Smoothlv Shaded Renderings of Polyhedral Objects on Raster Displays

by

Tom Duff
Computer Graphics Laboratory
New York Institute of Technology
Old Westbury, New York, 11568

Abstract

The appearance of raster-scan renderings of polyhedral approximations to curved surfaces can be enhanced greatly by shading them in a manner that varies smoothly across each polygon and which matches the correct shade at each vertex. Henri Gouraud and Bui Tuong-Phong have previously described two methods of computing such shading functions. While the computations in Gouraud's method are quite quick to perform, the renderings produced often exhibit pronounced Mach bands, and animated sequences tend to have annoying fluctuations in intensity. Phong's method tries to cure these problems, but is much more expensive computationally. We will show a new, faster method of computing Phong shading, and discuss the relationship of this method to Gouraud shading. We will also exhibit representatives of a class of surfaces for which Phong shading surprisingly produces worse Mach bands than Gouraud shading.

1. Introduction

Henri Gouraud has shown, in [Gouraud 71], that the appearance of raster-scan renderings of polyhedrally approximated curved surfaces can greatly be enhanced by shading each point on the surface with a colour computed by linear interpolation of "exact" shading values at the vertices of the polygon containing the point. Bui Tuong-Phong, in [Phong 73], noted several problems with Gouraud shading, and proposed a new shading method which in some measure remedied those difficulties. The nature of some of these problems, and the extent to which Phong was successful will be discussed below. As a byproduct of this investigation, we will demonstrate an improved method of calculating the Phong shading function.

2. Terminology

We will use the letters P, Q, and R to designate the normals to a surface at three given points. L will denote a vector which indicates the direction of a light source which is considered to be arbitrarily distant (or "at infinity"), so that L is independant of the point upon which the light shines. E is used for the vector in the direction of the viewer's eye, which is also at infinity. E, L, P, Q and R are always taken to be unit vectors. That is

$$E \cdot E = L \cdot L = P \cdot P = Q \cdot Q = R \cdot R = 1$$

Note that this means that, for example, $L \cdot P = \cos\theta$, where θ is the angle between L and P.

3. Illumination

Any method for rendering surfaces must somehow model the way in which the light which impinges on the surface is re-directed toward the viewer's eye. One of the simplest lighting models is Lambert's Law for diffuse reflectors, which states that the intensity perceived by an observer is independant of the observer's position, and varies directly with the cosine of the angle between the light direction and the normal to the surface at a given point. That is,

$$i = kL \cdot P, \qquad (1)$$

where i is the amount of reflected light, and k is a value which depends on the intensity of the light and the reflectance of the surface.

Most surfaces are not perfect diffusers, and exhibit a certain amount of specular reflection. The specular component is greatest when the surface reflects the light directly at the observer's eye, and drops off as the light moves away from that direction at a rate which depends on the properties of the surface being modelled. For surfaces with isotropic reflectance properties (not, for example, brushed aluminum or hair), the rate at which the specular reflection drops off may be expressed as a function of the cosine of the angle between the normal P and the direction of maximum highlight H. This cosine is just H·P. H is defined by

$$H = \frac{E+L}{|E+L|}.$$

[Phong 73] proposes the illumination function

$$i = dL \cdot P + s(H \cdot P)^b + a, \qquad (2)$$

where d and s are the maximum intensities of the diffuse and specular components, b is related to the "shininess" of the surface (large values of b give pinpoint highlights, small values give broader ones), and a is the contribution of the ambient light in the environment. This, and several other illumination models are discussed in [Blinn 77], which is a good detailed introduction to the whole subject of Illumination.

4. Gouraud and Phong Shading

In detail, Gouraud's shading method is as follows: We are given a collection of polygons with shading values associated with each vertex. The vertex shading values may of course be assigned by any means whatever, but for our purposes, let us assume that we are given the normal vector at each vertex, and can compute a shading value from it. At the points at which a given scanline intersects the edges of a polygon, we compute a shading value by linearly interpolating the shading values at the endpoints of the edges. Shading values for the interior points of a polygon are determined by linear interpolation along scanlines of the values computed where the scanlines meet the edges of the polygon.

