Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Molecular Graphics and Modelling 23 (2004) 233–238
Journal of Molecular Graphics and Modelling
www.elsevier.com/locate/JMGM

Hierarchical molecular modelling with ellipsoids

Nelson Max*

Lawrence Livermore National Laboratory, 7000 East Avenue, Livermore, CA 94550, USA

Received 27 May 2004; received in revised form 18 June 2004; accepted 1 July 2004

Abstract

Protein and DNA structures are represented at varying levels of details using ellipsoidal RGBA textured splats. The splat texture at each level is generated by rendering its children in a hierarchical model, from a distribution of viewing directions, and averaging the result. For rendering, the ellipsoids to be used are chosen adaptively, depending on the distance to the viewpoint. This technique is applied to visualize DNA coiling around nucleosomes in chromosomes.

Keywords: Splat; Level of detail; Ellipsoid; Hierarchical model

1. Introduction

There are several models for molecular visualization, including: (a) stick models for the bonds; (b) the ribbon and arrow models pioneered by Jane Richardson for secondary structure; (c) colored space filling Van der Waals sphere models; and (d) the solvent accessible surfaces of Richards, which are smoothed versions of the space-filling sphere models. Computer graphics renderings of (a)–(c) can be produced by MOLSCRIPT [1], and of (d) by the methods in [2,3].

Macromolecular assemblies within a cell can be very complex, including multiple chains of protein, DNA, RNA, and other molecules. When examining the quaternary structure of such assemblies, level of detail simplification is useful to get an overall visualization at interactive rates. This paper describes a hierarchical simplification of the space-filling models, using ellipsoidal textured splats.

This approach is similar to the QSplat algorithm of Rusinkiewicz and Levoy [4], where a KD tree was used to construct a hierarchy of spheres to represent the surface of a solid object. But molecular assemblies already have a biochemically meaningful hierarchy, with amino acids built from atoms, protein chains built from amino acids, and multi-chain assemblies built from separate chains. The current system uses this natural hierarchy. It uses ellipsoids instead of spheres to better approximate the shapes of the objects, and textures to approximate their colors and opacities.

The ellipsoids are projected by the methods described in [5]. The main innovation in this paper is that the color/opacity texture for each object's splat, rather than a Gaussian, is constructed from the average of rendered images of the object's children in the hierarchy, from a distribution of viewing directions.

2. Nucleosome structure

The nucleosome is a core assembly of eight histone protein chains, around which almost two full coils of DNA are wrapped, giving the disc-like particles shown in Fig. 1. These particles, connected by linker stretches of DNA, are arranged in a helical pattern to form the 30 nm chromatin fiber shown in Fig. 2, which is itself further coiled when a chromosome condenses during cell division. This organized coiling keeps the very long DNA strands in a chromosome from getting tangled, and also helps control transcription.

In the early 1980s, I participated in the production of a dome-screen stereo computer animated film for the Fujitsu pavilion at Expo'85 in Tsukuba, Japan, which showed DNA coiling into a condensed chromosome, at varying levels of

* Tel.: +1 925 422 4074; fax: +1 925 422 6287.
E-mail address: max2@llnl.gov.]

1093-3263/$ – see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.jmgm.2004.07.001

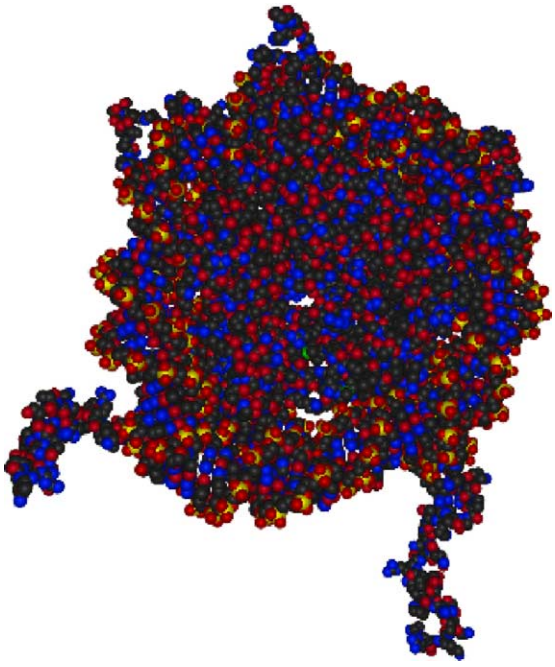

Fig. 1. Nucleosome with all non-hydrogen atom spheres.

