GLOBAL AND LOCAL DEFORMATIONS OF SOLID PRIMITIVES

Alan H. Barr
Computer Science Department †
California Institute of Technology
Pasadena, California

Abstract

New hierarchical solid modeling operations are developed, which simulate twisting, bending, tapering, or similar transformations of geometric objects. The chief result is that the normal vector of an arbitrarily deformed smooth surface can be calculated directly from the surface normal vector of the undeformed surface and a transformation matrix. Deformations are easily combined in a hierarchical structure, creating complex objects from simpler ones. The position vectors and normal vectors in the simpler objects are used to calculate the position and normal vectors in the more complex forms; each level in the deformation hierarchy requires an additional matrix multiply for the normal vector calculation. Deformations are important and highly intuitive operations which ease the control and rendering of large families of three-dimensional geometric shapes.

KEYWORDS: Computational Geometry, Solid Modeling, Deformation

Introduction

Modeling hierarchies are a convenient and efficient way to represent geometric objects, allowing users to combine simpler graphical primitives and operators into more complex forms. The leaf-nodes in the hierarchy are the hardware/firmware commands on the equipment which draws the vectors, changes the colors of individual pixels, and operates on lists of line segments or polygons. With the appropriate algorithms and interfaces, users can develop a strong intuitive feeling for the results of a manipulation, can think in terms of each operation, and are able to create the objects and scenes which they desire.

In this paper, we introduce globally and locally defined deformations as new hierarchical operations for use in solid modeling. These operations extend the conventional operations of rotation, translation, Boolean union, intersection and difference. In section one, the transformation rules for tangent vectors and for normal vectors are shown. In section two, several examples of deformation functions are listed. A method is shown in section three to convert arbitrary local representations of deformations to global representations, for space curves and surfaces. Finally, in section four, applications of the methods to the rendering process are described, opening future research directions in ray-tracing algorithms. Appendix A contains a derivation of the normal vector transformation rule.

Deformations allow the user to treat a solid as if it were constructed from a special type of topological putty or clay, which may be bent, twisted, tapered, compressed, expanded, and otherwise transformed repeatedly into a final shape. They are highly intuitive and easily visualized operations which simulate some important manufacturing processes for fabricating objects, such as the bending of bar stock and sheet metal. Deformations can be incorporated into traditional CAD/CAM solid modeling and surface patch methods, reducing the data storage requirements for simulating flexible geometric objects, such as objects made of metal, fabric or rubber.

† Previous address, Raster Technologies Inc., N. Billerica, Mass.

Although it is possible to use these techniques to accurately model the physical properties of different elastic materials with the partial differential equations of elasticity and plasticity theory, simpler mathematical deformation methods exist. These simpler methods have reduced computational needs, are widely applicable in modeling, and are described in the examples section. It is beyond the scope of this paper to formulate the mathematical details of exact mechanical descriptions of physical deformation properties of materials.

1.0 Background and Derivations.

A **globally specified deformation** of a three dimensional solid is a mathematical function $\underline{F}$ which explicitly modifies the global coordinates of points in space. Points in the undeformed solid are called (small) $\underline{x}$, while points in the deformed solid are called (capital) $\underline{X}$. Mathematically, this is represented by the equation

$$\underline{X} = \underline{F}(\underline{x}). \qquad [Equation\ 1.1a]$$

The x, y, and z components of the three dimensional vector $\underline{x}$ are designated x_1, x_2, and x_3. (For notational convenience, x_1, x_2, and x_3 and x, y, and z are used interchangably. A similar convention holds for the upper case forms.)

A **locally specified deformation** modifies the tangent space of the solid. Differential vectors in the substance of the solid are rotated and/or skewed; these vectors are integrated to obtain the global position. The differential vectors can be thought of as separate chain-links which can rotate and stretch; the local specification of the deformation is the rotation and skewing matrix function. The position of the end-link in the chain is the vector sum of the previous links, as shown in section three.

Tangent vectors and normal vectors are the two most important vectors used in modeling — the former for delineating and constructing the local geometry, and the latter for obtaining surface orientation and lighting information. Tangent and normal vectors on the undeformed surface may be transformed into the tangent and normal vectors on the deformed surface; the algebraic manipulations for the transformation rules involve a single multiplication by the Jacobian matrix $\underline{\underline{J}}$ of the transformation function $\underline{F}$. In this paper, the term "tangent transformation" substitutes for "contravariant transformation" and is the transformation rule for the tangent vectors. The term "normal transformation" substitutes for "covariant transformation" and is the transformation rule for the normal vectors.

The Jacobian matrix $\underline{\underline{J}}$ for the transformation function $\underline{X} = \underline{F}(\underline{x})$ is a function of $\underline{x}$, and is calculated by taking partial derivatives of $\underline{F}$ with respect to the coordinates x_1, x_2, and x_3:

$$\underline{J}_i(\underline{x}) = \frac{\partial \underline{F}(\underline{x})}{\partial x_i} \qquad [Equation\ 1.1b]$$

In other words, the i^{th} column of $\underline{\underline{J}}$ is obtained by the partial derivative of $\underline{F}(\underline{x})$ with respect to x_i.

