Preprint
UCRL-JC-146852

Multiresolution Distance Volumes for Progressive Surface Compression

D. A. Laney, M. Bertram, M. A. Duchaineau, N. L. Max

This article was submitted to *1st International Symposium on 3D Data Processing Visualization and Transmission, Padova, Italy, June 19-21, 2002*

January 14, 2002

This work was performed under the auspices of the United States Department of Energy by the University of California, Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

Multiresolution Distance Volumes for Progressive Surface Compression

Daniel Laney, Martin Bertram, Mark Duchaineau, and Nelson Max

***Abstract*—**

Surfaces generated by scientific simulation and range scanning can reach into the billions of polygons. Such surfaces must be aggressively compressed, but at the same time should provide for level of detail queries. Progressive compression techniques based on subdivision surfaces produce impressive results on range scanned models. However, these methods require the construction of a base mesh which parameterizes the surface to be compressed and encodes the topology of the surface. For complex surfaces with high genus and/or a large number of components, the computation of an appropriate base mesh is difficult and often infeasible.

We present a surface compression method that stores surfaces as wavelet-compressed signed-distance volumes. Our method avoids the costly base-mesh construction step and offers several improvements over previous attempts at compressing signed-distance functions, including an $\mathcal{O}(n)$ distance transform, a new zero set initialization method for triangle meshes, and a specialized thresholding algorithm. We demonstrate the potential of sampled distance volumes for surface compression and progressive reconstruction for complex high genus surfaces.

I. Introduction

THE rapid increase in computing power and advancements in surface acquisition techniques have enabled the creation of meshes of 200 million triangles and larger [1], [2], [3]. This has led to a dilemma in surface visualization: meshes of this size and complexity require both efficient compression techniques and a capacity for level-of-detail interrogation. Simulation codes running on the newest supercomputers will be able to generate meshes measured in billions of triangles. These surfaces will be complex and time-varying, multiplying the storage requirements and complicating the compression algorithms. Progressive compression algorithms enable both efficient compression and level of detail reconstruction. A progressive compression algorithm re-orders the bit stream in such a way that the most relevant information is near the front of the stream. Thus, with a small number of bits a usable approximation of a surface can be obtained for interaction and browsing. As a user finds areas of interest, more bits can be decompressed to provide the needed details. This paper presents a system for progressively compressing surfaces via a signed-distance representation.

Recently, subdivision surfaces have been shown to be effective for mesh compression as the connectivity information only needs to be stored for the base mesh. The work of Khodakovsky [4] and Bertram [5] show that wavelet-based techniques on subdivision surfaces result in competitive compression rates and allow for progressive decompression.

There exist two open problems with subdivision based approaches. First, a coarse base mesh is required before the subdivision can be applied. As surfaces become large and complex, with an ever greater number of components, the construction of the base mesh becomes problematic. Even if a base mesh is produced, a surface with hundreds or thousands of components requires topological modification in order to achieve usable progressive reconstructions. The hybrid mesh representation proposed by Guskov [6] may alleviate the problems of base mesh construction for these cases, but it requires user interaction in the construction process. The second open problem is the extension of subdivision based techniques to time-varying data. The topology of the surface is explicitly represented in the base mesh, causing difficulties when time is introduced to the representation. The work of Shamir [7] provides a starting point for multiresolution time varying surface representations. However, the suggested representation appears to be difficult to reconcile with multiresolution compression methods.

In this paper we advocate an alternative approach to surface compression which is based on a signed-distance volume representation [8], [9] . A signed-distance volume is a trivariate distance function encoding the minimum distance to a surface for each volume sample. The sign changes as the surface is crossed. Figure 1 depicts the data flow in our system. The resulting compressed surface is reconstructed by extracting the isosurface with zero distance.

The signed-distance representation does not directly specify the topology of the surface. This freedom from storing the topology increases the potential for using sim-

Daniel Laney and Nelson Max are with the Dept. of Applied Science at UC Davis and Lawrence Livermore National Laboratory
Mark Duchaineau is with Lawrence Livermore National Laboratory
Email:{laney1, duchaineau1, max2}@llnl.gov
Email: bertram@informatik.uni-kl.de

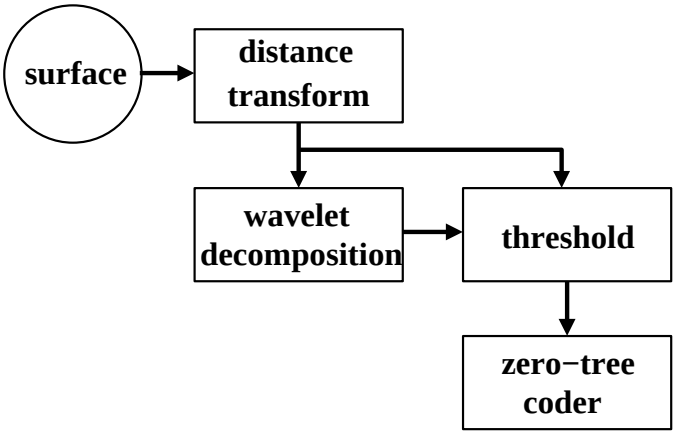

Fig. 1. The compression system comprises four modules. The input surface is transformed into a distance representation then decomposed into wavelet coefficients. These coefficients are thresholded using the distance information to retain the surface geometry. Finally, a zero-tree coder produces the progressive bit stream.

ple algorithms that will extend elegantly to surfaces that have high genus and are time varying. We achieve a multiresolution representation by applying a wavelet decomposition to the implicit function. Compressing multiresolution signed-distance functions has been studied in [10], [11]. Our contribution is a complete system that overcomes the problems of existing distance based surface compression methods. The main features of our method are as follows:

1. **Progressiveness:** We generate a progressive encoding of the distance function which can be partially reconstructed from the most relevant bits of the most relevant wavelet coefficients. Thus, approximations of the surface can be viewed without requiring a full decompression. In addition, the implicit nature of our representation facilitates topology modification to reduce the complexity of the approximate surfaces beyond what is attainable by subdivision surface methods.
2. **Scalability:** The method is not limited by the need to re-map a complex surface to a base mesh with subdivision connectivity. Our algorithm constructs regularly sampled distance volumes and performs a wavelet analysis. Thus, surfaces with large numbers of components and complex topology can be represented as long as appropriate sampling rates are used. Our thresholding method removes wavelet coefficients that do not contribute to the zero set resulting in a size related to the surface complexity.
3. **Simplicity:** The distance volume representation dispenses with a lot of the algorithmic complexity associated with base mesh construction and explicit topology tracking. All operations in our method are performed on regularly sampled volumes. The distance transform algorithm is based on the propagation technique of Breen [12] . The algorithm is simple to implement and can be coded efficiently. The wavelet transform is the common biorthogonal B-spline wavelet transform [13] .
4. **Autonomy:** The algorithm requires only a desired bit count in order to produce a compressed file. This is in contrast to subdivision methods which may require explicit base mesh vertex positioning for sharp features [4] , editing operations [6] , or multiple fitting parameters for obtaining the base mesh [14].

II. Related Work

Traditional approaches to surface compression have focused on polygonal mesh compression techniques. Early methods focused on the encoding of mesh connectivity information [15], with vertex information being compressed via quantization and prediction schemes. Surface simplification algorithms [16] were introduced to reduce the complexity of scanned surfaces so that editing operations and compression techniques could be applied. However, these approaches lacked the level-of-detail reconstruction required for interactive viewing and remote visualization applications. Multiresolution mesh representations enabling both compression and level-of-detail rendering [17], [18], [19], [14] have been developed to overcome the limitations of traditional mesh compression and simplification techniques [16]. Multiresolution methods allow selective refinement operations to adjust surface geometry according to a user-defined error criterion or a prescribed triangle budget. Progressive mesh representations such as the edge-contraction approaches of Hoppe [17] and Pajarola [20] encode the connectivity of the mesh.

Wavelet transforms have been used to obtain multiresolution representations of scalar volume data for rendering and compression [21], [22], [23], [24], [25] . Tao [26] described a system for progressively transmitting volume data encoded as wavelet coefficients. Time-varying volumes were treated with wavelet techniques in the work of Westermann [27] . Volume compression techniques [1], [28] based on wavelet transforms have been used the visualization of large data sets. The present work uses standard wavelet transforms on volumetric data but is not concerned with representing the entire volume. We retain only the minimal number of wavelet coefficients necessary to represent a surface. The error criteria used in the quantization step are markedly different than those used in the reviewed work due to the fact that we are concerned with

minimizing the geometric error of the derived surface.

Multiresolution techniques have also been investigated in the implicit surface literature. Velho [29] proposed a multi-scale implicit representation based on a biorthogonal B-spline wavelet transform. Their technique produces a representation based only on B-spline scaling functions. They eliminate the wavelet coefficients by projecting the wavelets onto the scaling basis functions at the next finer scale. This eliminates the wavelet coefficients at the cost of an increased number of scaling coefficients. The tradeoff is that all modeling and rendering operations are performed on a hierarchical B-spline representation. The work does not explicitly treat the problem of compressing the resulting data.

Closely related to the previous work is the technique of Grisoni [10], [11] . They represent the field function of an implicit surface as a sampled volume and apply a wavelet transform to obtain a multiresolution representation. Following Velho's method, they project the wavelets onto the scaling basis at the next finer level producing a data structure with only scaling coefficients. Their thresholding scheme operates on the projected coefficients. The location of the wavelet coefficients is not considered in the thresholding process. This allows thresholding of coefficients affecting the reconstructed surface at the expense of retaining some small coefficients far from the surface. They propose a sparse storage scheme based on a hash table storing the location and value of each coefficient in a packed three byte block. Coefficients at coarser scales require fewer bits for encoding position and thus increase the number bits available for quantizing the coefficient value. This data structure does not provide control over the number of bits allocated to the wavelet coefficients, which could impair the accuracy and efficiency of the compression. The present work provides both a location-based thresholding scheme and an adaptive bit allocation method that reduces geometric error and improves compression.

III. Signed-Distance Volumes

A signed-distance volume encodes the minimum distance to a surface for each sample point. The distance changes sign at the surface so that negative values lie on one side and positive values on the other. Given a closed shape, the sign determines whether a point is inside or outside of the shape. For isosurfaces, the notion of inside and outside is not helpful as the surface may exit the distance volume. In these cases the sign of the distance is determined by the isovalue without relying on notions of inside/outside. Signed-distance functions vary smoothly across the transition from inside to outside which make them prime candidates for wavelet decomposition. We formally define the signed distance from a surface Y as:

$$\mathrm{dist}(x) := \mathrm{sign}(x) * \min_{\forall y \in Y}(\parallel x - y \parallel) \qquad (1)$$

where $\mathrm{sign}(x)$ is negative on one side of the surface and positive on the other. Most scanned objects are single closed components making an inside/outside relation easy to define by insuring triangle normal vectors are oriented consistently. Isosurfaces from trilinearly interpolated scientific data also have this property although an isosurface may have a boundary on the boundary of the sampled volume. In such cases the boundary of the distance volume must coincide with the boundary of the scientific data. In the remainder of the paper we will use $\mathrm{d}(x)$ to denote the approximate distance as computed by a distance transform algorithm.

