that allows for both global (e.g., shape) changes as well as local modifications preserving the overall topological structure.

Global illumination and radiosity. Tensor product wavelets have been used to rapidly solve physical simulation problems such as global illumination and radiosity computation [GSCH93, HSA91, CSSD94]. Bidirectional reflectance distribution functions (BRDFs) are used to model light reflectance in global illumination. BRDFs are defined over a hemispherical domain but are typically modeled either with an approximation parametrized over the grid [CSSD94] or with spherical harmonics [SAWG91]. Using second generation wavelets, and in particular, the spherical wavelets described in Section 2.4, it is possible to approximate BRDFs with wavelets truly parametrized over a hemispherical domain. The use of spherical wavelets in the more general setting of spherical radiance distributions, such as those of Gondek *et al.* [GMN94], can lead to improved radiance computation. As a separate application, subdivision wavelets can be used to cluster groups of polygons into an appropriate approximation, as an alternative to the method of Smiths *et al.* [SAG94]. This can be used as a preprocessing step to the algorithm of Hanrahan *et al.* [HSA91] for accelerating hierarchical radiosity computation.

Surface optimization. The multiple levels of approximation produced by wavelet techniques offer a sort of multigrid technique for optimization. Pentland [Pen92] shows that wavelets permit multigrid techniques to be extended for surface interpolation and optimization problems over regular grids. Meyers [Mey94a, Mey94b] uses wavelet approximations of curves to implement a multigrid technique for area minimization of contour tilings [MSS92]. The resulting technique runs in $O(n \log n)$ time, and produces results that are comparable to the standard $O(n^2 \log n)$ global optimization algorithm of Fuchs *et al.* [FKU77]. A similar speedup of one-dimensional optimizations using wavelet-based multigrid techniques is reported by Gortler *et al.* [GSCH93]. The subdivision wavelets of Lounsbery *et al.* [LDW94, Lou94] can also be used to accelerate optimization over surfaces of arbitrary topology.

Volume morphing. Morphing, or gradually transforming a three-dimensional object into another of similar topological type, is another area to which multiresolution analysis has been applied. As described in [Lou94], wavelets provide certain key elements of morphing which make wavelet domain morphing preferable when compared to the traditional spatial domain morphing (e.g., due to fewer high frequency distortions). Kent *et al.* [Ken92] describe a technique for homeomorphic transformation between two shapes parametrized over the sphere. Second generation wavelets can be used to generalize this technique to allow transformations between topologically similar shapes of arbitrary topological type. Hughes [Hug92] develops a method for smoothly transforming between volumetric shapes of arbitrary topology, parametrized over the regular $\mathbb{R}^3$ domain. His technique involves computing a Forier analysis of the shapes first, followed by a smooth removal of high-frequency data and interpolation between the low-frequency features of the two shapes. The same approach can be used in conjunction with multiresolution (rather than Forier) analysis in order to obtain similar wavelet-based morphing technique. In [HWK94] wavelets are used to decompose the volumetric datasets into a set of frequency band, followed by a smooth interpolation in each band and reconstruction to form the morphed model. The authors also present a technique for establishing an appropriate voxel correspondence, which in conjunction with the wavelet-domain interpolation produces smooth volume morphing with less distortion than spatial morphing.

5 Summary and Conclusion

In this paper we have reviewed the basic classical wavelet theory, as well as the more general and fairly recent second generation wavelets and the lifting scheme for their construction. We have given an example of using the lifting scheme to construct second generation wavelets defined on a sphere, as well as their application to compression of functions defined on the sphere (such as elevation data).

We have also covered multiresolution analysis techniques for arbitrary topology meshes, based on the above mentioned wavelets, and surveyed a wide range of applications to computer graphics, including polyhedral compression, global illumination and radiosity, multiresolution editing, morphing, animation, and support of continuous level-of-detail. The large variety of applications that wavelets have been successfully applied to indicates that they are establishing themselves as a basic tool in computer graphics, which provides simple, efficient, and compact representations of geometric objects, along with a fast manipulation of them.

References

[BBS94] D. Berman, J. Bartell, and D. Salesin. Multiresolution painting and compositing. In *Computer Graphics Annual Conference Series*, pages 85–90, July 1994.

[CC78] E. Catmul and J. Clark. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design*, 10(6):350–355, 1978.

[CS92] C. K. Chui and X. Shi. Wavelets and multiscale interpolation. In T. Lyche and L. L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design II*. Academic Press, 1992.

[CSSD94] P. H. Christensen, E. J. Stollnitz, D. H. Salesin, and T. D. DeRose. Wavelet radiance. In *Proceedings of the Fifth Eurographics Workshop on Rendering*, pages 287–302, Darmstadt, Germany, 1994.

[CSSD96] Per H. Christensen, Eric J. Stollnitz, David H. Salesin, and Tony D. DeRose. Global illumination of glossy environments using wavelets and importance. *ACM Transactions on Graphics*, 15(1):37–71, Jan. 1996.