When the surface of an object is given by a parametric function of two variables u and v,

$$\underline{x} = \underline{x}(u, v), \qquad [Equation\ 1.1c]$$

any tangent vector to the surface may be obtained from linear combinations of partial derivatives of $\underline{x}$ with respect to u and v. The normal vector direction may be obtained from the cross product of two linearly independent surface tangent vectors.

The **tangent vector transformation rule** is a restatement of the chain rule in multidimensional calculus. The new vector derivative is equal to the Jacobian matrix times the old derivative.

In matrix form, this is expressed as:

$$\frac{\partial \underline{X}}{\partial u} = \underline{\underline{J}} \frac{\partial \underline{x}}{\partial u} \qquad [Equation\ 1.2a]$$

This is equivalent in component form to:

$$X_{i,u} = \sum_{j=1}^{3} J_{ij} x_{j,u} \qquad [Equations\ 1.2b]$$

In other words, the new tangent vector $\partial \underline{X}/\partial u$ is equal to the Jacobian matrix $\underline{\underline{J}}$ times the old tangent vector $\partial \underline{x}/\partial u$

The **normal vector transformation rule** involves the inverse transpose of the Jacobian matrix. A derivation of this result is found in Appendix A.

[Equation 1.3]

$$\underline{n}^{(X)} = \det J \underline{\underline{J}}^{-1T} \underline{n}^{(x)}$$

Of course, since only the direction of the normal vector is important, it is not necessary to compute the value of the determinant in practice, although it sometimes is implicitly calculated as shown in Appendix A. As is well known from calculus, the determinant of the Jacobian is the local volume ratio at each point in the transformation, between the deformed region and the undeformed region.

2.0 Examples of Deformations.

Example 2.1: Scaling. One of the simplest deformations is a change in the length of the three global components parallel to the coordinate axes. This produces an orthogonal scaling operation :

$$X = a_1 x \quad Y = a_2 y \quad Z = a_3 z \qquad [\textit{Equation } 2.1a]$$

The components of the Jacobian matrix are given by

$$J_{ij} = \frac{\partial X_i}{\partial x_j},$$

so

$$\underline{\underline{J}} = \begin{pmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & a_3 \end{pmatrix} \qquad [\textit{Equation } 2.1b]$$

The volume change of a region scaled by this transformation is obtained from the Jacobian determinant, which is $a_1 a_2 a_3$. The normal transformation matrix is the inverse transpose of the Jacobian matrix (optionally times the determinant of the Jacobian matrix), and is given by:

$$\det J \;\; \underline{\underline{J}}^{-1T} = \begin{pmatrix} a_2 a_3 & 0 & 0 \\ 0 & a_1 a_3 & 0 \\ 0 & 0 & a_1 a_2 \end{pmatrix}$$

Without the factor of the determinant, the normal transformation matrix is:

$$\underline{\underline{J}}^{-1T} = \begin{pmatrix} 1/a_1 & 0 & 0 \\ 0 & 1/a_2 & 0 \\ 0 & 0 & 1/a_3 \end{pmatrix}$$

To obtain the new normal vector at any point on the surface of an object subjected to this deformation, we multiply the original normal vector by either of the above normal transformation matrices. The new **unit** normal vector is easily obtained by dividing the output components by the magnitude of the vector.

For instance, consider converting a point $[x_1, x_2, x_3]^T$ lying on a roughly spherical surface centered at the origin, with normal vector $[n_1, n_2, n_3]^T$. The transformed surface point on the resulting ellipsoidal shape is $[a_1 x_1, a_2 x_2, a_3 x_3]^T$ and the transformed normal vector is parallel to $[n_1/a_1, n_2/a_2, n_3/a_3]^T$. The volume ratio between the shapes is $a_1 a_2 a_3$.

The scaling transformation is a special case of general affine transformations, in which the Jacobian matrix is a constant matrix. Affine transformations include skewing, rotation, and scaling transformations. When the transformation consists of pure rotation, it is interesting to note that the inverse of the matrix is equal to its transpose. For pure rotation, this means that the tangent vector and the normal vector are transformed by a single matrix. For more general affine transformations, pairs of constant matrices are required.

Example 2.2: Global Tapering along the Z Axis. Tapering is similar to scaling, by differentially changing the length of two global components without changing the length of the third. In figure 2.2, the function $f(z)$ is a piecewise linear function which decreases as z increases (from page bottom to the top). The magnitude of the tapering rate progressively increases from figure 2.2 a through figure 2.2 d. When the tapering function $f(z) = 1$, the portion of the deformed object is unchanged; the object increases in size as a function of z when $f'(z) > 0$, and decreases in size when $f'(z) < 0$. The object passes through a singularity at $f(z) = 0$ and becomes everted when $f(z) < 0$.

$$r = f(z), \quad X = rx, \quad Y = ry, \quad Z = z \qquad [\textit{Equation } 2.2a]$$

The tangent transformation matrix is given by:

$$\underline{\underline{J}} = \begin{pmatrix} r & 0 & f'(z)x \\ 0 & r & f'(z)y \\ 0 & 0 & 1 \end{pmatrix} \qquad [\textit{Equation } 2.2b]$$

The local volumetric rate of expansion, from the determinant, is r^2.

