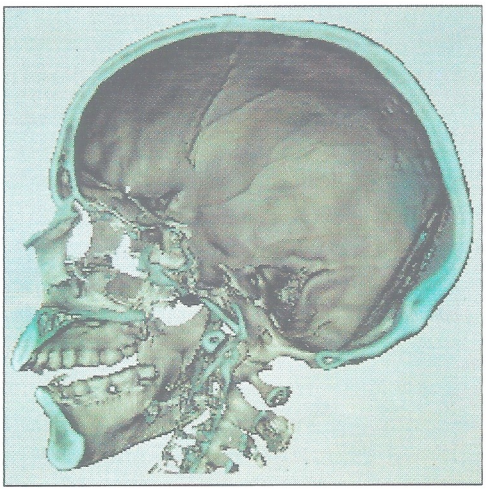
reprinted from
IEEE Computer Graphics and applications

Surface Rendering

Investigation of Medical 3D-Rendering Algorithms

Ulf Tiede, Karl Heinz Hoehne, Michael Bomans, Andreas Pommert, Martin Riemer, and Gunnar Wiebecke,
Institute of Mathematics and Computer Science in Medicine,
University Hospital Eppendorf, Hamburg, West Germany

Several algorithms help to visualize medical gray-level volumes. For clinical applications it is vital that the generated images represent reality. We have investigated the quality of surface-shading algorithms using computer-simulated test objects and we are presenting the results here. For the group of surface-shading algorithms, we tested *z*-buffer gradient, gray-level gradient, adaptive gray-level gradient, and marching cubes with two extensions. We will discuss transparent visualization using transparent gray-level gradient shading as an example. We found that gray-level gradient shading and marching cubes did not differ greatly, except for thin objects, where adaptive gray-level gradient shading was better. At present there is no way to assure the fidelity of transparent shading. Nevertheless it is useful when no other surface at all can be determined. Use of a combination of shading methods appears to yield the best visualization of the respective objects.



Computer tomography (CT) and magnetic resonance imaging (MRI) are the medical imaging modalities that deliver cross-sectional images of the human body. If we record sequences of adjacent images, we have a 3D volume representation of that part of the anatomy under consideration. Unlike 2D imaging, the presentation of gray-level volumes is subject to a number of choices. As those objects present in the volume may obscure each other, one has to decide which object to visualize. Furthermore, one can choose the form of presentation, for example, transparent, as cut planes or as surfaces. If one uses a surface visualization, there are different ways of determining the surface, the surface normals from the gray-level volume, and the subsequent shading.

Several methods for the visualization of objects from gray-level volumes have been described, including Goldwasser et al.,[1] Lenz et al.,[2] Hoehne and Bernstein,[3] Hoehne et al.,[4,5] Smith et al.,[6] Lorensen and Cline,[7] Levoy,[8] Drebin et al.,[9] and Hoehne et al.[10,24] The images are more or less pretty, but experience shows that despite all their beauty they may not re-

0272-17-16/90/0300-0041$01.00 ©1990 IEEE

flect reality. For clinical applications it is essential that one is sure about the fidelity of the pictures or at least understands its limitations. Except for studies by Tiede et al.,[11] Magnussen et al.,[12] and Pommert et al.,[13,14] no attempts have been made to compare the different methods and assess their capability of representing the "true" object. It is therefore the objective of this paper to assess the quality of different surface rendering algorithms using quantitative and qualitative measures of image quality.

Method

The raw data typically are a spatial sequence of image matrices of 256^2 pixels (in some cases compressed from 512^2 pixels) acquired through computer tomography or magnetic resonance imaging.

To save storage space the gray-level data of the original volume is compressed to a dynamic range of 256 gray values. To achieve cubic volume elements, a linear interpolation of the intensity values between the original slices is performed. This results in a 3D array (voxel model), which is the basic data structure for the algorithms we are describing. As an extension, each voxel may contain not only a gray value but such attributes as membership to an organ or an intensity value delivered by an additional imaging modality (generalized voxel model, Hoehne et al.[15]).

The principle of the visualization algorithm

The gray-scale volume is scanned using a ray-casting algorithm from the desired direction of view using the program system Voxel-Man (Hoehne et al.[5]). A gray value derived from the intensity profile encountered by each ray is projected onto an image plane, thus forming the projection image.

We investigated the following two kinds of projections:

1. A surface voxel is identified by an intensity threshold or by its attribute gained in a previous segmentation step ("binary segmentation").
2. Opacity is assigned to each voxel. A semitransparent presentation is then produced based on these opacities ("fuzzy segmentation").

Determination of image quality

The image quality in 3D visualization consists of the following three components:

- The precision of the object segmentation
- The precision of the surface normals
- The quality of shading.

Since object identification by binary segmentation is not a matter of computer graphics, we do not treat this subject here, but we refer the reader to Bomans et al.[16] The main computer-graphics issue is the correct determination of the surface normals from the original pictorial data. So we emphasize this in our study. Since the quality of the surface normals is generally worse than those generated in classical 3D graphics, no special efforts have been made for shading so far. This is certainly a subject for future study. In this article we used Phong shading with the following formula (note that parameters for the test objects are in parentheses).

$$S(x,y,z) = K_a + (K_d(\vec{L}\cdot\vec{N}) + K_s(\vec{R}\cdot\vec{V})^n)\,(1 - K\frac{z}{Z_{max}})$$

K_a = fraction of ambient reflection ($K_a = 0.0$)
K_d = fraction of diffuse reflection ($K_d = 0.8$)
K_s = fraction of specular reflection ($K_s = 0.2$)
K = depth factor ($K = 0.7$)
n = exponent for modeling highlights ($n = 5$)
$\vec{N}$ = surface normal vector at location x,y,z
$\vec{L}$ = normalized vector in direction of light source
$\vec{V}$ = normalized vector in direction of observer
$\vec{R}$ = normalized vector in direction of reflected ray
Z_{max} = maximum depth of 3D scene

Determination of the quality of images taken from living subjects generally suffers from the fact that we do not know the "truth" precisely. There are basically two ways to test image quality. One is the use of cadavers or real phantoms; the other is the computer simulation of test objects. Both approaches are used in our lab. If we use cadavers, precise measurements can only be performed for bone. The acquisition parameters cannot be changed at later times. In addition, a quantitative comparison cannot be performed. Even the qualitative comparison with pictures of the real object under exactly the same conditions is tedious. Therefore our first steps in quality control using artificial objects have advanced much further. For binary segmentation we chose the following procedure:

1. Tomographic acquisition of a formally specified test object is simulated. As in CT and MRI, sampling is done in coplanar slices with adjustable distance and thickness. This way one can simulate the sampling artifact known as "partial volume effect." Test objects may be any combinations of such simple solid geometric structures as spheres, cones, cubes, and with a specified (at present homogenous) gray value.

2. 3D-image processing is performed just as it is for real clinical data. Different methods of determining surface normals are tested. All results are visualized using the Phong illumination model.
3. Finally, the results are inspected qualitatively and compared with the original object specification using quantitative quality measures.

As a quantitative measure of the accuracy of surface-normal determination we use the angles between computed and ideal surface-normal vectors (in the range of 0 to 90 degrees). They can be visualized in an error image, thus presenting the local deviations. In the research reported in this article we used a test object which consisted of a cone and a pyramid intersecting each other and having a slit of 0.5 voxels. Effects concerning thin surfaces were studied with a ball-shaped test object with a wall thickness of only one voxel. Artificial objects have a drawback; not all facets of human anatomy can be modeled in detail using them. Decisive advantages, however, are the easy choice of acquisition parameters and the possibility of modeling special morphological structures (such as sutures or thin layers).

For fuzzy segmentation (transparent visualization) an exact object definition does not exist. Nevertheless we can achieve a reasonably good visual impression of the object. Yet an exact quality measure is hard to find. We can, however, visualize the contribution of the individual voxels to the final image via a "reflection map" (see Wiebecke[17]). Here, in a plane perpendicular to the projection plane, the percentage of the reflected incoming light is depicted. So we can at least assess qualitatively which objects contribute to the final image.

