

Texture mapping: A new tool for molecular graphics

Michael Teschner, Christian Henn, Horst Vollhardt,
Stephan Reiling, and Jürgen Brickmann

Abstract

The real-time texture mapping capabilities of modern graphics workstations are explored with respect to their applications in a variety of relevant scenarios in interactive molecular modeling techniques. The common usage of texture mapping to reduce geometric complexity while enhancing realism is extended, opening new ways to visualize large amounts of molecular data in a comprehensive fashion. Thus, texture mapping may be employed to (1) display and filter multichannel information of structural properties on molecular surfaces, (2) improve the quality and accuracy of highly complex isodensity contours, (3) increase the rendering speed of space-filling atomic representations by two orders of magnitude and (4) apply volume-rendering techniques to large, three-dimensional density distributions in real time. Implementation of these novel techniques requires only moderate modifications or extensions to existing molecular modeling applications.

-
- For detailed reference see: *J.Mol.Graphics* 1994, **12**, 98.
 - See also the article of Paul Haerberli and Mark Segal about Texture Mapping (<http://www.sgi.com/misc/grafica/texmap/>).
-

Introduction

The complexity of molecular systems leads to huge amounts of data generated by experiments or simulations. For a better understanding of the structure and function of molecular systems, more advanced computer graphics capabilities are desirable, allowing one to go from a qualitative general overview toward more and more detailed aspects.

With the advent of general-purpose workstations supporting the advanced technique of texture mapping [1,2] in computer graphics hardware [3], a new door for the visualization of molecular systems has been opened. Integration of the texture-mapping concept in molecular graphics applications dramatically extends existing visualization capabilities. Moreover, solutions to unsolved problems in molecular visualization are offered, filling a gap in the analysis of molecular scenarios.

The texture-mapping method is characterized by its simplicity and elegance in obtaining quite complex results. The technique, which may often provide additional physical and chemical insight to molecular problems, is demonstrated in the following selected examples. Due to hardware support, all visual effects may be fine-tuned to ultimately capture a complicated three-dimensional (3D) system in a simple fashion. Furthermore, implementation of texture-

mapping capabilities is straightforward and simple, avoiding algorithmic complexity and the loss of compute performance.

Texture mapping has been widely used in visual simulation applications [4,5]. In these applications, the major concern is that there is a constant frame rate for the visual representation of 3D objects moving in real time and represented with a high degree of realism. In visual simulations, texture mapping is helpful, because it reduces geometric complexity and enhances the realism of the scene. This is achieved by projecting images onto the surfaces of a 3D objects.

At present, texture mapping is not widely used by the molecular graphics community. Noninteractive applications of the technique, such as computer animation, have provided a severe bias towards what texture mapping can be used (e.g., wooden [6] or marble surfaces for the simulating solid materials or fuzzy, stochastic patterns mapped on quadrics to visualize clouds [4,5]).

This article demonstrates that the new potential brought about by texture mapping may generate an enormous impact on molecular graphics by identifying the real-time visual representation of molecular surfaces and 3D structural properties as the most important application areas. Moreover, this article attempts to provide a new way of thinking about texture mapping for molecular problems.

Reverting to the original, strict and formal definition of texture mapping, the technique becomes a flexible, elegant and more general framework for the generation and modification of visual molecular information obtained from a variety of sources. Based on this abstract definition of texture mapping, selected problems in molecular visualization can be elucidated in the form of four different examples: (1) display and filtering of multichannel structural information on molecular surfaces, (2) quality and accuracy improvement necessary for the visualization of highly complex isodensity contours, (3) increased speed of the rendering step of space-filling atomic representations, and (4) application of volume-rendering techniques to large 3D density distributions in real time.

The texture-mapping concept

Texture mapping is a technique that applies an image to an object's surface as if the image were a decal or cellophane shrink-wrap. The image exists in a parametric coordinate space called the *texture space* [7,8,9]. The individual elements of a texture are called *texels*. Figure 1 shows how a simple texture is defined in two-dimensional (2D) texture space and mapped to 3D object space.

At the moment, the general benefit provided by the application of texture mapping is to add realism or reduce geometric complexity. The object material can be shown by wrapping a wood grain pattern around a rectangular solid to create a block of wood. By mapping an image of a house onto a few polygons, instead of drawing a large number of polygons to represent the house, the complexity is reduced.