The normal transformation matrix is given by:

$$r^2 \underline{\underline{J}}^{-1T} = \begin{pmatrix} r & 0 & 0 \\ 0 & r & 0 \\ -rf'(z)x & -rf'(z)y & r^2 \end{pmatrix}$$

The inverse transformation is given by:

$$r(Z) = f(Z), \quad x = X/r, \quad y = Y/r, \quad z = Z \qquad [\textit{Equation } 2.2c]$$

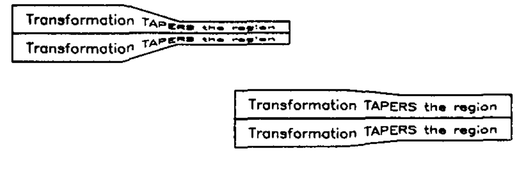

Figure 2.2 Progressive Tapering of a Ribbon

Example 2.3: Global Axial Twists. For some applications, it is useful to simulate global twisting of an object. A twist can be approximated as differential rotation, just as tapering is a differential scaling of the global basis vectors. We rotate one pair of global basis vectors as a function of height, without altering the third global basis vector. The deformation can be demonstrated by twisting a deck of cards, in which each card is rotated somewhat more than the card beneath it.

The global twist around the z axis is produced by the following equations:

$$\theta = f(z)$$
$$C_\theta = cos(\theta)$$
$$S_\theta = sin(\theta)$$

$$\begin{aligned} X &= xC_\theta - yS_\theta, \\ Y &= xS_\theta + yC_\theta, \\ Z &= z. \end{aligned} \qquad [Equation\quad 2.3a]$$

The twist proceeds along the z axis at a rate of $f'(z)$ radians per unit length in the z direction.

The tangent transformation matrix is given by

$$\underline{\underline{J}} = \begin{pmatrix} C_\theta & -S_\theta & -xS_\theta f'(z) - yC_\theta f'(z) \\ S_\theta & C_\theta & xC_\theta f'(z) - yS_\theta f'(z) \\ 0 & 0 & 1 \end{pmatrix}$$

Note that the determinant of the Jacobian matrix is unity, so that the twisting transformation preserves the volume of the original solid. This is consistent with our "card-deck" model of twisting, since each individual card retains its original volume.

The normal transformation matrix is given by:

$$\underline{\underline{J}}^{-1T} = \begin{pmatrix} C_\theta & -S_\theta & 0 \\ S_\theta & C_\theta & 0 \\ yf'(z) & -xf'(z) & 1 \end{pmatrix}$$

Our original deck of cards is a rectangular solid, with orthogonal normal vectors. We can see from the above transformation matrix that the normal vectors to the twisted deck will generally tilt out of the x-y plane.

Figures 2.3.1 a–d show the effect of a progressively increasing twist. In these line drawings of solids, vectors are hidden by the normal vector criterion—if the normal vector (as calculated by the above transformation matrix) faces the viewer, the line is drawn, otherwise, the line segment is not drawn. Figure 2.3.3 shows an object which has been twisted and tapered, while figures 2.3.4 and 2.3.2 show the results from twisting an object around an axis not within the object itself.

The inverse transformation is given by:

$$\begin{aligned} \theta &= f(Z), \\ x &= XC_\theta + YS_\theta, \\ y &= -XS_\theta + YC_\theta, \\ z &= Z \end{aligned} \qquad [Equation\quad 2.3b]$$

which is basically a twist in the opposite direction.

Figure 2.3.1 Progressive Twisting of a Ribbon

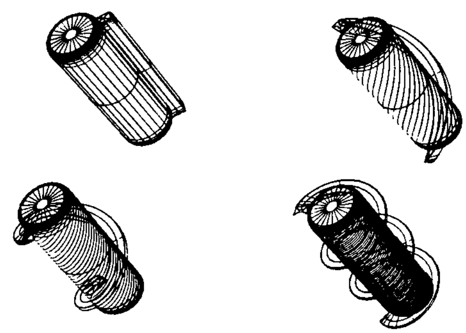

Figure 2.3.2 Progressive Twisting of Two Primitives

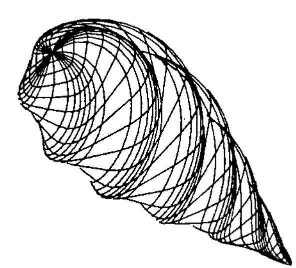

Figure 2.3.3 Twisting of a Tapered Primitive

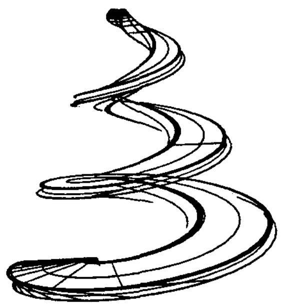

Figure 2.3.4 Tapering of a Twisted offset Primitive

Example 2.4: Global Linear Bends along the Y-Axis. For other applications, it is useful to have a simple simulation of bending.

