X.11

COMPUTING SURFACE NORMALS FOR 3D MODELS

Andrew S. Glassner
Xerox PARC
Palo Alto, California

Introduction

In this note we focus on polygonal approximations to smooth surfaces. Such surfaces may be made to appear smooth rather than faceted, using smooth-shading techniques such as those proposed by Gouraud (1971b) and Phong (1973). These techniques require a surface normal to be defined at each vertex in the model. This note discusses some methods for generating such normals.

Swept Contours

We begin with an important special class of shape: swept contours. Examples of these shapes are prisms and surfaces of revolution. Such shapes are defined by a planar curve (or contour), which is then translated along a path or rotated about an axis, as shown in Fig. 1. If our input consists only of the contour, how might we find a surface normal for points on the swept-out surface?

Reducing the dimension of a problem is often a good way to simplify its analysis. An easy way to eliminate one dimension for this problem is to generate normals for just the 2D contour curve, and then transform those normals with the curve as the contour is swept (see "Properties of

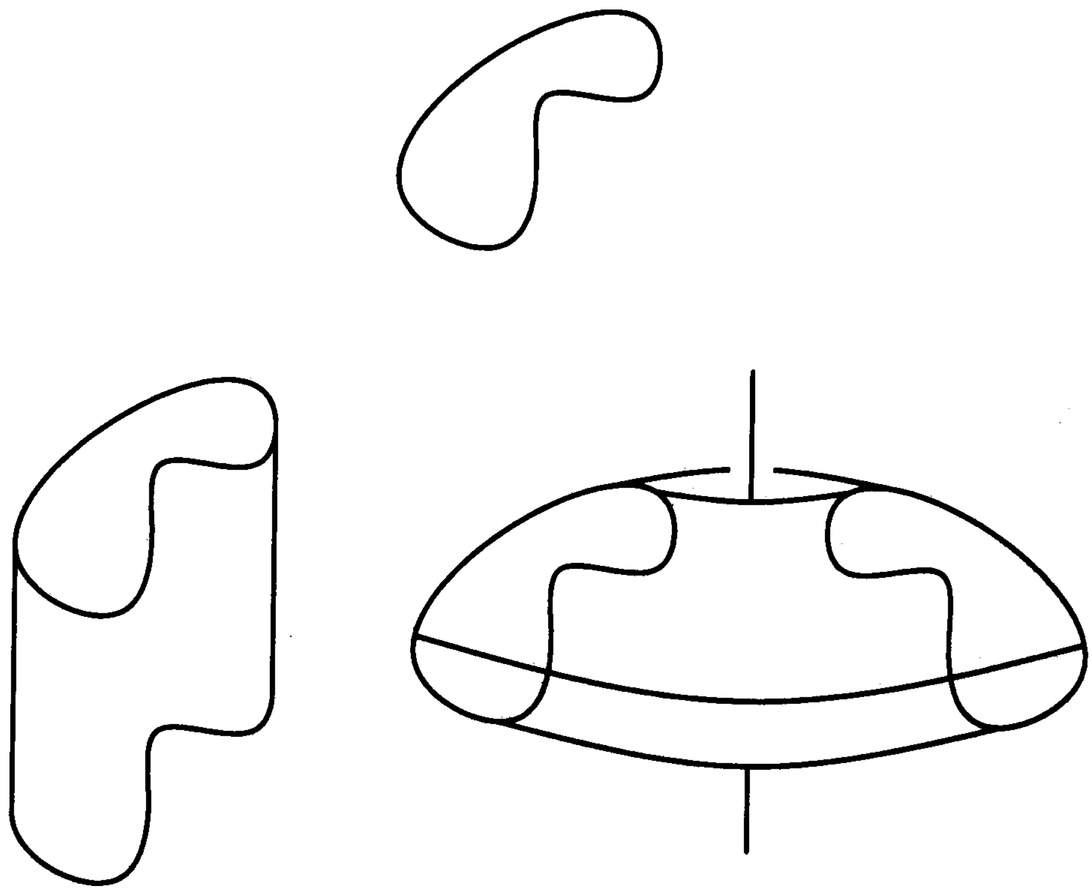

Figure 1. A contour, and the results of translation along a straight line and rotation about an axis.

Surface-Normal Transformations" in this volume). We now need only find planar normals to the planar contour.