The quality of the algorithms to be compared depends to a large extent on the acquisition and preprocessing parameters of the original data. To avoid overcrowding of the parameter space, we have confined ourselves to a small set of typical parameters. For CT we chose both a slice thickness and a distance of 2 mm; for MRI the corresponding values were 1.5 mm. In contrast to other investigations, all data were taken from living persons. Trilinear interpolation was used unless stated otherwise. For the artificial objects an isotropic data volume was chosen.

We are aware that not all aspects of image quality are covered this way. However, the relative differences between the tested algorithms can be assessed very well. We are also aware that we have not covered all the published algorithms. Furthermore, essential details have been omitted in some publications, thus rendering analysis impossible. We think our choice of algorithms is rather representative. Above all, we have tried to describe or to reference the algorithms in such a way that they can be reproduced by the reader.

Rendering "binary" surfaces

As is well known, the raw image of a surface can be produced by imaging the negative distance of the visible voxels to the observer (depth cuing or z-buffer shading).

For realistic images, of course, the surface-normal vectors have to be determined. A depth factor may be used additionally to distinguish surface locations at different distances.

Z-buffer gradient shading (*ZG*)

A fairly fast method is estimation of the surface normals from the z-buffer.[18] We used a modified version of this algorithm.[11,20]

Given the z-buffer, $Z(x,y)$, the components of the surface normal are approximated from the gradient vector

$$\nabla z = \left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, 1 \right) = (\varphi, \theta, r)$$

The partial derivative $\frac{\partial z}{\partial x}$ is approximated as the weighted sum of the backward and forward differences of the neighboring pixels in the z-buffer.

$$\frac{\partial z}{\partial x} \approx W(|\delta_{r_x}|)\,\mathrm{Sign}(|\delta_{r_x}|) + W(|\delta_{f_x}|)\,\mathrm{Sign}(\delta_{f_x})$$

$$W(\delta) = \begin{cases} \frac{\pi}{4}\left(\frac{\delta - a}{b - a}\right), & \text{if } a \le \delta \le m \\ 0, & \text{otherwise} \end{cases}$$

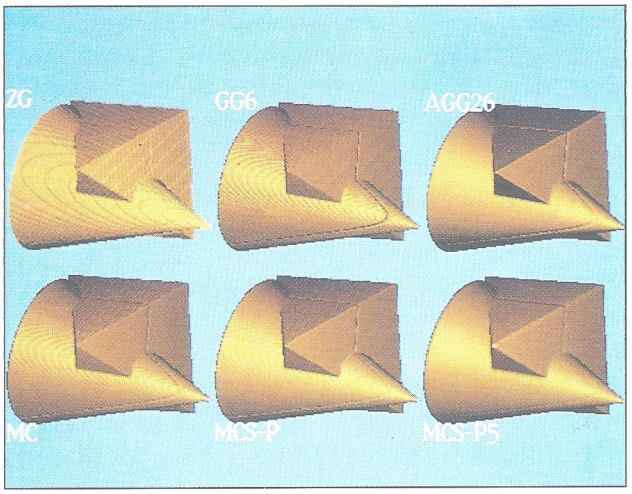
$$\delta_{r_x} = Z(x, y) - Z(x-1, y)$$

$$\delta_{f_x} = Z(x+1, y) - Z(x, y)$$

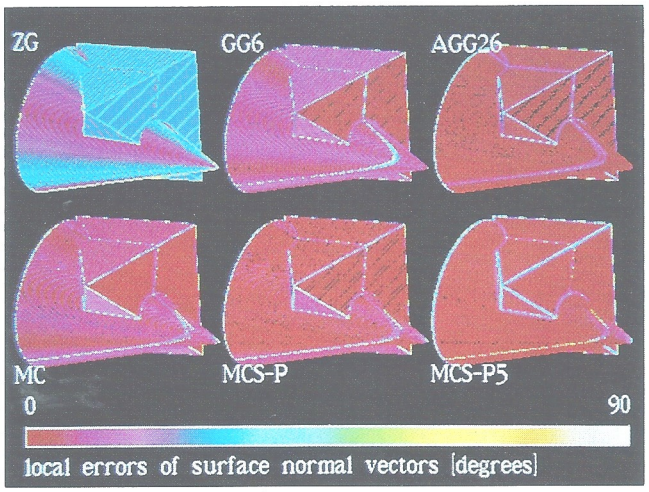
The computation of $\frac{\partial z}{\partial y}$, δ_{r_y}, and δ_{f_y} is analogous. The resulting spherical vector is converted to Cartesian coordinates. We use $a = 0$, $m = 8$, and $b = 12$.

This algorithm delivers fairly realistic images, but it suffers because the dynamic range and resolution of surface angles is low. This is because they are computed using the position of surface voxels in a 3×3 neighborhood, which does not allow a large number of choices.

Figure 1, case *ZG*, demonstrates the result for our test object. The object is rendered reasonably well, although the error image shows that the determined surface normals are fairly inaccurate. In fact, the visual impression is governed to a large extent by the

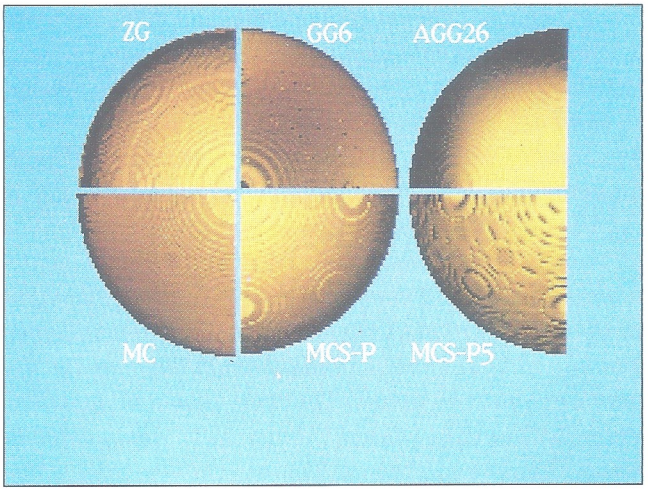


a

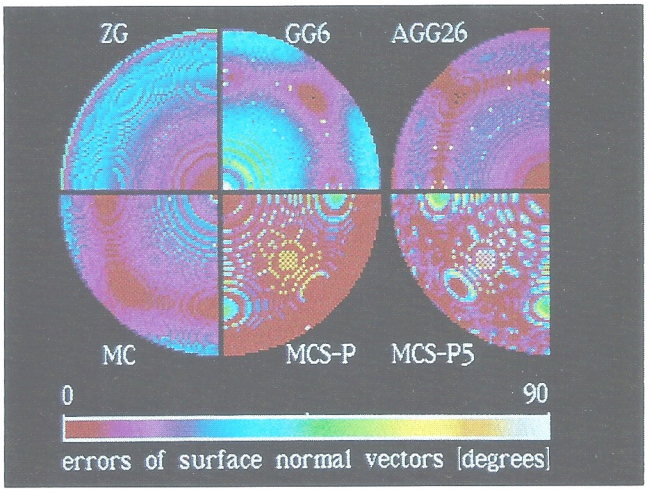


b

Figure 1. 3D views (a) and the corresponding error images (b) of a solid test object with a small gap. ZG—Z-buffer gradient shading; (GG6) gray-level gradient shading with six neighbors; (AGG26) adaptive gray-level gradient shading with up to 26 neighbors. Second row: marching cube segmentation. (MC) original gray level gradient shading; (MCS-P) Phong interpolation; (MCS-P5) Phong interpolation with five iterations.)



a



b

Figure 2. 3D-views (a) and the corresponding error images (b) of a spherical test object of one-voxel thickness. (ZG) Z-buffer gradient shading; (GG6) gray level gradient shading with six neighbors; (AGG26) adaptive gray-level gradient shading with up to 26 neighbors. Second row: marching cube segmentation. (MC) original gray level gradient shading; (MCS-P) Phong interpolation; (MCS-P5) Phong interpolation with five iterations.)

depth contribution. The slit is not detected. The same is true for the thin test object (Figure 2). If we consider an anatomical object rendered this way (Figure 3), we notice the low dynamic range and the discreteness of the surface. The eye socket, which is critical because of its thin bone, is presented quite well. Sutures are not recognizable.