The following equations represent an isotropic bend along a centerline parallel to the y-axis: the length of the centerline does not change during the bending process. The bending angle θ, is constant at the extremities, but changes linearly in the central region. In the bent region, the bending rate k, measured in radians per unit length, is constant, and the differential basis vectors are simultaneously rotated and translated around the third local basis vector. Outside the bent region, the deformation consists of a rigid body rotation and translation. The range of the bending deformation is controlled by y_{min}, and y_{max}, with the bent region corresponding to values of y such that $y_{min} \le y \le y_{max}$. The axis of the bend is located along $[s, y_0, 1/k]^T$, where s is the parameter of the line. The center of the bend occurs at $y = y_0$—i.e., where one would "put one's thumbs" to create the bend. The radius of curvature of the bend is $1/k$.

The bending angle θ is given by:

$$\theta = k(\hat{y} - y_0),$$
$$C_\theta = cos(\theta),$$
$$S_\theta = sin(\theta),$$

where

$$\hat{y} = \begin{cases} y_{min}, & \text{if } y \le y_{min} \\ y, & \text{if } y_{min} < y < y_{max} \\ y_{max}, & \text{if } y \ge y_{max} \end{cases}$$

The formula for this type of bending along the y axis centerline is given by the following relations:

[*Equation* 2.4a]

$$X = x$$

$$Y = \begin{cases} -S_\theta(z - \frac{1}{k}) + y_0, & y_{min} \le y \le y_{max}, \\ -S_\theta(z - \frac{1}{k}) + y_0 + C_\theta(y - y_{min}), & y < y_{min} \\ -S_\theta(z - \frac{1}{k}) + y_0 + C_\theta(y - y_{max}), & y > y_{max} \end{cases}$$

$$Z = \begin{cases} C_\theta(z - \frac{1}{k}) + \frac{1}{k}, & y_{min} \le y \le y_{max}, \\ C_\theta(z - \frac{1}{k}) + \frac{1}{k} + S_\theta(y - y_{min}), & y < y_{min} \\ C_\theta(z - \frac{1}{k}) + \frac{1}{k} + S_\theta(y - y_{max}), & y > y_{max} \end{cases}$$

These functions have continuous values at the boundaries of each of the three regions for y, and in the limit, for $k = 0$. However, there is a jump in the derivative of the bending angle θ at the $y = y_{min}$ and $y = y_{max}$ boundaries. The discontinuities may be eliminated by using a smooth function for θ as a function of y, but the transformation matrices would need to be re-derived.

The tangent transformation matrix is given by:

$$\underline{\underline{J}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & C_\theta(1 - \hat{k}z) & -S_\theta \\ 0 & S_\theta(1 - \hat{k}z) & C_\theta \end{pmatrix}$$

where

$$\hat{k} = \begin{cases} k, & \text{if } \hat{y} = y \\ 0, & \text{if } \hat{y} \neq y. \end{cases}$$

The local rate of expansion, as obtained from the determinant, is $1 - \hat{k}z$.

The normal transformation matrix is given by:

$$(1 - \hat{k}z)\underline{\underline{J}}^{-1T} = \begin{pmatrix} 1 - \hat{k}z & 0 & 0 \\ 0 & C_\theta & -S_\theta(1 - \hat{k}z) \\ 0 & S_\theta & C_\theta(1 - \hat{k}z) \end{pmatrix}$$

The inverse transformation is given by:

[*Equation* 2.4b]

$$\theta_{min} = k(y_{min} - y_0)$$

$$\theta_{max} = k(y_{max} - y_0)$$

$$\hat{\theta} = -tan^{-1}\left(\frac{Y - y_0}{Z - \frac{1}{k}}\right)$$

$$\theta = \begin{cases} \theta_{min}, & \text{if } \theta < \hat{\theta}_{min} \\ \hat{\theta}, & \text{if } \theta_{min} \le \hat{\theta} \le \theta_{max} \\ \theta_{max}, & \text{if } \hat{\theta} > \theta_{max} \end{cases}$$

$$x = X$$

$$\hat{y} = \frac{\theta}{k} + y_0$$

$$y = \begin{cases} \hat{y}, & y_{min} < \hat{y} < y_{max} \\ (Y - y_0)C_\theta + (z - \frac{1}{k})S_\theta + \hat{y}, & \hat{y} = y_{min} \text{ or } y_{max} \end{cases}$$

$$z = \begin{cases} \frac{1}{k} + ((Y - y_0)^2 + (Z - \frac{1}{k})^2)^{1/2}, & y_{min} < \hat{y} < y_{max} \\ -(Y - y_0)S_\theta + (z - \frac{1}{k})C_\theta + \hat{y}, & \hat{y} = y_{min} \text{ or } y_{max} \end{cases}$$