detail. (See [6] and [7].) At that time, full structural information for the nucleosome was not known, so we just used spherical splats for the histone proteins. We used geometrical morphing for the DNA representation, from a ball-and-stick representation, to a ''ladder'' representation, and then into a single thick line. In the lowest level of detail, the chromatin fiber was rendered using opaque cylindrical splats with shading and highlights, feathered into transparency at their ends, so that they could join into an approximation of a curved tube. (See Fig. 4 of [7].) This is similar to the way the current system renders the DNA double helix at a lower level of detail. The method of sorting the objects in depth and compositing them from back to front is also similar to the system presented here.

Since that time, a high resolution X-ray crystallography structure of the nucleosome with 147 base pairs of DNA has been solved [8], with atomic coordinates for all atoms except hydrogen. This structure was obtained from entry 1KX5 in the Protein Data Bank [9]. The 12 base pair double helix of DNA from entry 119D was used as the linker to connect the nucleosomes. It was rotated and translated to continue one end of the nucleosome DNA. The union of these two structures has a total of 14,103 atoms. The individual chains are identified, as well as each amino acid in the proteins and each nucleotide in the DNA.

3. Level of detail hierarchy

The first (highest) level of detail is the individual atoms shown in Fig. 1, which form the leaves of the hierarchy tree. The splats for the atoms use a separate RGBA texture for each atom type. So, the data structure for an atom contains the center, radius, and texture square index into the larger texture shown in Fig. 3. The other nodes in the hierarchy tree at lower levels of detail are for ellipsoidal objects with a center, a bounding sphere radius for view frustum culling and level of detail choice, a texture square index, a linked list of children, and six floats for the upper triangular elements of the symmetric moment of inertia matrix of the unweighted set of centers of the child objects, which defines the ellipsoid.

All the object textures are blocks in the single 256×256 texture shown in Fig. 3. The atom textures have an anti-aliased partially transparent profile curve, and a highlight in the center. Counting across rows starting with position 1 at the upper left, the atom textures are in positions 25–30, in the order nitrogen, carbon, phosphate oxygen, sulfur, phosphorus, and non-phosphate oxygen. I used two different colors for oxygen atoms: an orange-red for ones in the phosphate groups in the DNA sugar–phosphate backbone, and a purple-red for all the others. This distinguished the phosphate groups, and thus the DNA backbone.

The atoms (and also the other ellipsoids) are sorted in depth, and composited from back to front, by applying the corresponding texture to a polygon surrounding the projection of the atom sphere. The radius for each atom type is the atom's Van der Waals radius, giving the standard space filling molecular models. The intersection curves of these spheres are not generated by compositing the planar texture splats, so the spheres pop unnaturally in front of each other as the viewpoint changes. Therefore, I have added the option of rendering the front faces of an enclosing polyhedron, using the same texture. Latitude and longitude divisions are chosen at 45° intervals, dividing the visible half of a sphere, with the North pole pointing towards the viewer, into a triangle fan of eight triangles, surrounded by a quad strip of eight quads. With Z-buffering, this compromise gives only a rough approximation to the true intersection curve, but is quick to render, almost as quick as using a single square. Since the polyhedron is circumscribed around the sphere, the shape of the texture map circle still gives a smooth profile to the sphere; only the intersection curve appears somewhat jagged when viewed up close. The lower levels of detail use softer-edged ellipsoids of more similar colors, so the pop-

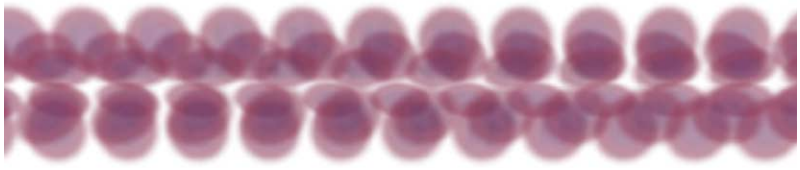

Fig. 2. A 30 nm chromatin fiber, rendered with a single splat per nucleosome.