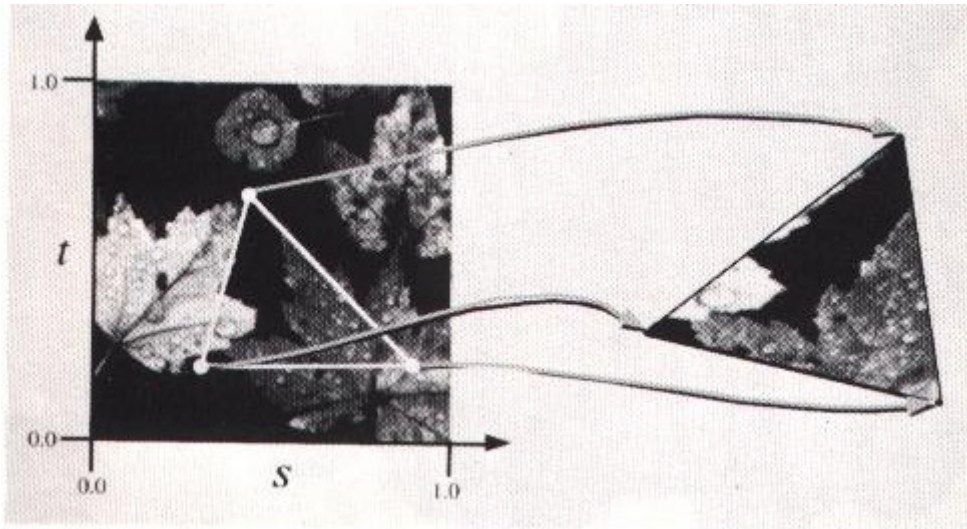


Figure 1: The basic principle of texture mapping. The texture (left) is defined in a normalized coordinate system. The mapping of the texture onto the geometry (right) is done on a per vertex basis.

Texture mapping, in a more abstract definition, can be used in a variety of ways to enhance visual representation. Two basic components are needed for the texture-mapping procedure - the texture, which is defined in a texture space, and the 3D geometry, defined on a per-vertex basis in object coordinates. 3D geometry could be a set of points and vertices, but the real value of texture mapping emerges on drawing filled polygons, such as triangles or higher order polygons.

Texture mapping assigns a coordinate in the texture space to each vertex of the 3D object . The mapping describes where the texels are placed in object space. This is not, however, a one-to-one mapping to screen pixels. It can be magnified or reduced through interpolation. This is done during the rendering step, where pixel information generated by lighting and coloring is modified, based on the texture information.

Assignment of texture coordinates is often done in an explicit fashion (i.e., specific texture values are attached to each object vertex). Alternatively, texture coordinates can be generated automatically. One method is based on the distance from a reference plane in world or object space, while another method uses the object's normals to determine texture coordinates depending on the direction in which it is viewed.

Texture space can be one-dimensional (1D), 2D or 3D (Figure 2). Analogous to the pixel in screen space, each element in texture space is called a *texel* (texture element). Current hardware implementations offer flexibility with respect to how the information stored with each texel is interpreted. Multichannel colors, transparency or even look-up indices, corresponding to a color look-up table, are supported [3,10].

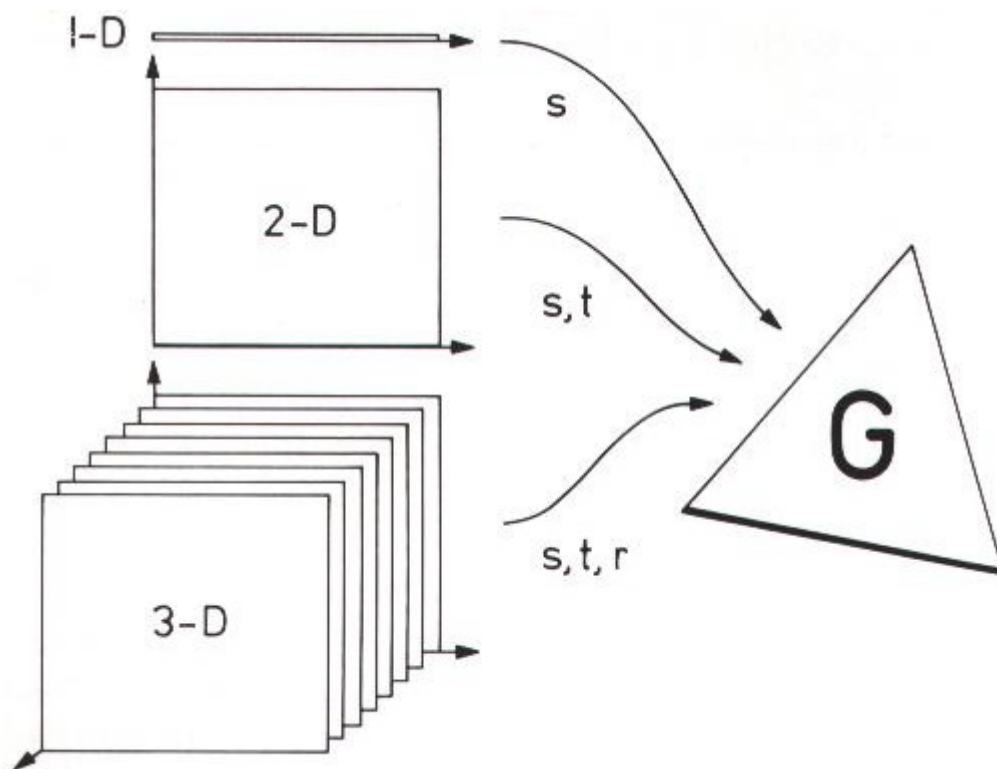


Figure 2: *The dimensionality of the texture space is independent of the geometry.*

As a basic definition, the texture space can be regarded as a special memory, where a variety of information can be stored and then linked to object representations in real 3D space.