A. Error Metrics

Surface errors are required to study the rate distortion properties of our algorithm. We adopt the L^2 error metric used in [4] and measured by the METRO tool [30]. The error is defined by taking the max of $\mathrm{d}(X,Y)$ and $\mathrm{d}(Y,X)$, where $\mathrm{d}(X,Y)$ is the distance between to surfaces X and Y defined as:

$$\mathrm{d}(X,Y) = \left(\frac{1}{\mathrm{area}(X)} \int_{x \in X} \mathrm{d}(x,Y)^2 dx\right)^{1/2} \qquad (2)$$

where $\mathrm{d}(x,Y)$ is the Euclidean distance from a point $x \in X$ to the closest point on Y. All errors reported in this paper are relative to the bounding box diagonal length.

IV. The Distance Transform

We apply a distance-transform algorithm to surfaces defined by triangle meshes and to isosurfaces from regularly sampled volumetric data. The transform produces an approximation of the actual distance function using the closest-point propagation algorithm of Breen [12] . The distance volume is first initialized with closest-point information for all cells intersecting the surface to be encoded (the *zero set*). These are the only explicit computations with respect to the input surface. Once the zero set is initialized a propagation algorithm assigns the closest points to the rest of the volume samples. In addition, the zero set signs are also initialized and this information is propagated along with the closest points.

The propagation technique is essentially a point sampling approach, as the approximation is produced with respect to the initial set of closest points in the zero set. We begin by describing the propagation algorithm, then describe the zero set initialization methods for scanned and scientific surfaces.

A. Closest Point Propagation

The closest point propagation algorithm relies on the following heuristic: the closest point of a sample s will in most cases be geometrically close to the closest points of the neighbors of s. Our algorithm differs from that of Breen [12] , since we use a simple queue instead of a priority queue. Let $\mathrm{d}(s)$ be the current approximate distance assigned to sample s. Let $\mathrm{cp}(s)$ denote the closest point assigned to sample s such that $\mathrm{d}(s) = \|s - \mathrm{cp}(s)\|$. Finally, when initializing the zero set for meshes the sign of some distance values may be ambiguous. Each sample contains a flag $\mathrm{amb}(s)$ which is set if the sign is ambiguous. The propagation algorithm is as follows:

```
For all distance samples s set
  d(s) = max float
Initialize the zero set of the distance
  field as described in the following
  sections.
Place all zero set samples in a queue Q
while Q is not empty do
    Let s = front(Q)
    For each 26-neighbor t of s do
        If ||cp(s) − t|| < d(t) then
            cp(t) ← cp(s)
            Push t onto Q
        End if
        If s has a sign and amb(s) = false
          then assign the sign of s to t
    end for
end while
```

Breen et. al. presented a scheme based on a priority-queue that always examines the sample with the smallest distance, insuring that a sample is visited only once by the algorithm. This leads to an expected running time of $\mathrm{O}(n \log n)$ where n is the average size of the zero set of samples. The algorithm presented here may set the distance value of a sample a small number of times but uses a simple and fast array based queue and runs in time $\mathrm{O}(n)$. On average each distance sample is updated only 1.2 times for the surfaces we examined.

B. Zero Set Initialization of Isosurfaces

Our current implementation produces signed-distance volumes of isosurfaces defined on regularly sampled scalar fields. Instead of formally defining the isosurface with respect to trilinear interpolation, we compute closest points based on local gradient estimates. The distance approximation is constructed at the same resolution as the initial scalar field. The algorithm examines each volume cell in the scalar field. If the cell contains the isosurface, then the distance samples at the cell corners are initialized with closest point information. Once a distance sample has been initialized it is not reinitialized later for another incident cell.

We denote the scalar field by $\mathrm{f}(s)$. Let f_0 denote the isovalue of the desired isosurface. We define a linear approximation about a sample s as $\hat{\mathrm{f}}(s') = \mathrm{f}(s) + \nabla \mathrm{f}(s)(s' - s)$ and compute the closest point:

$$\mathrm{cp}(s) = s + \frac{f_0 - \mathrm{f}(s)}{\|\nabla \mathrm{f}(s)\|} \nabla \mathrm{f}(s) \tag{3}$$

The scalar field gradient at a given sample point s is estimated by central differencing. The sign of the distance is positive if $\mathrm{f}(s) > f_0$ and negative otherwise. This approximation is inaccurate for high curvature regions but can be computed very efficiently. Greater accuracy can be obtained by performing Newton iterations, or by extracting a mesh and applying the technique in the next section.

C. Zero Set Initialization of Meshes

The zero set initialization for triangle meshes operates on individual triangles. The algorithm does not use edge or vertex adjacency information. The zero-set initialization algorithm proceeds as follows:

```
For every triangle in the input mesh:
    Compute the bounding box of the
      triangle
    For each cell wholly or partially
      included in the bounding box:
        For each distance sample s of the
          cell:
            Compute cp(s)
            Record whether cp(s) lies on a
              face, edge, or vertex.
            Let δ = ||cp(s) − s||
            If δ ≤ √3 * cellwidth and δ < d(s)
              then:
                place the sample on the
                  queue of zero set
                  samples.
                Set a flag indicating
                  that the sample is
                  queued to prohibit
                  duplication.
                Select the sign of this
                  sample according to the
                  rules presented below.
            end if
        end for
    end for
end
```

The sign of a given sample is computed based on the location of its closest point (vertex, edge, or face). Figure 2a

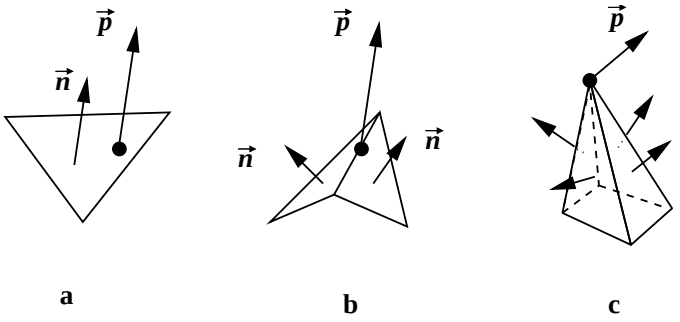

Fig. 2. Zero set initialization for triangle meshes: the sign of a distance sample is determined using the vector $\vec{p}$ from the closest point to the sample and the triangle normal vectors $\vec{n}$.

