

Wavelet-Based Volume Morphing

Taosong He, Sidney Wang, and Arie Kaufman

Department of Computer Science
State University of New York at Stony Brook
Stony Brook, NY 11794-4400

Published in:

Proceedings of Visualization '94
Washington D.C., October 1994
pp. 85 – 92

BiBTeX Entry:

```
@InProceedings{ He94Wavelet-Based,  
  author    = "T. He and S. Wang and A. Kaufman",  
  title     = "Wavelet-Based Volume Morphing",  
  year      = 1994,  
  month     = oct,  
  booktitle = "Proceedings of Visualization 94",  
  pages     = "85--92",  
  address   = "Washington, D.C."  
}
```

Wavelet-Based Volume Morphing

Taosong He, Sidney Wang, and Arie Kaufman

Department of Computer Science

State University of New York at Stony Brook

Stony Brook, NY 11794-4400

Abstract

This paper presents a technique for performing volume morphing between two volumetric datasets in the wavelet domain. The idea is to decompose the volumetric datasets into a set of frequency bands, apply smooth interpolation to each band, and reconstruct to form the morphed model. In addition, a technique for establishing a suitable correspondence among object voxels is presented. The combination of these two techniques results in a smooth transition between the two datasets and produces morphed volume with fewer high frequency distortions than those obtained from spatial domain volume morphing.

1. Introduction and motivation

Recently, 3D metamorphosis, the process of simulating the deformation of one 3D model to another, has gained popularity in animation and shape design. Previously published techniques [5, 6] deal mainly with the metamorphosis between two polygonal-based models. The general method of these algorithms is to displace the vertices, edges, and faces of the first model over time to coincide in position with the corresponding vertices, edges, and faces of the second model. However, establishing the suitable correspondence among surface elements is complex. In addition, these algorithms generally impose topological restrictions on the models in order to maintain the face connectivity during morphing.

Motivated in part by the difficulties presented in morphing surface-based 3D models and in part by the desire to morph sampled/simulated datasets directly, the volume graphics [4] approach represents the 3D models in voxel space and performs volume morphing between the two volumetric models. One of the main advantages of this approach is that the topology restriction on the datasets is eliminated, since there is no explicit topology description of the volume and the voxel correspondence can be directly established between any two volumes. However, the problem of finding the appropriate correspondence among voxels still exists. One naive solution is simply to cross-dissolve between the two volumes over time. In other words, to morph from model $g(x, y, z)$ to $f(x, y, z)$, a new model $k_t(x, y, z) = (1 - t)g(x, y, z) + t f(x, y, z)$ is formed.

Although simple, this method is often ineffective for creating the smooth transition from one model to another. For example, given two concentric spheres with identical iso-values but with different radii, an ideal morphing from the larger sphere to the smaller one should be a sphere with constant iso-value but a gradually shrinking radius. But the naive technique described above would generate a sudden shrinkage from the large sphere to the small one at a certain time T when a surface rendering method is employed. This is because when $0 \leq t < T$, the region between the two spheres has the density value above the iso-value, and at time $t = T$, the density values of the region uniformly drop below the iso-value.

Another problem when performing volume morphing is that the direct interpolation of the high-frequency components in the models might cause distortion and unsatisfactory results. Hughes [3] deals with this problem by performing volume morphing in the Fourier domain. Basically, his approach takes the first volumetric model, gradually removes the high frequencies, interpolates over to the low frequencies of the second model, and then smoothly adds in the high frequencies of the second model. Although effective in reducing high frequency distortion, the technique does not solve the problem of unsmooth transformation of iso-surfaces because the Fourier transform does not localize in spatial domain. In Hughes' implementation, in order to have a smooth transition during morphing, the voxel values of the entire volume are modified according to the distance to the nearest iso-surface. Hence, new datasets need to be created solely for the morphing application.

In this paper, a technique for performing volume morphing in the wavelet domain is introduced. Since wavelet transform localizes in both frequency domain and spatial domain, the problems of high frequency distortion and unsmooth transformation can be alleviated simultaneously. The idea is to decompose the volumes into a set of frequency bands, apply smooth interpolation between the volumes to each band, and then reconstruct the morphed volume. Furthermore, the decomposition and reconstruction processes are accomplished in a multiresolution fashion so that high frequency distortion

can be adjusted to the desired level. By taking advantage of the spatial information within each frequency band, we can extract and correspond the object voxels of the first model to the object voxels of the second model intelligently. In the next section, the volume correspondence problem is presented. In Section 3 wavelet theory is briefly introduced. Wavelet application to volume morphing is described in Section 4.

2. The correspondence problem

Unlike polygonal-based modeling, where every surface element of the mesh contributes to the modeling of a part of an object, voxel-based modeling captures not only the object itself but also the space surrounding the object. Hence, in order to have a gradual deformation from one object to another, it is essential to map only those voxels which belong to parts of an object. In our implementation, iso-values are used to distinguish these voxels from empty space. The algorithm for solving the correspondence problem is first described in 1D space, followed by the extension of the algorithm into 3D space. Given an object A in a 1D raster and an object B in another 1D raster, the first step of the algorithm is to classify these two rasters into segments of object and non-object. Without loss of generality, let object A consist of m object segments, object B consist of n object segments, and $m \geq n$ (Figure 1). As in the case of surface-based morphing, the two important quality criterion are maintaining the correct topology and minimizing the shape distortion during transformation. First, to satisfy the topology criterion, each object segment in A can be mapped to only one object segment in B . This restriction is needed to ensure that the number of object segments does not increase during morphing. Under the condition that the first criterion is satisfied, object segments of A should be distributed onto object segments of B as "evenly" as possible to minimize the shape distortion.