Application fields for texture-mapping techniques

A) Molecular surfaces

The detailed study of molecular surfaces have become important tools for the interpretation of properties beyond the molecular structure, in terms of atomic coordinates [11-14]. Models of molecular surfaces are either based on a spherical probe rolled over the atomic structure [15,16] or on isocontouring of 3D density grids [17]. Both approaches use algorithms capable of generating 3D objects with solid surfaces composed of triangles [18,19].

The combination of these forms of molecular representation with texture-mapping capabilities offers a variety of new ways for visualizing molecular properties.

B) Volumetric properties

Visualization of large 3D data sets has always been a challenge to interactive computer graphics applications. This is true not only for the field of molecular modeling, but also for other areas of scientific visualization with potentially high demands on interactivity and visual clarity. Two such areas can be found in medical applications: CT scanning and MR imaging.

Two approaches are used to generate visual information out of 3D data sets: extraction of a discrete isocontour surface [20] or voxel-based rendering of the volume [21] using a color look-up table to attribute irrelevant parts of the volume with transparency.

Applications

A) Color coding

Color coding is a popular means of displaying scalar information on a surface [22]. In interactive molecular graphics, high-contrast color code variation on sparsely tessellated geometry has so far been a serious problem. While standard hardware implementations allow the specification of an RGB color triplet per surface vertex, color interpolation on pixels in between is performed in a linear way [23]. As a consequence, all entries in the color code lying outside the linear color ramp joining two RGB triplets are never taken into account. With a highly tessellated surface, this problem can be reduced. An alignment of the surface vertices with the expected color code change or multipass rendering may remove such artifacts completely. However, these methods demand large numbers of polygons or extreme algorithmic complexity, both undesired in interactive applications.

A solution to this problem is provided by representing the color ramp as a 1D texture. Mapping the calculated property into texture space instead of color space ensures that the coloring evaluated at every pixel is taken from information lying in between the values of the relevant vertices (Figure 3). High contrast variation in the color code is then possible, even on sparsely tessellated surfaces (Figure 4 and 5). It is important to note that, although the texture is 1D, it is possible to tackle a 3D problem, because the dimensionality of the texture space does not affect the object space.

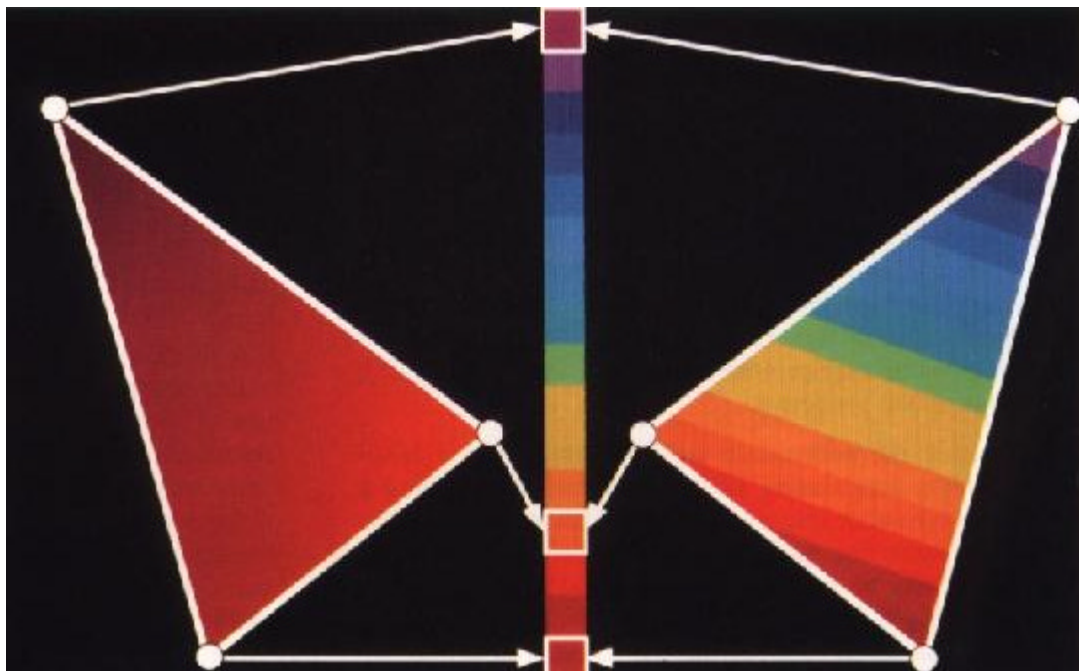


Figure 3: The difference between interpolation in color space (left) and in texture space (right).