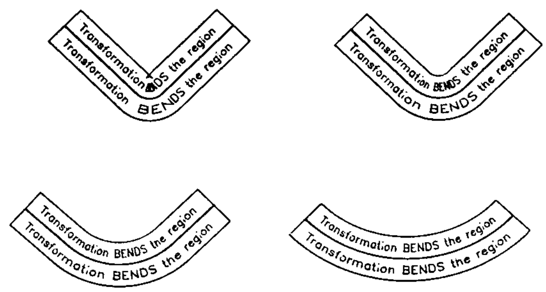

Figure 2.4.2 Progressive Change in Bending Range of a Region

In figure 2.4.2, a constant 90° bend is produced by varying the range and the bend rate. In other words, $k(y_{max} - y_{min}) = \pi/2$ in each of the examples. In figure 2.4.3, a twisted object is subjected to a progressive bend to produce a Moebius band. Figures 2.4.4 a and b show a hierarchy of tapering, twisting, and bending, by superimposing a bend on the objects in figures 2.3.2 and 2.3.3. In figure 2.4.5, a chair is made from six primitives using seven bends. The details of the crimp in the coordinate systems is shown in figures 2.4.6 a - b.

However, the type of bending shown in the figures does not retain all of the generality that true bending requires. Some materials are anisotropic and have an intrinsic "grain" or directionality in them. Although this is beyond the scope of this paper, it is interesting to note that the tangent and normal transformation rules may still be utilized.

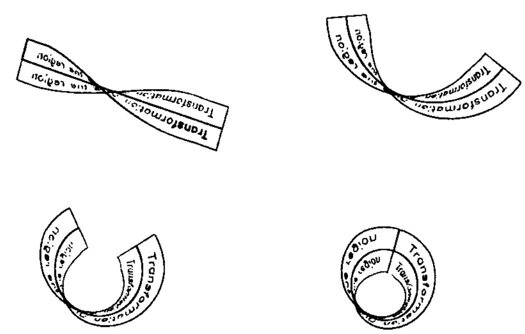

Figure 2.4.3 Moebius band is produced with a twist and a bend

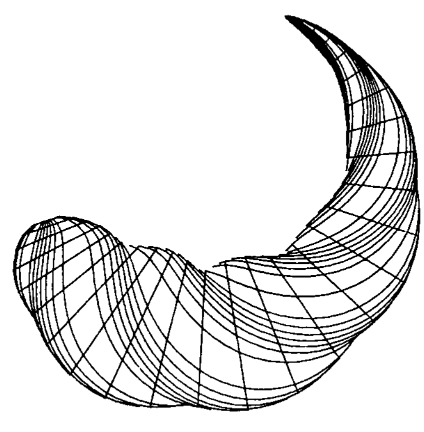

Figure 2.4.4 a Bent, Twisted, Tapered Primitive

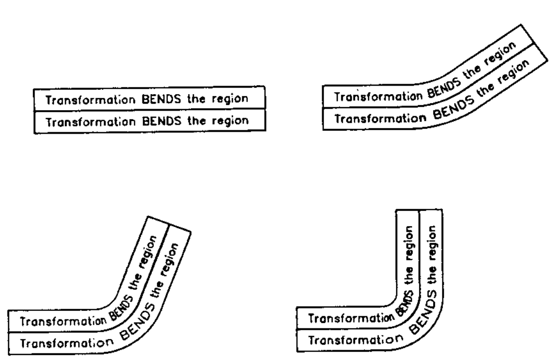

Figure 2.4.1 Progressive Bending of a Region

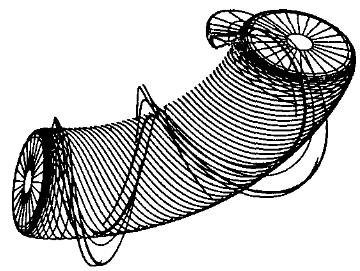

Figure 2.4.4 b Bent, Twisted Primitive

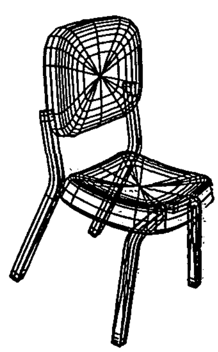

Figure 2.4.5 Chair Model, with six primitives and seven bends.

