4-1979

Representation of Articulable, Quasi-Rigid, Three-Dimensional Objects

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

Joseph O'Rourke

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-79-24.

Representation of Articulable, Quasi-Rigid, Three-Dimensional Objects

Abstract

Many three-dimensional objects consist of quasi-rigid pieces connected at articulable joints. The choice of a suitable three-dimensional representation for such objects, for example, the human body, is considered with respect to properties required during movement of the jointed segments. We discuss the coordinate systems involved, representation of the surface of the object, and maintenance of proper segment orientation during movement. Generalization to more complex forms is discussed.

Disciplines

Computer Engineering | Computer Sciences

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-79-24.

MS-CIS-79-24

Representation of Articulable, Quasi-Rigid, Three-Dimensional Objects

Norman I. Badler
Joseph O'Rourke

April 1979

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

Presented at the NSF sponsored Workshop on Three-Dimensional Object Representations
University of Pennsylvania, April 1979

Abstract

Many three-dimensional objects consist of quasi-rigid pieces connected at articulable joints. The choice of a suitable three-dimensional representation for such objects, for example, the human body, is considered with respect to properties required during movement of the jointed segments. We discuss the coordinate systems involved, representation of the surface of the object, and maintenance of proper segment orientation during movement. Generalization to more complex forms is discussed.

The Representation

As noted in the Introduction, we shall limit our discussion to movements which are skeletal, that is, which can be realized by a model composed of joints and segments. Figure 1 is a representation of the human body skeleton where segments are drawn as lines connecting joint nodes. We assume the object is decomposable into a tree structure.

It will be useful to define a standard position for the body in order to fix certain coordinate system relationships which we shall use later. The standard position for the human body will be standing upright, feet flat, toes forward, and hands at sides, palms toward thighs (Figure 2). Initially, we assume the body is standing on the "ground," at the origin of a rectangular coordinate system. In this position, the Z-axis of the environment points "up" (that is, opposed to gravity), the X-axis points "forward," and the Y-axis points "left." We shall frequently refer to world coordinates as "global" or "absolute."

Segments provide the rigid but articulated skeletal framework of the body and determine its external (surface) appearance. These functions are incorporated into the "segment" definition whose fields describe the relationship between the joints connected by a segment and also store information related to physical description of the associated body surface:

```
type segment = (name:  sequence character;
                proximal:  joint;
                distal:  joint;
                distance:  real;
                local directions: (forward:  vector;
                                   left:  vector;
                                   up:  vector);
```

Introduction

Many three-dimensional object representations appear to be quite adequate for rigid objects, yet become limited or nearly useless when those same objects are quasi-rigid or contain articulable joints. (We shall loosely define quasi-rigid as objects which may change form yet retain some notion of "ends" and a space curve axis which connects them.) We discuss the origin of the representational difficulty and present a solution for one particular type of object: those which consist of quasi-rigid segments connected at articulable point-like joints. The human body is an example of such an object. We shall assume that all body joints are effectively modelled by points, although this is not strictly true (for example, see [9]).

The human body consists of a two-level hierarchy of three-dimensional components: a skeleton of assumed rigid bones and a surface of flexible skin. A representation of the skeleton is easy to achieve with three-dimensional lines, but the resulting form is unsatisfactory in appearance and ambiguous in form and movement [4]. Alternatively, a representation of the surface without the skeleton would make jointed movement difficult to simulate. Given surface points, for example, computation of the trajectories of each point near a joint is at best non-obvious; often this problem is "solved" by ignoring it [4]. Both skeleton and surface are needed to effectively model the adjustment of surface shape to movement of the underlying skeletal structure. To model this hierarchy, we must have a representation for the object segments and joints, the surface and coordinate system of each segment, and well-defined mathematical relations between adjacent segment coordinate systems during movement.

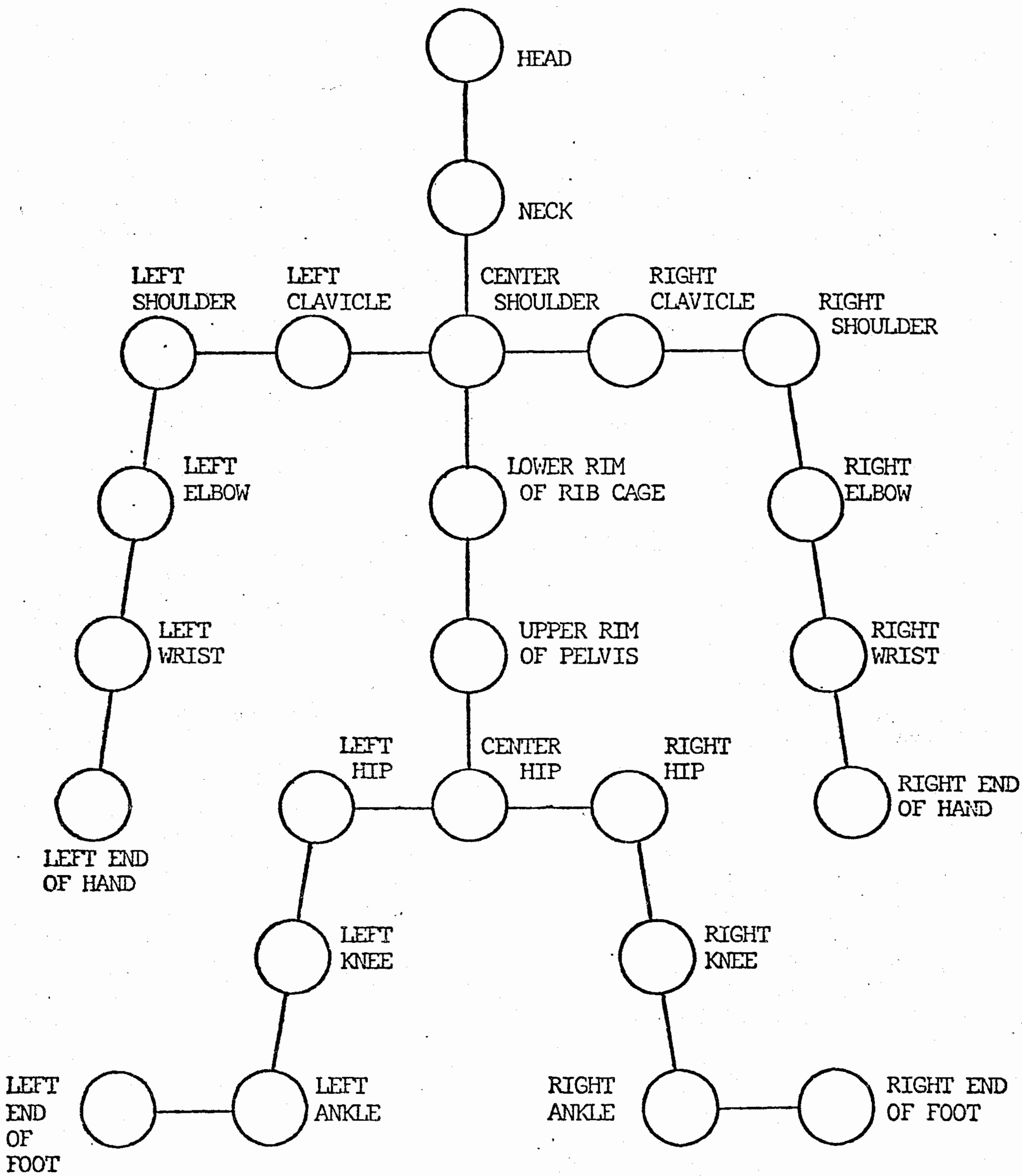


Figure 1. Joints in the body tree.