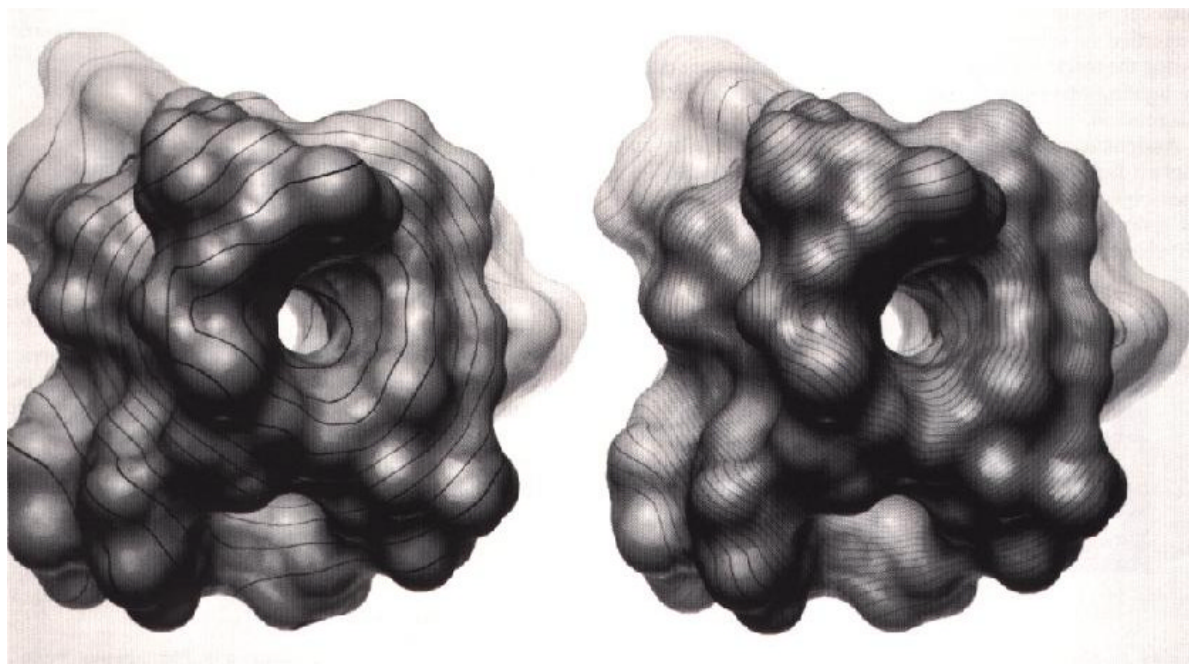


Figure 4: *Isocontouring of the hydrophobic potential on the solvent-accessible surface of Gramicidine A. Using a different scaling for the texture space results in a different spacing of the isocontour lines.*

The independence of texture and object coordinate space is well suited to accommodate immediate changes to the meaning of the color (i.e., by applying simple 3D transformations in texture space). Translation allows readjustment of the zero line of the color code, while scaling of the texture changes the range of the mapping. Such modifications may be performed in real time.

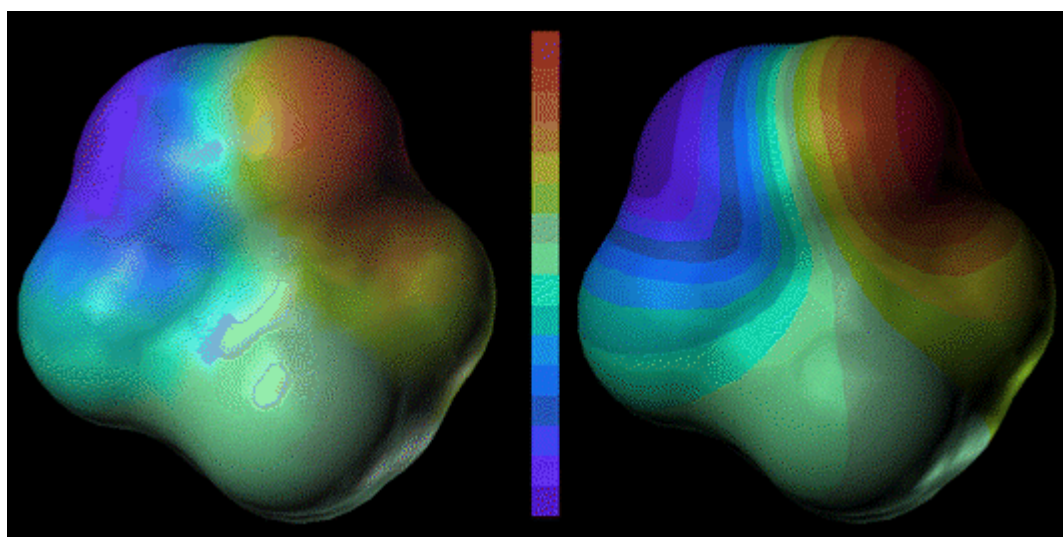


Figure 5: *Solvent-accessible surface of ethanol, color coded against the electrostatic potential, using the traditional color coding (left) and using texture mapping (right). With texture mapping, the resulting sharp transitions from one color value to the next significantly improve the accuracy of rendering. In addition, the sharp transitions help to visually understand the object's 3D shape.*

Similar to the 1D texture used as a color code on a molecular surface, the texture space may be extended to two or even three dimensions, incorporating additional information with each additional dimension, such that a maximum of three independent properties can simultaneously be visualized simultaneously. However, special care must be taken not to overload the surface with too much information. Combining two sources of information that together do not provide new insight into the properties of the molecular system does not make much sense either.

