

University of Pennsylvania ScholarlyCommons

**Technical Reports (CIS)** 

Department of Computer & Information Science

7-1978

# A Human Body Modelling System for Motion Studies

Norman I. Badler University of Pennsylvania, badler@seas.upenn.edu

Joseph O'Rourke

Hasida Toltzis

Follow this and additional works at: https://repository.upenn.edu/cis\_reports

Part of the Computer Engineering Commons, and the Computer Sciences Commons

### **Recommended Citation**

Norman I. Badler, Joseph O'Rourke, and Hasida Toltzis, "A Human Body Modelling System for Motion Studies", . July 1978.

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-78-27.

This paper is posted at ScholarlyCommons. https://repository.upenn.edu/cis\_reports/1000 For more information, please contact repository@pobox.upenn.edu.

## A Human Body Modelling System for Motion Studies

## Abstract

The need to visualize and interpret human body movement data from experiments and simulations has led to the development of a new three-dimensional representation for the human body. Based on a skeleton of joints and segments, the model is manipulated by specifying joint positions with respect to arbitrary frames of reference. The external form is modelled as the union of overlapping spheres which define the surface of each segment. The properties of the segment and sphere model include: an ability to utilize any connected portion of the body in order to examine selected movements without computing movements of undesired parts, a naming mechanism for describing parts within a segment, and a collision detection algorithm for finding contacts or illegal intersections of the body with itself or other objects. Several display algorithms are possible, including inexpensive hidden surface removal. The spherical body model can also be easily combined with planar polygon object environments.

## Disciplines

Computer Engineering | Computer Sciences

## Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-78-27.

· ~

A Human Body Modelling System

for Motion Studies

Norman I. Badler Joseph O'Rourke Hasida Toltzis

Department of Computer and Information Science Moore School of Electrical Engineering University of Pennsylvania Philadelphia, PA 19104

July 1978

Movement Project Report No. 13 (Supercedes Movement Project Reports 3 and 6.)

A Human Body Modelling System

for Motion Studies

Norman I. Badler Joseph O'Rourke

Hasida Toltzis

Department of Computer and Information Science Moore School of Electrical Engineering University of Pennsylvania, Philadelphia, PA 19104

#### Abstract

The need to visualize and interpret human body movement data from experiments and simulations has led to the development of a new threedimensional representation for the human body. Based on a skeleton of joints and segments, the model is manipulated by specifying joint positions with respect to arbitrary frames of reference. The external form is modelled as the union of overlapping spheres which define the surface of each segment. The properties of the segment and sphere model include: an ability to utilize any connected portion of the body in order to examine selected movements without computing movements of undesired parts, a naming mechanism for describing parts within a segment, and a collision detection algorithm for finding contacts or illegal intersections of the body with itself or other objects. Several display algorithms are possible, including inexpensive hidden surface removal. The spherical body model can also be easily combined with planar polygon object environments.

#### 1. Introduction

Human movement data is collected or generated in many different disciplines. Measurements of body positions or joint locations during movement are recorded for further study in physiology, anthropology, kinesics, biomechanics, dance, and engineering. Computer programs may simulate body movements to obtain position data in otherwise dangerous environments such as vehicle crash studies (10,15,23), or where effective automatic observation techniques are lacking such as in dance (26).

When observational data must be compared with simulated data (for example, to validate the simulation) it may be useful to visualize the movements on a computer model of the body. While conventional graphs may be adequate to diagram movements of individual body parts, a global impression can be best achieved by displaying all body movements simultaneously. A visual animation of the body in motion is essential whenever the mass of data collected or generated is too great to assimilate piecemeal, or when the movement complexity begins to involve positions of other body parts or the environment.

Visualizing human movement on a computer graphics display requires a computational model of the body which is realistically formed, jointed, and three-dimensional. Dissatisfication with existing body models and "stick figure" displays led to the development of a new human model for a computer with certain distinct advantages in display realism, movement definition, contact and collision detection, and cost-effectiveness in a real-time animation playback environment.

Our original motivation to develop a human body model came not from engineering considerations, but from an investigation of human movement itself.