Gray-level gradient shading (GG)

Much better results can be achieved if the gray-scale data are used for determination of the surface normals. As a consequence of the tomographic data-acquisition technique the gray values in the neighborhood of a surface voxel reflect the relative average of the various (usually two) tissue types (for

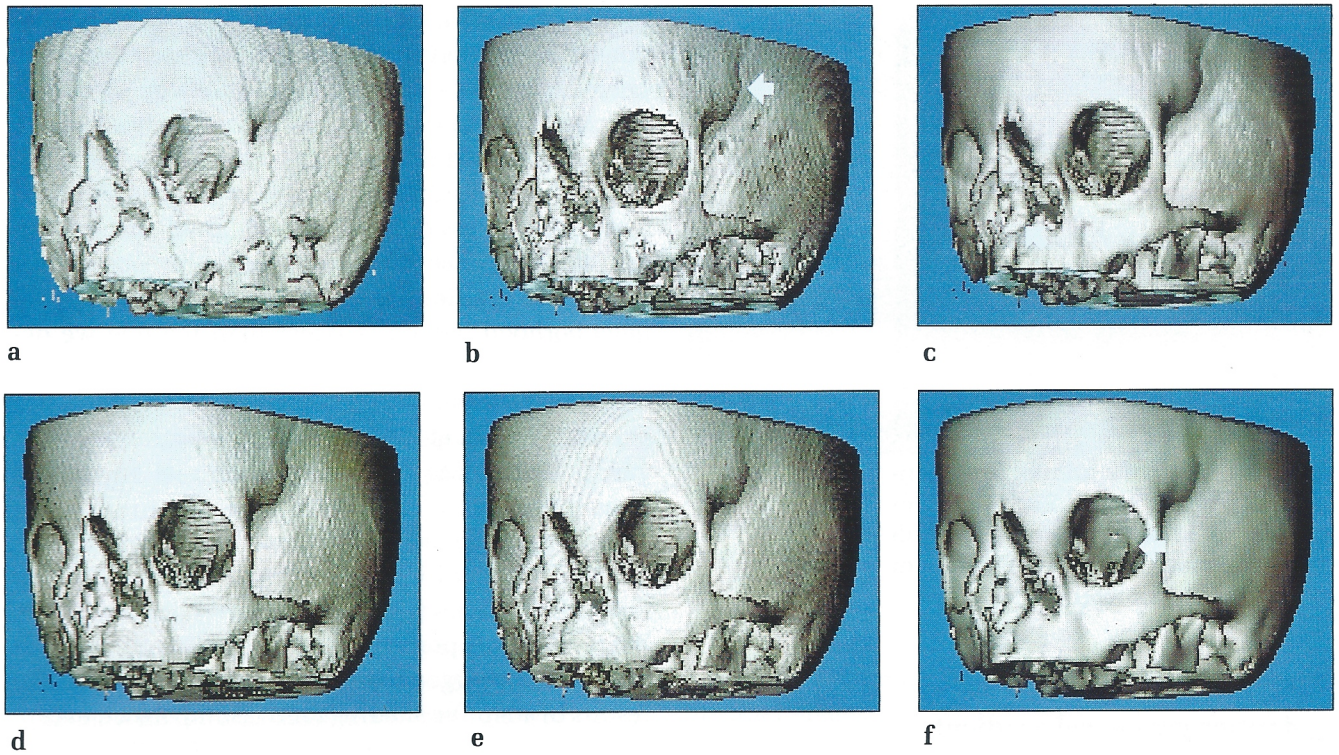


Figure 3. View of the skull of a three-month-old child with very thin bones and a facial cleft reconstructed from 46 computer tomograms. (a) *Z*-buffer gradient shading, (b) gray-level gradient shading with six neighbors, (c) adaptive gray-level gradient shading with up to 26 neighbors. Bottom row: marching cube segmentation. (d) original gray-level gradient shading, (e) Phong interpolation, (f) Phong interpolation with five iterations.

example, air/skin, soft tissue/bone) in the voxels immediately adjacent to the voxel in question. These relative volumes are related to the surface inclination. Thus the gray-level gradient can be considered a measure of surface inclination. This idea of gray-level gradient shading has been described by Barillot et al.,[21] and independently by Hoehne and Bernstein[3] and Tiede et al.[11,19] Our procedure is as follows:

Given the gray level g of a surface voxel at a location, i,j,k the gray-level gradient is computed as

$$G_x = g_{(i+1,j,k)} - g_{(i-1,j,k)}$$
$$G_y = g_{(i,j+1,k)} - g_{(i,j-1,k)}$$
$$G_z = g_{(i,j,k+1)} - g_{(i,j,k-1)}$$

The components of the surface normals are normalized as

$$N_u = \frac{G_u}{\sqrt{(G_x)^2 + (G_y)^2 + (G_z)^2}}, \quad u = x, y, z$$

The gradients are typically computed either from six central neighbors in the 3×3×3 neighborhood or from all 26 neighbors. In the latter case the algorithm is identical to the Zucker-Hummel operator for edge detection (see Ballard and Brown[22]).

Due to the high dynamic range of the gray levels a continuous shading is now possible, leading to a more realistic impression of the objects. Small details not visible with the z-buffer gradient shading such as the slit in the solid artificial object (Figure 1 case GG6) become apparent, even if only six neighbors are used. The thin surface test object, however, shows considerable error (see Figure 2).

In solid regions of real objects such details as sutures now become visible (see Figure 3). In addition, aliasing effects present in z-buffer gradient shading, especially during animation, do not occur. For the thin bone in the eye socket or the nasal septum and parts of the lateral head, this method leads to artifacts. This is because the gray level is no longer governed by membership in an object class but rather by the thickness of the object. As a consequence the gray-level threshold is not representative for the surface nor is the gradient representative for its inclination. The inclusion of 26 neighbors improves the quality of the solid regions, but does not show thin objects better.

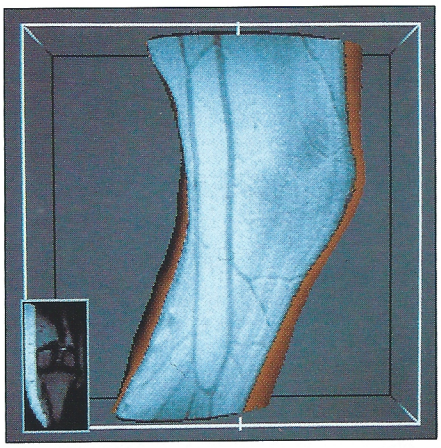


Figure 4. View of a human knee generated from a set of 26 MR tomograms. A "look through" projection is performed for a layer of 2 cm below the skin surface.

Adaptive gray-level gradient shading (AGG)

For thin objects even a six-voxel neighborhood may be too large. Therefore, we tried an adaptive selection of the neighborhood (see Pommert et al.[13]). An algorithm, which adapts to the thickness of the object using only 3 to 6 adjacent voxels, computes G_x (G_y and G_z correspondingly) in the following way: If the gray value of the surface voxel at (i,j,k) is greater (smaller) than the gray value of its neighbors at $(i-1,j,k)$ and $(i+1,j,k)$, it takes the gradient between the surface-voxel itself and the neighbor with the lower (higher) gray value; otherwise it takes the gradient between both neighbors. This principle can be generalized by choosing the neighborhood from all 26 neighbors. For solid test objects the effect is the same as in the non-adaptive case (see Figure 1, case AGG26). One obtains a smooth, nearly alias-free surface with minimal deviation from reality. For the thin ball the error is greatly reduced (see Figure 2). For a real object, the algorithm yields improved results in the critical regions (see Figure 3).

Marching cubes algorithm (MC)

An algorithm that combines classical surface-representation—via triangles—and surface-normal determination—via gray-level gradients—has been described by Lorensen and Cline.[7] They consider an octuple of neighboring voxels representing a cube. Depending on whether the voxels are within or outside the object, a surface of up to four triangles is placed inside the cube. The surface normals at the triangle vertices are computed via a linear interpolation of the gray-level gradients at the cube edges. The polygonal-surface description can then be shaded in the classical way (for example, using Phong shading). As demonstrated in Figure 1b, case MC, the solid test object shows results comparable to those of the gray-level gradient shading. Obviously, problems arise at thin surfaces (see Figure 2), probably because the gray-level gradient is calculated from a $4\times4\times4$ neighborhood, which is larger than the one we described before. Further, non-closed surfaces may occur if the voxel cubes belonging to an object are connected only at one edge. The appearance of the gray-level shading is similar to the real object (see Figure 3).