[Dau92] Ingrid Daubechies. Ten lectures on wavelets. In *CBMS-NSF Regional Conf. Series in Appl. Math.*, volume 61, Philadelphia, PA, 1992. Society for Industrial and Applied Mathematics.

[DJL92] R. A. DeVore, B. Jawerth, and B. Lucier. Image compression through wavelet transform coding. *IEEE Transactions on Information Theory*, 38(2):719–746, March 1992.

[DLG90] N. Dyn, D. Levin, and J. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics*, 9(2):160–169, April 1990.

[Don92] David L. Donoho. *Interpolating Wavelet Transforms.* Department of Statistics,Stanford University, preprint edition, 1992.

[DS96] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. Technical report, Bell Laboratories, Lucent Technologies, 1996.

[EDD+95] Matthias Eck, Tony DeRose, Tom Duchamp, Hugues Hoppe, Michael Lounsbery, and Werner Stuetzle. Multiresolution analysis of arbitrary meshes. In *Proceedings of SIGGRAPH '95*, pages 173–182, New York, 1995. ACM.

[FB88] D. Forsey and R. Bartels. Hierarchical b-spline refinement. *Computer Graphics*, 22(4):205–212, 1988.

[FKU77] Henry Fuchs, Z. M. Kedem, and S. P. Uselton. Optimal surface reconstruction from planar contours. *Communications of the ACM*, 20(10):693–702, October 1977.

[FS93] T. A. Funkhouser and C. H. Séquin. Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments. In *Computer Graphics Annual Conference Series*, pages 247–254, August 1993.

[FS94] Adam Finkelstein and David H. Salesin. Multiresolution curves. In *Proceedings of SIGGRAPH '94*, pages 261–268, New York, 1994. ACM.

[GC95] S. J. Gortler and M. F. Cohen. Hierarchical and variational geometric modeling with wavelets. In *Proceedings Symposium on Interactive 3D Graphics*, pages 35–42, April 1995.

[GMN94] J. S. Gondek, G. W. Meyer, and J. G. Newman. Wavelength dependent reflectance functions. In *Computer Graphics Annual Conference Series*, pages 213–220, July 1994.

[GSCH93] Steven J. Gortler, Peter Schröder, Michael F. Cohen, and Pat Hanrahan. Wavelet radiosity. In *Proceedings of SIGGRAPH '93*, pages 221–230, New York, 1993. ACM.

[HDH+94] H. Hoppe, T. Duchamp, M. Halstead, H. Jin, J. McDonald, J. Schweitzer, and W. Stuetzle. Piecewise smooth surface reconstruction. In *Computer Graphics Annual Conference Series*, pages 295–302, July 1994.

[HDMS93] H. Hoppe, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimization. In *Computer Graphics Annual Conference Series*, pages 19–26, Aug. 1993.

[HG94] P. S. Heckbert and M. Garland. Multiresolution modeling for fast rendering. In *Proceedings of Graphics Interface*, pages 43–50, 1994.

[HKD93] M. Halstead, M. Kass, and T. DeRose. Efficient, fair interpolation using catmull-clark surfaces. In *Computer Graphics Annual Conference Series*, pages 35–44, August 1993.

[Hop94] H. Hoppe. *Surface Reconstruction from Unorganized Points*. PhD thesis, University of Washington, Seattle, Washington, June 1994. TR 94-06-01.

[HSA91] P. Hanrahan, D. Salzman, and L. Aupperle. A rapid hierarchical radiosity algorithm. *Computer Graphics*, 25(4):197–206, July 1991.

[Hug92] J. F. Hughes. Scheduled forier volume morphing. *Computer Graphics*, 26(2):43–46, 1992.

[HWK94] Taosong He, Sidney Wang, and Arie Kaufman. Wavelet-based volume morphing. In *Proceedings of Visualization '94*, pages 85–92, Los Alamitos, CA, 1994. IEEE Computer Society Press.

[JM91] R. Q. Jia and C. A. Mitcchelli. Using refinement equations for the construction of pre-wavelets ii: Powers of two. In P. J. Laurent, A. LeMéhauté, and L. L. Schumaker, editors, *Curves and Surfaces*. Academic Press, 1991.

[Ken92] J. R. Kent. Shape transformation for polyhedral objects. *Computer Graphics*, 26(2):47–54, 1992.

[KL97] Krasimir Kolarov and William Lynch. Compression of functions defined on surfaces of 3d objects. In *Proceedings of IEEE Data Compression Conference (DCC '97)*, pages 281–290, Snowbird, Utah, March 1997.

[LDW94] M. Lounsbery, Tony D. DeRose, and J. Warren. Multiresolution analysis for surfaces of arbitrary topological type. Technical Report Technical Report 93-10-05b, Department of Computer Science and Engineering, University of Washington, January 1994.