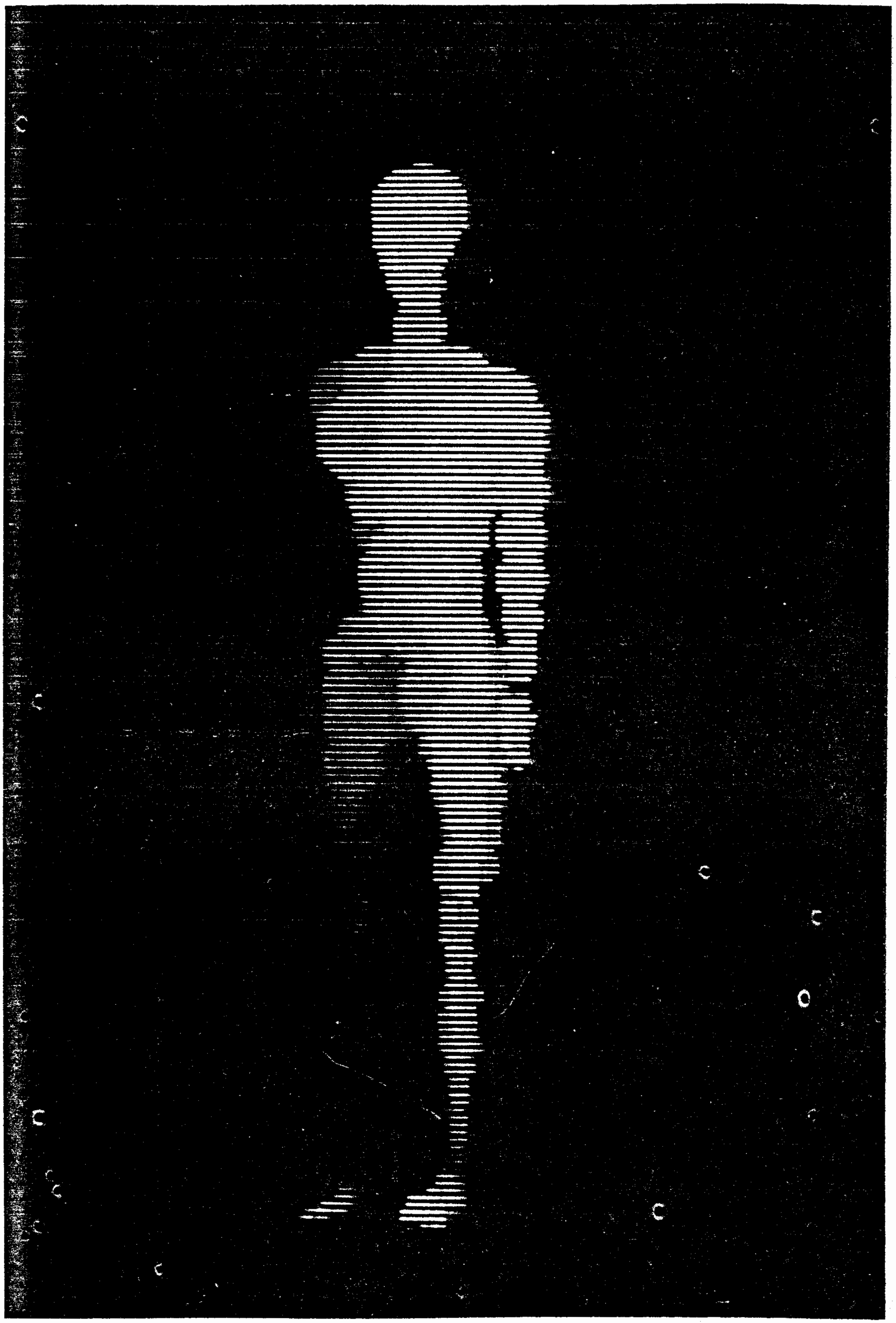


Figure 2. Standard position of the body.

```
                 twist:  (positive limit:  real;
                          negative limit:  real;
                          current:  real);
                 orientation:  vector;
                 stop:  proc (vector) logical;
                 enclosure:  sphere;
                 centroid:  vector;
                 mass:  real;
                 surface:  sequence sphere).

type sphere    = (origin:  vector;
                  radius:  real;
                  features:  sequence feature).

type feature   = (name:  sequence character;
                  direction:  vector).

type joint     = (name:  sequence character;
                  connection:  (proximal:  segment;
                                distal:  powerset segment);
                  location:  (current:  vector;
                              valid:  logical) ).

type vector    = (X:  real;
                  Y:  real;
                  Z:  real).
```

Each segment has an internal name, given as a character string. Of the two joints connected by the segment, the one which lies closer to the root of the body tree (as defined above) is called the proximal joint and the other is called the distal joint. Each joint connects a proximal segment with a non-empty set of distal segments unless it is an extremity. A joint has a character string name and a location which defines its current position in the absolute coordinate system. A logical flag indicates whether or not this location is currently valid, since proximal segments may be moved without immediately recomputing the joint location. (Additional fields in the joint definition are used for movement simulation [2], but these are not relevant to this discussion.)

A cross of axes is defined for a segment such that the proximal joint is the origin and the ray connecting the proximal and distal joints defines the Z-axis direction. The coordinates of the distal joint are therefore (0,0,distance) where "distance" is the length between the joints. The X-axis is chosen to lie in a vertical plane perpendicular to the global Y-axis through the proximal joint and also to have a positive forward component when the body is in the standard position (Figure 3). (If the Z-axis and the front direction coincide, then the X-axis is chosen to point upwards.) The Y-axis is the direction which yields a right-handed coordinate system.

As a segment moves, its cross of axes moves rigidly with it. Certain directions are defined in this local coordinate system to correspond to the conventional "front," "left," and "up" directions of the segment. For example, the negative Z-axis is "up" for the lower arm and the positive X-axis is "front" for the head, independent of the current orientation of these segments in the overall body position. Since a body segment may be capable of twisting along its Z-axis, the cross of axes may be rotated at the distal joint (Figure 4). Positive and negative rotation limits from the normal position of the X-axis (established above) define admissible twists.