Figure 2 shows a contour and a distinguished point *P* for which we wish to find a normal; we discuss three approaches. Technique A finds the normals of the two segments adjacent to *P*, and averages those (Fig. 2a). Technique B finds the line joining the two vertices adjacent to *P*, and uses the normal of that line as the normal at *P* (Fig. 2b). Both of these approaches are implicitly using the Mean Value Theorem, which guarantees that somewhere between two points on a continuous curve, the curve obtains a slope parallel to the line through those points. We can take a more direct, constructive approach to this result. Recall that the derivative of a curve at a point may be computed as the limit of the slope of a line that passes on either side of that point; that is,

$$\frac{\mathbf{d}f}{\mathbf{d}x}(x_0) \approx \frac{f(x_0 + \epsilon) - f(x_0 - \epsilon)}{(x_0 + \epsilon) - (x_0 - \epsilon)}.$$

Let us directly compute $x_0 + \epsilon$ and $x_0 - \epsilon$ as those points that are some small distance ϵ away from P along the lines joining P; these are on the circle of radius ϵ centered at P. Then the slope of that line may be used as the average slope at P, as shown in Fig. 2c. The choice of ϵ may be made somewhat arbitrarily; about 1/2 the distance along the shorter of the two edges has worked well for me. Note that this is a distance along the *nonzero* edges radiating from P; the next defined point after P may be P itself (this can happen when one wishes to define a double point for an interpolated curve).

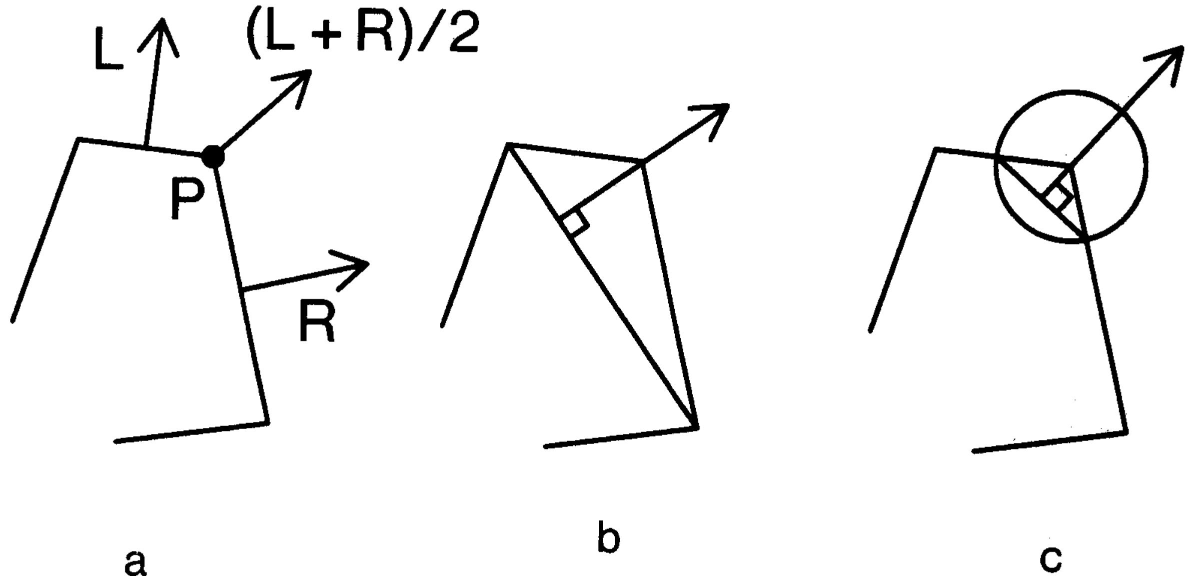

Figure 2. Different methods for computing the vertex normal on a contour.

3D Polyhedra

The more general problem of computing normals at vertices of arbitrary 3D polyhedra may also be approached in several ways. By far the most common approach is that suggested by Gouraud (1971), which simply averages the normal of each polygon that shares that vertex (see Fig. 3).

We may adapt the latter two techniques of the previous section, but there is a problem. In 2D, we found two points that together determined a unique line; this is because only two edges could leave a vertex in a profile curve. In 3D, we need exactly three points to determine a plane. If a vertex has three edges radiating from it, we may find the vertex on the far end of each edge, pass a plane through these points, and use the plane normal as the vertex normal (see Fig. 4a). The problem with this approach is that many vertices will have more or less than three edges. If a vertex has only two edges, the two neighboring vertices do not determine a unique plane; if a vertex has more than three edges, there typically will be no single plane that will pass through all the neighbor vertices. One