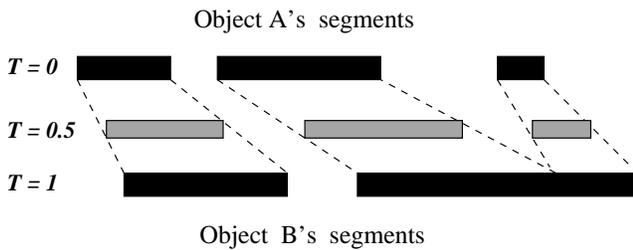


Figure 1: 1D Correspondence Problem

Formally, the 1D correspondence problem is stated as follows. Given:

$$A = \left\{ \overline{a_1 a_1'}, \overline{a_2 a_2'}, \dots, \overline{a_m a_m'} \right\},$$

$$B = \left\{ \overline{b_1 b_1'}, \overline{b_2 b_2'}, \dots, \overline{b_n b_n'} \right\}, \quad m \geq n \quad (1)$$

determine:

$$P = \{p_1, p_2, \dots, p_n\} \text{ and}$$

$$W = \{w_1, w_2, \dots, w_n\} \quad (2)$$

where

$$p_i = \left\{ x_j, \dots, x_k \right\}, \quad 1 \leq i \leq n, \quad 1 \leq j \leq k \leq m,$$

$$\text{and } x_l \in A \text{ for } j \leq l \leq k \quad (3)$$

$$w_i = \sum_{j \leq l \leq k} (a_l' - a_l) \quad (4)$$

subject to:

$$1. (x_u \in p_{i_1} \wedge x_v \in p_{i_2} \wedge i_1 < i_2) \rightarrow (u < v)$$

for all u, v (5)

and

$$2. \min \sum_{i=1}^n \left(\frac{w_i}{(b_i' - b_i)} - \frac{\sum_{i=1}^m (a_i' - a_i)}{\sum_{i=1}^n (b_i' - b_i)} \right)^2. \quad (6)$$

In Equation 2, the set P represents the partition of A 's object segments into n partitions, and the set W represents the corresponding weight for each member of P . As shown in Equation 4, the weight for each partition is the total length of object segments within each partition. Equation 5 guarantees that the partition is in consecutive order from left to right. Equation 6 insures that object segments of A are "evenly" distributed onto object segments of B by minimizing the variance.

In our implementation, dynamic programming has been used to solve the 1D correspondence problem. Once the correspondences have been established for the object segments in A and B , each object segment in B needs to be partitioned to accommodate the corresponding object segments from A . For example, if

```

Correspond_3D ( A, B )
Volume_Data *A, *B;
{
  Z1 = Object_Segments ( A, Z_AXIS );
  Z2 = Object_Segments ( B, Z_AXIS );

  Correspond_1D ( Z1, Z2 );

  for each non-empty scan plane u in A
  {
    u' = the corresponding scan plane in B;
    Y1 = Object_Segments ( u , Y_AXIS );
    Y2 = Object_Segments ( u' , Y_AXIS );

    Correspond_1D ( Y1, Y2 );

    for each non-empty scan line v in u
    {
      v' = the corresponding scan line in u' ;
      X1 = Object_Segments ( v , X_AXIS );
      X2 = Object_Segments ( v' , X_AXIS );

      Correspond_1D ( X1, X2 );
    }
  }
}

```

Figure 2: Pseudo-code for the 3D Correspondence

$$p_i = \left\{ \overline{a_j a_j'}, \dots, \overline{a_k a_k'} \right\} \quad (7)$$

then segment $\overline{b_i b_i'}$ of B needs to be partitioned into $k - j + 1$ sub-segments of lengths

$$\left\{ (a_j' - a_j) r, \dots, (a_k' - a_k) r \right\} \quad (8)$$

where

$$r = \frac{(b_i' - b_i)}{\sum_{l=j}^k (a_l' - a_l)}. \quad (9)$$

The correspondence problem in 3D space is essentially accomplished by applying the above 1D

correspondence algorithm, which we call `Correspond_1D`, to each of the three principal axes in an nested fashion. The 3D correspondence algorithm, described in the pseudo-code presented in Figure 2, establishes a mapping from object scan-planes of A to object scan-planes of B , from object scan-lines of A to object scan-lines of B , and finally from object voxels of A to object voxels of B . With these correspondence relations, volume morphing is achieved through the interpolations over time of the corresponding scan-planes, scan-lines, and voxels.

3. Wavelet theory

Generally, the high frequency components in 3D functions represented by volumetric data tend to generate small wiggles on the iso-surfaces of the models [3]. If the morphing algorithm described above is directly performed in the spatial domain, these wiggles could cause distortions on the iso-surfaces of the intermediate functions (see Figure 3a). Recently, wavelet theory, which is rooted in time-frequency analysis, has been widely used in a variety of applications, such as shape description of volumetric objects [8] and radiosity [2]. Since a wavelet transform has local property in both spatial and frequency domain, it is an ideal solution to the problem of high frequency distortion during morphing. In this section wavelet theory is briefly introduced, then in Section 4 the wavelet-based morphing algorithm is presented.

Multiresolution signal analysis decomposes a function into a smooth approximation of the original function and a set of detailed information at different resolutions [7]. Formally, let $L^2(R)$ denote all functions with finite energy; the smooth approximation of a function $f \in L^2(R)$ at any resolution 2^i is a projection denoted as $A_{2^i}: L^2(R) \rightarrow V_{2^i}$, $V_{2^i} \in L^2(R)$, and the detail of f at any higher resolution 2^j is a projection of f onto a subspace O_{2^j} of $L^2(R)$ denoted as $P_{2^j}: L^2(R) \rightarrow O_{2^j}$, $j \geq i$. Consequently, the finest detailed information is contained in P_{2^j} with the highest resolution. By choosing the appropriate projection functions such that O_{2^j} are orthogonal to both each other and V_{2^i} , we have $V_0 = L^2(R)$ and $L^2(R) = \bigoplus_{j=i}^1 O_{2^j} \oplus V_{2^i}$ (when O_{2^j} is the orthogonal complement of V_{2^j} , $V_{2^{j+1}}$ is written as $V_{2^{j+1}} = V_{2^j} \oplus O_{2^j}$). For the discrete functions, it can be proven that there exist two families of functions:

$$\psi_{j,n} = 2^{-j/2} \psi(2^j t - n)_{n \in \mathbb{Z}} \quad (10)$$

$$\phi_{j,n} = 2^{-j/2} \phi(2^j t - n)_{n \in \mathbb{Z}}, \quad (11)$$

which constitute the orthonormal basis of V_{2^j} and O_{2^j} , respectively. $\psi_{j,n}$ are called *wavelets* and $\phi_{j,n}$ are the corresponding *scaling functions*.

