Physics-Based Shape Modeling And Shape Recovery Using Multiresolution Subdivision Surfaces

Chhandomay Mandal* Hong Qin† Baba C. Vemuri*
*Department of Computer and Information Science and Engineering
University of Florida
†Department of Computer Science
State University of New York at Stony Brook

Abstract

The popularity of physics-based modeling is increasing during the last few years in computer graphics, animation, modeling and scientific visualization areas, as it can overcome some of the shortcomings of the traditional geometric modeling techniques. In this paper, we propose a powerful physics-based deformable model based on the multiresolution subdivision surfaces. The proposed model is much more general than the previously published subdivision surface-based deformable models. It allows the user to recover shapes from a set of unorganized points in 3D, and then directly edit the recovered smooth shape hierarchically at any desired level. Alternatively, the user can use the model for shape modeling or shape recovery purposes as well as a stand-alone system, thereby providing a very flexible system to the user. We present the mathematical formulation and implementation details of the deformable multiresolution subdivision surfaces along with experimental results to show the importance of the proposed model in shape modeling and shape recovery.

1 Introduction

Shape recovery and shape modeling are traditionally viewed as two distinct areas in computer graphics and geometric modeling literature. However, there are potential benefits if these two can be combined in a unified framework. For example, the modeler starts from scratch to build a specific model in a typical geometric modeling scenario. First, a rough shape is modeled and then it is fine-tuned by manipulating control vertex positions to obtain the desired effects. This turns out a cumbersome process in general. On the other hand, shape recovery using state-of-the-art methods yield large polygonal meshes which are very difficult to manipulate, especially for global changes in shape. Multiresolution analysis on these large meshes becomes a necessity to obtain a simplified mesh in order to change the global shape. However, most often the resulting meshes from a shape recovery application are not directly amenable to multiresolution analysis. Computationally expensive re-meshing techniques are needed to convert these meshes into a specific type of meshes on which multiresolution analysis can be performed. This specific type of mesh is typically known as mesh with "subdivision-connectivity," implying a topologically equivalent mesh with the same connectivity as of the given mesh can be obtained by recursive subdivision of a very simple known initial mesh.

In this paper, we present a deformable multiresolution subdivision surface model that conveniently combines shape recovery and shape modeling in a unified framework. In this framework, the modeler can scan in 3D points of a prototype model, recover the shape using the proposed dynamic subdivision surface model, and edit at any desired resolution using physics-based force tools. Thus, the modeler is relieved of the burden of both building an initial model and editing a cumbersome huge polygonal mesh. The shape recovery process starts with a smooth subdivision surface model which has a simple initial mesh. This model is deformed using synthesized forces from the given data points and is automatically refined using some pre-defined error in fit criteria. The initial mesh of the final recovered smooth shape has subdivision connectivity in-built, as it is obtained by subdivision refinement, and therefore no re-meshing is needed for multiresolution analysis. The dynamic framework for subdivision surface-based wavelets makes multiresolution editing using physics-based force tools easy to perform. These advantages of shape modeling and shape recovery in a unified framework is further illustrated in subsequent sections. However, prior to the technical details of our proposed scheme, we shall briefly review the previous work on physics-based modeling as well as on subdivision surfaces.

2 Background and Previous Work

Deformable surfaces typically use large number of degrees of freedom for representing the model geometry. A large variety of shapes can be modeled using this type of model, but handling a large number of degrees of freedom can be cumbersome. However, the large degrees of freedom of a deformable model are embedded in a physics-based framework to allow only a "physically meaningful" model behavior. Various types of energies are assigned to the model using the degrees of freedom, and the model "deforms" to find an equilibrium state with minimal energy. The motion equation is derived using Lagrangian dynamics [11], where various energies associated with model gives rise to internal and external forces. The equilibrium state of the model is a model position where the internal deformation force becomes equal to the externally applied force. These physics-based deformable models are useful for modeling where the modeler can deform a surface by applying synthesized forces, and for data fitting where external forces are synthesized from a given data set such that the model approximates the given data at equilibrium.

The free-form deformable models were first introduced to the computer graphics community in Terzopoulos et al. [28] and were further developed by Terzopoulos and Fleischer [27], Pentland and Williams [20], Metaxas and Terzopoulos [19]. Celniker and Gossard [4] developed a system for interactive free-form design based on the finite element optimization of energy functionals proposed in Terzopoulos and Fleischer [27]. Bloor and Wilson [1, 2], Celniker and Welch [5] and Welch and Witkin [30] proposed deformable B-spline curves and surfaces which can be designed by imposing the shape criteria via the minimization of the energy functionals subject to hard or soft geometric constraints through Lagrange multipliers or penalty methods. Terzopoulos and Qin [29] developed dynamic NURBS (D-NURBS) which are very sophisticated models suitable for representing a wide variety of free-form as well as standard analytic shapes. The D-NURBS have the advantage of interactive and direct manipulation of NURBS curves and surfaces, resulting in physically meaningful hence intuitively predictable motion and shape variation.

A severe limitation of most of the existing deformable models is that they are defined on a rectangular parametric domain. In a real world scenario, complex surfaces may not be globally parameterizable. Hence, it can be very difficult to model surfaces with arbitrary number of holes, i.e., surfaces of arbitrary genus, using these deformable models. In the geometric modeling paradigm, this problem is solved using two different techniques : (1) explicit patching using parametric surfaces and (2) subdivision surfaces.

The explicit patching technique involves partitioning the model surface into a collection of individual parametric surface patches. Adjacent surface patches are then explicitly stitched together using continuity constraints. This explicit stitching process is very complicated for modeling smooth surfaces of arbitrary topology due to the continuity constraints which need to be satisfied along the patch boundaries.

Subdivision surfaces are simple procedural models which offer an alternate representation for the smooth surfaces of arbitrary topology. A typical recursive subdivision scheme produces a visually pleasing smooth surface in the limit by repeated application of a fixed set of refinement rules on an user-defined initial control mesh. The initial control mesh is a simple polygonal mesh of the same topological type as that of the smooth surface to be modeled. At each step of subdivision, a *finer* polygonal mesh with more vertices and faces is constructed from the previous one via the refinement process, and the smooth surface is obtained in the limit. However, a few subdivision steps on the initial mesh generally suffice to approximate the smooth surface for all practical purposes. Various sets of rules lead to different subdivision schemes which mainly differ in the smoothness property of the resulting limit surface and/or the type of initial mesh (i.e. triangular, quadrilateral etc.) chosen.