shows the simplest case where the closest point lies on a face. In this case the sign is given by the sign of $\vec{n} \cdot \vec{p}$ where $\vec{n}$ is the triangle normal and $\vec{p}$ is the vector from the closest point to the distance sample. If the closest point lies on an edge as in Figure 2b, then there are are two dot products (one for each triangle sharing the edge). The absolute values of the dot products are compared and the sign of the larger dot product is taken. Finally, a closest point which coincides with a vertex of the mesh as in Figure 2c may be ambiguous if some dot products are negative and others are positive. However, one cannot use the edge test in this situation. In these cases the distance sample is marked as *ambiguous* and no sign information is propagated for it. After the distance transform has completed the ambiguous samples are revisited and the following heuristic is applied: the signs of the 26-neighbors are examined and the sign of the majority of the neighbors is assigned to the sample. Our method is very similar to a recently presented technique [31]. However, their algorithm does not detect the ambiguous vertex closest points and initializes the incorrect sign.

V. Wavelet Transforms

A wavelet transform [32] decomposes a signal into a sequence of wavelet coefficients representing the details of the signal at several levels of resolution. These coefficients are often of small value and can be compressed efficiently.

A. Fast Wavelet Transform

We apply the fast wavelet transform of Mallat [33] to signed-distance volumes. The left half of Fig. 3 shows one step of decomposition algorithm. At each step a low pass filter $\tilde{h}$ produces a set of scaling coefficients (a_{j+1}) which coarsely approximate the input data. Additionally, a high pass filter $\tilde{g}$ produces a set of wavelet coefficients (d_{j+1}) representing the details lost in the coarse approxi-

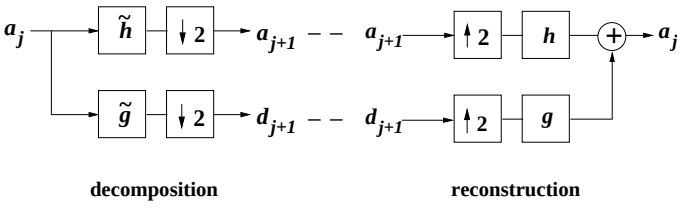

Fig. 3. (Left) Decomposition filter bank with low pass filter $\tilde{h}$ and high pass filter $\tilde{g}$. (Right) Reconstruction filter bank with low pass filter h and high pass filter g.

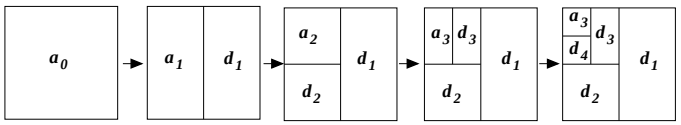

Fig. 4. (a) Two dimensional extension of the filter bank by alternating the directions of the filtering steps.

mation. These two filtering steps are repeated recursively on the coarse approximations to obtain a multiresolution representation. At each stage the size of the data is down sampled by a factor of two. We will use the term *subband* to refer to a set of wavelet coefficients generated by one step of the transform.

The original data can be reconstructed by reversing the process with another set set of filters h and g in the general case. The right half of Fig. 3 shows one reconstruction step. The low pass filter h is similar to a subdivision operator, smoothing the coarse approximations and increasing their resolution (the number of samples). The high pass filter g re-introduces the details encoded by $\tilde{g}$ and enables exact reconstruction of the data. Both filters are preceded by an up sampling by two which inserts zeros between each pair of input values. In the present work we chose the linear B-spline wavelets with the following filters:

$$\tilde{h}[n] = -\frac{1}{8}\ \frac{1}{4}\ \frac{3}{4}\ \frac{1}{4}\ -\frac{1}{8} \quad (4)$$

$$h[n] = \frac{1}{2}\ 1\ \frac{1}{2} \quad (5)$$

$$\tilde{g}[n] = \frac{1}{2}\ -1\ \frac{1}{2} \quad (6)$$

$$g[n] = \frac{1}{8}\ \frac{1}{4}\ -\frac{3}{4}\ \frac{1}{4}\ \frac{1}{8} \quad (7)$$

The 2D extension of the algorithm in Fig. 3 is shown in Fig. 4. The one dimensional transform is alternately applied to each dimension, creating subbands 1 and 3 in the x direction (d_1, d_3) and subbands 2 and 4 in the y direction (d_2, d_4). The 3D case follows the same pattern as Fig. 4 except that the transform directions cycle through the x, y, and z directions. Some readers may note that in

image processing applications the high pass coefficients resulting from the x direction filtering are processed by the filter bank a second time in the y direction yielding three sub-bands per level in the 2D case and seven subbands in 3D (after a z pass). In contrast, our approach generates one subband per level for data of any dimension.

We chose the direct extension of the filter bank algorithm for three reasons: first, in our experiments on image data the compression rates were better. Second, fewer operations need to be performed on the data. Third, the first subband of wavelet coefficients (produced in the first high pass filtering) effect distance values in the x direction only. The support of these wavelets is thus local in memory, enabling simple and local optimization schemes for allocating bits to these values.

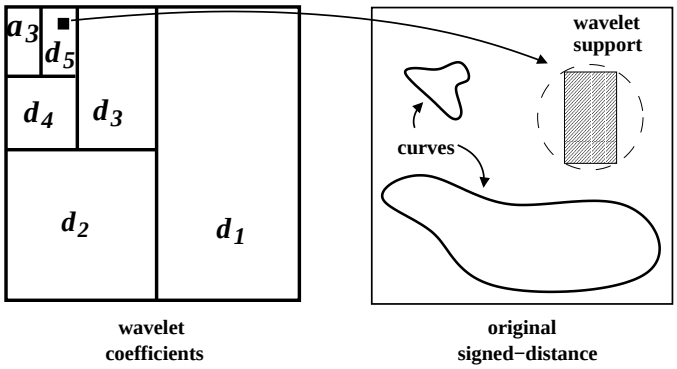

Fig. 5. Distance based thresholding: the coefficient in d_3 is set to zero because its support shown on the right does not overlap the curves.