Phong's shading procedure is similar to Gouraud's, except that instead of interpolating shading values, we interpolate the normals given at each corner, and evaluate the illumination function at each point. This is considerably more expensive, since three normal components must be computed, rather than one shading value, and the vector at each point must be normalized before evaluating the shading function.

There are several problems with these shading methods. First, if an object should have a highlight at any point but a vertex, Gouraud shading will misplace the highlight, or omit it altogether, since the shading values at interior points must interpolate those at the corners. This problem is so severe as to render Gouraud shading unusable when a non-Lambert's Law lighting model is used, unless a large number of polygons are used in the object definition. Figures 1a and b show a highly specular sphere, approximated by a small number of polygons, Gouraud and Phong shaded, respectively. Since Phong's method interpolates normals rather than shading values, the intensity values at each point need not lie between those at the corners. It should be clear that any polygon on which we expect to see a highlight (i.e. those polygons for which the normals at the corners surround the H vector), will in fact exhibit a highlight.

A problem shared by Phong and Gouraud shading is that if an object and its light source are rotated together in the image plane, the shading of the object can change, contrary to expectation. For example, if the polygon ABCD in Figure 2 is viewed from an angle such that 1' is a scanline, the shading value at point P will depend only on the values at points A, B and C. In particular, moving point D will not change the shading value computed at P. If, however, the viewer is rotated 90 degrees, so that 1' is a scanline of his view, moving D can change the shading value, since the ratio in which 1' cuts AD can change as D moves.

This problem can be ameliorated by interpolating shading values (or normals) in a rotation-independant manner. Such an interpolator can be constructed as follows: let $A_i, (1 \le i \le n)$ be the points of a polygon, and let S_i be the shading value (or normal) at A_i. Further, let T be any point inside the polygon, and let $f_i(T)$ be any function which is non-zero at A_i and zero at all other A_j. Then

$$\frac{\sum_{i=1}^{n} f_i(T) S_i}{\sum_{i=1}^{n} f_i(T)}$$

is an appropriate interpolator. One possible choice for $f_i(T)$ is

$$(1-D_i(T)) \prod_{j \ne i} D_j(T),$$

where $D_i(T)$ is the distance from A_i to T. Of course, in the Phong shading case, the vector computed by this formula must be re-normalized. For all but the simplest of applications, this function is probably too slow to use -- it is certainly out of the question for high-resolution animation applications.

A quicker scheme, which works only for quadrilaterals is as follows: Let $A_i, (1 \le i \le 4)$ be the four corners of the quadrilateral, and let S_i be the shading values (or normals) at the four corners. Compute a linear transformation of homogeneous co-ordinates which maps A_1 into (0,0), A_2 into (1,0), A_3 into (1,1) and A_4 into (0,1). Use this transformation to transform each point T of the quadrilateral into a point (u,v) $0 \le u,v \le 1$. Then, an appropriate shading value (or normal) is

$$(1-v)((1-u)S_1+uS_2) + v((1-u)S_4+uS_3)$$

This is relatively cheap, compared to the more general scheme discussed above, but is still quite expensive in the absolute.

5. Mach Bands

Another problem with Gouraud shading which Phong hoped to overcome is the Mach bands which appear where adjacent Gouraud shaded polygons meet. Mach bands are light or dark lines which the eye perceives at places where the spatial derivitive of the shading function is discontinuous or changes quickly. The eye enhances these discontinuities in order to make it easier to see the edges of objects. Mach bands are readily apparent in Figure 3, which shows a Gouraud shaded display of a sphere approximated by 72 polygons. (Note: because Mach bands are an optical illusion whose presence depends on the spatial derivative of the image intensity being large, they tend to disappear on close examination, since as your eye approaches an image, the apparent spatial frequencies in the picture decrease. Thus, paradoxically, Mach bands are more visible when pictures are viewed from a greater distance.)