A lot of research have been carried out on subdivision surfaces during the last two decades. The subdivision techniques of Doo and Sabin [8] and Catmull and Clark [3] generalize the idea of obtaining uniform biquadratic and bicubic B-spline patches respectively from a rectangular control mesh. Loop [15] presented a similar subdivision scheme based on the generalization of quartic triangular B-splines for triangular meshes. Hoppe et al. [13] extended his work to produce piecewise smooth surfaces with selected discontinuities. Halstead et al. [12] proposed an algorithm to construct a Catmull-Clark subdivision surface that interpolates the vertices of a mesh of arbitrary topology. Peters and Reif [21] proposed a simple subdivision scheme for smoothing polyhedra. Most recently, non-uniform Doo-Sabin and Catmull-Clark surfaces that generalize non-uniform tensor product B-spline surfaces to arbitrary topologies were introduced by Sederberg et al. [24]. All the schemes mentioned above generalize recursive subdivision schemes for generating limit surfaces with a known parameterization. Some other important subdivision techniques were developed by Dyn et al. [9], Zorin et al. [31], and Kobbelt et al. [14]. The mathematical properties of various types of subdivision surfaces are discussed in a number of papers. We refer the reader to Mandal [17] for a detailed list of references.

3 Motivation

Recursive subdivision surfaces are powerful for representing smooth geometric shapes of arbitrary topology. However, they constitute a purely geometric representation, and furthermore, conventional geometric modeling with subdivision surfaces may be difficult for representing highly complicated objects. For example, modelers are faced with the tedium of indirect shape modification and refinement through time-consuming operations on a large number of (most often irregular) control vertices when using typical subdivision surface-based modeling schemes. Despite the advent of advanced 3D graphics interaction tools, these indirect geometric operations remain non-intuitive and laborious in general. In addition, it may not be enough to obtain the most "fair" surface that interpolates a set of (ordered or unorganized) data points. A certain number of local features such as bulges or inflections may be strongly desired while requiring geometric objects to satisfy global smoothness constraints in geometric modeling and computer graphics applications.

In contrast, physics-based modeling provides a superior approach to shape modeling that can overcome most of these limitations associated with traditional geometric modeling approaches. Free-form deformable models governed by the laws of continuum mechanics are of particular interest in this context. These dynamic models respond to externally applied forces in a very intuitive manner. The dynamic formulation marries the model geometry with time, mass, damping and constraints via a force balance equation. Dynamic models produce smooth, natural motions which are easy to control. In addition, they facilitate interaction – especially direct manipulation of complex geometries. Furthermore, the equilibrium state of the model is characterized by a minimum of the deformation energy of the model subject to the imposed constraints. The deformation energy functionals can be formulated to satisfy local and global modeling criteria, and geometric constraints relevant to shape design can also be imposed.

It is apparent from the above discussion that a powerful model can be developed if the advantages of directly and intuitively manipulating a smooth surface using physics-based force tools and modeling arbitrary topology using subdivision surfaces can be combined together. Recently, some research has been carried out in this direction. DeRose et al. [7] used a physics-based subdivision surface model in their wonderful short animation film "Geri's Game." They assigned material properties to control meshes at various subdivision levels in order to simulate cloth dynamics using subdivision surfaces. Note that, they assign physical properties on the control meshes at various levels of subdivision and not on the limit surface itself, and modelers had to put a considerable amount of time to figure out correct material property values at different subdivision levels. An alternate approach is proposed in Mandal et al. [18, 22], where material properties are assigned on the smooth limit surface obtained via subdivision process, and they reported a compact representation of a complex smooth shape. However, they have to increase the degrees of freedom (control vertices) in order to capture the details present in the shape being modeled. This approach is named as "multilevel approach" by them. In this paper, we use a similar technique for locally parameterizing the smooth limit surface as in Mandal et al. [18, 22], but the proposed technique, which we call "multiresolution approach" is significantly different and much more powerful. We need to compare these two approaches in detail in order to show the significance of the approach proposed here, and we devote the next section for this purpose.

4 Multilevel Dynamics vs. Multiresolution Dynamics

Mandal et al. [18, 22] provides a dynamic framework for subdivision surfaces where the users can directly manipulate the smooth limit surface generated via subdivision process by applying synthesized forces. In any subdivision scheme, an initial (control) mesh is refined using specific subdivision rules to obtain a smooth surface in the limit. To develop the dynamic framework, this smooth limit surface is parameterized over the domain defined by the ini-

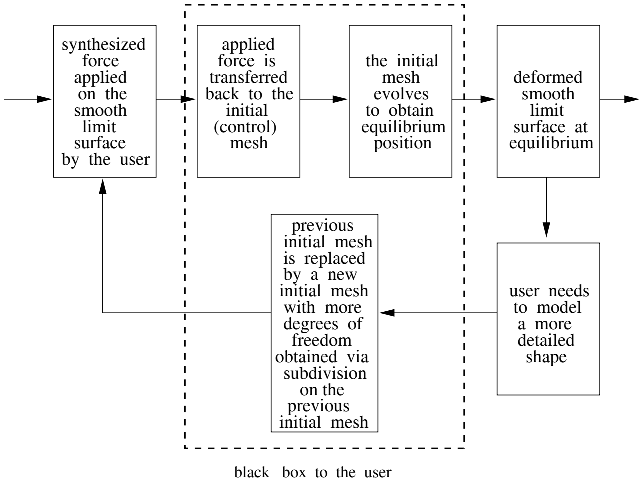


Figure 1: Schematic block diagram of the multilevel dynamics approach.

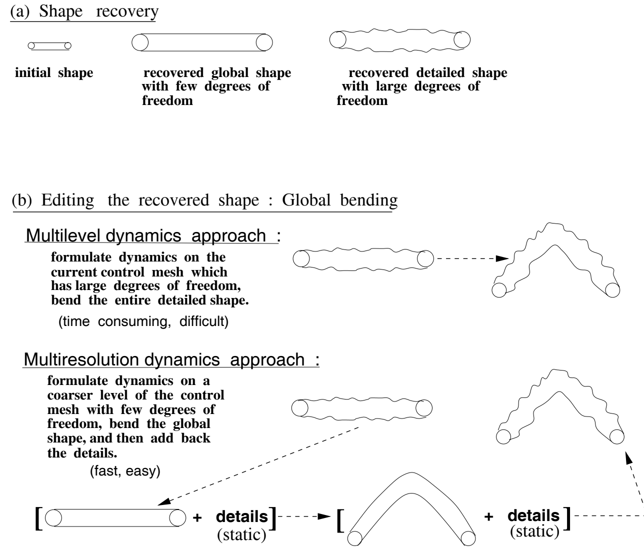


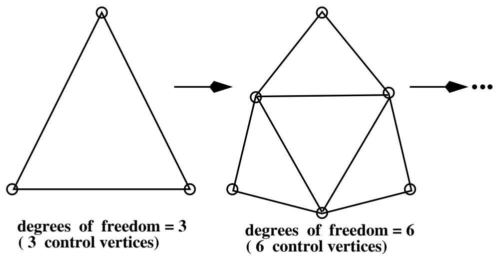
Figure 2: Multilevel dynamics vs. multiresolution dynamics.