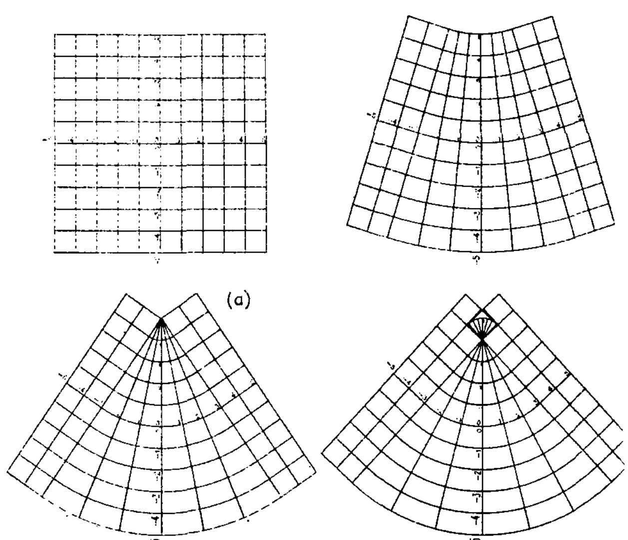

Figure 2.4.6 Details of the Bend near the Crimp

3.0 Converting Local Representations to Global Representations.

In this section, a method for generating more general shapes is addressed. The Jacobian matrix $\underline{\underline{J}}(\underline{x})$ is assumed to be known as a function of x_1, x_2, and x_3, but a closed form expression for the corresponding coordinate deformation function $\underline{X} = \underline{F}(\underline{x})$ is not known (i.e., in terms of standard mathematical functions). The basic method involves

(1) the conversion of the undeformed input shape into its tangent vectors by differentiation,

(2) transforming the tangent vectors via the tangent transformation rule into the tangent vectors of the deformed object, and then

(3) integrating the new tangent vectors to obtain the new position vectors of the deformed space curve, surface, or solid.

This "local-to-global" operation converts the local tangent vectors and Jacobian matrix into the global position vectors. The absolute position in space of the deformed object is defined within an arbitrary integration constant vector .

The above method provides a completely general description of deformation, and may be directly coupled to the output from the elasticity equations, finite element analysis, or other advanced mathematical models of deformable entities describing a profoundly general collection of shapes. The integrations outlined above need not be calculated explicitly in a ray-tracing environment: a multidimensional Newton's method can use the Jacobian matrix directly.

3.1 Transformations of Space Curves. Given a space curve, parameterized by a single variable s,

$$\underline{x} = \underline{x}(s), \quad s_0 \leq s \leq s_1$$

a new curve $\underline{X}(s)$ is desired which is the deformed version of $\underline{x}(s)$. The Jacobian matrix $\underline{\underline{J}}(s)$ or $\underline{\underline{J}}(\underline{x}(s))$ is assumed to be known, but the coordinate transformation function $\underline{X} = \underline{F}(\underline{x})$ is assumed to be unavailable. As stated above, the equation for $\underline{X}(s)$ may bé derived from the fact that,

(1) by definition, the position $\underline{X}(s)$ is a constant vector plus the integral of the derivative of the position, i.e.,

[*Equation* 3.1*a*]

$$\underline{X}(s) = \int_0^s \underline{X}'(\tilde{s})d\tilde{s} + \underline{x}_0,$$

(2) the derivative of the position is obtained via the tangent transformation rule, Equation 1.2 a, so

[*Equation* 3.1*b*]

$$\underline{X}(s) = \int_0^s \underline{\underline{J}}(\underline{x}(\tilde{s}))\underline{x}'(\tilde{s})d\tilde{s} + \underline{x}_0$$

where $\underline{\underline{J}}(\underline{x}(s))$ is the Jacobian matrix which depends upon the value of s, and $\underline{x}'(s)$ is the arclength derivative (a tangent vector) of the input curve $\underline{x}(s)$. At each point in the untransformed curve, $\underline{x}(s)$, the tangent vectors $\underline{x}'(s)$ are rotated and skewed to a new orientation in the transformed curve: the curve can be bent and twisted with or without being being stretched. For this case, any matrix function which allows the integral to be evaluated may serve as a Jacobian, since there is only one path along which to integrate.

For inextensible bending and twisting transformations of the space curve, with no stretching at any point of the curve, the Jacobian matrix $\underline{\underline{J}}(s)$ must be a varying rotation matrix function. (Even though this is not a constant affine rotation, the matrix function for the tangent vector transformation rule is identical to that used for the normal vector transformation rule.)

3.2 Transformations of 3-D surfaces and solids. The representation of a transformed surface or solid can be obtained much in the same manner as a space curve. First, an origin $\underline{O}$ is chosen in the object to be deformed. For each point $\underline{x}$ in the surface of the object, a piecewise smooth space curve is chosen, which connects the origin $\underline{O}$ to the input point $\underline{x}$. The space curve is then subjected to the deformation as in section 3.1. If $\underline{\underline{J}}(\underline{x})$ is in fact the Jacobian of some (unspecified) deformation function $\underline{X} = \underline{F}(\underline{x})$, the transformation from $\underline{x}$ to $\underline{X}$ is unique: all smooth paths connecting $\underline{O}$ and $\underline{x}$ will be equivalent. Since the equation of the surface is given by $\underline{x} = \underline{x}(u, v)$, the space curve in the surface may be obtained by selecting two functions of a single variable, say s , for u and for v. i.e.,

$$u = u(s)$$

$$v = v(s)$$

so that the space curve in the surface $\hat{\underline{x}}(s)$ is obtained by substituting the values of u and v into the equation for $\underline{x}$.