While investigating techniques for the representation of human movement information in a computer, we were led to interaction with professional movement analyists, recorders (notators), and performers. These people are primarily interested in a human body animation as a visualization of notated movement, in both a verification mode and a demonstration or tutorial mode (26). In either case, the constraints which arose from discussions on the form and movement of the body took the following forms:

- . The three-dimensionality of the body, both in space and in substance, must be visible and unquestionable.
  - The movements must be displayed in real-time for teaching purposes, although slower rates are acceptable for verification of notated movements or study of transient effects (for example, in crash studies).
  - The model could be positioned with respect to any rectangular coordinate system and, in particular, any body part could be treated as a reference segment from which other movements are computed.
- Any connected subset of the body could be animated, that is, the movements could be observed in a "logical window" of body parts and not even computed for others out of the set.
- . The presence of environment objects as well as other people must be allowed, displayed, and used for collision detection.
- Collision detection should be simple and should return concise information on the location of any interference or contacts.

The first and last conditions especially preclude the use of stick-figure models since the surfaces of the body are extremely important in movement interpretation and collision detection.

Besides these criteria we imposed our own graphical design consideration that the display generation process should be as device-independent as possible, yet support a variety of fundamentally different graphic displays. In particular, we wish to use video raster devices as well as line-drawing refresh cathode ray display tubes. Moreover, the efficiency of the display algorithms should not be undermined by this flexibility. We feel that we have achieved this goal with our model; indeed it is the first proposed for a raster device that promises to be suitable for low cost graphic display hardware. Each of these points will be addressed in the following sections. After discussing existing or potential models based on various object representation schemes, we will describe our model and how it can be used, among other things, to generate different visual displays, name particular body parts and surfaces, and locate collisions.

#### 2. Modelling Schemes

Representing the form of the human body with computer data structures is not a simple task. A variety of models are in use among the set of possible representation structures, but each has certain faults to balance its advantages. Our model fits this description, too, but we intend to show that whatever disadvantages it has are certainly less severe than those of the others.

We can categorize the schemes for modelling complex three-dimensional objects into two broad groups (2), then further divide these into specific representations based on different primitives:

#### I. Surface representations

A. Surface points

- B. Planar polygon surface patches
- C. Curved surface patches

(bicubic or other mathematical formulations)

#### II. Volumetric representations

- A. Polyhedral decomposition
- B. Object algebras
- C. Cylinders
- D. Ellipsoids
- E. Spheres

A surface representation maybe either a collection of surface points or a partition into a number of primitive patches. In a volumetric representation, on the other hand, the surface is approximated by the visible portions of primitive solid objects which may arbitrarily overlap and combine.

A surface point representation is used in two human models by Fetter (9),

one having 300 points, the other, 3000. Subsets of the body have been digitized as planar polygons by Parke (21) and Catmull (6). The advantages of planar polygons are their ability to model detail and the ease of display of the edge network by line-drawing devices. Outweighing these factors, however, are the very large number of such polygons which would be required to model the human body, the cost of hidden line removal or surface shading, and perhaps most importantly, the unrealistic surface deformations caused by joint movements. For example, in an animation of finger flexion in a polygonsurface hand (6), the fingers become thinner as they bend. No provision is made in the graphical data structure to modify the plane vertices as the movements are executed, and specification of the appropriately interpolated transformations which might provide each vertex with a realistic movement would be non-trivial.

Representing curved surfaces by a partition into bicubic or other mathematical curve patches (24) solves some of the problems posed by polygonal patches. An experimental human model constructed of curved patches has been animated by Wessler (30). The number of patches can be drastically reduced and the surfaces are inherently smooth, avoiding the intensity smoothing which gives polyhedra the appearance of uniform curvature (5,12). Hidden surface removal is, however, no easier (7). Otherwise a grid network must be displayed and this may be taxing on the observer who must interpret the image and perhaps on the display unit, too. The joint deformation problem is potentially solvable, since the patches will adjust to changes in their boundary curves, but certain representations could produce singularities or strange shapes if deformed past certain limits.

The failures of surface partitioning schemes are partially rectified by using a volumetric representation. Ryan (25) constructs a model based on six-sided polyhedra (usually rectangular prisms). Object algebras such as described by Braid (4) or Goldstein (11) include in the primitive set curved solids such as cylinders, ellipsoids, and spheres, although by their generality do not lend themselves to computationally simple display techniques.