Using wavelets and scaling functions, the *discrete detail signal* and *discrete approximation* at resolution 2^j are respectively defined as:

$$(D_{2^j} f)_n = 2^{-j/2} \langle f(u), \psi_{j,n} \rangle \quad (12)$$

$$(A_{2^j}^d f)_n = 2^{-j/2} \langle f(u), \phi_{j,n} \rangle \quad (13)$$

and the detailed information and smooth approximation are:

$$P_{O_{2^j}} f = \sum_{n=-\infty}^{\infty} (D_{2^j} f)_n \psi(2^j t - n) \quad (14)$$

$$A_{2^j} f = \sum_{n=-\infty}^{\infty} (A_{2^j}^d f)_n \phi(2^j t - n). \quad (15)$$

Instead of calculating the inner products in Equations 12 and 13, a pyramidal algorithm [7] is applied for the decomposition of the function (Figure 4a), where $\bar{H} = H(-n)$ and $\bar{G} = G(-n)$. The impulse response of the filters used is defined as:

$$H(n) = \langle 2^{-1} \phi(2^{-1} u), \phi(u - n) \rangle \quad (16)$$

$$G(n) = \langle 2^{-1} \psi(2^{-1} u), \psi(u - n) \rangle \quad (17)$$

By repeating the algorithm for $-1 \geq j \geq -M$, both the discrete detail signal and the discrete approximation at resolution 2^j can be computed. Using the same pair of filters, the original discrete samples can be computed by the reverse pyramidal algorithm, as shown in Figure 4b.

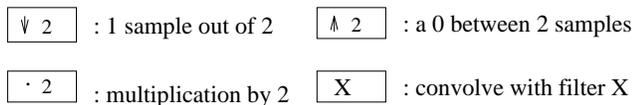
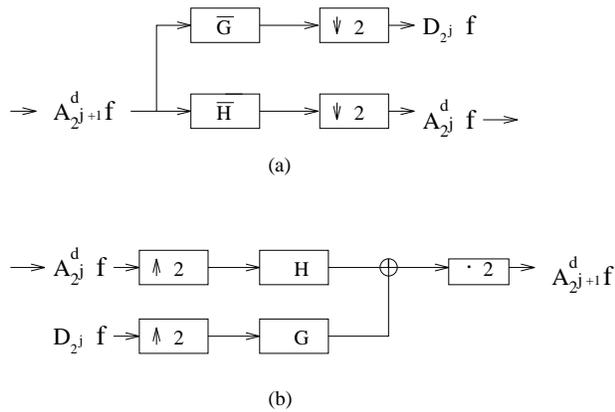


Figure 4: Wavelet Decomposition and Reconstruction

Wavelet theory can be easily expanded to any dimension by constructing high dimension orthonormal wavelets using the tensor product of several subspaces of $L^2(R)$ [7, 8]. To decompose or reconstruct a 3D function, the one dimensional pyramidal algorithm described in Figure 4 is applied sequentially along the principal axes. Since the convolution along each axis is separable, for a volume of size n^3 , the decomposition and reconstruction can be implemented in $O(n^3)$ time, which is asymptotically optimal. Figure 5 shows that the smooth approximation of a volume at resolution 2^{j+1} decomposes into a smooth approximation at resolution 2^j and the discrete detail signals along seven orientations. Since the wavelets and scaling functions are orthogonal, the multiresolution representation

$$(A_{2^{-M}}^d f, (D_{2^j}^i f)_{-M \leq j \leq -1, 1 \leq i \leq 7}) \quad (18)$$

has the same total number of samples as the original signal $A_1^d f$.

4. Wavelets for morphing

Equation 10 and Equation 11 indicate that both the wavelets and scaling functions are the translation and dilation of a mother function $\psi(t)$ or $\phi(t)$. It can thus be proven that the wavelet decomposition has the property of both spatial and frequency locality. The algorithm shown in Figure 4a can be interpreted as the separation of the detailed information, which corresponds to high pass filtering, and the generation of a smooth function, which corresponds to low pass filtering. In addition, a signal that is nonzero only during a finite time span has a wavelet transform whose nonzero elements are concentrated around that time. For a 3D volume, this means that the spatial information, which is essential for volume morphing, is maintained in wavelet domain.

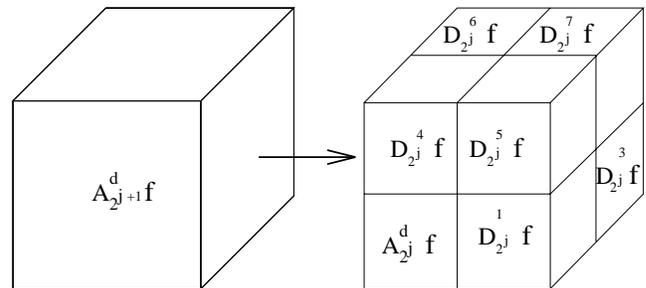


Figure 5: Decomposition of a Volume Discrete Approximation by a 3D Wavelet Transform

The basic idea of wavelet-based morphing is to solve the correspondence problem between the general shape of the objects without the interference of high frequencies. To achieve this, the two volumes f and g are first decomposed into smooth approximation at resolution 2^{-M} and the detailed information:

$$(A_{2^{-M}}f, (P_{O_{2^j}}f)_{-M \leq j \leq -1}) \text{ and} \\ (A_{2^{-M}}g, (P_{O_{2^j}}g)_{-M \leq j \leq -1}). \quad (19)$$

Next, the direct morphing algorithm, which is described in Section 2, is applied on the smooth approximation $A_{2^{-M}}f$ and $A_{2^{-M}}g$ to generate a smooth approximation $A_{2^{-M}}k$ of an intermediate model. Then, the same correspondence relations found between $A_{2^{-M}}f$ and $A_{2^{-M}}g$ are employed for the interpolation between the detailed information of f and g to generate $P_{O_{2^j}}k$. Finally, the morphed model k is reconstructed from the smooth approximation $A_{2^{-M}}k$ and detailed information $P_{O_{2^j}}k$.