A segment's orientation is given as a vector which is the position of the distal joint in the cross of axes of the proximal segment of the proximal joint (Figure 5). Movements of the proximal segment are naturally transmitted to the distal segment through the common joint with no further computation. In addition, a "stop" function specifies whether or not a particular orientation is admissible. It may be used, for example, to limit movement to part of one plane, as at the elbow or knee.

Segments carry information to aid collision detection and support computations. The "enclosure" is the minimum sphere which includes the entire surface of the segment. It is used to approximate the location of the segment when comparing it against other (non-adjacent) segments and is fixed when the body model is

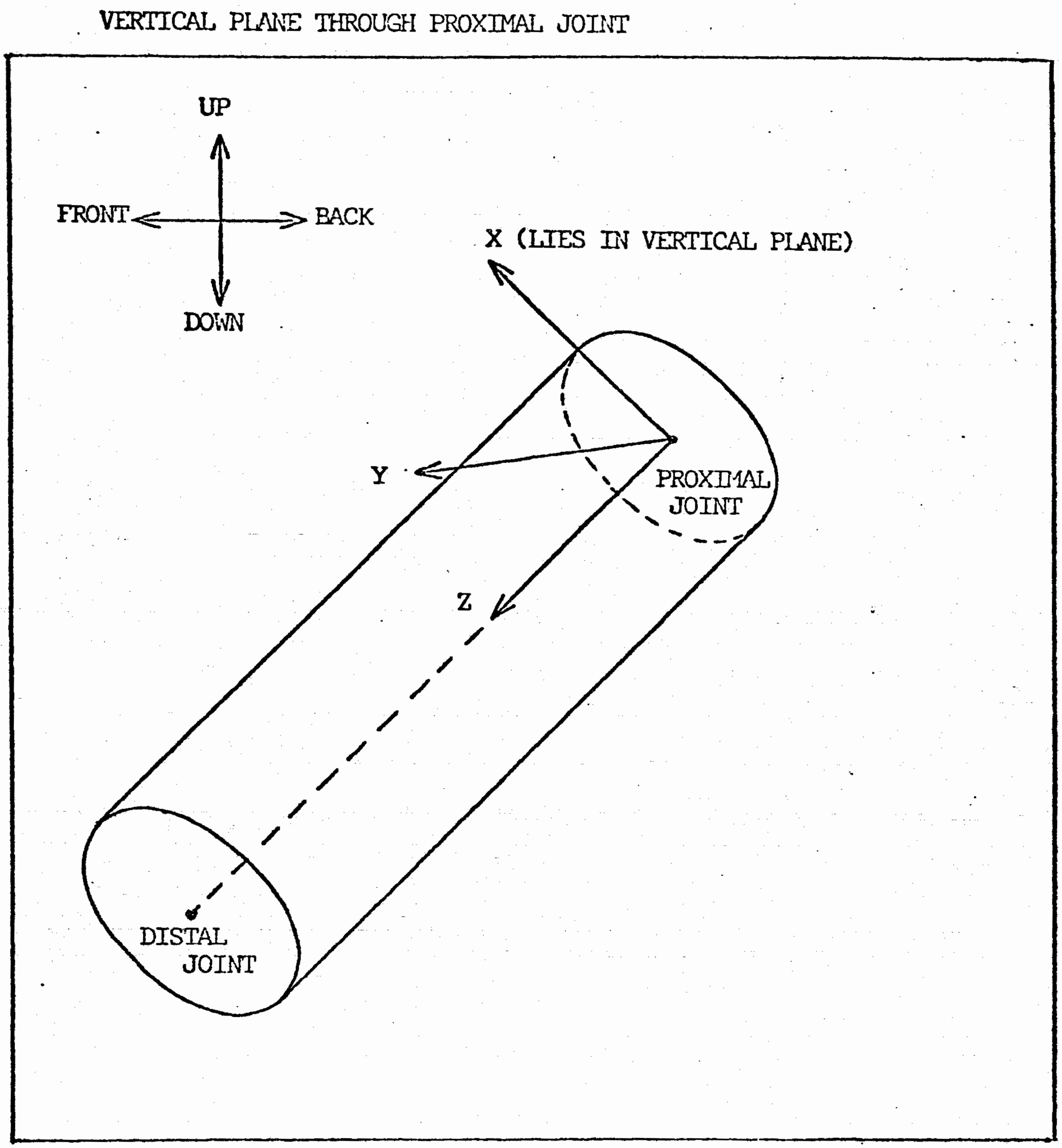


Figure 3. Segment cross of axes.