B. Thresholding

The goal of the thresholding step is to reduce the number of values that need to be coded without introducing geometric errors. An aggressive thresholding method is required for efficient distance volume compression. Our method removes all wavelet coefficients that do not contribute to the reconstructed surface. Thresholding too many coefficients could result in spurious surface components appearing in the distance field. Currently, we do not have a formalism that allows us to prove that new components or handles are not added under the method we present. For complicated surfaces a verification step can be performed that checks the original distance volume against the distance volume reconstructed after the thresholding step and warns of any irregularities.

The thresholding method proceeds as follows:

```
For each subband:
    Determine the support of wavelets in
      this subband with respect to the
      reconstructed samples.
    Compute the radius r of a bounding
      sphere for the wavelet support.
    For each wavelet coefficient within
      the subband:
        Compute the sample s in the
          reconstructed volume
          corresponding to the center of
          the wavelet basis function
        if d(s) > r set the wavelet
          coefficient to zero
    end for
end for
```

Figure 5 illustrates the basic thresholding operation in 2D. On the left is the wavelet transformed signed-distance field showing three wavelet subbands. On the right we have the original signed-distance field as computed for the two curves shown inside. The wavelet coefficient in d_3 can be thresholded because its support does not effect the curves being represented.

The wavelet support radii are easily computed as they depend only on the lengths of the filters and the number of applications of those filters. A more accurate thresholding can be accomplished by explicitly testing the distance samples in the wavelet support to determine if any belong to the zero set. This more expensive test would be applied after the sphere test. However, this scheme was not implemented for this paper.

VI. Zero-Tree Coding

A progressive wavelet coder should send the most significant bits of the most significant wavelet coefficients first. This amounts to encoding the locations of the significant coefficients as efficiently as possible. A zero tree coder [34] generates a progressive bit stream by utilizing the property that wavelets decay in magnitude at finer resolutions. That is, if one defines a hierarchy of wavelet coefficients from one subband to the next it is likely that the child coefficients will be smaller than the parent.

A zero tree is defined as a hierarchy of coefficients for which $c \leq T$ for every coefficient c in the hierarchy, where T is a threshold used to determine the significance of any given coefficient. The zero tree relation is defined for a quadtree-like hierarchy in 2D and an octree-like hierarchy in 3D. Figure 6 depicts the 2D case for our subband ordering. Two hierarchies are shown, one for the x direction (subbands 1, 3, and 5) and one for the y direction (subbands 2 and 4). A zero tree coder is particularly well suited to the distance based thresholding method as the thresholded coefficients are spatially contiguous.

The coding algorithm repeatedly traverses the wavelet

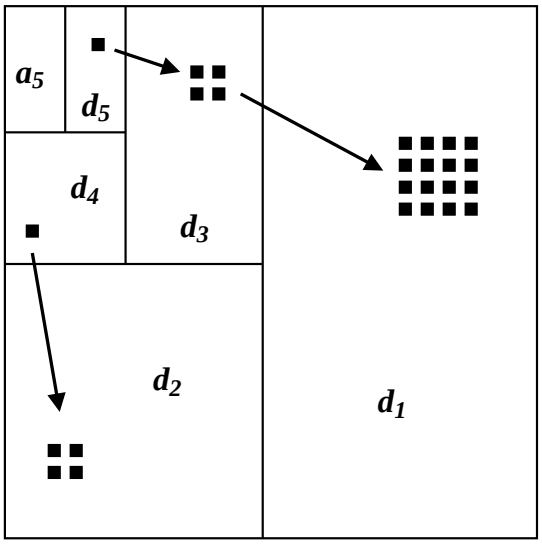

Fig. 6. Hierarchies of wavelet coefficients for the zero tree relation. If all of the coefficients are less than a given threshold, the entire tree can be skipped until later in the coding process.

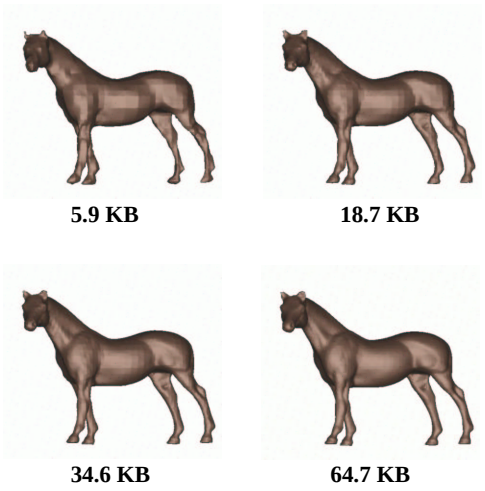

Fig. 7. Progressive reconstructions of the horse model from a compressed distance volume of 64x138x115 samples.