The neighborhood for gradient determination is reduced if we modify the algorithm so the surface segmentation is done with the marching cube algorithm, but rendering is performed using the Phong interpolation of the polygonal surface. The result looks like results of adaptive shading for the solid object, except at sharp edges (see Figure 1, case MCS-P). For thin surfaces the problem of non-closed surfaces becomes more prominent (Figure 2). However, these artifacts are not visible in the real object (Figure 3).

The modified version can now be extended by applying the original Phong interpolation scheme repetitively, so an even larger neighborhood of adjacent polygons are used in the surface-normal computation (see Tiede[20]). The errors at sharp edges are enhanced, while homogenous areas are presented with nearly no error (Figure 1 case MCS-P5). Though the iterative Phong interpolation produces strong local deviations for thin objects (see Figure 2), the appearance of real objects, especially the eye socket, becomes significantly smoother. On the other hand, the suture is almost smoothed out (Figure 3).

"Fuzzy" surface rendering

There is no doubt that an object that is opaque in reality should be rendered as an opaque object to get the highest fidelity. Exceptions that justify transparent rendering are cases where we really want to look into an object that does not have distinguishable constituents or when we are not able to determine the true surfaces automatically. The latter is often true if we want to visualize soft tissue structures in detail.

"Look through" projections

The simplest way of looking into an object is to assign a certain opacity to each gray level and to integrate over the opacities. If applied in a straight for-

ward manner, this is a step back to the old X-ray projection technique. If applied in a selective way, it is nevertheless helpful in special cases. As an example, vessels below the skin surface can be visualized by just integrating over a certain range below the surface (Figure 4), thus the subcutaneous vessels become visible selectively, the remaining skin delivering a 3D clue. It is, of course, hard to give a measure for the fidelity of these images.

Transparent gray-level gradient shading (TGG)

More sophisticated ways of transparent rendering have been described by Levoy[8] and Drebin et al.[9] These algorithms have the following features in common:

1. Certain opacities are assigned to certain gray-level ranges (for example, bone and muscle).
2. Opacities at surfaces are enhanced by weighting them with the gray-level gradient.
3. Using a ray-casting algorithm, transparent views are produced by shading all visible voxels using the gray-level gradient method.

We used an algorithm that determines the intensity in the image plane $I(x,y)$ using the following recursive equations:

$$I(x,y) = F(x,y,0,1)$$

$$F(x,y,z,l) =$$

$$\begin{cases} l \cdot \text{background value}, & \text{if } l \leq t \text{ or } z \geq z_{max} \\ l \cdot O(x,y,z) \cdot S(x,y,z) + F(x,y,z+1, l \cdot (1 - O(x,y,z))), & \text{otherwise} \end{cases}$$

t = minimum remaining fraction of the incoming light
z_{max} = maximum depth of 3D scene
l = fraction of incoming light
$O(x,y,z) = W(g_{(x,y,z)}) \cdot G(x,y,z)$ – opacity
$W(g_{(x,y,z)})$ = weight assigned to the gray level g
$G(x,y,z) = \sqrt{Gx^2 + Gy^2 + Gz^2} / \sqrt{255^2 + 255^2 + 255^2}$ = magnitude of gray level gradient for six neighbors
$S(x,y,z)$ = surface shading component (see section on determination of image quality)

The resulting image depends strongly on the weighting function W. As a tool for assessing the choice of parameters we compute an auxiliary image $R = l \cdot O(x, y, z)$, which we call reflection map. This map shows the relative contribution of the voxels to the final image.

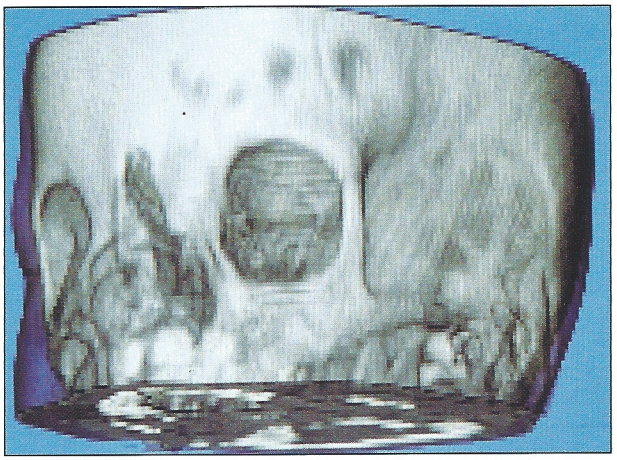


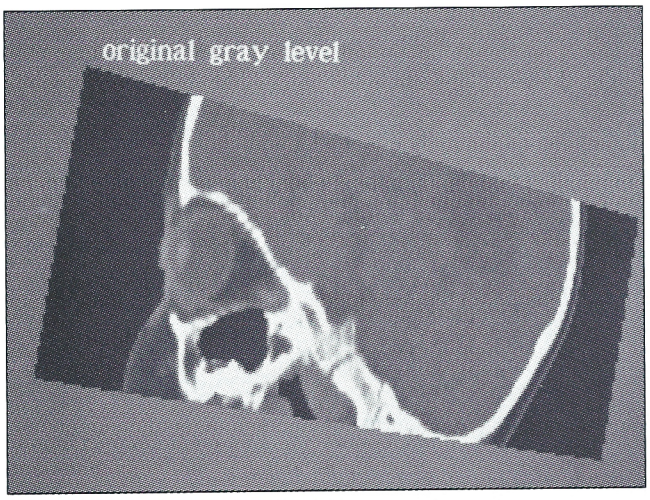
Figure 5. The skull as in Figure. 3 rendered with transparent gray-level gradient shading (TGG).

In contrast to the algorithm originally proposed by Levoy,[8] our algorithm traverses the volume in a front-to-back manner and allows use of a stopping mechanism. Thus, computing time can be greatly reduced.

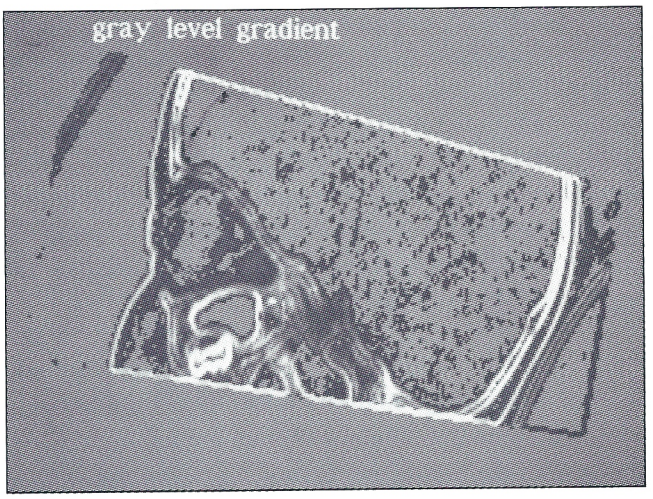
We have applied this method extensively to CT, as well as to MRI volumes. The motivation for CT is that possibly critical regions (thin bone, etc.) could be visualized better, as claimed by Levoy.[8] As an example, Figure 5 shows the skull shown in Figure 3 with the skin nearly transparent and the bone semitransparent. In fact, the bone looks good and the critical eye socket looks smoother than it does with binary segmentation (Figure 5b). If, however, we look at the reflection map (Figure 6), we notice, especially in the eye socket, that voxels from all over the object are contributing to the image.

One can object that the weighting function could have been chosen more properly. This is possibly true, but our experience is that various parameter settings deliver good-looking but different images, and there is no algorithmic way of determining the "correct" parameter set. We are experimenting with semiautomatic procedures. Compared to binary segmentation, the computation time is much higher, because, depending on the opacity, the number of voxels to be rendered may become very large. Therefore, we do not recommend transparent rendering for objects that are segmentable, such as bone in CT.