tial mesh and hence, the limit surface can be expressed as a function of the initial mesh. The users apply synthesized forces on the smooth limit surface which is transferred back on the initial mesh via a transformation matrix. The initial mesh evolves over time in response to the applied forces, and consequently the smooth limit surface deforms to obtain an equilibrium position where all forces are balanced. Even though various types of forces can be applied on the limit surface to obtain a desired effect, the possible shapes that can be obtained via evolution is directly related to the number of control vertices present in the initial mesh.

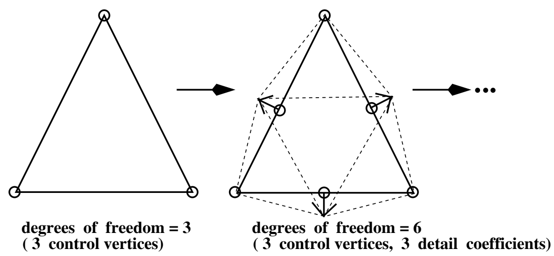
The concept of multilevel dynamics is introduced in Mandal et al. [18, 22] in order to have varying number of control vertices in the initial mesh, and a wide range of shapes with arbitrary topology can be modeled within their dynamic framework. The concept of multilevel dynamics is illustrated with block diagrams in Fig.1. This multilevel dynamics can be considered as a subdivision surface-based coarse-to-fine modeling framework and can be summarized as follows. The user starts with a smooth limit surface which has a very simple initial mesh with few control vertices. This smooth surface is deformed by applying synthesized forces so that the limit surface at equilibrium would look like the final desired shape but without the details in it. Then, the number of control vertices in the initial mesh is increased by replacing the initial mesh at equilibrium by another one obtained via one subdivision step on the old initial mesh, and the old initial mesh is discarded. Both of these old and new initial mesh represent the same limit surface, but the latter has more control vertices. Now, this new initial mesh, and consequently the smooth limit surface, can be deformed to obtain a shape at equilibrium which has more details. This process is repeated till the deformed smooth surface has all the desired details.

The limitation of the multilevel dynamics approach is that once a finer resolution is chosen, the modeler can not opt for a lower resolution as the dynamics is defined on the current initial mesh. There is no way of having the dynamics defined on a simpler initial mesh, at the same time preserving the already obtained details in the model within the multilevel dynamics approach. This is potentially a severe restriction. For example, let us imagine the scenario depicted in Fig.2. The modeler has recovered a long straight pipe-like shape with lots of detail by synthesized force application on a smooth limit surface. This smooth limit surface initially had a simple initial mesh with few vertices, but the current initial mesh after many model subdivision steps has large number of control vertices. Now, if the modeler wants to bend this straight pipe-like shape, the synthesized forces applied on the smooth limit surface need to deform an initial mesh with a large number of control vertices. It would have been much easier and faster if a lower resolution version of the current initial mesh could be deformed with the details being added back when the shape has bent to the desired extent in response to the applied synthesized forces. Also, this dynamic framework can integrate shape recovery and shape modeling in a unified framework where the recovered shape can easily be edited as well. However, if the modeler wants a global change in the recovered shape, a lower resolution control mesh (than the one obtained via shape recovery process) may be required, and the details need to be preserved at the same time.

A multiresolution representation of the initial mesh is proposed in this paper to solve the above-mentioned problem. At equilibrium, an initial mesh is subdivided to obtain an initial mesh with more control vertices (degrees of freedom). Instead of developing the dynamics on this newly obtained initial mesh as in the multilevel dynamics approach, the dynamics is developed on a representation which views the newly obtained initial mesh as the previous initial mesh plus detail (wavelet) coefficients needed to get the current initial mesh (see Fig.3). This is essentially developing the dynamics on a multiresolution representation of the evolving initial mesh which has undergone model subdivision step(s) to increase the degrees of freedom. It may be noted that both the multilevel and the multiresolution approach have same degrees of freedom to represent the initial mesh obtained after a model subdivision step, but the degrees of freedom for the multilevel dynamics approach are the control vertex positions in the newly obtained mesh whereas, the degrees of freedom for the current approach are the control vertex positions in the previous initial mesh and the detailed coefficients necessary to obtain the current initial mesh from the previous one. The multiresolution approach has the benefit that the modeler can opt for a lower resolution initial mesh later, on which the dynamics can be defined, at the same time preserving the detailed coefficients of the initial mesh on which the dynamics is currently defined. This concept will be detailed further in later sections, but first an overview of the multiresolution analysis is provided so that the reader can easily follow the rest of the material presented in the paper.



(a) Multilevel approach



(b) Multiresolution approach

Figure 3: Representation of the degrees of freedom in multilevel dynamics and multiresolution dynamics approach.

5 Overview of Multiresolution Analysis and Wavelets

The basic idea of multiresolution analysis is to decompose a complicated function into a lower resolution version along with a set of detailed coefficients (also known as wavelet coefficients) necessary to recover the original function. Multiresolution analysis and wavelets have widespread applications in diverse areas like image and signal processing, physics, numerical analysis, computer vision and computer graphics. An excellent collection of references on applications of multiresolution analysis in computer graphics can be found in Stollnitz et al. [25]. The idea of multiresolution analysis is explained next with a simple example.

Let $\mathbf{p}^j$ be a vector containing functional values of some arbitrary function at n discrete points. A lower resolution representation of the function using m $(m < n)$ functional values can be obtained from $\mathbf{p}^j$ by doing certain weighted averages of the n functional values stored in $\mathbf{p}^j$. This is essentially a low pass filtering followed by down-sampling, and the entire operation can be expressed as

$$\mathbf{p}^{j-1} = \mathbf{A}^j \mathbf{p}^j, \tag{1}$$

where $\mathbf{p}^{j-1}$ is the m-valued vector representing a lower resolution version and the matrix $\mathbf{A}^j$ is of size (m, n) and implements the low pass filter along with the down-sampler.