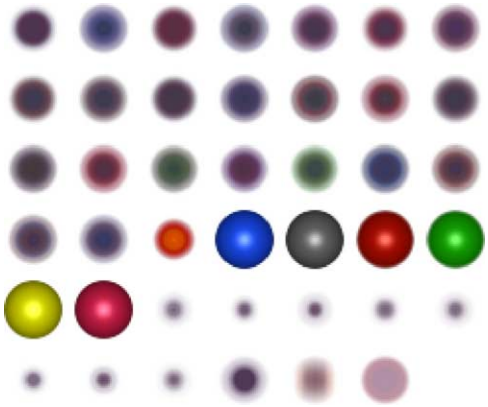

Fig. 3. Splat textures. In row order starting from the upper left are splats for 19 amino acids, four nucleosides, phosphate, six atoms, eight protein chains, the union of all eight protein chains, a cylinder for DNA, and the complete nucleosome.

ping is less of a problem, and a single textured rectangle is used. Z-buffer testing is turned off, and visibility is determined only by the depth-sorted compositing order.

The next level of detail in the protein hierarchy is the amino acids, which are joined together to make a protein chain. There are 20 common amino acid types, only 19 of which appear in the nucleosome structure. Each type has a certain amount of flexibility, and can assume different shapes (rotamers) depending on the environment in the protein. So, each instance of an amino acid has its own moment of inertia matrix, describing its shape as well as its orientation. The inverse of this matrix gives the quadratic form for the ellipsoid equation. But there is only one texture square for each of the amino acid types, constructed from one representative shape, giving the first 19 texture squares (Fig. 3).

If a 3D ellipsoid is projected in perspective, the result is a 2D ellipse. The mathematics for the ellipsoid, its perspective projection, and its enlargement for anti-aliasing, are given in [5]. The 2D ellipse is enclosed in a rectangle determined by its major and minor axes, and composited using the appropriate texture map. Finding the major and minor axes amounts to finding the eigenvectors of a quadratic form, which is relatively simple in 2D. Neither the inverse of a 3×3 matrix, nor 3D eigenvectors, are required.

The textures for the ellipsoids are created in preprocessing steps, with the higher levels of detail first. Each ellipsoid should represent the average appearance of its collection of child objects, as seen from different viewing directions. Therefore, for a sample set of viewing directions, the child objects, whose textures have already been determined, are rendered, and the RGBA images are read back from the frame buffer.

On each image, the corresponding projected ellipse is conceptually superimposed. Its quadratic form gives the square of the Mahalanobis distance to the center, with contours of equal distance representing concentric ellipses. For each image, histograms of average color and opacity as a function of the Mahalanobis distance are constructed. A weighted average of these histograms is taken, and accessed to determine the color and opacity for the pixels in the corresponding circularly symmetric texture in Fig. 3, as a function of the radius from the center of the texture square. The resulting texture gives an approximation to the average appearance of the child objects from any viewing angle, and its opacity gives the average coverage.

The sampled viewing directions are taken at equally spaced latitude and longitude angles on a unit sphere. The solid angles of the spherical triangle and quadrilateral regions corresponding to these samples are unequal, but are easily computed, and are used as weights in computing the weighted average of the histograms.

For DNA, the unit of structure corresponding to an amino acid is a nucleotide, consisting of a phosphate group attached to a nucleoside. The nucleoside, in turn, is made up of a sugar and one of four bases. The nucleotides are linked into a chain via their phosphates. In the DNA double helix, the sugar–phosphate linkages form two spiral backbones around the outside of the helix cylinder, and the bases stack inside, in planes approximately perpendicular to the cylinder axis.