To establish the correspondence between the smooth approximation $A_{2^{-M}}f$ and $A_{2^{-M}}g$, a natural approach is to first reconstruct the smooth approximation of the functions from the discrete approximations $A_{2^{-M}}^d f$ and $A_{2^{-M}}^d g$ at resolution 2^{-M} back to the original resolution using the expansion of Equation 15 in 3D. Then, the correspondence problem can be solved at the original resolution. This approach can generate good results, but is computationally expensive because of the reconstruction process.

Another approach is to directly establish the correspondence between the discrete approximation $A_{2^{-M}}^d f$ and $A_{2^{-M}}^d g$ (see Figure 6). The reason why this method is reasonable is that the filter \bar{H} used in the decomposition can be seen as a low pass filter. Consequently, $A_{2^{-M}}^d f$ and $A_{2^{-M}}^d g$ can be interpreted as the representation of the original functions at a lower resolution, and the correspondence between $A_{2^{-M}}^d f$ and $A_{2^{-M}}^d g$ presents the relation between the general shape of the two objects. In addition, given the correspondence, since the scaling

functions $\psi(2^j t - n)$ are all the translations of a single function $\psi(2^j t)$, interpolation before or after reconstruction using Equation 15 would generate the same intermediate model. Unlike the first approach, there is no need to reconstruct the smooth approximation at the original resolution. In addition, since the time of solving the correspondence problem depends on the size of volume, it is much cheaper to establish the correspondence at a lower resolution.

As for the high frequency components, the discrete detail signals are interpolated at the same resolution and the same orientation using the same correspondence relation found between the smooth approximation. Again, the theoretical base of this approach is the spatial and frequency locality of the wavelet transform. To establish the correspondence between $D_{2^j}^i f$ and $D_{2^j}^i g$ when $j > M$, we treat the subvolumes with size of $(2^{j-M})^3$ as unit volume elements in $D_{2^j}^i f$ and $D_{2^j}^i g$.

Once we have the multiresolution representation, different interpolation schedules, similar to [3], are applied at different resolutions. The effect is the blending of the general shape of the two models, with the gradual removal of the high frequencies of the first model and the gradual appearance of the high frequencies of the second model. This is demonstrated in Figure 3b and Figure 7b, where schedules are designed so that finer details of the first model disappear faster than the coarser details, while the coarser details are blended in before the finer details for the second model.

An advantage of wavelet multiresolution representation is that the detailed information along seven different orientations is separately saved in $D_{2^j}^i f$ ($1 \leq i \leq 7$). Although not implemented yet, different schedules can be designed for the detail signals at the same resolution but with different orientations, since visual sensitivity depends not only on the frequency components but also on the orientation of the stimulus [7]. In summary, by designing different schedules, high frequency

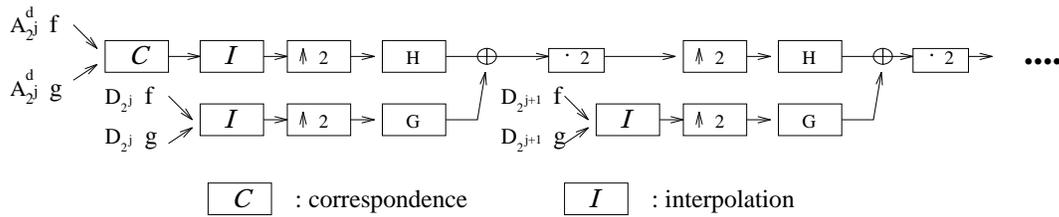


Figure 6: Wavelet Morphing Algorithm

distortion can be adjusted to the desired level (even magnified, if desired) and different morphing effects can be achieved.

Another flexibility of the wavelet-based morphing is the wide selection of wavelets that can be employed, since there is an infinite number of wavelets with different characteristics. In our implementation, the Battle-Lemarie wavelet is used for its symmetry and exponential decay.

5. Results and conclusions

We have presented a technique for performing volume morphing in the wavelet domain. The advantage of our method over a Fourier volume morphing [3] is that our approach not only effectively reduces high frequency distortion, but also establishes a suitable correspondence between the two volumetric datasets without the data modification process.

The wavelet-based morphing technique presented can be applied to sampled, simulated and modeled geometric datasets. Two sequences of morphing from a CT scanned lobster to an MRI head are illustrated in Figures 3a and 3b. In Figure 3a, volume morphing is performed directly in the spatial domain, while in Figure 3b our technique of wavelet-based volume morphing is applied to the datasets. The alleviation of high frequency distortion is most apparent during the middle stages of the animation in Figure 3b, where the morphing is performed mainly on the general shape of the models. Similarly, the comparison between spatial and wavelet-based morphing for geometric datasets is shown in Figures 7a and 7b, where in these two sequences a binary voxelized goblet is morphed into a binary voxelized torus. It is shown that in these two figures the correct topology is maintained during morphing. Furthermore, in Figure 7b our technique of wavelet-based morphing gradually alleviates the high frequency distortion caused by the aliasing existed in the original binary voxelized models. Binary voxelized models were used here just to demonstrate the effectiveness of the wavelet-based volume morphing. In practice, volume-sampled voxelized models [9] are used as original models, resulting in even smoother wavelet-based morphing.

The multiresolution representation of the volume can be explored for an adaptive morphing. We are currently building a "previewer" so that the morphing can be interactively performed at low resolutions. This kind of "preview" tool is very useful for adjusting morphing parameters in the interpolation schedules. By taking advantage of the spatial locality property of wavelet transform, subvolumes of the models can be selected for morphing. Thus, similar to 2D feature-based morphing [1], 3D features of one volume can be extracted and mapped to

the desired feature of another. We are currently developing a user interface to accomplish this task. Other future work includes the investigation of the wavelet selection for the specific morphing effect, and the design of interpolation schedules for the information along different orientations.