$$\hat{\underline{x}}(s) = \underline{x}(u(s), v(s))$$

This space curve is then transformed as shown above, in Equation 3.1 b. The space curve should be piecewise differentiable, so that the derivatives can be evaluated and integrated. The equation for the deformed curve is

[*Equation* 3.2.1]

$$\underline{X}(u(s), v(s)) =$$
$$\int_0^s \underline{\underline{J}}(\underline{x}(u(\hat{s}), v(\hat{s})))\underline{x}'(u(\hat{s}), v(\hat{s}))d\hat{s} + x_0$$

Expanding the above equation, using the fact that the symbol $'$ means d/ds, and using the multidimensional chain rule, we obtain

$$\underline{X}(u(s), v(s)) =$$
$$\int_0^s \underline{\underline{J}}(\underline{x}(u(\hat{s}), v(\hat{s})))(\frac{\partial \underline{x}}{\partial u}u'(\hat{s}) + \frac{\partial \underline{x}}{\partial v}v'(\hat{s}))d\hat{s} + x_0$$

As stated before, for consistency, $\underline{\underline{J}}$ must be the Jacobian matrix of some global function $\underline{F}(\underline{x})$, so that the results are independent of the path connecting $\underline{O}$ and $\underline{x}$, and so that the tangent and normal vector transformation rules apply. The test for the "Jacobian-ness" of the matrix, (in the absence of a prespecified deformation function $\underline{F}(\underline{x})$) depends on the partial derivatives of the columns of $\underline{\underline{J}}(\underline{x})$

The columns must satisfy

$$\underline{J}_{i,j} = \underline{J}_{j,i} \qquad [\textit{Equation } 3.2.2]$$

In other words, the partial derivative of the i^{th} column of $\underline{\underline{J}}$ with respect to x_j must be equal to the partial derivative of the j^{th} column of $\underline{\underline{J}}$ with respect to x_i. (The underlying principle to prove this result is a multiple-integration path consistency requirement. The integrand must be an exact differential.) The values of the Jacobian may be directly related to the material properties of the substance to be modeled, and may utilize the plasticity and elasticity equations.

4.0 Applications to Rendering

To obtain a set of control points and normal vectors with which to create surface patches like polygons or spline patches, we sample the deformed surface parametrically, With the appropriate sampling, the patches can faithfully tesselate the desired object, with more detail where the surface is highly curved, and less detail where the surface is flat.

First, the object is sampled with a raw grid of parametric u-v values. This raw parametric sampling of the surface is then refined using normal vector criteria, as calculated by the transformation rule: the surface is recursively subdivided when the adjacent normal vectors diverge too greatly. Dot products which are far enough from unity indicate that more recursive detail is necessary in that region.

In this way, patch-oriented methods like depth-buffer and scan-line encoding schemes are effective. These algorithms are linear in terms of the total surface area and total number of patches. The direct subdivison approach is not as well-suited to ray tracing, since the total number of operations is quadratic in the number of ray comparisons and objects.

The incident ray can be intersected with the deformed primitive analytically, to reduce the number of objects. In addition, it is possible to use the inverse deformation to undeform the primitives and trace along the deformed rays. (See figures 4.1 and 4.2). This reduces the dimensionality of the parameter search from three to one, indicating a tremendous saving in numerical complexity.

The Jacobian techniques in this paper aid the traditional solution methods to find roots of non-linear ray equations (in the context of ray-tracing deformed objects), including the multidimensional Newton-Raphson method, the method of regula falsi, and the one-dimensional Newton's methods in N-space. (See [ACTON].) The analysis of rendering deformed primitives using these techniques is left to a future study.

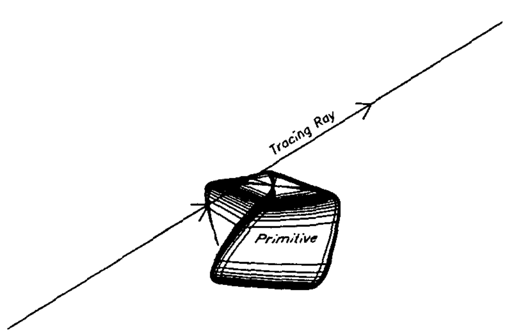

Figure 4.1 Deformed primitive, in undeformed space.

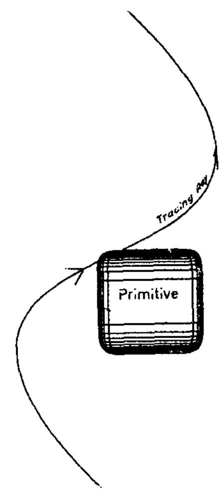

Figure 4.2 Undeformed primitive, in its undeformed coordinate system, showing path of ray

Appendix A: Proof of the normal vector transformation rule.

A short derivation in cross product and dot product style demonstrates the normal vector transformation rule.