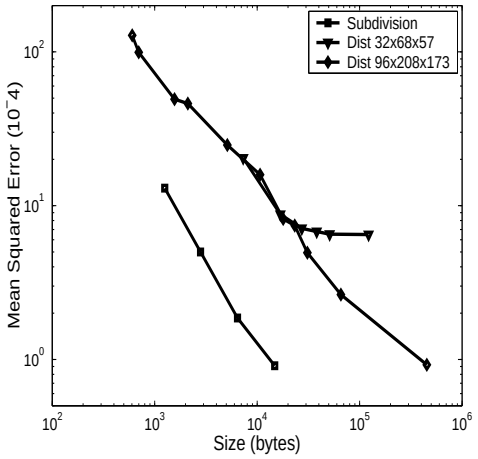

Fig. 8. Comparison of compression rates of subdivision surfaces and signed-distance volumes.

coefficients in a predefined order. At any point in the coding process the wavelet coefficients are divided into two groups: those that are not yet significant, and those that have been found to be significant during the current traversal or a preceding one. The threshold T starts at half the value of the largest wavelet coefficient, and is divided in half after each traversal. Thus, each traversal progressively adds more wavelet coefficients to the significant group and removes them from the insignificant group. A zero tree is coded by a single symbol the informs the decoder that every coefficient in the hierarchy can be skipped at that stage during decoding. Zero trees efficiently encode the positions of the insignificant coefficients. Once a coefficient is deemed significant, its sign is transmitted and another bit is added to its representation following each subsequent traversal of the insignificant coefficients. Our implementation follows [34] which contains pseudocode and a small worked example of the algorithm.

VII. Results

We applied our algorithm to two surfaces. All file sizes are the result of applying the gzip utility to the progressive bit-streams resulting from our zero tree coder. A horse model was used to compare the performance of the signed-distance volume approach to the subdivision surface approach in [4]. An isosurface was used to demonstrate the ability of our algorithm to compress complex surfaces with many components.

Figure 7 shows four progressive reconstructions of the horse model for increasing file sizes. The original horse model consists of $48,485$ vertices and $145,450$ triangles. A good lossless non-progressive mesh compression algorithm [35] generates between $8-15$ bits per vertex. Thus, one could expect the lossless non-progressive compression of the horse mesh to produce a file between $48-80$ kilobytes in size. Our progressive coding begins to represent the muscle structure of the horse at about 30 kilobytes. Our algorithm is of course a lossy compression technique.

Figure 8 plots the mean squared geometric error defined in section III-A for the subdivision surface method of Khodakovsky [4] and our approach. The errors are scaled to the bounding box diagonal. As expected, the subdivision surface approach fairs better than the distance volume method for this smooth model. The base mesh used for the subdivision based method consists of 122 ver-

Data Set	Zero Set (s)	Propagation (s)
Horse (32x68x57)	74	2
Turbulence (256x256x128)	17	157

TABLE I
ELAPSED TIMES FOR THE ZERO SET INITIALIZATION AND CLOSEST POINT PROPAGATION STEPS OF THE DISTANCE TRANSFORM.

tices. This small size enables quite efficient compression. Our method produces a very efficient encoding of the approximate signed-distance as the original distance function comprised $3,454,464$ floating point values in the case of the $96x208x173$ distance volume.

Figure 9 shows four reconstructions of a complex isosurface compressed with our system. The data set depicts the turbulent mixing of two fluids [1] at a resolution of 2048x2048x1920. The isosurface shown here derives from a subset of 256x256x384 samples. The distance transform was computed at the same resolution as the subset of the scalar field. Our system retains the major features of the surface for small sizes and achieves good compression through topological modification. This surface is less well suited to subdivision surface techniques as the size of the base mesh is much larger due to the complex topology and large number of components.

Finally, we present performance results for our system. Table I shows the time required for each stage of the distance transform. All times are reported for an SGI O2 work station with 320 megabytes of ram and a 200 megahertz MIPS R5000 processor. The zero set initialization cost of triangle meshes is due to the large number of intersection computations. In contrast, the simple linear approximation used for scalar field data is ideal. The propagation time is manageable even for large data sets. Some improvement is possible if the distance samples are reordered to improve cache coherence when visiting a sample's 26-neighborhood. The zerotree coder took from $0.5-3$ seconds for the horse model at various resolutions and geometric errors. The isosurface coding took between $1-20$ seconds depending on the number of bits produced.

VIII. DISCUSSION

Our algorithm successfully produces a progressive encoding of a signed-distance volume. However, subdivision surface approaches still produce very compact surface representations. One drawback of our current implementation is the use of the gzip utility as a back-end. This is less efficient than a compression technique specifically tailored for wavelet transforms. An arithmetic coder applied to the coefficient magnitudes should reduce the file sizes even further.

The geometric error for small bit counts can be improved by modifying the ordering of the bits. A standard zero tree coder assumes an $L_{\inf}$ error because the significance test depends only on the value of the coefficient and not on the size of the corresponding wavelet support. It is possible for wavelet coefficients at the finest level to have bits emitted along with coefficients at a coarser level. However, the mean-squared geometric error metric integrates over the surface area, implying that the significance test should include the support of the wavelet. Incorporating the support of the wavelets would insure that all of the early bits increase the accuracy of the coarser scale wavelets instead of potentially adding fine scale wavelets, thus improving the overall error.

Finally, the encoding process should be modified so that the topology is simplified at early stages and is refined as more bits are added. Our implementation does not track topology changes and allows both simplifications and refinements to occur. This can produce holes in thin shapes at small numbers of bits that disappear later in the decoding process. A method for ordering the topology changes and increasing the significance of the wavelet coefficients affecting those areas could mitigate this problem.

IX. CONCLUSION

We have presented an algorithm that produces progressively compressed signed-distance volumes. Our method progressively compresses surfaces without the re-meshing that is required for subdivision based approaches. Our representation does not explicitly represent the surface topology, enabling topological modification without complicating the data structures used in the implementation. We believe our approach is best suited to surfaces with complicated topology and many components. Time-varying surfaces pose many problems that can be overcome with an implicit representation. We believe our compression techniques can be extended to the time domain to produce an efficient yet simple surface representation.

X. ACKNOWLEDGMENTS

This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48.