The lower resolution $\mathbf{p}^{j-1}$ has fewer functional values than $\mathbf{p}^j$, and hence some amount of detail present in $\mathbf{p}^j$ is lost due to the low pass filtering process. If the low pass filter $\mathbf{A}^j$ is chosen appropriately, it is possible to capture the lost detail by high pass filtering followed by a down-sampling of the original functional values stored in $\mathbf{p}^j$. This operation can be expressed as

$$\mathbf{q}^{j-1} = \mathbf{B}^j \mathbf{p}^j, \tag{2}$$

where $\mathbf{q}^{j-1}$ is a $(n-m)$-valued vector representing detailed coefficients necessary to recover $\mathbf{p}^j$ from $\mathbf{p}^{j-1}$. The matrix $\mathbf{B}^j$ is of

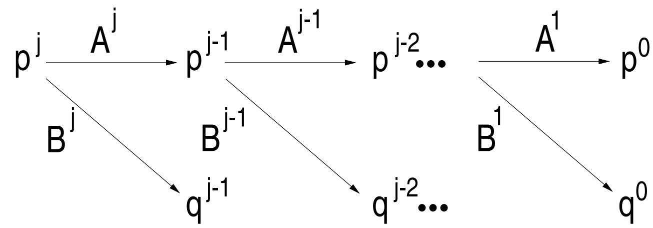


Figure 4: The filter bank.

size $(n-m, n)$ and implements the high pass filter along with the down-sampler. The high pass filter $\mathbf{B}^j$ is related to the low pass filter $\mathbf{A}^j$. These two filters, $\mathbf{A}^j$ and $\mathbf{B}^j$, are known as *analysis filters* and the process of splitting a high resolution vector $\mathbf{p}^j$ into a low resolution vector $\mathbf{p}^{j-1}$ and a vector $\mathbf{q}^{j-1}$ storing detail coefficients is known as *decomposition*.

The original vector $\mathbf{p}^j$ can be reconstructed from the vectors $\mathbf{p}^{j-1}$ and $\mathbf{q}^{j-1}$ if the synthesis filters are chosen correctly. This is called the *reconstruction* process, which can be expressed as

$$\mathbf{p}^j = \mathbf{P}^j \mathbf{p}^{j-1} + \mathbf{Q}^j \mathbf{q}^{j-1}, \tag{3}$$

where $\mathbf{P}^j$ is a refinement matrix of size (n, m) and $\mathbf{Q}^j$ is a perturbation matrix of size $(n, m-n)$. The refinement matrix $\mathbf{P}^j$ encodes the rules for obtaining a vector of larger size from a given vector and the perturbation matrix $\mathbf{Q}^j$ encodes how to obtain a perturbation of $(\mathbf{p}^j - \mathbf{P}^j\mathbf{p}^{j-1})$ from the given detailed coefficient vector $\mathbf{q}^{j-1}$. The matrices $\mathbf{P}^j$ and $\mathbf{Q}^j$ are collectively called *synthesis filters*, and are related to the analysis filters.

The process of decomposition can be continued recursively as shown in Fig.4 and is known as *filter bank*. The original vector is decomposed into a hierarchy of lower resolution parts $\mathbf{p}^{j-1}, \mathbf{p}^{j-2}, \ldots, \mathbf{p}^0$ and detail parts $\mathbf{q}^{j-1}, \mathbf{q}^{j-2}, \ldots, \mathbf{q}^0$. The original vector $\mathbf{p}^j$ can be recovered from the sequence $\mathbf{p}^0, \mathbf{q}^0, \mathbf{q}^1, \ldots, \mathbf{q}^{j-2}, \mathbf{q}^{j-1}$. This sequence has the same size as that of the original vector and is known as the *wavelet transform* of the original vector.

The analysis and synthesis filters are designed in such a fashion that the low resolution versions are good approximations of the original function in a least square sense. The decomposition and reconstruction processes should have time complexity $O(n)$, where n is the size of the vector being decomposed or reconstructed. A detail (wavelet) coefficient value should also be related to the error introduced in the approximation when that particular coefficient is set to zero.

Multiresolution analysis is a framework for developing these analysis and synthesis filters. It involves derivation of a sequence of nested linear spaces $V^0 \subset V^1 \subset V^2 \subset \ldots$ such that the resolution of the functions contained in some space V^j increases with increasing j. Also, there exists an orthogonal complement space W^{j-1} for each V^{j-1}, $j = 1, 2, \ldots$ such that $V^{j-1} \oplus W^{j-1} = V^j$, i.e., the linear space V^{j-1} and its orthogonal complement space W^{j-1} together span the linear space V^j. The inner product between any two functions at some level j, $j = 0, 1, 2, \ldots$ needs to be defined in order to derive the orthogonal complement space.

Let $\phi_1^j, \phi_2^j, \ldots, \phi_n^j$ be a set of basis functions spanning the linear space V^j. These basis functions are also called *scaling functions* for level j. Now, $V^{j-1} \subset V^j$ and hence the basis functions ϕ_1^{j-1}, $\phi_2^{j-1}, \ldots, \phi_m^{j-1}$, spanning the linear space V^{j-1}, can be expressed as a linear combination of the basis functions than span the linear space V^j. This refinability of scaling functions is used to construct the refinement matrix $\mathbf{P}^j$, which should satisfy the expression

$$\mathbf{\Phi}^{j-1} = \mathbf{\Phi}^j \mathbf{P}^j, \tag{4}$$

where $\mathbf{\Phi}^{j-1} = [\phi_1^{j-1}, \phi_2^{j-1}, \ldots, \phi_m^{j-1}]$, $\mathbf{\Phi}^j = [\phi_1^j, \phi_2^j, \ldots, \phi_n^j]$, and $\mathbf{P}^j$ is the (n, m) refinement matrix introduced earlier (Eqn.3). Similarly, let ψ_1^{j-1}, ψ_2^{j-1}, ..., ψ_{n-m}^{j-1} be a set of basis functions that span the linear space W^{j-1}. These basis functions, also known as *wavelets* at level $j-1$, together with the basis functions of linear space V^{j-1} form a basis for the linear space V^j. Next, the perturbation matrix $\mathbf{Q}^j$ can be constructed such that it satisfies the expression

$$\mathbf{\Psi}^{j-1} = \mathbf{\Phi}^j \mathbf{Q}^j, \tag{5}$$

where $\mathbf{\Psi}^{j-1} = [\psi_1^{j-1}, \psi_2^{j-1}, \ldots, \psi_{n-m}^{j-1}]$, and $\mathbf{Q}^j$ is the $(n, n-m)$ perturbation matrix introduced in Eqn.3. A more compact expression may be obtained by combining Eqn.4 and 5, which can be written as

$$[\mathbf{\Phi}^{j-1} \mid \mathbf{\Psi}^{j-1}] \quad = \quad \mathbf{\Phi}^j \quad [\mathbf{P}^j \mid \mathbf{Q}^j]. \tag{6}$$