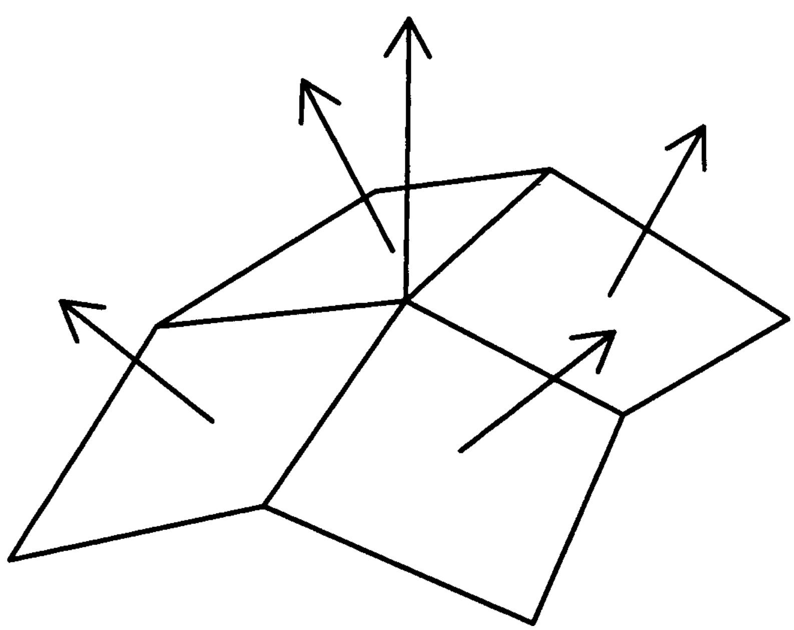

Figure 3. Computing a vertex normal by averaging neighbor polygon normals.

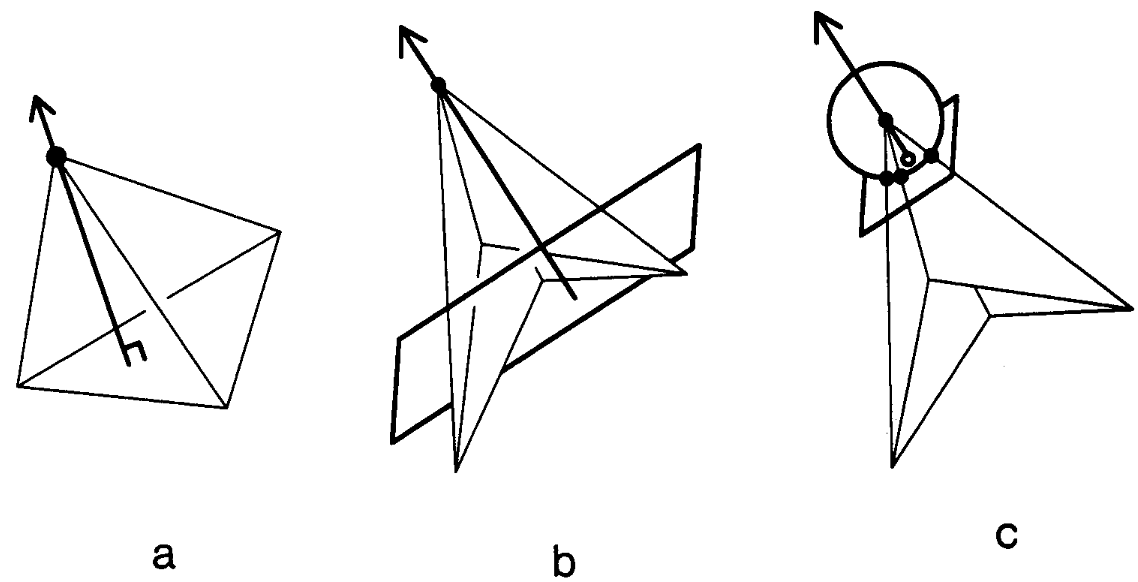

Figure 4. Different methods for computing the vertex normal on a polyhedron.

approach in this latter case is to find the least-squares solution to the points, resulting in the plane that most nearly interpolates the points (Fig. 4b). Techniques for finding this plane are well-known; see Lawson and Hanson (1974).

This approximate solution may also be applied to a more local approximation of the derivative, found by traveling only some distance along each edge sharing *P* (Fig. 4c).

In practice, the simple averaging of the normals of adjacent polygons works quite well, as long as the polygons interpolate the surface "reasonably." (It is interesting to consider just what that word means in this context.) If the polygons are a poor approximation to the surface, then I think that flattened silhouettes and badly approximated intersections with other surfaces will usually be more objectionable symptoms than poor shading. These latter effects may be alleviated with the technique of Max (1989).