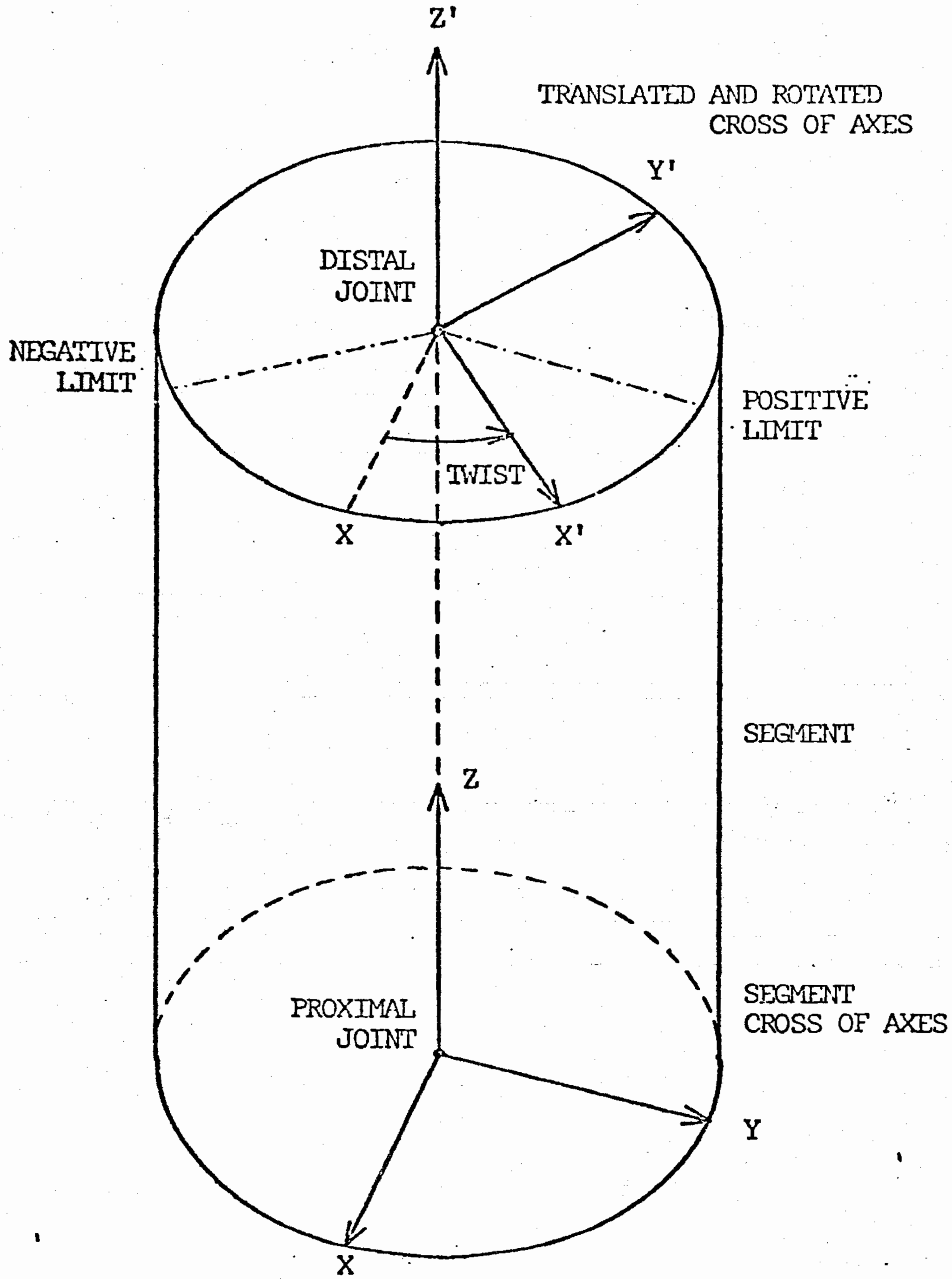


Figure 4. Effect of segment twist on cross of axes.

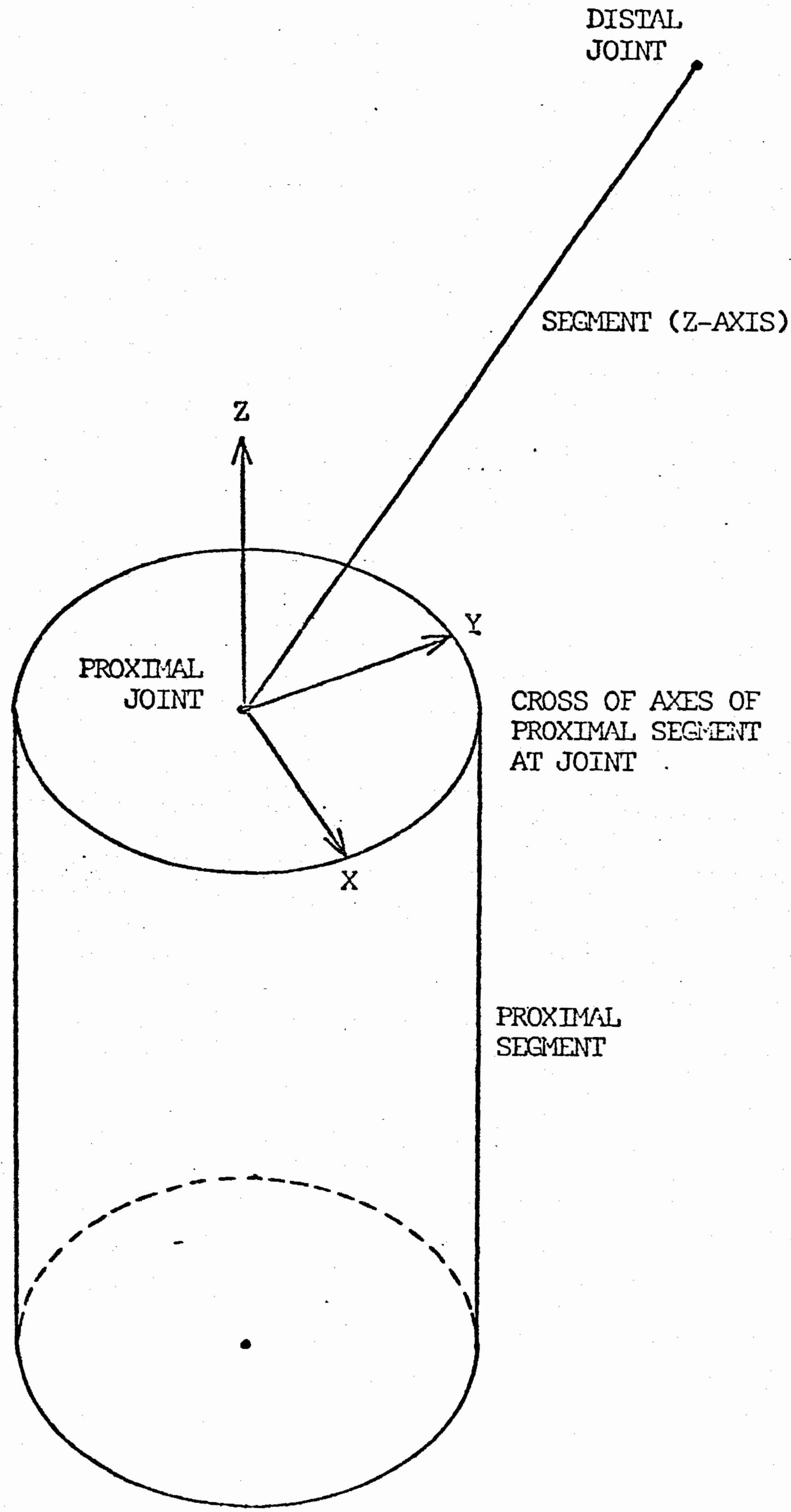


Figure 5. Orientation of segment with respect to proximal segment.

dimensioned [3]. A segment also has a fixed centroid (a vector in the local cross of axes) and a mass value which is used to compute the overall center of gravity for the body.

The body is positioned with respect to the environment through a chain of special instances of joints and segments. The environment is regarded as such a segment establishing the global reference system. It only articulates with a single distal joint named "center of gravity". This joint, in turn, connects the environment segment to a distal segment named "whole body". Finally, this "whole body" segment connects to the distal joint at the root of the body tree, generally the center hip. If necessary, however, any other joint may be substituted as the root, since the tree structure of Figure 1 is undirected. (Figure 6 illustrates these relationships.)

Normally, the center of gravity is the only joint allowed to translate with respect to its adjacent segments, since there are no physical connections between it, the environment, and the "whole body" segment. Other segments may be allowed to change length, provided the information in their definitions is modified accordingly; for example, the "enclosure" of the segment will change and, more importantly, the representation of the surface may require interpolation or augmentation.

The "whole body" segment is a device used to model the relationship between the center of gravity and the visible object segments. No surface definition is associated with the "whole body" segment, but it does have an orientation. It effectively models the "stance" orientation in Labanotation [6], maintaining the notion of "front" direction for the whole body even if individual parts of the body should twist away from that direction during movement.