Models are in use which are based on cylindrical volumes: Evans uses them in the Combinan model (8), and Potter and Wilmert in the Calspan model (22). Cylinders are a natural component in an object such as the body where many axes already exist, although "blobby" parts such as the head or hand may be hard to model. A rather small number of cylinders is therefore sufficient. The real difficulty occurs when the cylinder caps are smoothed at the joints. The Calspan model takes special pains to achieve rounded corners, but the best results are nevertheless obtained only from side or front views. Hidden parts of the cylinders are not removed either.

The cylinder end problem can be solved by abandoning cylinders for ellipsoids. Herbison-Evans (13) uses such a model to produce animated cartoons of human figures. He shows that hidden parts of the ellipsoid can be removed with a modest amount of computation since ellipsoids orthogonally project into ellipses. That advantage, however, precludes using a perspective projection which is often an important depth cue. Not all body parts are well approximated by ellipsoids either, and surface shading would also be costly.

The projection problem is solved if a primitive is used whose form is unchanged from any view or perspective. The only primitive meeting this

criterion is the sphere since its projection is always a circle or disk. If the body is decomposed into overlapping spheres, non-cylindrical parts can be modelled as easily as cylindrical ones. Directionality is rightfully a property of sets of spherical primitives and is not imposed by the representation itself. Although the number of spheres required to represent the body is greater than the number of cylinders, there are compensating savings in display computation and hidden surface removal. Joint deformation and smoothing problems also disappear. Since this is our proposal for a new body model, we will describe its properties in detail.

#### 3. The Spherical Model

Because of the difficulties of the other modelling methods, we are representing the human body as a collection of overlapping spheres. These are associated with a "skeleton" structure of segments and articulating joints. The desirable properties of this model include its ability to handle joint deformations, generate solid-appearing displays, allow for automatic decomposition of objects into spheres, and permit efficient collision detection. A modest number of spheres suffices for a complete body model. Our present model consists of about 310 spheres (Fig. 1), but 60 have been used for a satisfactory rendering. More spheres may be added for increased smoothness or anatomical accuracy.

The skeletal structure of the body is used as a framework for the set of spheres so that the problems associated with pure surface representations are avoided. In fact, if a sphere is placed exactly at a joint of the body, then the two adjoining body segments join smoothly regardless of their relative orientation. Unlike cylindrical approximations, the sphere model can be easily refined to any desired degree, and because spheres have no inherent directionality, they are better than cylinders for modelling shapes such as the head.

When a sphere is projected onto the two-dimensional viewing plane it always appears as a disk or circle. In an orthogonal projection the radii remain fixed, while in a perspective projection the radii decrease in inverse proportion to the depth. In no case is the circular boundary affected. There are efficient methods for drawing disks or circles (1,14).

The spherical representation offers economical hidden surface removal and shading effects on a raster display. After transformation into the

viewing coordiante system (19), hidden surface removal may be effected by a variety of methods. The simplest is a variant of the depth-priority algorithm (18). Each sphere is written into a frame buffer as a solid disk, and the depth value of its frontmost point is stored as well. As each sphere is processed (the order is arbitrary), only those picture elements with a lesser (closer) depth are inserted, thus parts of spheres which are hidden are overwritten in the frame buffer by the spheres in front. By changing the gray value of the disk according to its depth in space from the observer to make closer spheres lighter and distant spheres darker, depth cueing and shading effects are generated. As long as the body decomposition is fine enough, overlapping spheres will tend to be close in depth and therefore have nearly identical gray values. The result is a quite smoothly shaded picture (Fig. 2). Notice that the surface of each sphere need not be shaded to appear curved, it is sufficient to render the disk uniformly.

An anti-aliasing algorithm for disks can be used to soften the edges and effectively double the apparent resolution. Our algorithm computes the pixel area covered at each disk boundary and shades that pixel value in a proporitionalamount. Figure 3 shows the result of anti-aliasing the display of Fig. 2. Since the resolution of the Ramtek display used to produce these images is only 240 rows by 320 points (the body covering only about half that height in these figures) we are effectively using a very low resolution (and therefore less expensive) graphics display.