A rather useful application of multiple property mapping is the introduction of transparency. This is illustrated in analyzing the local flexibility of a structure, according to the crystallographically determined B-factors. The more rigid the structure is, the more opaque the surface will be displayed. Increasing transparency indicates higher floppiness of the domains. Such a transparency map may well be combined with any other color-coded property, as it is of interest to study the dynamic properties of a molecule in many different contexts.

By filtering property information on a molecular surface, one is able to generate more insight in two different ways:

- (1) The filter allows the scientist to distinguish between important and irrelevant information.
- (2) The filter puts an otherwise qualitative property into a quantitative context (e.g., the standard deviation from a mean value may provide a hint as to accuracy of a represented property).

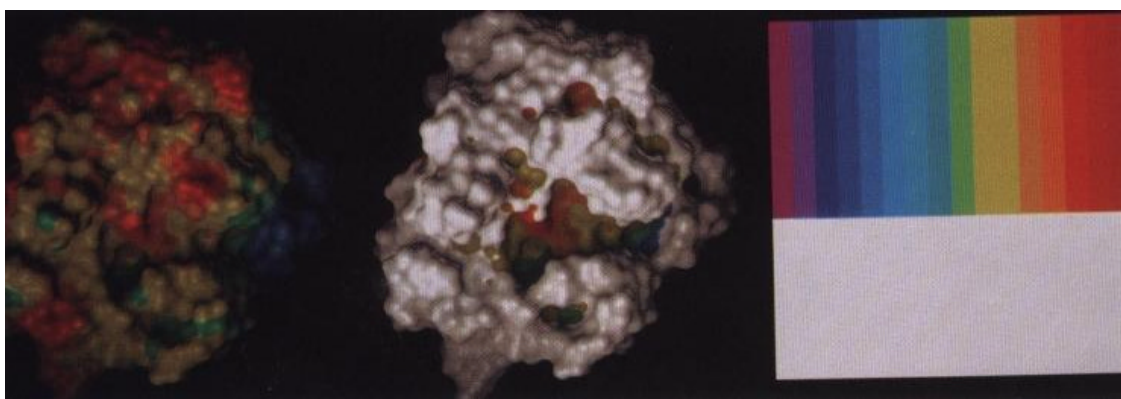


Figure 6: *The solvent-accessible surface of Trypsin, color coded against the electrostatic potential, using the texture-mapping approach (left). This needs only one dimension in texture space. With coding the electric field in the second dimension, it is possible to filter the potential with the strength of the field. Only those areas are color coded where the electric field is above a given threshold.*

In both cases, the information can be filtered using a delta-type function, suppressing all information not exceeding a specific threshold, or else a continuous filter may be used to allow for a more fine-grained quantification (Figure 6).

Filtering may be implemented analogously to the color-coding technique earlier, if one uses 2D or 3D texture maps. A useful application results from the filtering of properties such as the electrostatic potential and the electric field. Taking the absolute value of the electric field, the filter threshold easily pinpoints the areas of the highest local field gradient, which is often

used in identifying the binding site of an inhibitor without additional interaction of the scientist. Another application includes filtering based on the statistical error of a represented property, such as putting the displayed property into a quantitative context.

B) Real-time Phong shading

Electron density maps are often quite complex. They can exhibit a high degree of local curvature, which is a major challenge for accurate visualization. This is where the simple Gouraud shading [23] approach often fails. It destroys information in the highly curved regions of the surface because the linear interpolation of vertex colors cannot account for the local shadow variation due to the lighting model.

Correct perception of the curvature can only be achieved with computationally much more demanding shading techniques (e.g., Phong algorithms [24]). In contrast to linear interpolation of vertex colors, the Phong shading approach interpolates the normal vectors for each pixel of a given geometric primitive, computing the lighting equation in the subsequent step. Attempts have been made to overcome some of the computationally intensive steps of the procedure [25], but their performance was insufficient to be a reasonable alternative to Gouraud shading in real-time applications.

A much more simpler solution is possible using a special texture-mapping technique called *normal mapping*. In contrast to Phong shading, the interpolation is not performed on the normals directly. However, normals are used to automatically generate texture coordinates based on the orientation of the surface vertices in the world coordinate system. The subsequent interpolation then takes place in 2D texture space, using the image of a perfectly rendered sphere as the texture (Figure 7 and 8).

The visual result compares exactly to a surface computed with the original Phong approach. Although this trick is only applicable for infinitely distant light sources, it is a tremendous aid for the visualization of highly complex surfaces.