6. Acknowledgments

This work has been supported in part by the National Science Foundation under grant number CCR-9205047 and the Department of Energy under the PICS grant. The MRI head data for Figure 3 is courtesy of Siemens, Princeton, NJ, and the CT lobster data is courtesy of AVS, Waltham, MA. The authors would like to thank Steve Skiena for his suggestions on the correspondence problem.

7. References

1. Beier, T. and Neely, S., "Feature-based Image Metamorphosis", *Computer Graphics*, **26**, 2 (July 1992), 35-42.
2. Gortler, S. J., Schroder, P., Cohen, M. F. and Hanrahan, P., "Wavelet Radiosity", *SIGGRAPH '93*, August 1993, 221-230.
3. Hughes, J. F., "Scheduled Fourier Volume Morphing", *Computer Graphics*, **26**, 2 (July 1992), 43-46.
4. Kaufman, A., Cohen, D. and Yagel, R., "Volume Graphics", *IEEE Computer*, **26**, 7 (July 1993), 51-64.
5. Kaul, A. and Rossignac, J., "Solid-Interpolating Deformations: Construction and Animations of PIPs", *Proceeding of EUROGRAPHICS'91*, September 1991, 493-505.
6. Kent, J., Parent, R. and Carlson, W., "Establishing Correspondence by Topological Merging: A New Approach to 3-D Shape Transformation", *Proceedings of Graphics Interface '91*, June 1991, 271-278.
7. Mallat, S. G., "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **11**, 7 (July 1989), 674-693.
8. Muraki, S., "Volume Data and Wavelet Transform", *IEEE Computer Graphics and Applications*, **13**, 4 (July 1993), 50-56.
9. Wang, S. W. and Kaufman, A. E., "Volume Sampled Voxelization of Geometric Primitives", *Proceedings Visualization '93*, San Jose, CA, October 1993, 78-84.

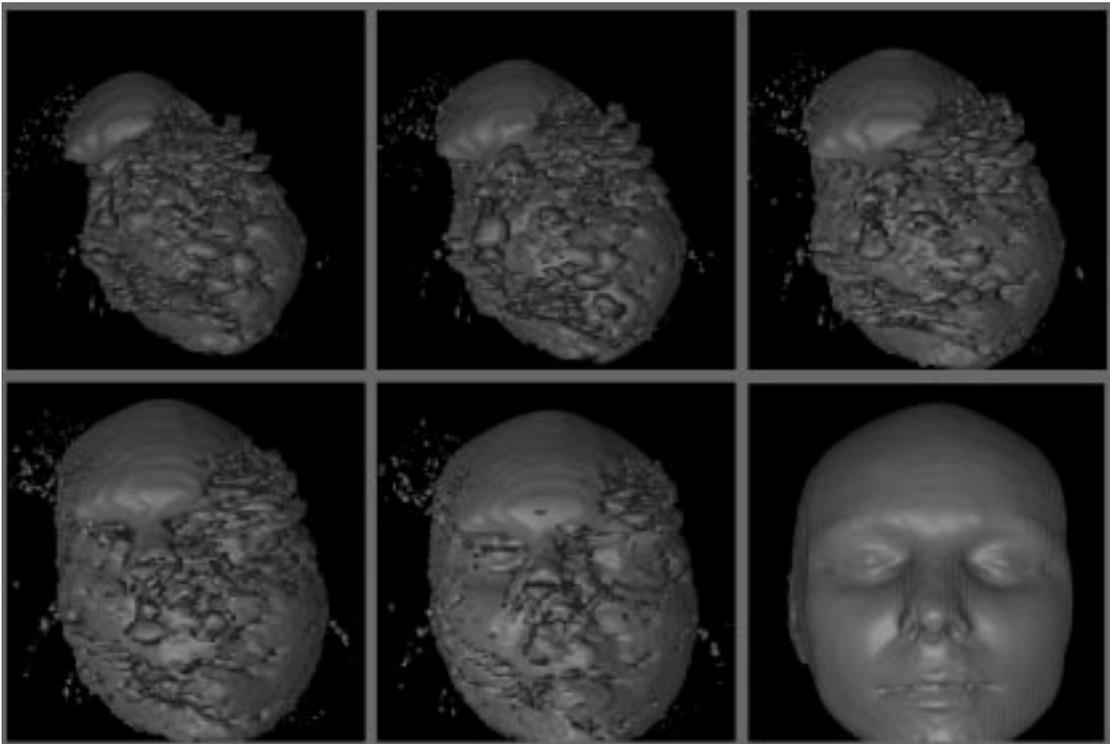
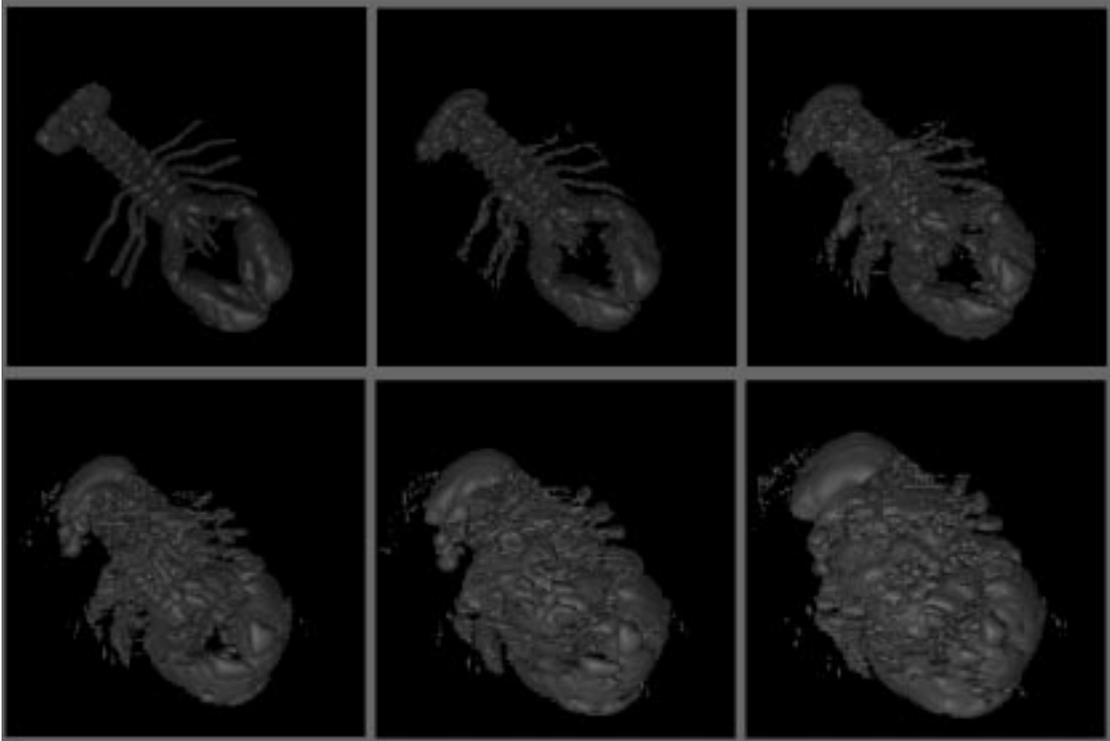