I chose to separate out the phosphate groups in the object hierarchy, in order to delineate the spiral backbones. Since phosphorus is colored yellow and the phosphate oxygens are colored orange-red, this gives orange spherical splats for the phosphates. The four nucleoside textures are next after the amino acids in Fig. 3, followed by the orange phosphate texture. An image of the nucleosome at the level of detail showing amino acids, phosphates, and nucleosides, is shown in Fig. 4. The four nucleosides are approximated by long flattened ellipsoids, so the base stacking is still visible at this level of detail.

The next lower level of detail uses one ellipsoid for each of the eight protein chains in the nucleosome core. These are shown in the 8 positions after the atoms in the texture in Fig. 3. Since some of these chains are long and extended, they are not well approximated by ellipsoids, and diffuse density remains outside the central core of the texture. I truncated the texture at a smaller radius to remove some of this excess density. The following texture in Fig. 3 is for the whole nucleosome core, the union of these eight chains.

For the DNA at this level of detail, I used a structure consisting of 10 base pairs. Since this many base pairs gives a full turn of the DNA, the major axis of the resulting ellipsoid will point approximately along the axis of the double helix.

In order to give the DNA helix the appearance of a bent cylindrical tube, cylindrical symmetry was assumed. The color and opacity histograms were created as a function of the projection, onto the minor axis direction of the 2D ellipsoid, of the vector from the image rectangle center to the pixel. The resulting 2D texture of vertical stripes was feathered to zero at the top and bottom, using a circular window, so that the splats of the DNA would fit together to

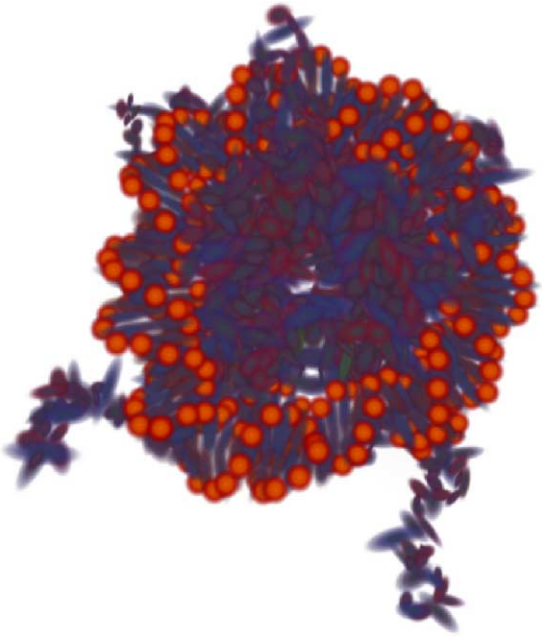

Fig. 4. The nucleosome at the level of detail showing amino acids, phosphates, and nucleosides.

form an approximation to a curved cylindrical tube. This texture is shown as the next to last one in Fig. 3. Note the orange edges, caused by the projection of the orange phosphate spheres concentrated at the profile of the double helix. A nucleosome at this level of detail is shown in Fig. 5. In addition to the eight histone protein chains, a 50% dense image of the full nucleosome core was used to partially fill in the gaps between the protein chain ellipsoids.

At the lowest level of detail, the whole nucleosome is represented by a single ellipsoidal splat, whose texture, the final one in Fig. 3, was constructed by viewing the structure of Fig. 5 from multiple directions. Fig. 2, showing a chain of nucleosomes in a chromatin fiber, was drawn using this level of detail.

Fig. 5. The nucleosome at the level of detail showing protein chains and DNA cylinders.

4. Hierarchy traversal

The rendering step begins by constructing the list of objects to be sorted. In order to get a smooth "cross dissolve" transition between the levels of detail, this list may contain both an object and its children, each with opacity weights for the cross dissolve. These weights are used on the polygon vertices, and multiply the colors and opacities from the texture map. The opacity of an object at an internal hierarchy node begins to fade out in favor of its children when the ratio of its boundary sphere radius to its distance from the viewpoint is greater than a threshold T1, and completely fades out when this ratio reaches a larger threshold T2. When the hierarchy is being traversed, the opacity of a node's parent is also known, and the opacity actually used for a node is the minimum of its own opacity, and one minus the opacity of its parent. This makes each ellipsoid visible only for an appropriate interval of viewing distances. Pseudocode for the hierarchy traversal thus uses the recursive routine putSort routine below, initially called for the root node of the hierarchy. Here, putList puts an object, with an associated opacity factor for the cross-dissolve, into the list to be sorted in depth.

```
putSort(node *n, float ap) {
     if(sphere(n->center, n->radius) is outside
          the view frustum) return;
     float an = max(0., (min 1.,
          (T2 - n->radius/n->depth)/(T2 - T1)));
     if(an > 0.) putList(n, min(an, 1. - ap));
     if(an < 1.)
          for all children *c of *n
               putSort(c, an);
     }
```