An alternative display technique, but which involves more computation, is an adaptation of Watkins' hidden surface algorithm (29). The display is produced a scan line at a time by computing only the frontmost, visible







-

Figure 2. Human body model drawn with solid disks.



Figure 3. Human body model drawn with solid disks and anti-aliased.

portions of the spheres intersecting the current horizontal cutting plane. A modification of this technique has been programmed in an attempt to generate outline drawings of the body model (20) for a vector drawing display. The idea is to use the structure of the body to decide when a sphere boundary should be ignored or treated as a visible edge. Thus an arm positioned across the front of the body should appear in outline and properly occluding the body edges, yet visibly connected to the body (Fig. 4). More work needs to be done on this algorithm to improve output quality; its cost, however, probably precludes it from use in real-time animation anyway.

For a vector drawing display each sphere is drawn as a circle. In spite of the fact that no hidden surface are removed, depth cueing on the circle intensities and animation combine to produce an excellent three-dimensional effect. Since circle drawing generates a sizable number of graphic commands, a display with a built-in circle generator is preferred. One such graphic display, the Vector General 3404 (28), not only provides such a circle generator, but also allows the circles to be positioned in any depth plane and intensity controlled automatically.

As with most surface or volume representations, the most difficult step is data entry. We have implemented an algorithm for the decomposition of curved surface objects into overlapping spheres (20). Starting with any number of cross-sectional outlines of a portion of the object, the algorithm provides a set of spheres to fit the surface (as described by the outlines) within some tolerance. The maximum and minimum sizes for the spheres can also be specified. The algorithm has been used to construct part of our human body model and could be employed to produce other models with different body frames. For example, two skull x-rays were used to digitize three crosssections. (Fig. 5). One particular decomposition











#### 4. The Body Model

In order to position the model for a display a segment structure is used as a "skeleton." This consists of a number of body segments connected by joints where each segment is conceptually a three-dimensional mass of arbitrary shape defined by a set of spheres, and each joint is the point where two segments connect. Sliding joints, such as in the human shoulder, are not permitted since the joint must remain fixed relative to each of the adjacent segments. Each joint may connect only two segments, but a single segment may have connections through any number of joints. The current model uses 20 segments and 19 joints without articulating fingers and toes. The model may be easily extended, however, since all structural parameters are in generalizable arrays. The stage or ground is considered a segment so that the model may be related to the environment.

The segments are organized into a tree structure with segments as nodes and joints as edges. One segment is designated as the <u>reference</u> <u>segment</u> and becomes the root of the tree. An important feature of this organization is the ability to deal with a connected subset of the tree. By specifying a subset of the set of segments as a <u>logical window</u> (as opposed to the viewing window of the actual display) the user can restrict the model in order to examine some particular body area. Only segments within the logical window or which lie along a path from the reference segment to some segment within the logical window are retained.

The shape and size of each segment is described independently of the other segments. For each segment we associated a local coordinate system, which is rigidly embedded within the segment and moves together with it. The origin of a local coordinate system is at the center of gravity of its

segment and coordinate systems are all oriented so that with the human standing upright and feet flat, toes forward, hands at sides and palms toward thighs, the z-axis points upward, the x-axis points forward, and the y-axis points to the left. Using these local coordinate systems, joint locations and the shape of the segment can be determined. Each joint is completely specified by giving its coordinates in the coordinate systems of the adjacent segments. The external surface shape of a segment is the union of surfaces of the spheres, each defined by a center and radius in the local coordinate system. Adjacent segments cannot physically address the same spheres, but the spheres from one segment may arbitrarily overlap those from another.

With each local coordinate system we can associate a table specifying the directions for back, front, top, bottom, right, and left for the segment itself. Thus for the right hand segment "back" is -y, "top" is -z, and "left" is x. This allows any point on the surface of the segment to be "named" by a local direction independent of the segment's orientation in space. Such a feature is very useful for describing points or collision or contact in the model.

In addition to local direction naming, a sphere can be optionally named as a specific subpart of a segment (perhaps restricted to a certain direction). Thus we could label the spheres at each joint or name exterior features such as nose, ear, heel, or fingernail. In each case we can indicate the central directionality of each feature in a table: for example, the sphere for a fingernail would be associated with the "back" direction of its segment.