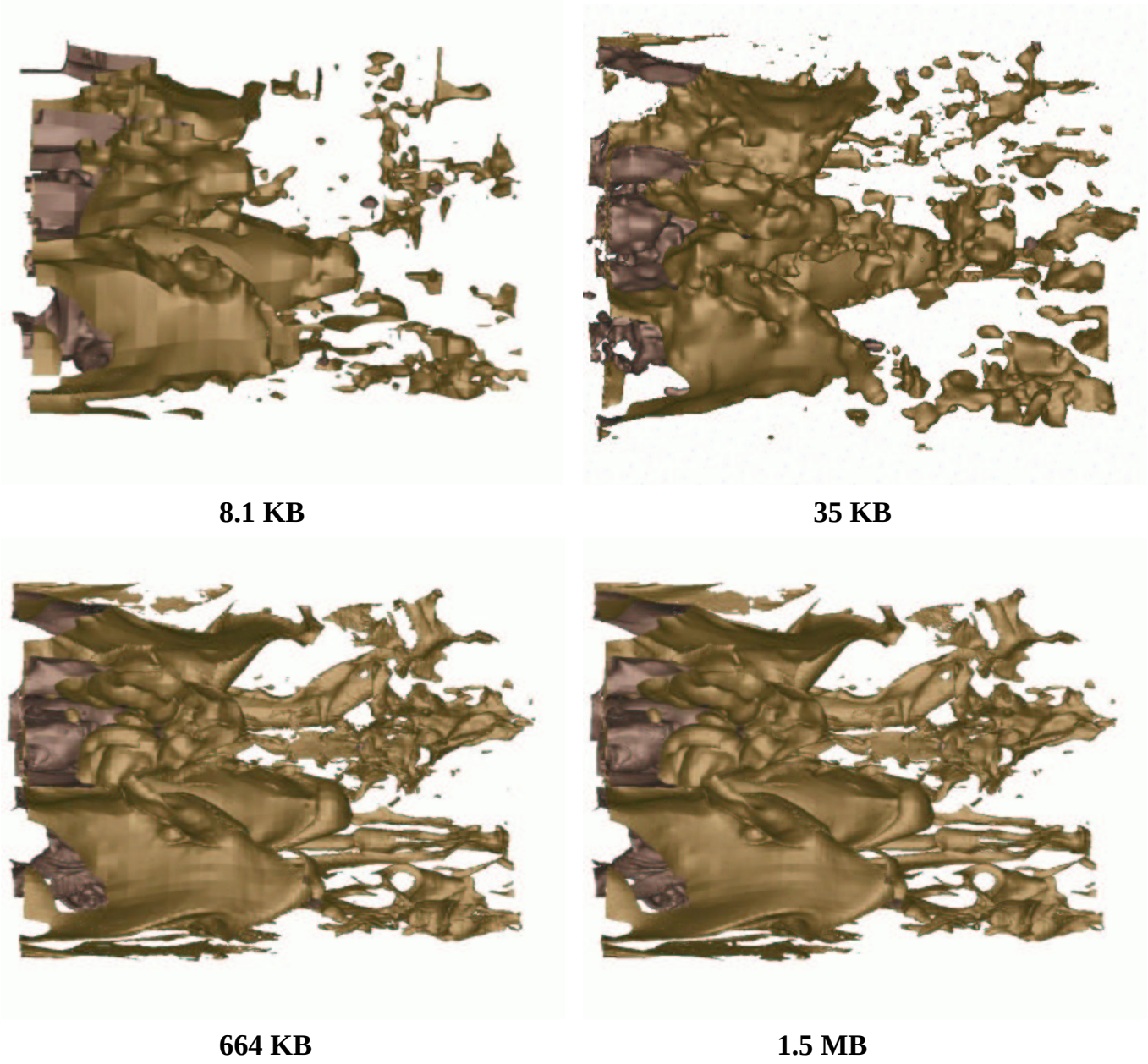

Fig. 9. Progressive reconstructions of an isosurface of a turbulent mixing simulation.

REFERENCES

[1] A. A. Mirin, R. H. Cohen, B. C. Curtis, W. P. Dannevik, A. M. Dimits, M. A. Duchaineau, D. E. Eliason, D. R. Schikore, S. E. Anderson, D. H. Porter, P. R. Woodward, L. J. Shieh, and S. W. White, "Very high resolution simulation of compressible turbulence on the IBM-SP system," in *Super Computing 1999, Oregon*, ACM, Ed. 1999, ACM Press and IEEE Computer Society Press.

[2] Marc A. Levoy, "The Digital Michelangelo Project," *Computer Graphics Forum*, vol. 18, no. 3, Sept. 1999.

[3] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy Ginsberg, Jonathan Shade, and Duane Fulk, "The digital michelangelo project: 3D scanning of large statues," in *Siggraph 2000, Computer Graphics Proceedings*, Kurt Akeley, Ed. 2000, Annual Conference Series, pp. 131–144, ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[4] Andrei Khodakovsky, Peter Schröder, and Wim Sweldens, "Progressive geometry compression," in *Siggraph 2000, Computer Graphics Proceedings*, Kurt Akeley, Ed. 2000, Annual Conference Series, pp. 271–278, ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[5] Martin Bertram, Mark A. Duchaineau, Bernd Hamann, and Kenneth I. Joy, "Generalized b-spline subdivision-surface wavelets for geometry compression," *Submitted for publication*, 2001.

[6] Igor Guskov, Andrei Khodakovsky, Peter Schröder, and Wim Sweldons, "Hybrid meshes," in *Submitted for publication*, January 2001.

[7] Ariel Shamir, Valerio Pascucci, and Chandrajit Bajaj, "Multiresolution dynamic meshes with arbitrary deformations," in *Proc.*

of the 11th Ann. IEEE Visualization Conference (Vis) 2000, 2000, pp. 423–430.

[8] Ross T. Whitaker and David E. Breen, "Level-set models for the deformation of solid objects," in *Proc. of the 3rd International Workshop on Implicit Surfaces*, June 1998, pp. 19–35.