The surface of an undeformed object is given by a parametric function of two variables u and $v, \underline{x} = \underline{x}(u,v)$. The goal is to discover an expression for the normal vector to the surface after it has been subjected to the deformation $\underline{X} = \underline{F}(\underline{x})$.

We note that the inverse of an arbitrary three by three matrix $\underline{\underline{M}}$ may be obtained from the cross-products of pairs of its columns via:

$$[\underline{M}_1, \underline{M}_2, \underline{M}_3]^{-1} = \frac{[\underline{M}_2 \wedge \underline{M}_3, \underline{M}_3 \wedge \underline{M}_1, \underline{M}_1 \wedge \underline{M}_2]^T}{\underline{M}_1 \cdot (\underline{M}_2 \wedge \underline{M}_3)}.$$

We start the derivation using the fact that the normal vector is the cross product of independent surface tangent vectors:

$$\underline{n}^{(X)} = \frac{\partial \underline{X}}{\partial u} \wedge \frac{\partial \underline{X}}{\partial v} \qquad [Equation\ \ B.1d]$$

The tangent vectors for $\underline{X}(u,v)$ are expanded in terms of $\underline{x}(s,t)$.

$$\underline{n}^{(X)} = \left(\underline{\underline{J}} \frac{\partial \underline{x}}{\partial u}\right) \wedge \left(\underline{\underline{J}} \frac{\partial \underline{x}}{\partial v}\right)$$

Matrix multiplication is expanded, yielding

$$\underline{n}^{(X)} = \left(\sum_{i=1}^{3} \underline{J}_i x_{i,u}\right) \wedge \left(\sum_{j=1}^{3} \underline{J}_j x_{j,v}\right)$$

The summations are combined together:

$$= \sum_{i=1}^{3}\sum_{j=1}^{3}\left(\underline{J}_i \wedge \underline{J}_j\right) x_{i,s} x_{j,t}$$

Since the cross product of a vector with itself is the zero vector, and since for any vectors $\underline{b}$ and $\underline{c}$, $\underline{b}\wedge\underline{c} = -\underline{c}\wedge\underline{b}$, this expands to:

$$\underline{n}^{(X)} = \left(\underline{J}_2\wedge\underline{J}_3, \underline{J}_3\wedge\underline{J}_1, \underline{J}_1\wedge\underline{J}_2\right)\begin{pmatrix} x_{2,u}x_{3,v} - x_{3,u}x_{2,v} \\ x_{3,u}x_{1,v} - x_{1,u}x_{3,v} \\ x_{1,s}x_{2,v} - x_{2,u}x_{1,v} \end{pmatrix}$$

Thus,

$$\underline{n}^{(X)} = \left[\underline{J}_2\wedge\underline{J}_3, \underline{J}_3\wedge\underline{J}_1, \underline{J}_1\wedge\underline{J}_2\right]\underline{n}^{(x)}$$

Since $\det M = \underline{M}_1 \cdot (\underline{M}_2 \wedge \underline{M}_3)$ for an arbitrary matrix $\underline{\underline{M}}$,

$$\underline{n}^{(X)} = \det J \underline{\underline{J}}^{-1T} \underline{n}^{(x)}$$

In other words, the new normal vector $\underline{n}^{(X)}$ is expressed as a multiplication of matrix $\underline{\underline{J}}^{-1T}$ and the old normal vector $\underline{n}^{(x)}$.

Since only the direction of the normal vector is important, it is not necessary to compute the value of the determinant in practice, unless one needs the local volume ratio between corresponding points in the deformed and undeformed objects.

The fact that the normal vector follows this type of transformation rule makes it less expensive to calculate, increasing its applicability in a variety of modeling circumstances.

Acknowledgements

I would like to thank Dan Whelan, of the California Institute of Technology, and Olin Lathrop, of Raster Technologies Inc., for technical help with the typography and the illustrations.

Bibliography

1. Acton, F.S., Numerical Methods that Work, Harper and Row, 1970.
2. Barr, A.H., "Superquadrics and Angle-Preserving Transformations," IEEE Computer Graphics and Applications, Volume 1 number 1 1981.
3. Buck, R. C., Advanced Calculus, McGraw-Hill, 2nd edition, 1965
4. Faux, I.D., and M.J. Pratt, Computational Geometry for Design and Manufacture, Ellis Horwood Ltd., Wiley and Sons, 1979.
5. Franklin, W.R., and A.H. Barr, "Faster Calculation of Superquadric Shapes," IEEE Computer Graphics and Applications, Volume 1 number 3, 1981.
6. Kajiya, J.T., "Ray Tracing Parametric Patches," SigGraph 82 Conference Proceedings, Computer Graphics, Volume 16, Number 3, 1982.
7. Segel, L.A., Mathematics Applied to Continuum Mechanics, Macmillan Publishing Co., 1977.
8. Solkolnikoff, I.S., Mathematical Theory of Elasticity, McGraw Hill, 1956.