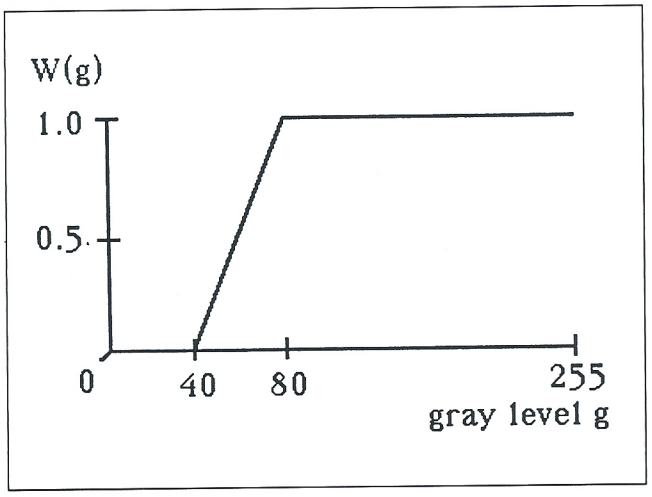
The situation is somewhat different in cases where there is no way to segment the objects automatically, as is often true with MRI data. As an example, we tried to visualize a 4D MRI data set of a human heart con-



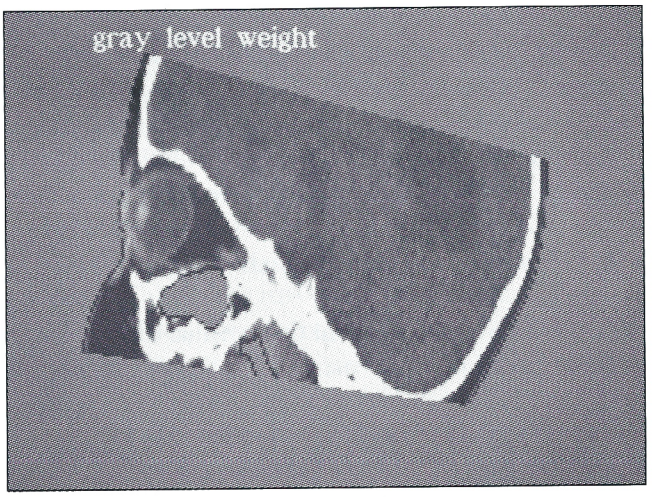
a



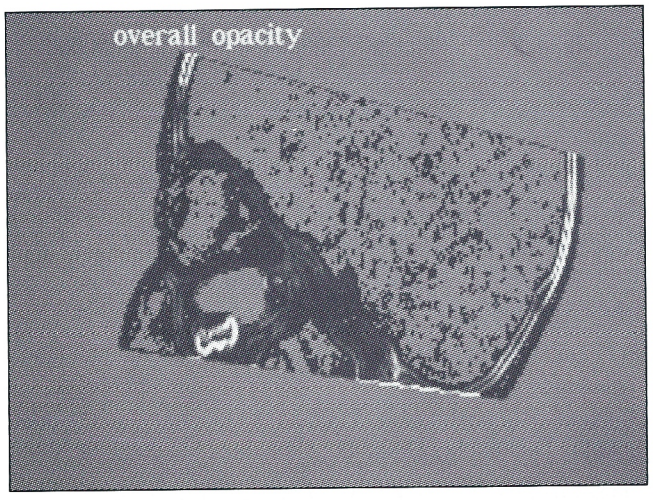
b



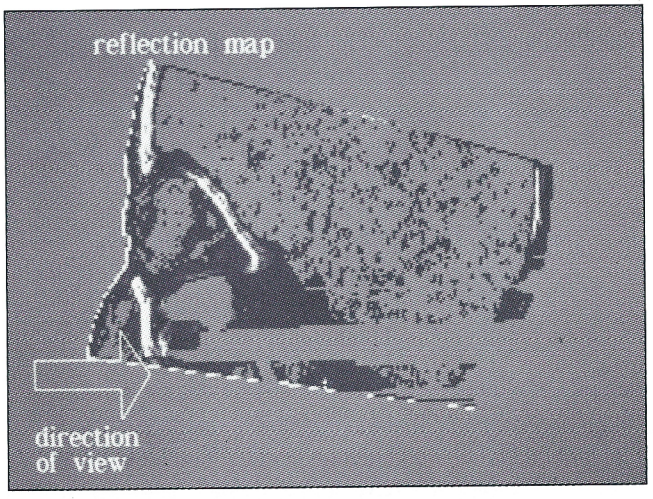
c



d

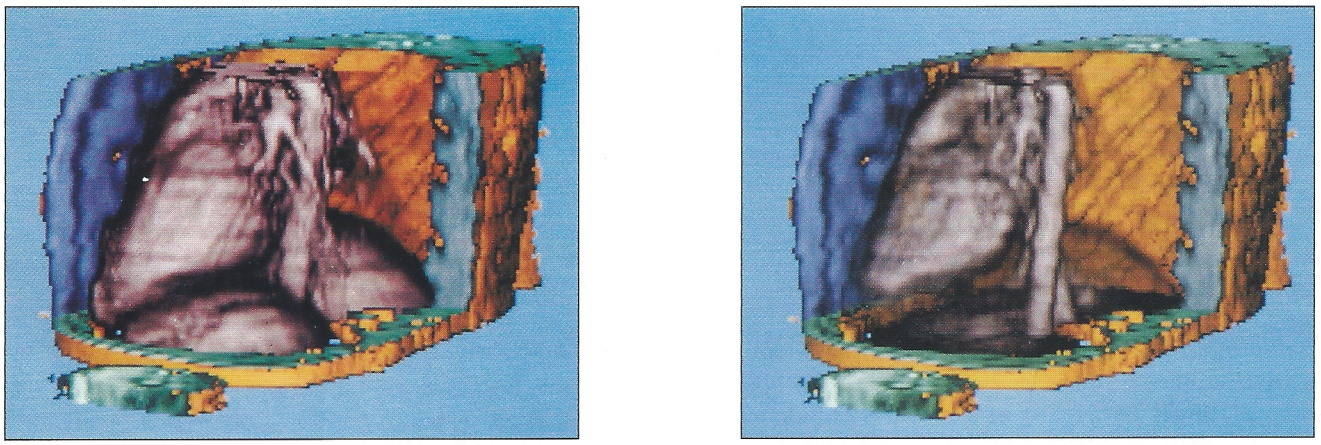


e



f

Figure 6. Parameters leading to the image in Figure 5, shown on a cross-sectional image. (a) original CT values (b) gray-level gradient $G(x,y,z)$, (c) weighting function $W(g)$, (d) weighted CT values. (e) overall opacity $O(x,y,z)$, (f) reflection map R showing the relative contribution of the voxels to the final image.)



a b

Figure 7. View of a human heart generated from 26 MR images using transparent gray-level gradient shading. (a) Emphasis on outer heart surface and vascular spaces. (b) Emphasis on blood pool.

sisting of 26 slices for each of 12 heart phases. Binary segmentation was of no value.

Using transparent gray-level gradient shading, we can obtain a fairly good visualization (see Figure 7). The contributions to the final image can be assessed by viewing the reflection map (see Figure 8). This way animated views of the heart or the vascular spaces have been generated from a living patient. These parts could never be seen before, even during surgery (see Hoehne et al[23]). Though a real contribution, the opacities derived from the gray levels have to be chosen carefully. This process is helped by showing their strength on a cross-sectional map.

Maximum intensity projection

If the objects to be presented become so tiny that neither a segmentation nor the determination of a surface normal is possible, the maximum intensity projection delivers reasonable results in special cases. For example, it is possible to produce image volumes in magnetic resonance imaging where the blood vessels are enhanced in intensity. If we record the maximum of the intensity along the rays as an image, we get a kind of projection image that has no depth cue per se. With motion and/or implantation into a surface image (see Figure 9) a 3D impression can be achieved.

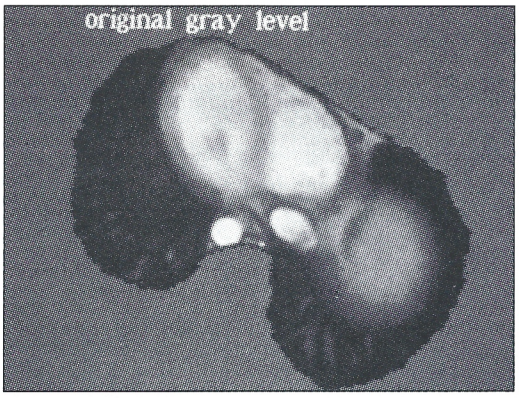
Conclusions

We have tried to assess different methods of visualizing tomographic volume data. For those cases where we can determine surfaces of the objects contained in a volume (binary segmentation), we have compared several rendering methods. *Z*-buffer gradient shading is easy to implement, and it is fast. However, its ability to reflect reality is poor. Nevertheless, it still has its justifications, for example, as a tool for fast previewing, or when no gray-level gradient can be calculated, as is the case with manually segmented objects, for instance.