Before the list of objects produced by the putList calls is sorted in depth, an extra displacement is added to the depth of all atoms. This causes the Z-buffered atom spheres to be composited before their parent splats, making the cross dissolve appear more natural. After the quicksort is completed, the spheres or ellipsoids are projected in back to front order, and the corresponding textured polygons are composited.

Instead of generating a root node for the chromatin fiber with multiple rotated and translated copies of the same nucleosome, the root node was a single nucleosome. Then the hierarchy traversal procedure putSort was called multiple times with different model-view matrices, to create a single output list to be sorted. This meant that it was not necessary to store moment of inertial matrices for all the instances of all ellipsoids in the fiber, nor 4×4 positioning matrices for each child object with respect to its parent object. The effect of the model-view rotation on the moment of inertia matrix is explained in [5].

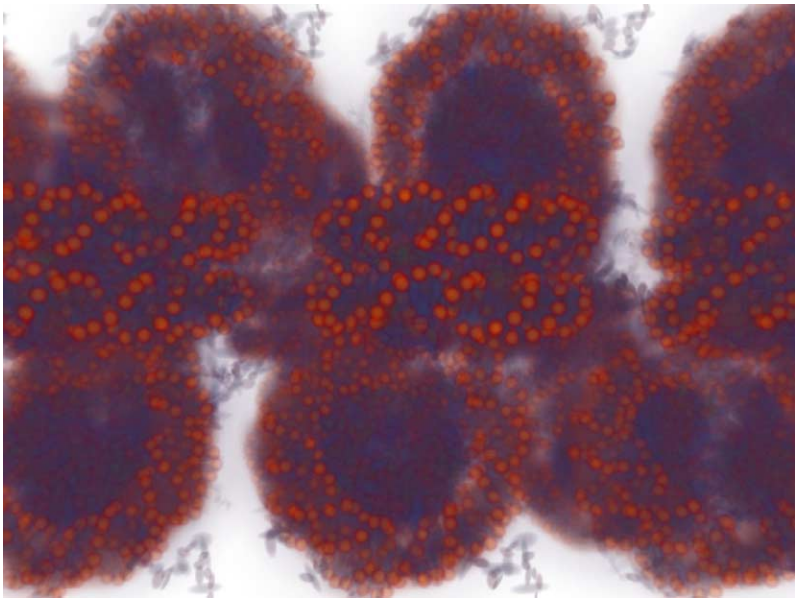

Fig. 6. The chromatin fiber, showing cross dissolves between levels of detail.

5. Results

Fig. 2 shows an early frame from an animation in which the camera moves toward a chromatin fiber formed from 80 nucleosomes. Fig. 6 shows a later frame, where the levels of detail shown in Figs. 4 and 5 are both present. This animation (see link at the end of this document), at 640 × 480 (Video) resolution, ran at an average speed of 9 frames/s using hardware textured polygon rendering on an nVidia 5900FXUltra graphics card, controlled by one processor of a 800 MHz dual Pentium4 Xeon PC.

Fig. 7 is from an animation looking and moving along the fiber, where all the levels of detail are used simultaneously (see link at the end of this document). It ran at 2 frames/s, since there was much more occluded geometry. Occlusion culling could probably speed this up.

6. Future work

As stated above, occlusion culling should be added.