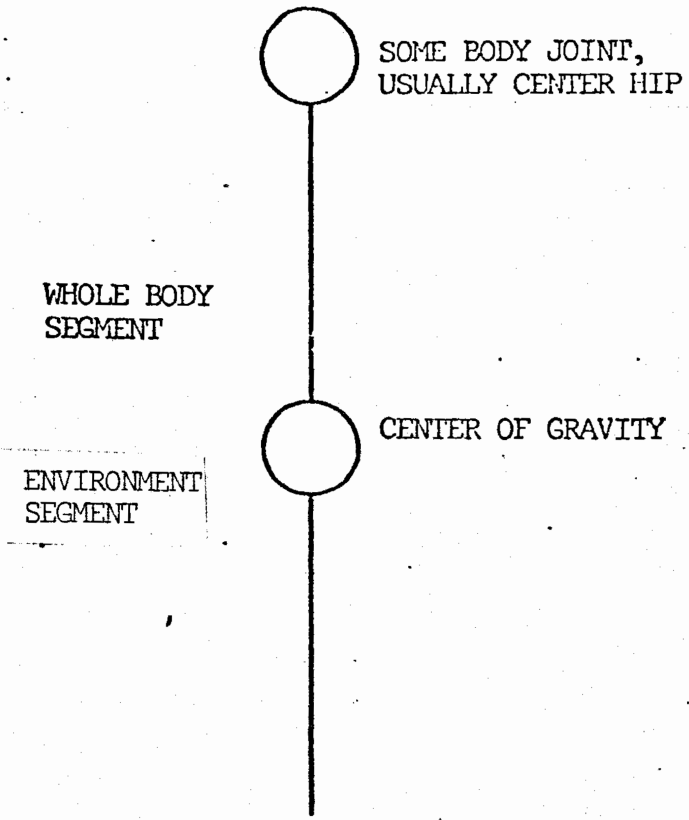


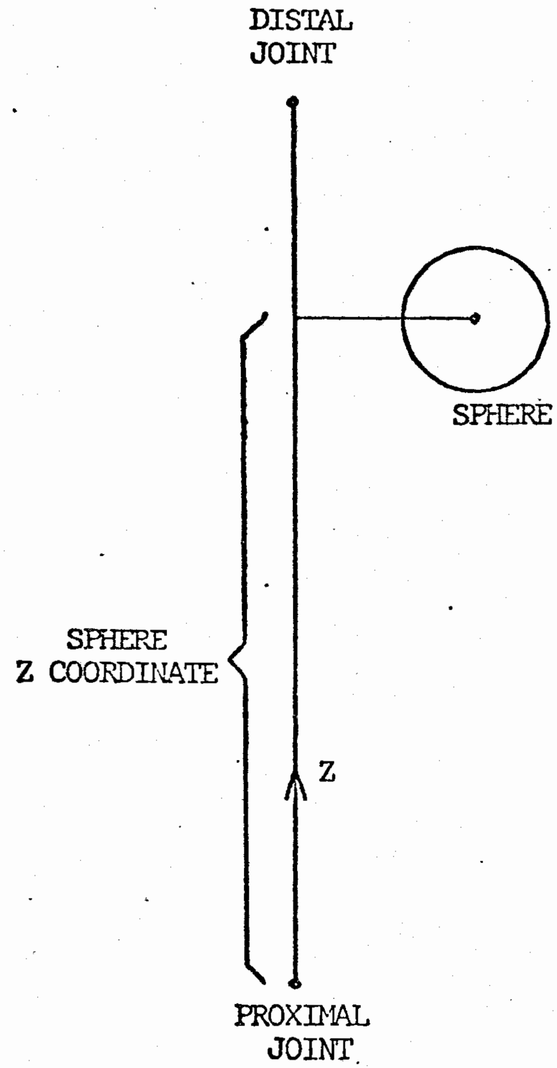
Figure 6. Relating the body to the environment.

The "skin" (surface) of a segment is defined by a set of overlapping spheres [7]. The origin of each sphere is given as a vector in the segment's cross of axes. If the segment is twisted, the sphere center is rotated about the Z-axis by an amount proportional to its distance along the segment from the proximal joint, which is just the Z component of the sphere's "origin" vector (Figure 7). Certain "features" of a segment may be distinguished by giving a point on a specific sphere a name which will allow it to be specified for a contact location or used in collision reports. The feature direction is indicated by a vector in the segment cross of axes; normally this direction from the sphere origin will define the surface perpendicular (Figure 8).

Standard Orientation

One difficulty with defining segment direction in terms of the cross of axes of the proximal segment is the ambiguity in orientation of the segment during movement. Consider the arm as it moves throughout its entire free range. There are certain orientations at each arm position which are more natural than others; in fact, some orientations possible in one arm position are not physically realizable in other positions. We therefore use a "standard" or default orientation at each possible position of the limbs; all other orientations must be explicitly specified as a deviation from the default.

If we imagine the arm moving freely throughout its angular range (without bending at the elbow), then the end of the hand will lie on a sphere centered at the shoulder joint. Of course, there are sections of the sphere which are inaccessible to the arm (due to angular limitations at the shoulder joint), but the entire sphere is required for an arbitrary joint and segment. The standard orientation is represented by a coordinate system at each point on the surface of



$$\text{TWIST OF SPHERE} = \frac{\text{CURRENT TWIST}}{\text{Z OF SPHERE ORIGIN}}$$

Figure 7. Transmitting segment twist to the surface.