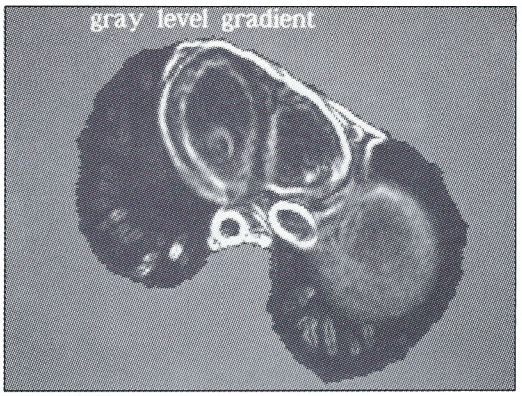
Quality is—at least for CT-data—much higher in gray-level gradient shading. The ability to visualize thin surfaces can be greatly enhanced if the gradients are computed according to the thickness of the objects (adaptive gray-level shading). The marching-cube algorithm delivers equally good quality, but it has problems with thin objects, and it is more expensive to compute. One of its obvious merits lies in its ability to connect the voxel world with the polygon world. The combination of marching-cube segmentation and Phong interpolation generates better results for real objects, but the problems with thin surfaces and sharp edges remain unsolved.

It was our experience that the attempt to overcome segmentation problems in critical regions with semi-transparent visualizations does not seem successful. If, however, a surface cannot be determined at all, transparent visualization is the only possibility. Here, the transparent gray-level gradient shading delivers useful results. Since we have the free choice of assigning opacities to the gray levels and/or the gray-level gradients, we cannot be sure about the fidelity of the images. The comparison of an original gray-level picture with the reflection map is helpful in determining an optimal parameter set.

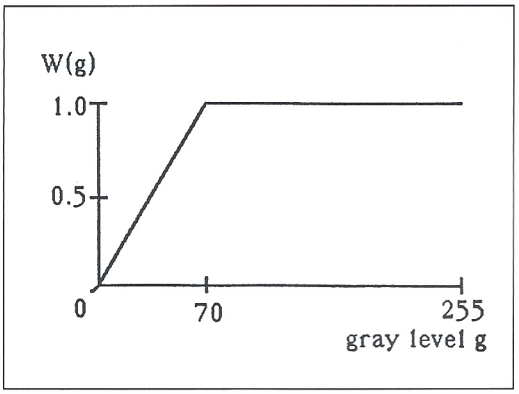
The question of whether an opaque or a transparent visualization of tomographic volumes is better can be answered as follows: Whenever surfaces are detectable, an opaque visualization is better, because it is



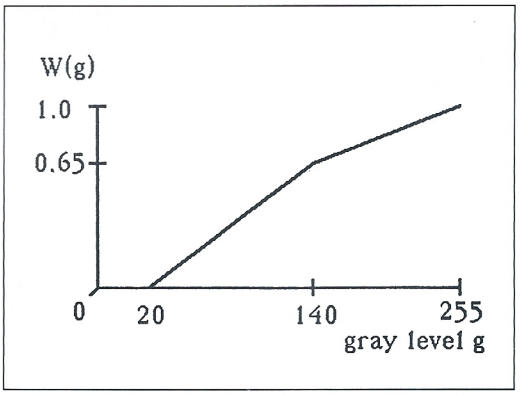
a



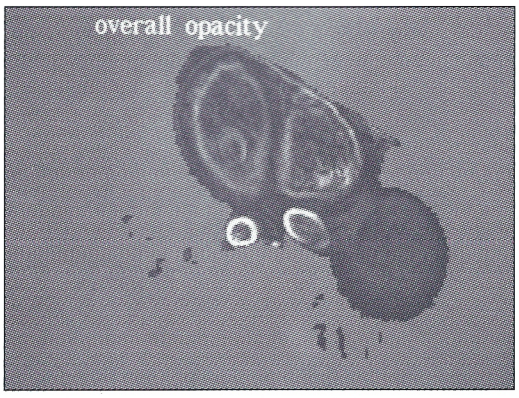
b



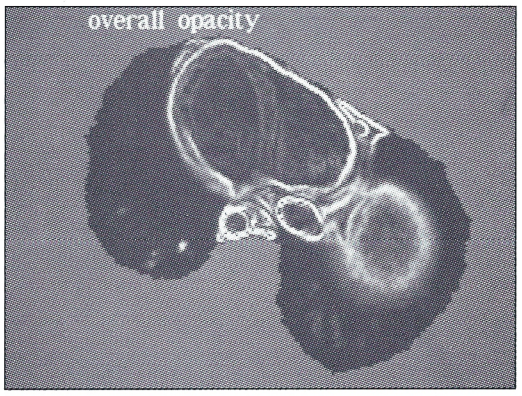
c



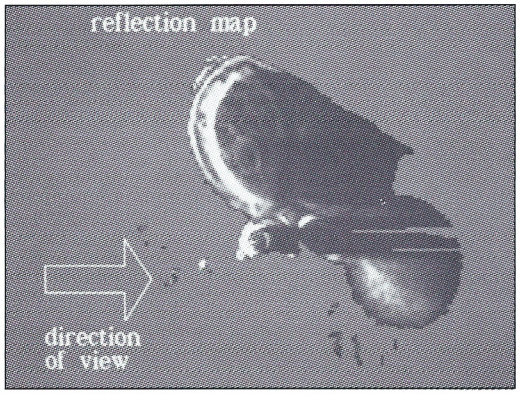
d



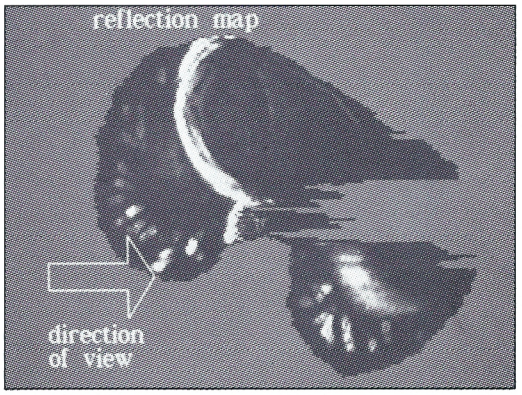
e



f



g



h

Figure 8. Parameters leading to the images in Figure. 7, shown on a cross-sectional image (left column corresponds to Figure. 7 left, right column to Figure. 7 right). (a) original MR values, (b) gray-level gradient $G(x,y,z)$, (c and d) Weighting functions $W(g)$; (e and f) Overall opacity $O(x,y,z)$; (g and h) Reflection maps R, showing the relative contribution of the voxels to the final image.

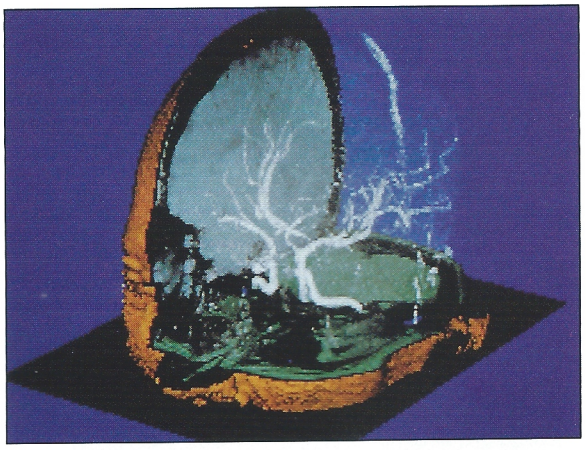


Figure 9. View of a human head generated from 120 MR tomograms. The blood vessels are made with maximum-intensity projection. The skin is made using gray-level gradient shading.