Figure 3a: Spatial Domain Volume Morphing from a CT Lobster to an MRI Head.

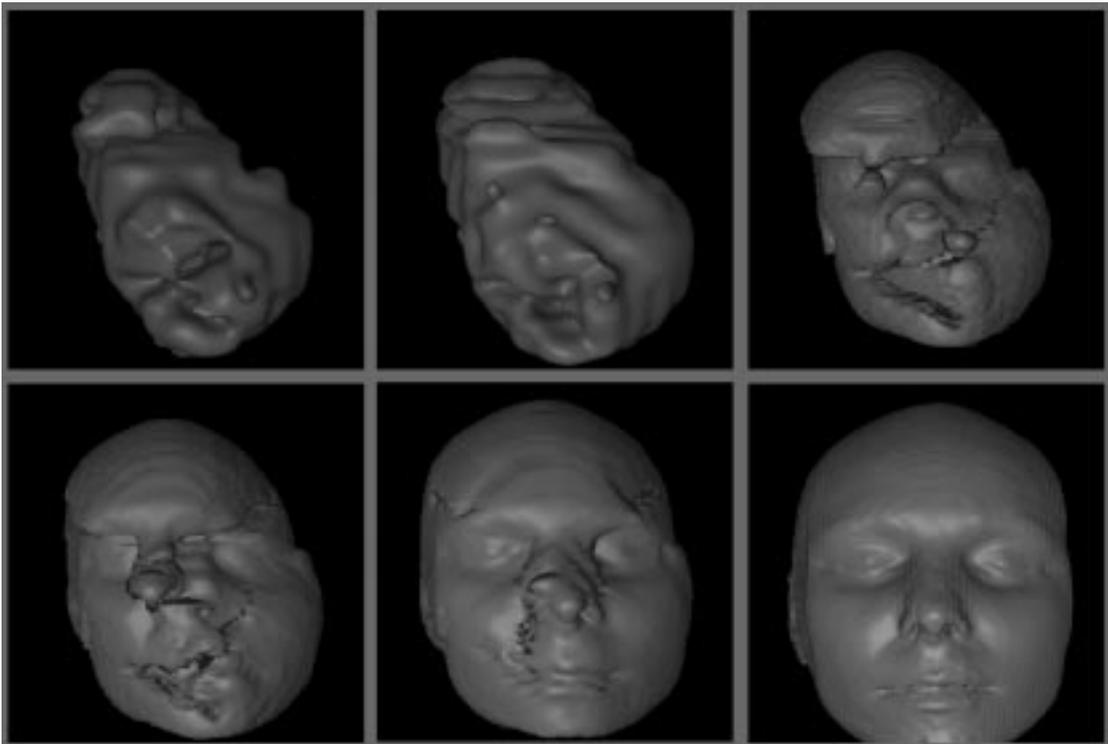
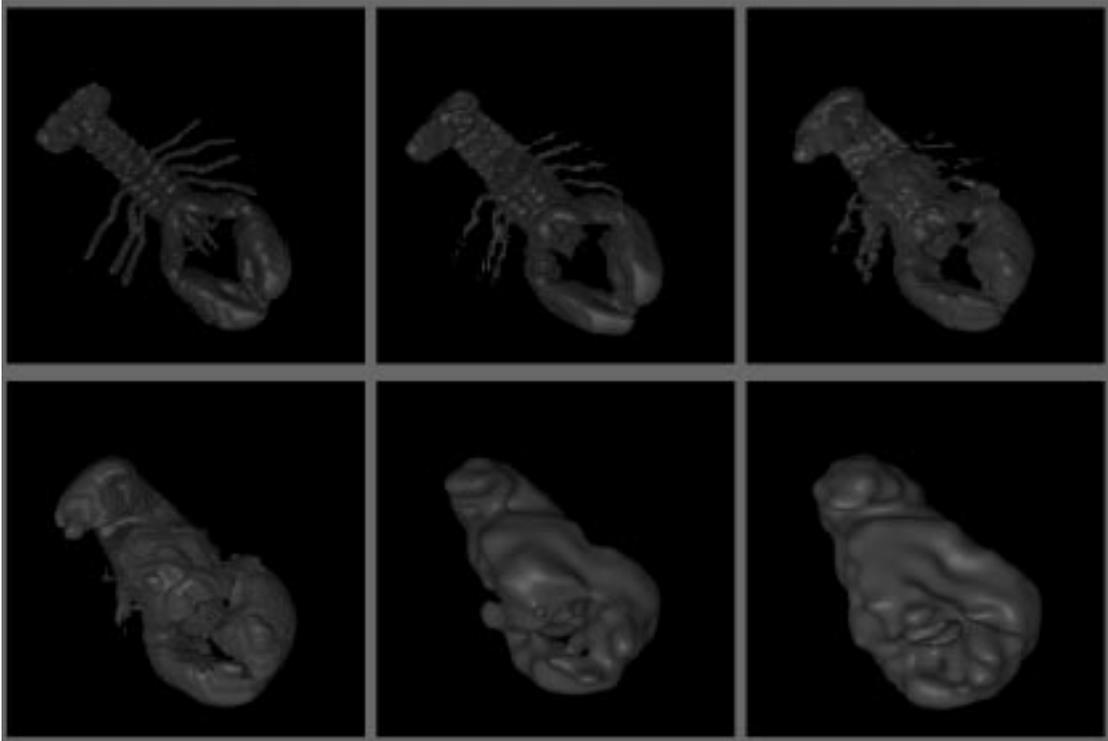