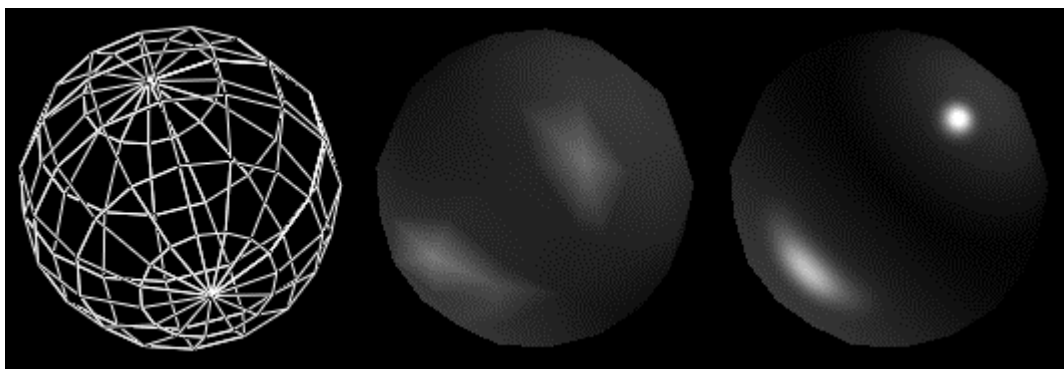


Figure 7: Differences between Gouraud shading and texture mapping for shaded surfaces. The tessellation in both cases is the same (left). In the Gouraud case (middle), the tessellation influences the shading, which is not valid for texture mapping.

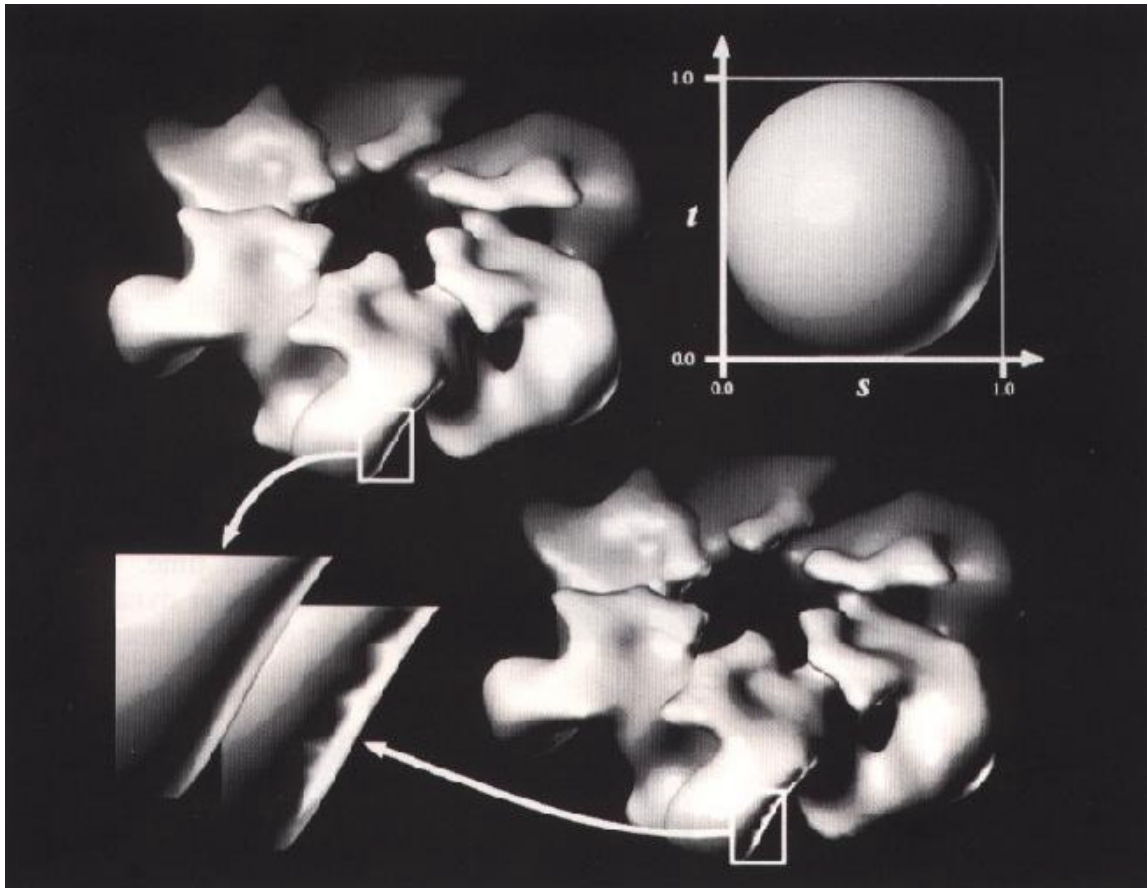


Figure 8: Application of Phong shading to arbitrary surfaces. The sphere is used as the texture. The example shown is a 3D surface reconstruction of an electron microscopy experiment visualizing the asymmetric unit membrane of the urinary bladder. The difference between Gouraud shading (right) and the texture mapping implementation of Phong shading (left) is very obvious, looking at the close-ups.

C) Real-time space-filling models

The use of high-quality CPK models [26] in interactive molecular-modeling applications is currently restricted to molecules of limited size. Spheres that are used to represent individual atoms need to be highly tessellated, which increases the geometric complexity of the model system. In order to display a simple sphere with sufficient accuracy, a few hundred Gouraud shaded triangles are required. The rendering of large molecules without degradation of the visual quality is impossible within reasonable effort.