When a new position is desired, the angles at each joint and the position in space of the reference segment are specified. Because all the segments

are part of a tree with the reference segment as the root, this information unambiguously determines the position and orientation in space of each body segment. To actually compute the global coordinates of the spheres, which are only defined in their local segment coordinate systems, transformations are chained along the paths of the tree. To facilitate this process homogeneous coordinates are used. Translations (from segment center of gravity to joint) are represented as  $4 \times 4$  matrices so that transformation chaining can be achieved by matrix multiplication. The result of chaining transformations out to a segment is a single  $4 \times 4$  matrix which when multiplied by the local coordinates of a point in the segment will yield the global coordinates. Once a position for the body is established, the transformation for a segment can be used for all the spheres defined for that segment.

The only exception to the application of the final transformation to the whole segment occurs if the segment itself is twisted along some axis as happens in the lower legs, forearms, and torso. In that case the sphere centers are transformed by a rotation proportional to the distance of their centers along the axis, that is, one end is not rotated about the axis at all, while the other end is rotated the full amount. The effect will be visible only when the segment is not perfectly circular in cross-section along the axis.

When successive positions of the body must be computed, as in an animation, some computation is avoided if the orientation of some joint is unchanged. The associated segment-to-segment transformation does not change either, so when chaining from the reference segment outward, the transformations need to be recomputed only after the first changed joint

orientation is encountered. From that joint onward, the remainder of the tree path must be updated.

#### 5. Collision Detection

One problem which is nontrivial in most object representations is the detection and localization of collisions (intersections) or contact between parts. Moreover, some collisions are "legal" in the sense that two adjacent segments will intersect to a varying degree about the common joint. These problems can be easily solved within the proposed model.

First, consider collisions between the model and itself. For adjacent segments we can define joint stops to give numerical limits to the segment orientation at a joint. For convenience this can be taken to be a cone, but more complex dependencies could be used. Illegal collision of adjacent segments therefore only requires a simple angle check, or at worst a function evaluation for a complex joint stop function.

The power of the segment and sphere model solves the case of non-adjacent segments. Let D denote the distance between the closest surfaces of two spheres defined as the distance between their centers less the sum of their radii. If D is zero, the spheres intersect at only one point, if D is negative the spheres share some non-zero volume, and if D is positive the spheres are disjoint. The collision detection algorithm follows easily. We associate a sphere or box with each body segment which encloses all spheres in its definition. To check whether two segments intersect, check whether their enclosing spheres or boxes overlap. If not, their individual spheres cannot intersect. Otherwise check whether any sphere of one of the segments (the one with fewer spheres) intersects the enclosing sphere or box of the other. If not, the segments cannot intersect. Otherwise check each sphere in one segment against each sphere of the other. Any intersections are determined at this point. These can be used as feedback to the simulation

program, or else simply printed in a textual report using the direction and subpart naming tables and changing the distance value to indicate contact or overlap.

It is a straightforward process to extend collision detection to one model against another spherically represented object or body. What is perhaps more surprising is that the process can be easily extended to check sphere intersections with object surfaces described by planar polygons. Sphere to plane distances are easy to compute. Should a sphere intersect a plane the problem reduces to testing a circle against a polygon. If the sphere intersects or lies within the minimum enclosing circle (or box) of the polygon, further tests determine whether the circle intersects any edge. If so, then we are done; if not, then the circle either lies totally within or totally outside the polygon. If outside, the original sphere does not intersect; if inside, the sphere does intersect. The resulting list of sphere-polygon intersections or contacts can now be used in the same fashion as the sphere-sphere collisions.

## 5. Applications

We are using the human body model in two research projects: one to transform movement notation into a graphic animation (26), the other to combine the human model with existing cockpit design and crash simulation systems. These two applications complement each other in the sense that the first involves simulation of all potential skeletally-mediated body movements, while the second manipulates the body in response to external physical forces or environmental constraints.