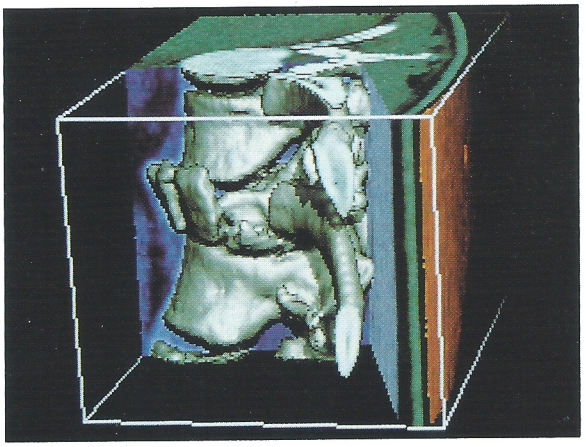


Figure 10. This view of a human spine with a broken vertebra was generated from 28 computer tomograms. The surfaces were shaded using gray-level gradient shading.

perceived better (decisively in static images). For further improvement, different rendering methods can be used selectively for different objects (see Figures 10 and 11). In addition, a visualization by multiple surfaces and cuts is very well suited to locating and measuring objects. Transparent visualization lacks this property, which is critical for clinical practice. And shouldn't there be a reason Leonardo da Vinci, who was an expert on human anatomy and a genius in visualization, did not use transparency (see Figure 12)? ■

Acknowledgments

We thank Prof. W.J. Holtje (Department of Craniofacial Surgery) and Dr. Jurgen Wening (Department of Traumatology) for many discussions. We are grateful to Ellen Vaske, Bernhard Pflesser, and Rainer Schubert for their support and to Renate Muller for her patient help in editing the paper. The MRI volume data sets have kindly been provided by Siemens Erlangen. The pictures of the mummy were produced in cooperation with the Department of Anthropology and the Department of Neuroradiology, University of Tubingen.

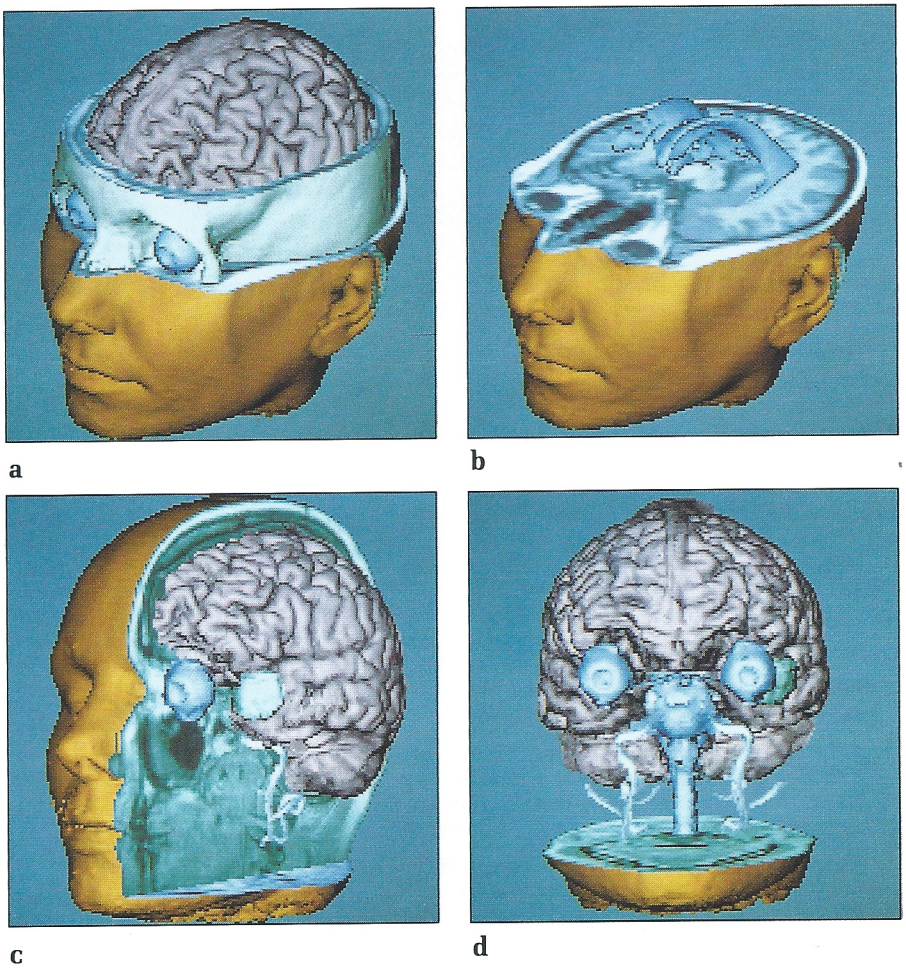


Figure 11. Different views of a head generated from 128 MR images. The skin, bone, ventricle (blue), and tumor (green) are shaded using gray-level gradient shading. The brain is rendered with transparent gray-level gradient shading, and a "look-through" projection of 3 voxels is used for the carotid artery (white).

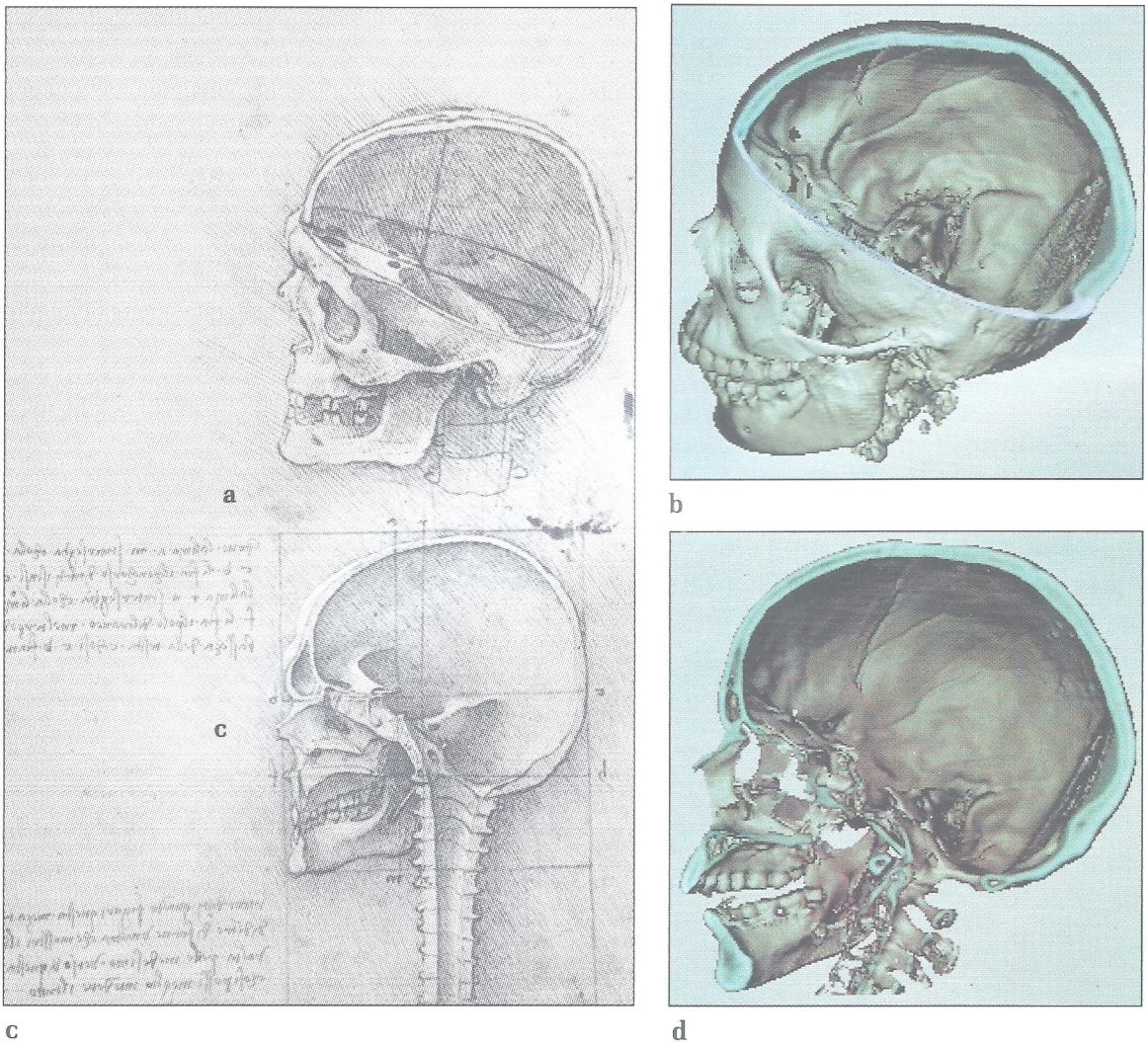


Figure 12. Leonardo da Vinci's famous drawings of a human head, together with analogous views of a 2,600-year-old mummy generated from 200 computer tomograms using gray-level gradient shading.