This problem can be solved with a special form of 2D texture mapping. Each atom is represented by a square on which there is an image of a perfectly rendered sphere. Areas of the square that are not covered by the actual sphere are drawn fully transparent. However, each sphere might intersect with other objects. To handle this problem appropriately, one can use a special feature allows correct, hidden line removal on a per-pixel basis [27].



Figure 9: *Fast spheres using texture mapping. The 13,767 spheres of a hydrated phospholipid bilayer (DLPE), part of a cell membrane can be rotated in real time. The configuration is taken from a molecular dynamics simulation performed by some of the authors.*

This approach reduces the geometric complexity to one squared polygon per atom, resulting in a performance at least two orders of magnitude better. Furthermore, the visual quality of the representation is raised to an optimum (Figure 9).

D) Volumetric molecular properties

Texture-oriented volumetric visualization is based on sampling a 3D texture volume with vertex-based geometry. The final coloring of the geometry reflects the result of the intersection with the texture.

The most simple application of this principle is that of a slice plane, defined either in object or in world coordinates. The slicing samples the texture to reflect the contents of the volume as if it were exposed by cutting the object with a knife. The orientation of such a slice may be modified interactively (Figure 10). A direct extension of the slice plane approach leads to the complete visualization of the entire volume. A stack of planes, oriented in parallel to the computer screen, is used to sample the entire 3D texture. The planes are drawn starting from the slice plane at the far end of the stack and then proceeding up to the viewer in regular, sufficiently small intervals. Geometric transformations of the volume are performed by manipulating the orientation of the texture, keeping the planes in screen-parallel orientation.

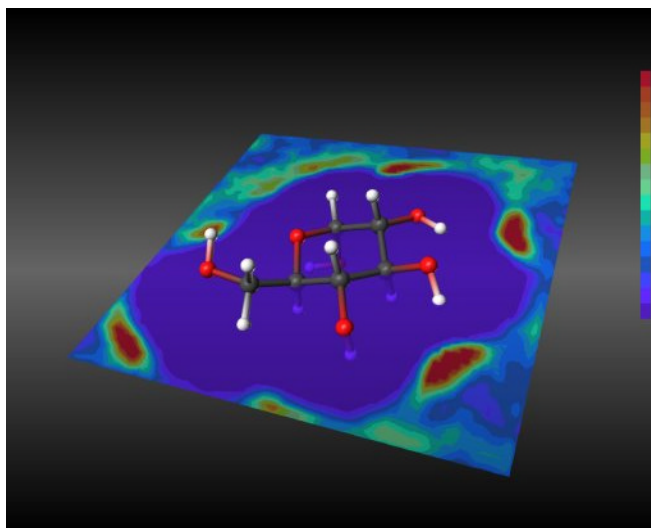


Figure 10: A slice plane cutting a volumetric density distribution. The property visualized is the probability of water being distributed around a sugar molecule. The slice plane may be reoriented within the property volume in real time.

This type of volume visualization technique is greatly enhanced by interactive updates of the color look-up table used in defining the texture. In fact a general-purpose color ramp editor may be used to vary the look-up colors or even to vary the transparency based on the scalar value at a given point in the 3D volume.

The concept of the slice plane can be extended to arbitrarily shaped objects (Figure 11). The idea is to probe a volumetric property and to display it wherever the primitive of the probing object cuts the volume. The probing geometry can be of any shape (e.g., a sphere) that collects information about the volumetric property at a certain distance from a specified point or it may be extended to describe a complete molecular surface.

The independence of the object's transformation from that of the 3D volume, treated as a 3D texture, offers complete freedom in orienting the surface with respect to the volume. This provides an opportunity to look at a molecular surface and have the displayed property updated, based on the current orientation of the object.

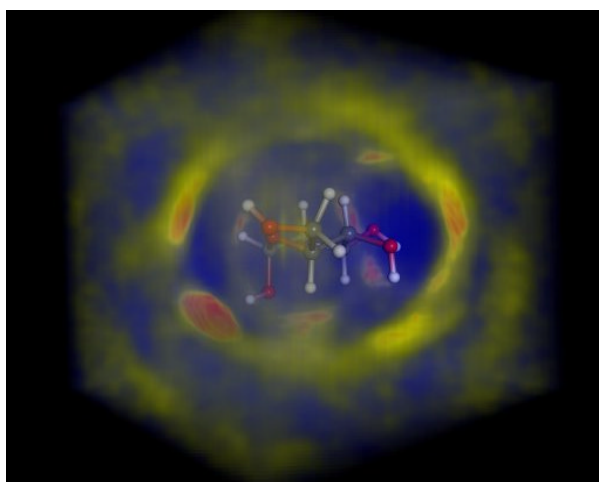


Figure 11: A slice plane cutting a volumetric density distribution. This time, a stack of slice planes is used to produce a volume-rendered representation.

Implementation

All of the examples presented earlier can be implemented using techniques other than texture mapping. The trade-off is either a complete loss of performance or an high level of algorithmic complexity. Both of them are undesirable.