The analysis filters $\mathbf{A}^j$ and $\mathbf{B}^j$ should satisfy the inverse relation

$$[\mathbf{\Phi}^{j-1} \mid \mathbf{\Psi}^{j-1}] \quad \left[\frac{\mathbf{A}^j}{\mathbf{B}^j}\right] \quad = \quad \mathbf{\Phi}^j. \tag{7}$$

Both the matrices $[\mathbf{P}^j \mid \mathbf{Q}^j]$ and $\left[\frac{\mathbf{A}^j}{\mathbf{B}^j}\right]$ are of size (n, n) and satisfy the relation

$$[\mathbf{P}^j \mid \mathbf{Q}^j] \quad = \quad \left[\frac{\mathbf{A}^j}{\mathbf{B}^j}\right]^{-1}. \tag{8}$$

5.1 Multiresolution Analysis for Surfaces of Arbitrary Topology

Classically, the multiresolution analysis was developed on infinite domains such as real line $\Re$ and plane $\Re^2$ [6]. However, many applications need multiresolution analysis on bounded intervals, and techniques for imposing boundary constrains have also been derived [10, 23]. The idea of generalizing multiresolution analysis on arbitrary manifolds was first introduced by Lounsbery et al. [16], and was further improved by Stollnitz et al. [25] using the concepts of *lifting scheme* [26]. In this paper, the multiresolution schemes derived by Stollnitz et al. [25] for surfaces of arbitrary topology is used to develop the multiresolution dynamic framework for subdivision surfaces. This involves (1) developing nested spaces using subdivision schemes, (2) selecting an inner product on arbitrary manifold and (3) constructing biorthogonal wavelets. We can not go into the details of how to construct the analysis and synthesis filters for multiresolution analysis on meshes of arbitrary topology due to space constraints. We used the "k-disc wavelets" developed in Stollnitz et al. [25], and the reader can find the expressions for the corresponding analysis and synthesis filters there.

6 Multiresolution Representation

The multiresolution analysis on arbitrary manifold was developed to obtain good low resolution approximations of complicated meshes. This is a top-down technique where one starts with a complicated arbitrary topology mesh and goes on obtaining lower resolution meshes using analysis filters. The complicated mesh on which multiresolution analysis is applied can be reconstructed using the synthesis filters. In this paper, the multiresolution technique is used in a novel bottom-up approach where an evolving control mesh of a dynamic subdivision surface is subdivided at equilibrium to obtain more degrees of freedom, and then, the resulting control mesh is treated as the previous control mesh and a collection of detail (wavelet) coefficients. Of course, the detail coefficients are zero when an control mesh is replaced by another control mesh obtained by a pure subdivision step on the previous mesh, but becomes non-zero as the new control mesh (represented as previous control mesh and wavelet coefficients) evolves over time in response with the synthesized force application. This idea is further illustrated in the rest of this section.

Let $\mathbf{s}$ be a smooth limit surface obtained via infinite number of subdivision steps on an initial mesh S^0, whose vertex positions are collected in $\mathbf{p}^0$. The limit surface can be written as

$$\mathbf{s} = \mathbf{J}^0 \mathbf{p}^0, \tag{9}$$

where $\mathbf{J}^0$ is a collection of basis functions. These basis functions are obtained as a limiting process of subdivision, and may or may not have analytic expressions. Mandal et al. [18] discusses at length on how to express the smooth limit surface resulted via subdivision using any type of subdivision rules in this form. Stollnitz et al. [25] discusses on how to perform mathematical operations correctly in case the basis functions obtained through subdivision do not have analytic representation. Now, if we obtain the control mesh S^1 by subdividing the control mesh S^0 once, then the same limit surface can be expressed as

$$\mathbf{s} = \mathbf{J}^1 \mathbf{p}^1, \tag{10}$$

where $\mathbf{p}^1$ is the collection of vertex positions of the mesh S^1 and $\mathbf{J}^1$ is the collection of corresponding basis functions. Since $\mathbf{p}^1 = \mathbf{P}^1 \mathbf{p}^0$, the limit surface $\mathbf{s}$ can also be written as

$$\mathbf{s} = \mathbf{J}^1 \mathbf{P}^1 \mathbf{p}^0. \tag{11}$$

The difference between Eqn.9 and Eqn.10 is that the limit surface $\mathbf{s}$ has more degrees of freedom (control vertices) for representation in the latter in comparison with the former. However, in Eqn.11, even though the limit surface is expressed using the same basis functions as that of S^1, the degrees of freedom remains the same as that of S^0. Eqn.11 can be modified in the following manner so that it uses same basis functions and same number of degrees of freedom as that of S^1.

$$\mathbf{s} = \mathbf{J}^1 \quad [\mathbf{P}^1 \, \mathbf{Q}^1] \quad \begin{bmatrix} \mathbf{p}^0 \\ \mathbf{q}^0 \end{bmatrix}, \tag{12}$$

where $\mathbf{q}^0$ is the collection of wavelet coefficients whose values are zero, $\mathbf{P}^1$ and $\mathbf{Q}^1$ are the synthesis matrices $\mathbf{P}^1_{lift}$ and $\mathbf{Q}^1_{lift}$ respectively for the "k-disk wavelets" [25]. The above formulation yields a multiresolution representation of the control mesh S^1 obtained via one subdivision step on the previous control mesh S^0. The mesh S^1 is represented as a mesh at level 0 along with the detail (wavelet) coefficients at level 0. As mentioned earlier, these coefficients would become non-zero when the smooth limit surface deforms over time due to synthesized force application. Similarly after one more subdivision step, the new control mesh S^2 can be viewed as control vertex positions at level 0, wavelet coefficients at level 0 and wavelet coefficients at level 1, and can be written as

$$\begin{aligned} \mathbf{s} &= \mathbf{J}^2 \ \mathbf{p}^2 \\ &= \mathbf{J}^2 \ [\mathbf{P}^2 \, \mathbf{Q}^2] \ \begin{bmatrix} \mathbf{p}^1 \\ \mathbf{q}^1 \end{bmatrix} \\ &= \mathbf{J}^2 \ [\mathbf{P}^2 \, \mathbf{Q}^2] \ \begin{bmatrix} \mathbf{P}^1 & \mathbf{Q}^1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \ \begin{bmatrix} \mathbf{p}^0 \\ \mathbf{q}^0 \\ \mathbf{q}^1 \end{bmatrix}, \end{aligned} \tag{13}$$