References

1. S.M. Goldwasser et al., "Physicians Workstation with Real-Time Performance," *CG&A*, Vol. 5, No. 9, Sept. 1985, pp. 44-57.
2. R. Lenz et al., "Presentation and Perception of 3D Images" in K.H. Hoehne, ed., *Pictorial Information Systems in Medicine*, NATO ASISeries F: Computer and Systems Science, Vol. 19,Springer-Verlag, Berlin, 1986, pp. 459-468.
3. K.H. Hoehne and R. Bernstein, "Shading 3D Images from CT Using Gray Level Gradients," *IEEE Trans. Medical Imaging*, Vol. MI-5, No. 1, March, 1986, CS Press, Los Alamitos, Calif., pp. 45-47.
4. K.H. Hoehne et al., "Combined Surface Display and Reformatting for the 3D-Analysis of Tomographic Data," *Investigative Radiology*, Vol. 22, No. 7, July, 1987, pp. 658-664.
5. K.H. Hoehne, M. Riemer, U. Tiede, "Viewing Operations for 3D-Tomographic Gray Level Data," in H.U. Lemke et al., eds., Computer Assisted Radiology (*Proc. CAR*), Springer-Verlag, Berlin, 1987, pp. 599-609.
6. W.K. Smith, D.S. Schlusselberg, D.J. Woodward, "Algorithms for Ray Tracing Volumetric Representations of Medical Image Data," in H.U. Lemke et al., eds., *Computer Assisted Radiology*, (Proc. CAR), Springer-Verlag, Berlin, 1987, pp. 882-887.
7. W.E. Lorensen and H.E. Cline, "Marching Cubes: A High Resolution 3D Surface Construction Algorithm," *Computer Graphics* (Proc. SIGGRAPH), Vol. 21, No. 3, July 1987, pp. 163-169.
8. M. Levoy, "Display of Surface from Volume Data," CG&A, Vol. 8, No. 3, May, 1988, pp. 29-37.
9. R.A. Drebin, L. Carpenter, and P. Hanrahan, "Volume Rendering," Computer Graphics (Proc. SIGGRAPH), Vol. 22, No. 3, Aug. 1988), pp. 65-74.
10. K.H. Hoehne et al., 3D Visualization of Tomographic Volume Data Using the Generalized Voxel Model," *Proc. Chapel Hill Wkshp. Volume Visualization*, Univ. of North Carolina, Chapel Hill, N.C., 1989, pp. 51-57.

11. U. Tiede, K.H. Hoehne, and M. Riemer, "Comparison of Surface Rendering Techniques for 3D Tomographic Objects," in U. Lemke, ed., *Computer Assisted Radiology*, (Proc. CAR), Springer-Verlag, Berlin, 1987, pp. 599-610.

12. M. Magnusson, R. Lenz, and P.E. Danielsson, "Evaluation of Methods for Shaded Surface Display of CT Volumes," *Proc. 9th ICPR*, 1988, pp. 1287-1294.

13. A. Pommert et al., "Image Quality in Voxel-Based Surface Shading," in H.U. Lemke, eds., *Computer Assisted Radiology* (Proc. CAR), Springer-Verlag, Berlin, 1989, pp. 737-741.

14. A. Pommert et al., "Simulation Studies for Quality Assurance of 3D Images from Computed Tomograms," in Todd-Pokropek and M.A. Viergever, eds., "The Formation, Handling and Evaluation of Medical Images," NATO ASI Series F, *Computer and Systems Sciences*, Springer-Verlag, Berlin, 1990, in press.

15. K.H. Hoehne et al., Display of Multiple 3D Objects Using the Generalized Voxel Model, *Proc SPIE 914*, SPIE, Bellingham, Wash., pp. 850-854.

16. M. Bomans et al., "3D Segmentation of MR Images of the Head for 3D Display," *IEEE Trans Med. Imaging*, 1990, in press.

17. G. Wiebecke, "3D Visualisierung von Schwer Segmentierbaren Tomographschen Volumendaten," master's thesis, Univ. Hamburg, Hamburg, W. Germany, 1990.

18. L.S. Chen et al., "Surface Shading in the Cuberille Environment," *CG&A*, Vol. 5, No. 12, December 1985, pp. 33-43.

19. U. Tiede et al., "Display Techniques for 3D Tomographic Volume Data," *Proc. NCGA 88*, Vol. III, NCGA, Fairfax, Va., pp. 188-197.

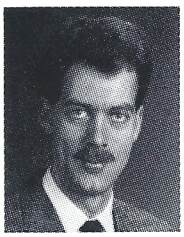
20. U. Tiede," Entwurf, Implementation und Vergleich Verschiedener Reprasentationsformen fur die 3D-Darstellung von medizinischen Objekten," master's thesis, Univ. Hamburg, Hamburg, W. Germany, 1988.

21. C. Barillot et al. (1985) "3-D Representation of Anatomic Structures from CT Examinations," *Biostereometrics 85*, Proc. SPIE 602, Bellingham. Wash., 1985, pp. 307-314.

22. D.H. Ballard, C.M. Brown, *Computer Vision*, Prentice-Hall, Englewood Cliffs, N.J., pp. 82-83.

23. K.H. Hoehne et al., "Dynamic Three-dimensional Display of the Beating Heart from Four Dimensional MR Imaging Data," 74th Assembly of RSNA, *Radiology*, Vol. 169, supplement, 1988, p. 472.

24. K.H. Hoehne et al., "3D Visualization of Tomographic Volume Data Using the Generalized Voxel Model," The Visual Computer, Vol.6, No. 1, February 1990, pp. 28-36.



Ulf Tiede is a research assistant at the Institute of Mathematics and Computer Science in Medicine at the University Hospital Eppendorf, Hamburg, W. Germany.
He is working on image processing, computer graphics algorithms and user-interface design for 3D medical workstations. His research interests include display techniques for voxel-based data and simulation of surgical therapy planning.

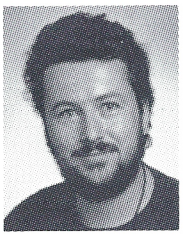
Tiede received his MS in computer science at the University of Hamburg in 1988.

Andreas Pommert is a research assistant at the Department of Computer Science in Medicine, University of Hamburg, where he is engaged in medical image processing and computer graphics. His primary research interests are in quality assurance for 3D display techniques. He is responsible for the research communication system installed at University Hospital Eppendorf in Hamburg. He has worked on several software projects in Germany and the USA.

Pommert received his MS in computer science at the University of Kiel in 1987.

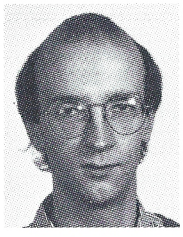
K.H. Hoehne is a professor of medical informatics and director of the Department of Computer Science in Medicine at the University of Hamburg. His current research interests include techniques for visualization and management of ictorial information for medical diagnosis, treatment, and education.

Hoehne received his MS in physics from the University of Wurzburg and his PhD from the University of Hamburg, Germany.



Martin Riemer is a staff member of the Institute of Mathematics and Computer Science in Medicine. He is in charge of the system programming and software engineering of the department's imaging computer facilities. His main interests are image processing and computer graphics of 3D and 4D data.

Riemer received his MSEE from the Fachhochschule Wedel, West Germany, in 1976.



Michael Bomans is a research assistant at the Department of Computer Science in Medicine, University of Hamburg. He is working on image processing and computer graphics, and his special research interests are 3D-edge detection and segmentation of tomographic volumes.

Bomans studied computer science at the University of Hamburg and received his MS in computer science in 1986.

Gunnar Wiebecke has been working at the Department of Computer Science in Medicine, University of Hamburg, since 1986. His research interests include medical applications and computer vision.

Wiebecke is a graduate student in computer science. Presently he is preparing his MS thesis on the visualization of 3D objects without explicit segmentation.

The authors can be contacted via E-mail: surname@IMDM.UKE.UNI-HAMBURG.DBP.DE.