The movement simulation system involves the compilation of a wellstructured symbolic movement notation, Labanotation (16), into a set of "primitive" movement concepts: directional movements or positions, rotations or twists, surface facing orientations, paths in space or shape configurations of body parts, and contacts or relationships between the body and itself, other people, or its environment (27). Instances of these concepts are presented as instructions to a simulator modelled as a collection of parallel processors, one for each joint of the body, and an additional processor for movements of the whole body. A monitor synchronizes and schedules the processors and supervises changes to the body position data base (3). The simulation output is a sequence of "snapshots" or frames of the body model performing the notated movements. By storing the joint angles in a data file, the model may be animated in real-time on a vector drawing display such as the Vector General 3404, since the necessary circle generator and transformation chaining are provided in firmware. Implementation of the simulator and the real-time display is currently in progress.

For the other applications, we are integrating the outputs of two systems with the body model. A cockpit design and evaluation program (25) provides

a planar polygon environment, while a crash simulator provides hypothetical movements of a vehicle occupant under various deceleration conditions (10). A typical cockpit is shown in Fig. 7, shaded to better illustrate the surfaces. Four frames from a typical sequence (Fig. 8) of a simulated crash (head-on at an oblique angle, with the occupant restrained only be a lap belt) may be drawn on a raster display. By registering the initial configuration of the body in the vehicle seat (Fig. 9), both may be displayed simultaneously throughout the simulated period (17). During this interval, a listing of the body collisions and contacts at any one time may be obtained from the algorithm previously described. Moreover, this listing indicates the specific body areas and object surfaces affected, since the planar surfaces are named by the vehicle designer.

Although the integration of body and vehicle may eventually be used to control the simulation, a more likely short-term goal is the presentation of the simulation data in a readily intepretable form. By generating each frame of the simulation and recording it on videotape, a permanent visual record may be obtained of any simulated movements. During the recording process, the real time between each frame may be extended to slow the actual motion. With enough care in the positioning of the observer for the computer generated sequence of images, actual experimental data from high-speed photography may be superimposed, allowing the simulation designer to verify the accuracy of the simulation program.













#### 6. Conclusions

The conceptual simplicity of the spherical decomposition of the human body provides many advantages to the user in hidden surface removal, collision detection, part naming, and device independent display. While a fast vector drawing device would be used primarily for preview or outline animation, we are particularly hopeful that the raster-based display will provide, for the first time, the possibility of saving computer simulation data on video tape or cassettes so that animation playback can be done offline without typing up large computers or expensive display devices.

Acknowledgement

The work reported here was supported in part by National Science Foundation Grants No. ENG75-10535 and MCS-76-19464 and ONR Contract N00014-78-C-0102. Development of the human body model has been considerably aided by artistic assistance from Michael O'Rourke. We also wish to thank Jim Weissman for the anti-aliasing algorithm, David Bourne for the skull data, and Bill Wood for the shaded surface algorithm.

#### References

- 1. Badler, N.I.; Disk generators for a raster display device, Computer Graphics and Image Processing 6, 1977, pp. 589-593.
- 2. Badler, N. and Bajcsy, R.: Three-dimensional representations for computer graphics and computer vision. <u>Computer Graphics</u>, <u>A Quarterly Journal</u>, Summer 1978.
- 3. Badler, N.I., O'Rourke, J., Smoliar, S.W., and Weber, L. The simulation of human movement by computer. Movement Project Report No. 14, Department of Computer and Information Science, University of Pennsylvania, July 1978.
- Braid, I.C.: The synthesis of solids bounded by many faces. <u>Comm. ACM</u> 18 (4), April 1975, pp. 209-216.
- 5. Bui Tuong Phong. Illumination for computer generated images. <u>Comm. ACM</u> 18 (6), June 1975, pp. 311-317.
- 6. Catmull, E.: A system for computer generated movies. <u>Proc. ACM Annual</u> Conf., vol. 1, 1972, 422-431.
- 7. Catmull, E.: Computer display of curved surfaces. Proc. Conf. on Computer Graphics, Pattern Recognition, and Data Structure, Los Angeles, CA, May 1975, pp. 11-17.
- 8. Evans, S.M.: User's guide for the programs of Combiman. Univ. of Dayton Research Institute. Report No. AMRL-TR-76-117, Univ. of Dayton, Dayton, Ohio, 1976.
- 9. Fetter, W.A.: A human figure computer graphics development. Proc. Eurocomp Congress, May 1974, Online, Brunel, England, 1974, pp. 476-488.
- 10. Fleck, J.T., Butler, F.E. and Volgel, S.L.: An improved three dimensional computer simulation of crash victims. Volume 1-Volume 4, Final report for contract No. DOT-HS-053-2-485, DOT Report Nos. DOT-HS-801, 507 through 510, NHTSA, April 1975.
- 11. Goldstein, R.A. and Nagel, R.: 3-D visual simulation. <u>Simulation</u> <u>16</u> (1), January 1971, pp. 25-30.
- 12. Gouraud, H.: Continuous shading of curved surfaces. <u>IEEE Trans. on</u> Computers, Vol. C-20, No. 6, June 1971.
- 13. Herbison-Evans, D.: NUDES 2: A Numeric Utility Displaying Ellipsoidal Solids, Version 2. <u>Computer Graphics, A Quarterly Journal</u>, Summer 1978.
- 14. Horn, B.K.P.: Circle generators for display devices. <u>Computer Graphics</u> and Image Processing 5, 1976, pp. 280-288.