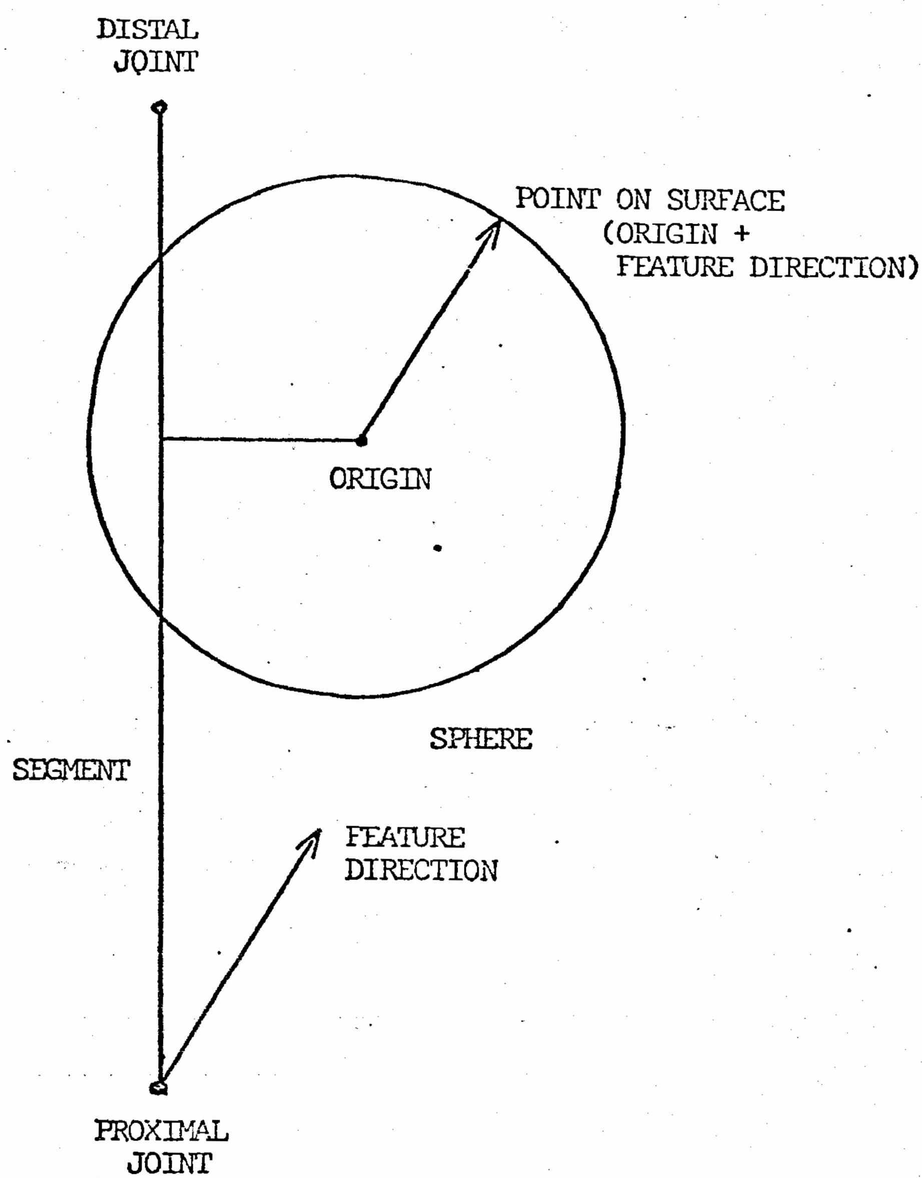


Figure 8. Specifying features.

the sphere: when the end of the hand is positioned at a particular point on the sphere's surface, then the standard orientation is given by the local coordinate system at that point. If the arm's position is specified by the spherical coordinates θ and ϕ, then the standard orientation is a function which assigns to each $<\theta,\phi>$ a particular coordinate system orientation.

Although this standard orientation function could be represented in a program as a table, with interpolation between entries, it would be much more efficient if it could be represented as a analytic function of θ and ϕ. We have found it easiest to work from the orientation which is most "natural" in terms of the matrix multiplications which transform coordinate systems. If we erect a coordinate system at the north pole of the sphere(i:e., at $\theta=0$ and $\phi=0$), and transport it to $<\theta,\phi>$ by twisting it by θ and then sliding it along a meridian by ϕ, then the coordinate system will have some particular orientation dictated by the matrices used in the transformations. We can use this coordinate system as our base, and then specify the standard orientation as some further twist from this base orientation. In this paradigm, the standard orientation can be represented as a function from $<\theta,\phi>$ to t, the further twist needed. For the arm and leg standard orientations (as culled from Labanotation [6]), this function is linear in both θ and ϕ, and only five pieces are required to define it over the entire sphere surface. The actual equations for the left arm are as follows (all angles are expressed in degrees):

θ_range	φ_range	Equation
[-90,90]	[0,90]	t = 180 - θ*(1-φ/90)
[-90,90]	[90,180]	t = -180*(1-θ/90) + (φ-270)*(θ/90)
[90,270]	[90,180]	t = (θ-180)*(1+(180-φ)/90)
[90,180]	[0,90]	t = 2*(θ-180)+φ/90 - (180+θ)*(1-φ/90)
[180,270]	[0,90]	t = (180-θ)*(1-3φ/90)+sgn(225-θ)*360*φ/90

The standard orientations vary continuously over the entire surface of the sphere, except for a seam in the middle of the back upper right octant, where the physical limitations of the joint dictate that the limb should change its direction of twist (figures 9,10, and 11).

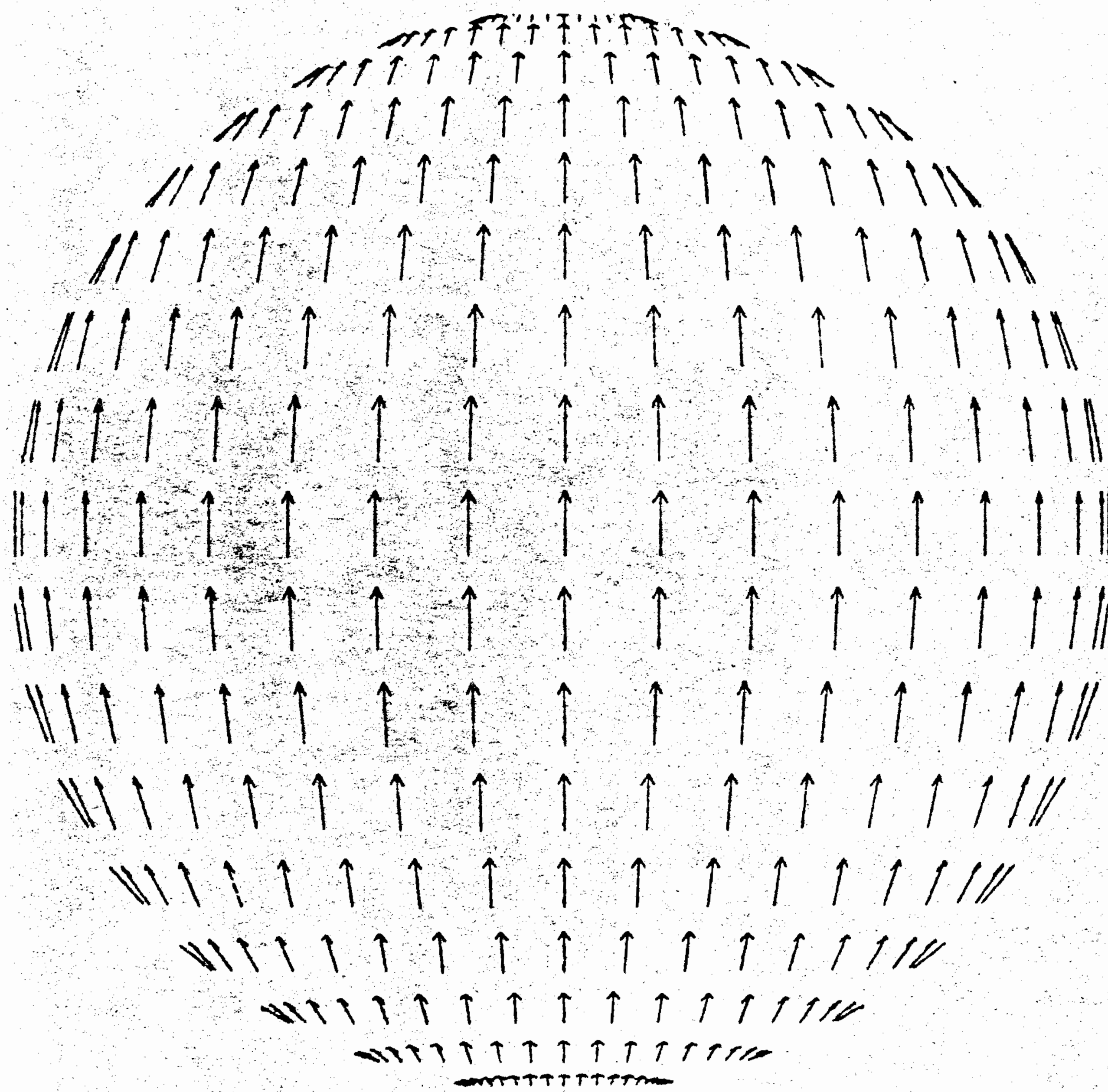


Figure 9. Front view of X-axis of standard orientation function: $-90 \leq \Theta \leq 90$, $0 \leq \phi \leq 180$.