where $\mathbf{J}^2$ is the collection of basis functions at level 2, $\mathbf{p}^2$, $\mathbf{p}^1$ and $\mathbf{p}^0$ are the control vertex positions at level 2, 1 and 0 respectively, $\mathbf{q}^1$ and $\mathbf{q}^0$ are the wavelet coefficients at level 1 and 0 respectively, $\mathbf{I}$ is the identity matrix and $\mathbf{0}$ is the zero matrix. This idea can be generalized for a control mesh S^j obtained after j-th subdivision and the corresponding expression can be written as

$$\mathbf{s} \quad = \quad \mathbf{J}^j \quad \mathbf{A}^j \quad \mathbf{p}_r^j, \tag{14}$$

where $(\mathbf{p}_r^j)^T = [\mathbf{p}^0 \mathbf{q}^0 \ldots \mathbf{q}^{j-2} \mathbf{q}^{j-1}]$, and

$$\mathbf{A}^j = [\mathbf{P}^j \, \mathbf{Q}^j] \begin{bmatrix} \mathbf{P}^{j-1} & \mathbf{Q}^{j-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}^{j-2} & \mathbf{Q}^{j-2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{I} \end{bmatrix} \cdots \begin{bmatrix} \mathbf{P}^0 & \mathbf{Q}^0 & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \ldots & \mathbf{I} \end{bmatrix}.$$

The multiresolution dynamics is developed using this formulation in the next section.

7 Dynamics

The smooth limit surface $\mathbf{s}$ can be made dynamic if the control vertex positions at level 0 and wavelet coefficients at level $1, 2, ..., (j-1)$ are functions of time. The velocity of the smooth surface, controlled by the mesh S^j obtained through j steps of model subdivision, is given by

$$\dot{\mathbf{s}}(\mathbf{x}, \mathbf{p}_r^j) = \mathbf{J}^j(\mathbf{x}) \, \mathbf{A}^j \, \dot{\mathbf{p}}_r^j, \tag{15}$$

where the overstruck dot denotes the time derivative and $\mathbf{x} \in S^j$. The Lagrangian motion equation is given by

$$\mathbf{M}^j \ddot{\mathbf{p}}_r^j + \mathbf{D}^j \dot{\mathbf{p}}_r^j + \mathbf{K}^j \mathbf{p}_r^p = \mathbf{f}_p^j, \tag{16}$$

where $\mathbf{M}^j$, $\mathbf{D}^j$ and $\mathbf{K}^j$ are the j-th level mass, damping and stiffness matrices respectively and $\mathbf{f}_p^j$ is the generalized force vector at level j. If $\mu(\mathbf{x})$ is the mass density of the subdivision surface model, then the mass matrix at j-th level is given by

$$\mathbf{M}^j = \int_{\mathbf{x} \in S^j} \mu(\mathbf{x}) (\mathbf{A}^j)^T (\mathbf{J}^j(\mathbf{x}))^T \mathbf{J}^j(\mathbf{x}) \mathbf{A}^j d\mathbf{x}.$$

The damping matrix $\mathbf{D}^j$ can be derived in a similar fashion, while the stiffness matrix $\mathbf{K}^j$ can be obtained following the techniques described in Mandal et al. [18]. The generalized force vector at level j can be written as

$$\mathbf{f}_p^j = \int_{\mathbf{x} \in S^j} (\mathbf{A}^j)^T (\mathbf{J}^j(\mathbf{x}))^T \mathbf{f}(\mathbf{x}, t) d\mathbf{x}.$$

8 Implementation Details

Multiresolution representation of the evolving control mesh achieves coarse-to-fine as well as fine-to-coarse manipulation of the smooth limit surface. For example, the user starts with a deformable smooth surface with a simple control mesh S^0 and directly manipulates the limit surface by applying synthesized forces. When an equilibrium is obtained, the user can increase the degrees of freedom by switching to a control mesh S^1 which is theoretically obtained by applying one subdivision step on S^0, but represented as a collection of vertex positions in S^0 along with wavelet coefficients at level 0 needed to reconstruct S^1. The wavelet coefficients are zero at the point of switching, but becomes non-zero as the control mesh S^1, represented as mentioned above, evolves over time. The user can further increase the degrees of freedom to obtain more localized effect by introducing control mesh S^2 which is represented as control vertex positions of mesh S^0, along with wavelet coefficients at level 0 and 1. Thus the user can have coarse-to-fine manipulation of the smooth limit surface. This facility is also present in the multilevel dynamics approach [18] as mentioned earlier. However, the multilevel dynamics approach does not support fine-to-coarse manipulation – it fails if the user wants to apply synthesized forces on a coarser control mesh at level j after moving up to level $j + k$. This can be achieved in multiresolution approach by simply using the dynamic equation of motion at level j, and applying synthesized forces on a smooth limit surface which is obtained by multiresolution synthesis using "time varying" control vertex positions at level 0 and wavelet coefficients at level $0, 1, \ldots, (j-1)$ and "static" wavelet coefficients at level $j, (j+1), \ldots, (j+k-1)$. If the user switches back to level $(j + k)$, then the control vertex positions at level 0 as well as all the wavelet coefficients at level $0, 1, \ldots, (j-1), j, (j+1), \ldots, (j+k-1)$ become function of time, and the system switches back to the motion equation of level $(j + k)$.

The actual implementation differs slightly from the formulation to achieve efficiency in coarse-to-fine and fine-to-coarse manipulation. In the implemented system, the user starts out with a smooth limit surface which has a simple control mesh with very few degrees of freedom. The model grows and when the equilibrium is achieved, one step of subdivision yields larger degrees of freedom. This larger degrees of freedom are the control vertex positions in this finer resolution and not control vertex positions in lower resolution along with wavelet coefficients in that lower resolution. The model can grow further, and if more subdivision step is needed to increase the degrees of freedom, it is handled in a similar fashion. If at any certain point of time the user needs to go back to a coarser level control mesh, wavelet decomposition is done on the higher resolution base mesh. This decomposition yields vertex positions in the coarser level mesh (which becomes the new base mesh) and non-zero wavelet coefficients. The lower resolution mesh evolves with time due to the force applied on the limit surface. It may be noted that the limit surface in this scenario is obtained by doing wavelet reconstruction with non-zero wavelet coefficients at some levels. The user can choose further coarser level control mesh by repeating the same process.