- Huston, R.L., Passerello, C.E., Harlow, M.W. and Winget, J.M.; The UCIN 3-D aircraft-occupant. <u>Aircraft Crashworthiness</u>, Saczalski et al., eds. Univ. Press of Virginia, Charlottesville, VA, 1975.
- 16. Hutchinson, A.: Labanotation. Second Edition, Theatre Arts Books, New York, 1960.
- 17. Isaacson, H. Spheres in polygon environments, MSc Thesis, Department of Computer and Information Science, University of Pennsylvania, 1978 (forthcoming).
- 18. Newell, M.E., Newell, R.G. and Sancha, T.L.: A solution to the hidden surface problem. <u>Proc. ACM Annual Conf.</u>, Vol. 1, 1972, pp. 443-450.
- 19. Newman, W. and Sproull, R. <u>Principles of Interactive Computer Graphics</u>. McGraw-Hill, 1973.
- 20. O'Rourke, J.: Representation and display of three dimensional objects with spheres. MSc Thesis, Dept. of Computer and Information Science, University of Pennsylvania, Philadelphia, PA, August 1977.
- 21. Parke, F.I.: Animation of faces. Proc. ACM Annual Conf. vol. 1, 1972, pp. 451-457.
- 22. Potter, T.E. and Willmert, K.D.: Three-dimensional human display model. Computer Graphics 9 (1), Spring 1975, pp. 102-110.
- 23. Robbins, D.H., Bennett, R.O., Jr. and Bowman, B.: User-oriented mathematical crash victim simulator. Highway Safety Research Institute, The Univ. of Michigan, SAE Paper No. 720962, <u>Proc. 16th</u> Stapp Car Crash Conf., 1972, pp. 128-148.
- 24. Rogers, D.F. and Adams, J.A.: <u>Mathematical Elements for Computer Graphics</u>. McGraw-Hill, New York, 1976.
- Ryan, P.W.: Cockpit geometry evaluation---Phase III final report: Volume 1: Program description and summary. Boeing, under ONR Contract N00014-71-C-0170, Boeing Report number D162-10125-3.
- 26. Smoliar, S.W. and Weber, L.: Using the computer for a semantic representation of Labanotation. In <u>Computing and the Humanities</u>, S. Lusignan and J.S. North (Eds.), University of Waterloo Press, 1977, pp. 253-261.
- 27. Smoliar, S.W., Weber, L., and Badler, N. An architecture for the simulation of human movement, Movement Project Report No. 8, Department of Computer and Information Science, University of Pennsylvania, 1977.

- 28. Vector General System 3404 Reference Manual, Vector General Corp., Woodland Hills, CA, 1977.
- 29. Watkins, G.S.: A real-time visible surface algorithm. Tech. Report UTEC-CSC-70-101, Dept. of Computer Science, University of Utah, Salt Lake City, UT, June 1970.
- 30. Wessler, Barry. Computer-assisted visual communication. Ph.D. Dissertation, Department of Computer Science, University of Utah, August 1973.