[LGC94] Z. Liu, S. J. Gortler, and M. F. Cohen. Hierarchical spacetime control. In *Computer Graphics Proceedings*, pages 35–42, July 1994.

[Loo87] C. T. Loop. Smooth subdivision surfaces based on triangles. Master's thesis, Department of Mathematics, University of Utah, August 1987.

[Lou94] M. Lounsbery. *Multiresolution analysis for surfaces of arbitrary topological type*. PhD thesis, University of Washington, October 1994. Updated version available as 93-10-05b, January, 1994.

[Luc92] B. J. Lucier. Wavelets and image compression. In T. Lyche and L. L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design II*. Academic Press, 1992.

[Mal89a] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.

[Mal89b] S.G. Mallat. Multifrequency channel decompositions of images and wavelet models. *IEEE Trans. Acoust. Speech Signal Process.*, 37(12):2091–2110, 1989.

[Mal92] S. G. Mallat. characterization of signals from multiscale edges. *IEEE Trans. Anal. Mach. Intell.*, 14:710–732, 1992.

[Mey90] Y. Meyer. *Ondelettes et Operateurs, I: Ondelettes, II: Operateurs de Calderon-Zygmund, III: (with R. Coifman) Operateurs multilineaires.* Hermann, Paris, 1990. English translation of first volume, *Wavelets and Operators*, was published by Cambridge University Press, 1993.

[Mey94a] D. Meyers. Multiresolution tiling. In *Proceedings of Graphics Interface*, pages 25–32, May 1994.

[Mey94b] D. Meyers. *Reconstruction of Surfaces from Planar Contours.* PhD thesis, University of Washington, Seattle, Washington, July 1994.

[MH92] S. Mallat and W. L. Hwang. Singularity detection and processing with wavelets. *IEEE Transactions on Information Theory*, 38(2):617–643, March 1992.

[MSS92] D. Meyers, S. Skinner, and K. Sloan. Surfaces from contours. *ACM Transactions on Graphics*, 11(3):228–258, July 1992.

[Mur92] Shigeru Muraki. Approximation and rendering of volume data using wavelet trasforms. In *Proceedings of Visualization*, pages 21–28, Los Alamitos, CA, 1992. IEEE Computer Society Press.

[NN84] Newhard and M. Nicol. Selective scene management in flight simulator visual systems. In *Proceedings of the Sixth Interservice/Industry Training Equipment Conference*, pages 22–24, 1984.

[Pen92] A. P. Pentland. Fast solutions to physical equilibrium and interpolation problems. *Visual Computer*, 8(5-6):303–314, June 1992.

[RB93] J. Rossignac and P. Borrel. Multi-resolution 3d approximation for rendering complex scenes. In B. Falcidieno and T. L. Kunii, editors, *Modeling in Computer Graphics: Methods and Applications.* Springer-Verlag, 1993.

[SAG94] B. Smiths, J. Arvo, and D. Greenberg. A clustering algorithm for radiosity in complex environments. In *Computer Graphics Annual Conference Series*, pages 435–442, July 1994.

[SAWG91] F. X. Sillion, J. R. Arvo, S. H. Westin, and D. P. Greenberg. A global illumination solution for general reflectance distributions. *Computer Graphics*, 25(4):187–196, July 1991.

[SDS96] Eric J. Stollnitz, Tony D. DeRose, and David H. Sallesin. *Wavelets for Computer Graphics: Theory and Applications.* Morgan Kauffman Publishers, Inc., San Francisco, CA, 1996.

[SS95a] P. Schröder and W. Sweldens. Spherical wavelets: Texture processing. In P. Hanrahan and W. Purgathofer, editors, *Rendering Techniques '95.* Springer Verlag, Wien, New York, August 1995.

[SS95b] Peter Schröder and Wim Sweldens. Spherical wavelets: efficiently representing functions on the sphere. In *Proceedings of SIGGRAPH '95*, pages 161–172, New York, 1995. ACM.

[Swe] W. Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, (to appear). Available on WWW as http://cm.bell-labs.com/who/wim/papers/papers.html#lift2.

[Swe95] W. Sweldens. The lifting scheme: A new philosophy in biorthogonal wavelet constructions. In A. F. Laine and M. Unser, editors, *Wavelet Applications in Signal and Image Processing III*, pages 68–79. Proc. SPIE 2569, 1995.

[SZL92] W. J. Schroeder, J. A. Zarge, and W. E. Lorenson. Decimation of triangle meshes. *Computer Graphics*, 26(2):65–70, 1992.

[Tur91] G. Turk. Generating textures on arbitrary surfaces using reaction-diffusion. *Computer Graphics*, 25(4):289–298, 1991.

[Tur92] G. Turk. Re-tiling polygonal surfaces. *Computer Graphics*, 26(2):55–64, 1992.

[ZSS97] D. Zorin, P. Schröder, and W. Sweldens. Interactive multiresolution mesh editing. *Computer Graphics Proceedings (SIGGRAPH 96)*, 1997.