Using texture mapping, it is sufficient to modify existing code fragments or to incorporate modest extensions, such as reading in a texture and activating texture coordinates. In other words, implementing texture mapping in existing molecular graphics applications is extremely simple and strongly encouraged. All examples were implemented on a Silicon Graphics Crimson Reality Engine [3] equipped with two raster managers. The programs were written in C, either in mixed mode GLX or pure GL. Sample code fragments are available from the authors.

Conclusions

The potential of hardware-supported texture mapping, available on general-purpose graphics workstations, has been demonstrated for a number of molecular graphics applications. Its main impact may be expected where (1) the geometric complexity must be reduced to get interactive performance, (2) the information density needs to be increased to reflect adequately the properties of a molecule, (3) irrelevant information has to be filtered; and (4) an improvement of the rendering quality is required to fully support correct perception of highly complex surfaces. It can be expected that in the near future even more uses for real-time texture mapping in molecular graphics will emerge.

Literature

- [1] Blinn, J.F., and Newell, M.E. Texture and reflection in computer generated images. *Communications of the ACM* 1976, **19**, 542-547.
- [2] Blinn, J.F. Simulation of wrinkled surfaces, *Computer Graphics* 1978, **12**, 286-292.
- [3] Akeley, K. Reality Engine Graphics, *Computer Graphics* 1993, **27**, 109-116.
- [4] Gardner, G.Y. Simulation of natural scenes using textured quadric surfaces, *Computer Graphics* 1984, **18**, 11-20.
- [5] Gardner, G.Y. Visual simulations of clouds, *Computer Graphics* 1985, **19**, 297-303.
- [6] Peachey, D.R. Solid texturing of complex surfaces, *Computer Graphics* 1985, **19**, 279-286.
- [7] Catmull, E. A subdivision algorithm for computer display of curved surfaces, Ph.D. dissertation, University of Utah, 1974.
- [8] Crow, F.C. Summed area tables for texture mapping, *Computer Graphics* 1984, **18**, 207-212.
- [9] Williams, L. Pyramidal parametrics, *Computer Graphics* 1983, **17**, 1-11.
- [10] Porter, T., and Duff, T. Compositing of digital images, *Computer Graphics* 1984, **18**, 253-259.
- [11] Langridge, R., Ferrin, T.E., Kunz, J.D., Connolly, M.L Real-time color graphics in studies of molecular interactions, *Science* 1981, **211**, 661.

- [12] Pang, L., Lucken, E.A.C., Weber, J. and Bernardelli, G. Structural investigations and modeling of cavities in clathrates. *J. Mol. Comput.-Aided Mol. Design* 1991, **5**, 285-291.
- [13] Freeman, C.M., Levine, S.M., Newsam, J.M., Sauer, J., Tomlinson, S.M., Brickmann, J. and Bell, R.G. In "Zeolite Computer Graphics," *Modelling of Structure and Reactivity in Zeolites* (C.R.A. Catlow, Ed) Academic Press, London (1992) 133-155.
- [14] Mezey, P. In "Molecular Surfaces," *Reviews in Computational Chemistry* (K. Lipkowitz and D. Boyd, Eds) VCH, Weinheim, (1990) 265-294.
- [15] Richards, F. Calculation of molecular volumes and areas for structures of known geometry, *Methods in Enzymology* 1985, **115**, 440-465.
- [16] Connolly, M. Analytical molecular surface calculation, *J. of Applied Cryst.* 1983, **16**, 548-558.
- [17] Goetze, T. The fractal dimension of molecular surfaces. Thesis, Technische Hochschule Darmstadt, in preparation
- [18] Heiden, W., Schlenkrich, M. and Brickmann, J. Triangulation algorithms for the representation of molecular surface properties. *J. Comput.-aided Mol. Design* 1990, **4**, 255-269.
- [19] Heiden, W., Goetze, T. and Brickmann, J. Fast generation of molecular surfaces from 3D data fields with an enhanced "marching cube" algorithm. *J. Comp. Chem.* 1993, **14**, 246-250.
- [20] Lorensen, W.E., and Cline, H.W. Marching cubes: a high-resolution surface construction algorithm, *Computer Graphics* 1987, **21**, 163-169.
- [21] Drebin, R., Carpenter, L., and Hanrahan, P. Volume Rendering, *Computer Graphics* 1988, **22**, 65-74.
- [22] Dill, J.C. An application of color graphics to the display of surface curvature, *Computer Graphics* 1981, **15**, 153-161.
- [23] Gouraud, H. Continuous shading of curved surfaces, *IEEE Transactions on Computers* 1971, **20**, 623-628.
- [24] Phong. B.T. Illumination for computer-generated pictures, *Communications of the ACM* 1987, **18**, 311-317.
- [25] Bishop, G., and Weimer, D.M. Fast Phong shading, *Computer Graphics* 1986, **20**, 103-106.
- [26] Corey, R.B. and Pauling, L., Molecular models of amino acids, peptides and proteins, *Rev. Sci. Instr* 1953, **24**, 621ff.
- [27] Fuchs, H, et.al. Fast spheres, shadows, textures, transparencies, and image enhancements in pixel-plane, *Computer Graphics* 1985, **19**, 111-120.