The sphere intersection curves are jagged when viewed in close-up, due to the rough polyhedral approximation. To fix this, the number of faces in the polyhedron could be increased as the spheres come closer. Alternatively, the sphere depth could be calculated at each pixel using a fragment program on the graphics processor. The ellipsoids could be handled the same way, as in [10]. This would reduce the slight popping visible when the ellipsoids change their sorting order during animation, and smooth the sphere intersection curves.

Protein domains are compact subsets of a protein chain, which may have evolved independently, and later joined by gene rearrangement. They can be more closely approximated by ellipsoids than can their whole protein chain, and

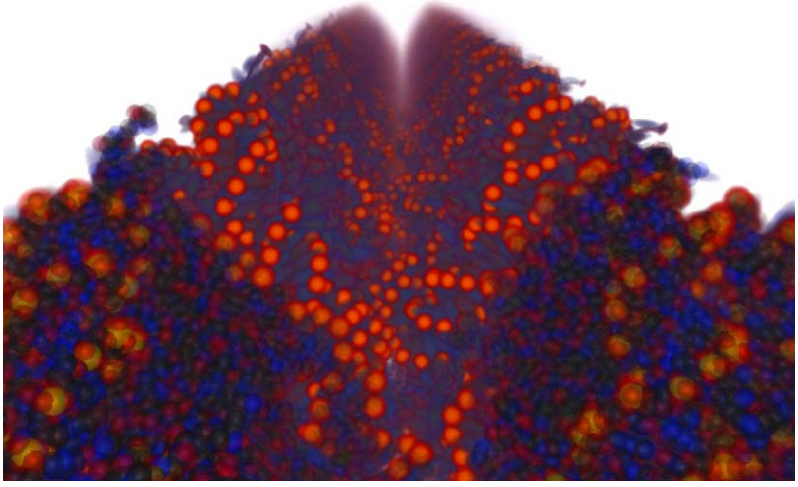

Fig. 7. View along a chromatin fiber showing all levels of detail.

could be an intermediate level of detail between the amino acids and the whole chain. This domain structure could be added to the hierarchy for files in which it is available (it was not for the 1KX5 nucleosome), or an automated domain subdivision algorithm could be written.

Acknowledgements

This work was performed under the auspices of the US Department of Energy by Lawrence Livermore National Laboratory under Contract W-7405-ENG-48. I thank the reviewers for comments which improved the paper, and Mike Pique of the Scripps Research Institute for giving me a multicolored physical ‘‘rapid prototype’’ model of the nucleosome, which helped me understand its structure.

Supplementary material

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jmgm.2004.07.001.

References

[1] P.J. Kraulis, MOLSCRIPT: a program to produce both detailed and schematic plots of protein structure, J. Appl. Crystallogr. 24 (1991) 946–950.
[2] M.L. Connolly, Analytic molecular surface calculation, J. Appl. Crystallogr. 16 (5) (1983) 548–558.
[3] A. Varshney, F. Brooks, W. Wright, Computing smooth molecular surfaces, IEEE Comput. Graph. Appl. 14 (5) (1994) 19–24.
[4] S. Rusinkiewicz, M. Levoy, A multiresolution point rendering system for large meshes, in: ACM Computer Graphics Proceedings, Annual Conference Series, 2000, pp. 343–352.
[5] M. Zwicker, H. Pfister, J. van Baar, M. Gross, E.W.A. Splatting, IEEE Trans. Visual. Comput. Graph. 8 (3) (2002) 223–238.
[6] F. Hirata, N. Max, D. Lerner, T. Okada, Computer graphics: the FACOM M-380 creates an OMNIMAX, stereo movie, Fujitsu 36 (2) (1985) 151–165, (in Japanese).
[7] N. Max, DNA animation, from atom to chromosome, J. Mol. Graph. 3 (2) (1985) 69–71, (with correct Figs. 1–3 on p. 126).
[8] K. Luger, A. Maeder, R. Richmond, D. Sargent, T. Richmond, Crystal structure of the nucleosome core particle at 2.8 Å resolution, Nature 389 (1997) 251–260.
[9] http://www.rcsb.org.pdb.
[10] S. Gumhold, Splatting illuminated ellipsoids with depth correction, in: Proceeding of Vision, Modelling, and Visualization, 2003, pp. 245–252.