In the implemented system, the user can achieve global deformation on a detailed mesh by going down couple of levels using wavelet decomposition, retaining the wavelet coefficients, and applying force on the limit surface (obtained through wavelet reconstruction process) and letting the coarser level mesh evolve. Similarly, to achieve local deformation on a coarser mesh, the user can move up couple of steps by doing subdivision (and adding non-zero wavelet coefficients at higher levels, if present) and applying synthesized forces in the region of interest.

The formulation using wavelet representation has mathematical niceties, but inefficient for implementation purposes. The synthesized force is applied on the limit surface. To evaluate the limit surface, wavelet reconstruction needs to be done at each time step starting from the original control mesh upto a specified level if an explicit wavelet representation of the evolving control mesh is maintained as mentioned in the formulation. On the other hand, if the evolving mesh is control vertex positions at that resolution, then the wavelet reconstruction starts from that resolution for evaluating the limit surface. Wavelet decomposition needs to be done while going down from finer to coarser resolution, but this happens only once in a while, and not at each time step. Therefore, the implemented version is much more efficient. It may be noted that the motion equation at any level is implemented using finite element techniques discussed in Mandal et al. [18] for various subdivision schemes.

9 Experimental Results

The proposed multiresolution dynamic subdivision surface model can be used in shape recovery and shape modeling applications independently in a similar way as in Mandal et al. [18]. However, the added benefit of the proposed model is that a recovered shape can be edited at any coarser or finer level (which is not permitted in the approach taken by Mandal et al. [18]). The proposed multiresolution dynamic subdivision techniques present hierarchical shape recovery and shape modeling within a common framework, where

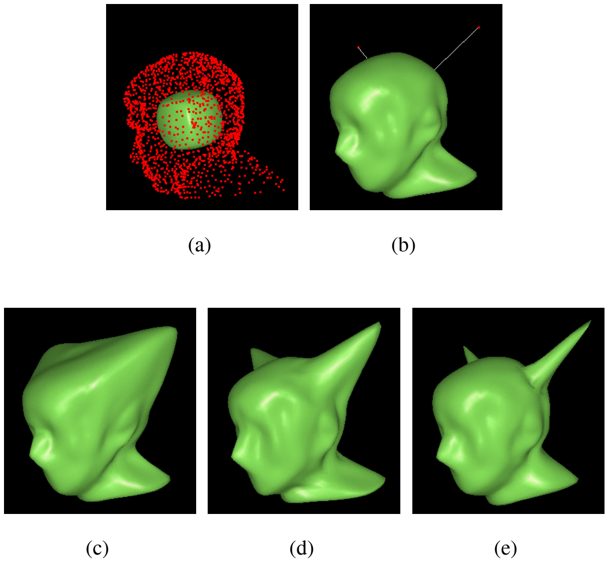


Figure 5: (a) The initialized model along with the head data set; (b) the fitted model with two subdivisions on the initial mesh, along with attached springs for editing. The model after editing (c) at lower resolution, (d) at the same resolution of the fitted model, and (e) at higher resolution.

the modeler can scan in 3D data points of a prototype model, recover the underlying shape from the point set, and then edit the recovered shape. The multiresolution representation of the evolving control mesh enables the modeler to edit the smooth limit surface at any desired level. Within the proposed multiresolution dynamic framework, the modeler does not need to build a model from scratch (unlike other shape modeling techniques), and there is no need of using computationally intensive remeshing techniques for multiresolution representation (unlike other shape recovery techniques).

The idea is illustrated with a set of examples. In the first example, we use a data set of a head comprising 1779 data points in 3D. We use butterfly subdivision technique [9] to build the multiresolution dynamic subdivision surface model. The initialization is a smooth surface resulted via butterfly subdivision technique on a control (initial) mesh of 24 triangular faces with 14 vertices (Fig.5(a)). The initialized smooth surface model is deformed by applying synthesized spring-based forces from the data points in 3D till the evolving model reaches an equilibrium. The system checks for the error in fit at equilibrium. If the error is more than the prescribed threshold, the degrees of freedom (control vertices) of the evolving model is increased using the butterfly subdivision technique based multiresolution dynamics approach. This process is repeated recursively until the prescribed error in fit is achieved. To obtain a fit with 3% error, the degrees of freedom of the evolving model had to be increased twice. The final fitted model (Fig.5(b)) has a control mesh with 384 triangular faces and 194 vertices. Once the smooth shape is recovered, it can be edited at any desired level. We show the effects of editing the recovered smooth shape at various levels using spring forces from two points in 3D for the head example (Fig.5(c)-(e)). The effect was more global in the lowest level edited, and the effect became increasingly local as the level of editing was increased. Note that, the user perceives a smooth surface at any editing resolution, but the effect of force application on the smooth surface varies with change in editing levels.

In the next example, we use an anvil data set comprising 3000

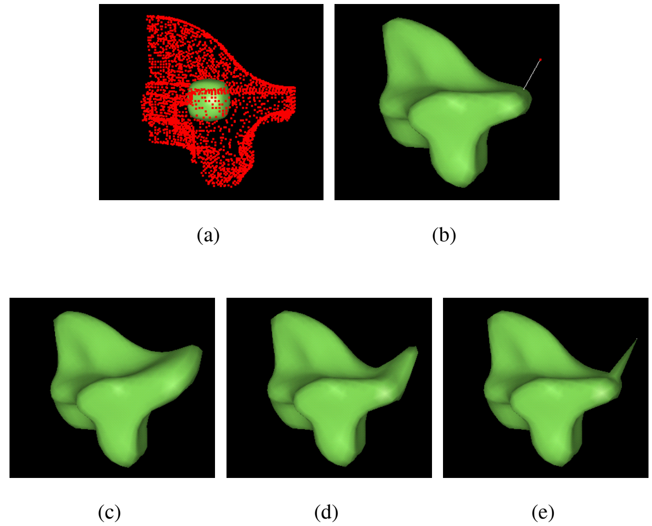


Figure 6: (a) The initialized model along with the anvil data set; (b) the fitted model with two subdivisions on the initial mesh, along with an attached spring for editing. The model after editing (c) at lower resolution, (d) at the same resolution of the fitted model, and (e) at higher resolution.

data points in 3D. As in the previous example, we initialize a simple deformable smooth subdivision surface model which has a control (initial) mesh of 24 triangular faces with 14 vertices. The initialized model along with the data set is shown in Fig.6(a). Springs are attached to the initialized model from the data points in 3D, and the initialized model is deformed due to the spring-based applied force. The degrees of freedom was increased twice through butterfly subdivision process in order to achieve a 3% error in fit, and the fitted model, which has a control mesh of 384 triangular faces and 194 vertices, is shown in Fig.6(b). Next, we edit the recovered smooth shape by applying spring forces from a point in 3D, and the editing effects at different resolutions are shown in Fig.6(c)-(e). As in the previous example, the editing effect shifted from global to localized effect as the editing resolution is increased.