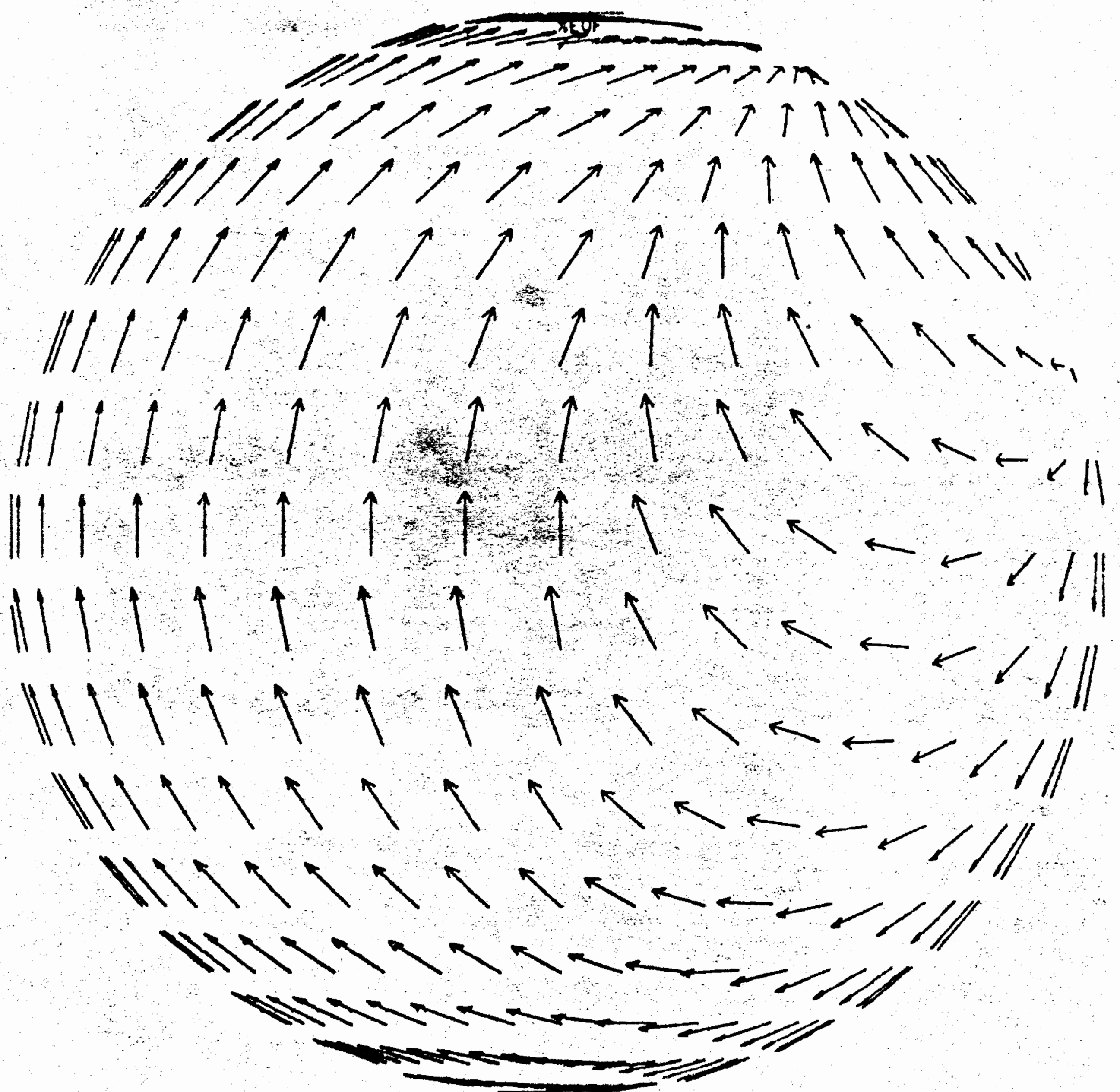


Figure 10. Side view of X-axis of standard orientation function:

$0 \leq \theta \leq 180$, $0 \leq \phi \leq 180$.