Figure 3b: Wavelet Domain Volume Morphing from a CT Lobster to an MRI Head.

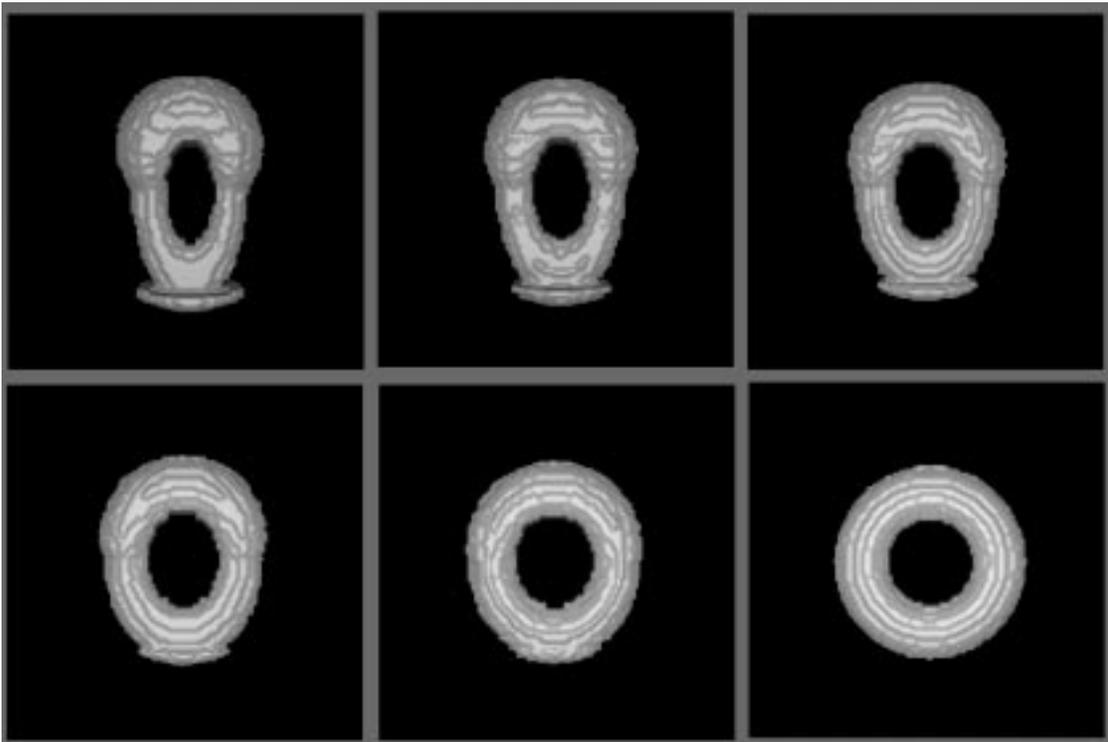
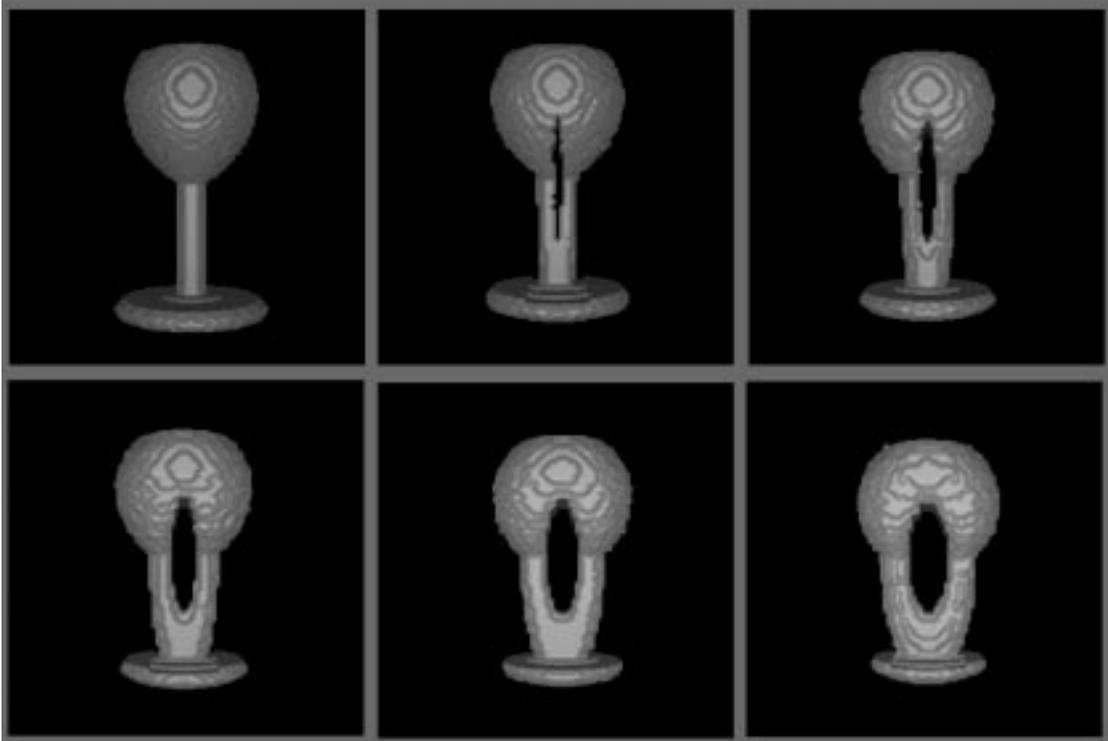


Figure 7a: Spatial Domain Volume Morphing from a Binary Voxelized Goblet to a Binary Voxelized Torus.