Phong's thesis discusses the Mach band problem, and a casual reading might lead one to believe that he had solved it. An informal survey of the staff of the Computer Graphics Lab revealed that the majority of those aware of the problem thought he had, in fact, solved it, although a little reflection will convince the reader that it is not so. The illustrations in Phong's thesis do show that in general the Mach bands are much improved by Phong shading.

It is interesting to compute the size of the shading derivative discontinuities at the edges of a Phong shaded polyhedron. For simplicity, we will use the Lambert's law illumination model, and consider the shading of a scan line which has three uniformly spaced edges crossing it. The normals at these edges we will call P, Q and R, from left to right. Let $S_{PQ}(a)$ be the shading function between P and Q. $S_{PQ}(0)$ is the shading value at P, and $S_{PQ}(1)$ is the shading value at Q. By equation 1 above, and remembering that $P\cdot P = Q\cdot Q = 1$:

$$S_{PQ}(a) = L\cdot\frac{aQ+(1-a)P}{|aQ+(1-a)P|} \qquad (3)$$

$$= L\cdot\frac{aQ+(1-a)P}{\sqrt{(aQ+(1-a)P)\cdot(aQ+(1-a)P)}}$$

$$= L\cdot\frac{aQ+(1-a)P}{\sqrt{a^2Q\cdot Q+2a(1-a)P\cdot Q+(1-a)^2P\cdot P}}$$

$$= L\cdot\frac{aQ+(1-a)P}{\sqrt{a^2+2aP\cdot Q-2a^2P\cdot Q+1-2a+a^2}}$$

$$= L\cdot\frac{aQ+(1-a)P}{\sqrt{2(1-P\cdot Q)a^2-2(1-P\cdot Q)a+1}} \qquad (4)$$

For the sake of compactness, let $B=2(1-P\cdot Q)$. The derivative $S'_{PQ}(a)$, obtained after much calculation, is

$$S'_{PQ}(a) =$$

$$L\cdot\frac{(Q-P)\sqrt{Ba^2-Ba+1}-(P+a(Q-P))\dfrac{2Ba-B}{2\sqrt{Ba^2-Ba+1}}}{Ba^2-Ba+1}$$

The left derivative of the shading function at Q is just

$$S'_{PQ}(1) = L\cdot((P\cdot Q)Q-P)$$

The right derivative, obtained similarly, is

$$S'_{QR}(0) = L\cdot(R-(Q\cdot R)Q)$$

Clearly, in general the left and right derivatives are different. Let D_P be the size of the discontinuity in the Phong derivative at Q.

$$D_P = S'_{QR}(0)-S'_{PQ}(1)$$

$$= L\cdot(R-(Q\cdot(P+R))Q+P)$$

Now, let us consider the analogous Gouraud shading function $G_{PQ}(a)$.

$$G_{PQ}(a) = (1-a)L\cdot P+a(L\cdot Q)$$

$$= L\cdot P+aL\cdot(Q-P) \qquad (5)$$

$$G'_{PQ}(a) = L\cdot(Q-P)$$

$$D_G = G'_{QR}(0)-G'_{PQ}(0)$$

$$= L\cdot(R-Q)-L\cdot(Q-P)$$

$$= L\cdot(R-2Q+P)$$

If Phong shading is to produce uniformly less severe Mach bands than Gouraud shading, then $|D_P|$ must be smaller than $|D_G|$. Certainly, this is usually the case. However, it is possible for $|D_P|$ to be larger than $|D_G|$, indicating that Phong shading can produce worse Mach bands than Gouraud shading. Consider, for example, sets of vectors of the following form, which might easily appear when rendering spheres or cylinders:

$$L = [0\ 0\ 1]$$
$$P = [.6\cos(-\theta)\ .6\sin(-\theta)\ .8]$$
$$= [.6\cos(\theta)\ -.6\sin(\theta)\ .8]$$