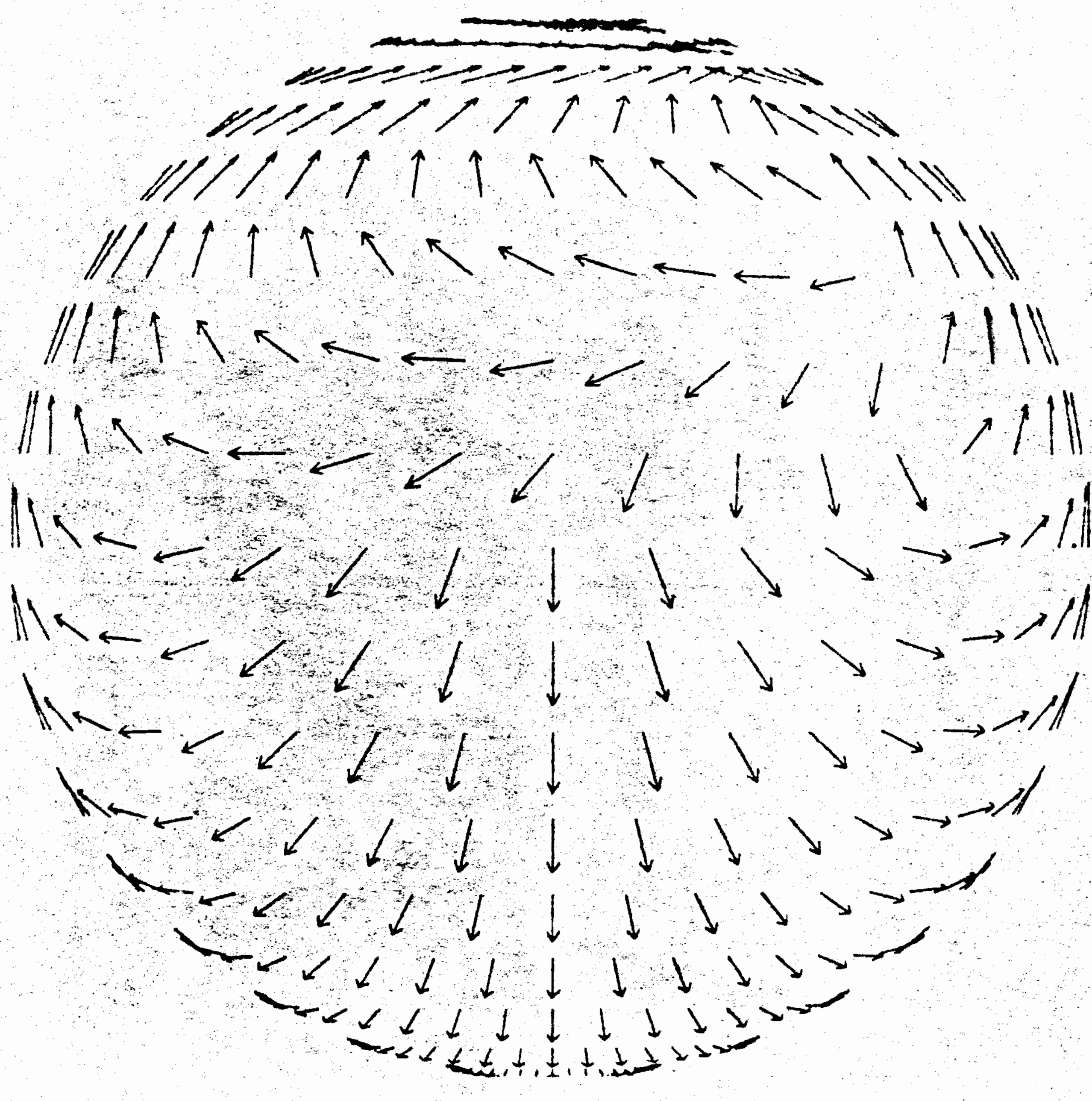


Figure 11. Back view of X axis of standard orientation function: $90 \leq \Theta \leq 270$, $0 \leq \phi \leq 180$. Note seam at $\Theta = 225$, $0 \leq \phi \leq 90$.

The equations for the right arm and the left and right legs are very similar.

When the limbs of the body are moved into some position, the orientation of the segments will in general depend on the path used to reach that position. So whenever the limbs are placed into a position by specifying the joint positions, the resulting orientation of the segments is computed and compared to the orientation given by the standard orientation function, and the segments are then adjusted to agree with the default orientation. If an orientation different from the default is desired, then a further adjustment is made.

Discussion

The advantages of using spheres to model the body surface are reviewed in references [1], [3], and [4]. The primary properties of the sphere representation are the preservation of joint shape during movement of the articulated segments, simple though realistic display of the model on various graphic devices, efficient collision detection between the body and itself or its environment, and existence of an algorithm for constructing spherical decompositions of an object [8].

Another property of spheres described in [1] and utilized in an implementation by O'Rourke [8] is the construction of a three-dimensional medial surface of an object [5]. This surface is the locus of centers of all maximal spheres within the object and (in continuous three-dimensional space with the Euclidean distance metric) consists of unions of isolated points, one-dimensional curves, two-dimensional surfaces, and branch points. We may consider branch points to be potential articulation points between the one- or two-dimensional structures. The body model just described is an example of branch points connecting one-dimensional segment axes. (Of course, these axes approximate the medial axis; there are many branch points which are artifacts of the segment shape and which are not articulated.) We see, however, a method for generalizing the structure:

- Extend the notion of segment to permit arbitrary, but smooth, curves in the segment axis.
- Extend the coordinatization to two-dimensional surfaces such as obtained from the medial transform.

The first is relatively straightforward: define a suitable parametrized space curve between the endpoints and move the coordinate system from one end to the other so that the Z-axis represents the partial derivative of the curved axis and the X- and Y- axes follow the conventions set out in the standard position discussion above.

The second extension is not difficult to imagine if we consider the two-dimensional surface to consist of a union of curved surface patches [10]. The local coordinate system of the patch is well-defined as long as the patch is locally smooth: two axes are determined by the partial derivatives in the original grid parameters at a point on the surface, the third (Z-axis) is the perpendicular to the surface at that point.

The reason for using the medial surface is to permit representation of the three-dimensional object surface by unions of spheres. For space curves, the spheres are defined as if the segment were linear; the function which transforms the line into the space curve also transforms the sphere centers. Likewise, the sphere centers may be defined on a square grid for a two-dimensional surface; if the grid is deformed by any of the various three-dimensional curved surface patch algorithms, the sphere centers will be likewise transformed. A good example of this situation arises with the palm of the hand: its flexibility might be modelled by a curved patch with articulation points for the wrist, fingers, and thumb. Notice that we need not consider patch-to-patch joints, only patch-to-space curve connections. The former would be handled by conventional patch boundary constraints and computation methods.

Conclusions

We have presented a method for representing three-dimensional articulated objects based on a hierarchy of linear skeletons and sphere surface unions and have demonstrated the usefulness of the method in modelling the human body. The method seems extensible to objects composed of more general "skeletons," namely those consisting of unions of space curves and curved surface patches. The coordinatization of articulated segments and means for extending it to the more general case were described. A standard orientation function to preserve segment orientation during movement was formulated. We are hopeful that the methodology proposed here will provide representations for movable and articulable objects and will prove descriptively and analytically useful in computer vision as well as in computer graphics.

Acknowledgement

This research is supported in part by NSF Grants MCS76-19464 and MCS78-07466.

References

[1] N.I. Badler and R. Bajcsy, "Three-dimensional representations for computer graphics and computer vision," Computer Graphics, vol. 12, pp. 153-160, August 1978.

[2] N.I. Badler, J. O'Rourke, S.W. Smoliar, and L. Weber, "The simulation of human movement by computer," Movement Project Report No. 14, Department of Computer and Information Science, University of Pennsylvania, July 1978.

[3] N.I. Badler, J. O'Rourke, and H. Toltzis, "A spherical representation of a human body for visualizing movement," to appear in IEEE Proceedings, 1979-80.

[4] N.I. Badler and S.W. Smoliar, "Digital representations of human movement," Computing Surveys, vol. 11, pp. 19-38, March 1979.

[5] H. Blum, "Biological shape and visual science," J. Theoretical Biology, vol. 38, pp. 205-287, 1973.

[6] A. Hutchinson, Labanotation. New York, NY: Theatre Arts Books, Second Edition, 1960.

[7] J. O'Rourke, "Representation and display of three-dimensional objects with spheres," MSE Thesis, Department of Computer and Information Science, University of Pennsylvania, August 1977.

[8] J. O'Rourke and N.I. Badler, "Decomposition of three-dimensional objects into spheres," to appear in IEEE-PAMI, vol. 1, July 1979.

[9] F.I. Parke and M. Friedell, "Interactive simulation of biomechanical systems: The kinematics and stresses of the human knee," in Proc.ACM Annual Conference, 1978, vol. 2, pp. 759-764.

[10] D.F. Rogers and J.A. Adams, Mathematical Elements for Computer Graphics. New York, NY: McGraw-Hill Book Co., 1976.