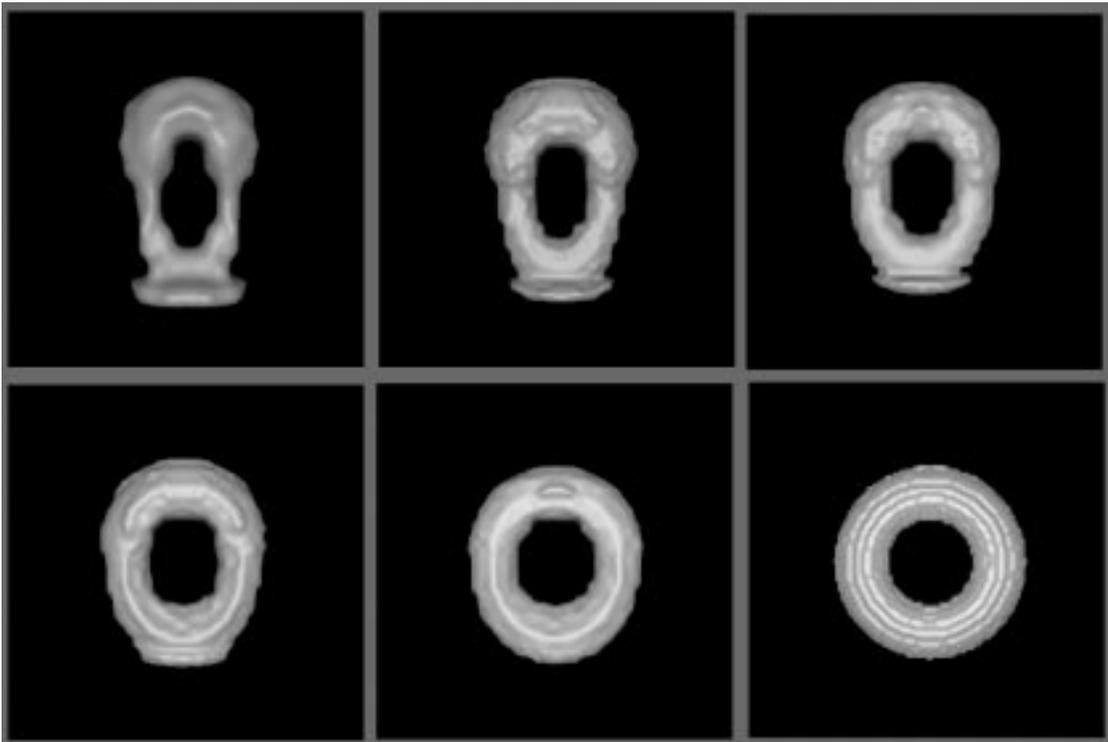
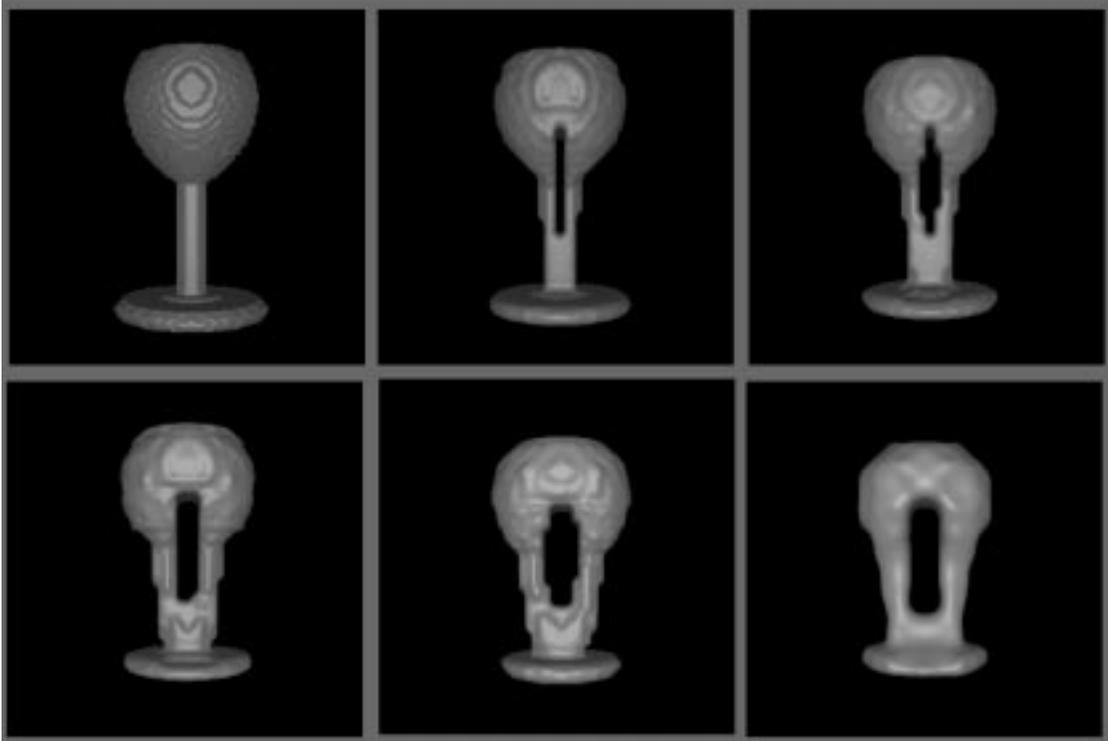


Figure 7b: Wavelet Domain Volume Morphing from a Binary Voxelized Goblet to a Binary Voxelized Torus.