10 Conclusions

We have presented a multiresolution dynamic framework for subdivision surfaces, which effectively combines the shape modeling and shape recovery problem under a unified framework. The proposed model reaps the benefits of easily controlled, intuitive and direct manipulation of smooth surfaces using physics-based force tools, as well as those of subdivision surfaces for representing surfaces of arbitrary topology. The multiresolution representation of the evolving control mesh aids coarse-to-fine as well as fine-to-coarse editing needs, and the model performs well in the combined shape recovery and shape modeling tasks. In future, we would like to test the model for more complicated real world shape recovery and shape modeling examples.

References

[1] M.I.J. Bloor and M.J. Wilson, “Representing PDE surfaces in terms of B-splines,” *Computer Aided Design*, vol. 22, no. 6, pp. 324 – 331, 1990.

[2] M.I.J. Bloor and M.J. Wilson, "Using partial differential equations to generate free-form surfaces," *Computer Aided Design*, vol. 22, no. 4, pp. 202 – 212, 1990.

[3] E. Catmull and J. Clark, "Recursively generated B-spline surfaces on arbitrary topological meshes," *Computer Aided Design*, vol. 10, no. 6, pp. 350 – 355, 1978.

[4] G. Celniker and D. Gossard, "Deformable curve and surface finite elements for free-form shape design," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 257 – 266, July, 1991.

[5] G. Celniker and W. Welch, "Linear constraints for deformable B-spline surfaces," in *Proceedings of the Symposium on Interactive 3D Graphics*, pp. 165 – 170. ACM, New York, 1992.

[6] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. 41, no. 7, pp. 909 – 996, 1988.

[7] T. DeRose, M. Kass and T. Truong, "Subdivision surfaces in character animation," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 85 – 94, July, 1998.

[8] D. Doo, "A subdivision algorithm for smoothing down irregularly shaped polyhedrons," in *Proceedings on Interactive Techniques in Computer Aided Design*, pp. 157 – 165, 1978.

[9] N. Dyn, D. Levin and J.A. Gregory, "A butterfly subdivision scheme for surface interpolation with tension control," *ACM Transactions on Graphics*, vol. 9, no. 2, pp. 160 – 169, April, 1990.

[10] A. Finkelstein and D.H. Salesin, "Multiresolution curves," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 261 – 268, July, 1994.

[11] B.R. Gossick, *Hamilton's Principle and Physical Systems*, Academic Press, New York, 1967.

[12] M. Halstead, M. Kass and T. DeRose, "Efficient, fair interpolation using Catmull-Clark surfaces," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 35 – 44, August, 1993.

[13] H. Hoppe, T. DeRose, T. Duchamp, M. Halstead, H. Jin, J. McDonald, J. Schweitzer and W. Stuetzle, "Piecewise smooth surface reconstruction," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 295 – 302, July, 1994.

[14] L. Kobbelt, "A variational approach to subdivision," *Computer-Aided Geometric Design*, vol. 13, pp. 743 – 761, 1996.

[15] C. Loop, *Smooth subdivision surfaces based on triangles*, M.S. thesis, University of Utah, Department of Mathematics, 1987.

[16] J.M. Lounsbery, T. DeRose and J. Warren, "Multiresolution analysis for surfaces of arbitrary topological type," *ACM Transactions on Graphics*, vol. 16, no. 1, pp. 34 – 73, January, 1997.

[17] C. Mandal, *A dynamic framework for subdivision surfaces*, Ph.D. thesis, University of Florida, Gainesville, 1998, UF-CISE-TR-98-022, Avaialable via http://www.cise.ufl.edu/tech-reports/tech-reports/tr98-abstracts.shtml.

[18] C. Mandal, H. Qin and B.C. Vemuri, "A novel FEM-based dynamic framework for subdivision surfaces," University of Florida Tech. Report UF-CISE-TR-98-021, Avaialable via http://www.cise.ufl.edu/tech-reports/tech-reports/tr98-abstracts.shtml.

[19] D. Metaxas and D. Terzopoulos, "Dynamic deformation of solid primitives with constraints," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 309 – 312, July, 1992.

[20] A. Pentland and J. Williams, "Good vibrations : Modal dynamics for graphics and animation," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 215 – 222, 1989.

[21] J. Peters and U. Reif, "The simplest subdivision scheme for smoothing polyhedra," *ACM Transactions on Graphics*, vol. 16, no. 4, pp. 420 – 431, October, 1997.

[22] H. Qin, C. Mandal and B.C. Vemuri, "Dynamic Catmull-Clark subdivision surfaces," *IEEE Transactions on Visualization and Computer Graphics*, vol. 4, no. 3, pp. 215 – 229, July - September, 1998.

[23] E. Quak and N. Weyrich, "Decomposition and reconstruction algorithms for spline wavelets on a bounded interval," CAT report 294, Center for Approximation Theory, Texas A&M University, April, 1993.

[24] T.W. Sederberg, J. Zheng, D. Sewell and M. Sabin, "Non-uniform recursive subdivision surfaces," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 387 – 394, July, 1998.

[25] E.J. Stollnitz, T.D. DeRose and D.H. Salesin, *Wavelets for computer graphics : theory and applications*, Morgan Kaufmann, San Francisco, 1996.

[26] W. Sweldens, "The lifting scheme : a custom-design construction of biorthogonal wavelets," Tech. Rep. 1994:7, Industrial Mathematics Initiative, Department of Mathematics, University of South Carolina, 1994.

[27] D. Terzopoulos and K. Fleischer, "Deformable models," *The Visual Computer*, vol. 4, no. 6, pp. 306 – 331, 1988.

[28] D. Terzopoulos, J. Platt, A. Barr and K. Fleischer, "Elastically deformable models," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 205 – 214, 1987.

[29] D. Terzopoulos and H. Qin, "Dynamic NURBS with geometric constraints for interactive sculpting," *ACM Transactions on Graphics*, vol. 13, no. 2, pp. 103 – 136, April, 1994.

[30] W. Welch and A. Witkin, "Variational surface modeling," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 157 – 166, July, 1992.

[31] D. Zorin, P. Schröder and W. Sweldens, "Interpolating subdivision for meshes with arbitrary topology," in *Computer Graphics Proceedings*, ACM SIGGRAPH, Annual Conference Series, pp. 189 – 192, August, 1996.