[9] Sarah F. Gibson, "Using distance maps for accurate surface representation in sampled volumes," in *Proceedings of the 1999 Conference on Volume Visualization*, Oct 1998, pp. 23–30.

[10] Laurent Grisoni and Christophe Schlick, "Multiresolution representation of implicit objects," in *Implicit Surfaces*, 1998, pp. 1–10.

[11] L. Grisoni, B. Crespin, and C. Schlick, "Multiresolution implicit representation of 3d objects," Tech. Rep. 1217-99, 1999.

[12] David E. Breen, Sean Mauch, and Ross T. Whitaker, "3d scan conversion of csg models into distance volumes," in *Proc. 1998 Symposium on Volume Visualization*, Oct 1998, pp. 7–14.

[13] W. Sweldens, "The lifting scheme: A new philosophy in biorthogonal wavelet constructions," in *Wavelet applications in signal and image processing III*, Andrew F. Laine, Michael A. Unser, and Mladen V. Wickerhauser, Eds., 1995, vol. 2569 of *Proceedings of SPIE*, pp. 68–79.

[14] Martin Bertram, Mark A. Duchaineau, Bernd Hamann, and Kenneth I. Joy, "Bicubic subdivision-surface wavelets for large-scale isosurface representation and visualization," in *Proc. of the 11th Ann. IEEE Visualization Conference (Vis) 2000*, 2000.

[15] G. Taubin and J. Rossignac, "3d geometry compression," 1999, p. Course Notes 21, ACM Siggraph.

[16] Michael Garland and Paul S. Heckbert, "Surface simplification using quadric error metrics," in *SIGGRAPH 97 Conference Proceedings*, Turner Whitted, Ed. ACM SIGGRAPH, Aug. 1997, Annual Conference Series, pp. 209–216, Addison Wesley, ISBN 0-89791-896-7.

[17] Hugues Hoppe, "Progressive meshes," in *SIGGRAPH 96 Conference Proceedings*, Holly Rushmeier, Ed. ACM SIGGRAPH, Aug. 1996, Annual Conference Series, pp. 99–108, Addison Wesley, held in New Orleans, Louisiana, 04-09 August 1996.

[18] Jovan Popović and Hugues Hoppe, "Progressive simplicial complexes," in *SIGGRAPH 97 Conference Proceedings*, Turner Whitted, Ed. ACM SIGGRAPH, Aug. 1997, Annual Conference Series, pp. 217–224, Addison Wesley, ISBN 0-89791-896-7.

[19] R. Pajarola and J. Rossignac, "SQUEEZE: Fast and progressive decompression of triangle meshes," in *Proceedings of the Conference on Computer Graphics International (CGI-00)*, Los Alamitos, CA, June 19–24 2000, pp. 173–182, IEEE.

[20] R. Pajarola and J. Rossignac, "Compressed progressive meshes," in *IEEE Transactions on Visualization and Computer Graphics*, Hans Hagen, Ed., vol. 6 (1), pp. 79–93. IEEE Computer Society, 2000.

[21] Shigeru Muraki, "Approximation and rendering of volume data using wavelet transforms," in *Proc. IEEE Visualization*, 1992, pp. 21–28.

[22] Shigeru Muraki, "Volume data and wavelet transforms," vol. 13, no. 4, pp. 50–56, July 1993.

[23] Shigeru Muraki, "Multiscale volume representation by a dog wavelet," in *IEEE Trans. on Visualization and Computer Graphics*, 1995, vol. 1, pp. 109–116.

[24] M. H. Gross, L. Lippert, R. Dittrich, and S. Häring, "Two methods for wavelet-based volume rendering," *Computers & Graphics*, vol. 21, no. 2, pp. 237–252, Mar. 1997, ISSN 0097-8493.

[25] R. Grosso, T. Ertl, and J. Aschoff, "Efficient data structures for volume rendering of wavelet-compressed data," in *Winter School of Computer Graphics 1996*, Feb. 1996, held at University of West Bohemia, Plzen, Czech Republic, 12-16 February 1996.

[26] H. Tao and R. J. Moorhead, "Progressive transmission of scientific data using biorthogonal wavelet transform," in *Proceedings of the Conference on Visualization*, R. Daniel Bergeron and Arie E. Kaufman, Eds., Los Alamitos, CA, USA, Oct. 1994, pp. 93–99, IEEE Computer Society Press.

[27] Rüdiger Westermann, "Compression domain rendering of time-resolved volume data," in *Proc. IEEE Visualization*, 1995.

[28] Insung Ihm and Sanghun Park, "Wavelet-based 3D compression scheme for very large volume data," in *Graphics Interface*, June 1998, pp. 107–116.

[29] Luiz Velho, Demetri Terzopoulos, and Jonas de Miranda Gomes, "Multiscale implicit models," in *Proc. of SIBGRAPI*, 1994, pp. 93–100.

[30] P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: Measuring error on simplified surfaces," *Computer Graphics Forum*, vol. 17, no. 2, pp. 167–174, 1998.

[31] Jian Huang, Yan Li, Roger Crawfis, Shao-Chiung Lu, and Shuh-Yuan Liou, "A complete distance field representation," in *Proceedings of the 2001 IEEE Conference on Visualization (VIS-01)*, Thomas Ertl, Ken Joy, and Amitabh Varshney, Eds., N.Y., Oct. 21–26 2001, pp. 247–254, ACM Press.

[32] Stéphane Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, 1998.

[33] Stéphane Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Patt. Recog. and Mach. Intell.*, vol. 11, no. 7, pp. 674–693, July 1989.

[34] A. Said and W. Perlman, "A new, fast, and efficient image codec based on set partioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, 1996.

[35] Costa Touma and Craig Gotsman, "Triangle mesh compression," in *Graphics Interface*, 1998, pp. 26–34.

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.