$$Q = [.6\cos(0)\ .6\sin(0)\ .8] = [.6\ 0\ .8]$$

$$R = [.6\cos(\Theta)\ .6\sin(\Theta)\ .8]$$

Now:

$$D_G = [0\ 0\ 1]\cdot(R-2Q+P) = .8-1.6+.8 = 0$$

and

$$D_P = [0\ 0\ 1]\cdot(R-(Q\cdot(P+R))Q+P) = .8-.8([.6\ 0\ .8]\cdot[1.2\cos(\Theta)\ 0\ 1.6])+.8 = 1.6-.8(.72\cos(\Theta)+1.28) = .576-.576\cos(\Theta)$$

Thus, except in the trivial case where $\Theta=0$, D_P is non-zero, and a Phong shaded surface will show a Mach band where the same surface Gouraud shaded will show none.

Figures 4a and 4b, which illustrate an extreme case of this phenomenon, were generated from the following (admittedly pathological) set of vectors:

$$P = [0\ .8\ .6]$$
$$Q = [\sqrt{.91}\ 0\ .3]$$
$$R = [0\ 1\ 0]$$
$$L = [0\ 0\ 1],$$

For this set, the value of D_G is 0, and D_P is .546. Figure 4a shows a surface with these normals Phong shaded, and Figure 4b shows the same surface Gouraud shaded. Every scanline of these surfaces has normal P at the left, Q in the middle and R on the right. As expected, the Phong shaded version evinces a pronounced Mach band, while none is visible in the Gouraud shaded version. To help show what occurs in more usual circumstances, Figures 5a and b show Phong and Gouraud shaded spheres, approximated by 800 polygons (40 around the equator by 20 from pole to pole). Slight Mach bands can be discerned near the right edge of both the Phong and Gouraud shaded versions. Figures 5c and d, included for comparison, show the same surface, first shaded using the exact normals for a sphere, and secondly using the actual normals to the polygons to produce a faceted rendering.

6. Speed of Rendering

The usual method of computing the Phong shading function involves evaluating the right hand side of equation 3 at each point inside each visible polygon in the scene. This can be computed with 6 multiplications, 7 additions, a square root and a division, if finite differences are used to eliminate the multiplications which appear to be needed to evaluate $aQ+(1-a)P$. Note, however, that equation 4 can be rewritten as

$$S_{PQ}(a) = \frac{aL\cdot Q+(1-a)L\cdot P}{\sqrt{2(1-P\cdot Q)a^2-2(1-P\cdot Q)a+1}} \qquad (6)$$

Notice that the numerator of the right hand side is just the Gouraud Shading function (equation 5), and the denominator is the square root of a quadratic in a. Because P and Q are unit vectors, $1-P\cdot Q$ is constrained to be between 0 and 2. Thus, the denominator cannot range lower than 0 or higher than 1 when $0\leq a\leq 1$. One rather surprising result of this observation is that, at least when using Lambert's Law, Phong shading always produces a brighter rendering than Gouraud shading -- independant of the shape and orientation of the surface, and independant of the light direction. Occasionally it has been suggested (although not, to my knowledge, in print) that Phong shading can be sped up at the expense of image quality by omitting the step of normalizing the normal vector at each point. Since this leaves us with a slow method for doing Gouraud shading, the merit of this suggestion is somewhat dubious.

The right hand side of equation 6 is computable much more quickly than the form in which the Phong shading function is usually given. is usually given; at each interior point we need do only 3 additions, a square root and a division, if differencing between adjacent pixels is used to best advantage. This saves 6 multiplies and 4 adds over the usual method. It should also be noted that the square root in the denominator can be computed to sufficient precision for display use by linear interpolation from a small table, since the denominator polynomial is guaranteed to be between 0 and 1. Use of a 64 element table gives 12 bits of precision, which is more than adequate for modern raster displays.

It should be noted that while the above algorithm is stated only for the Lambert's Law case, it is applicable to most other illumination schemes, since almost all of them compute the dot product of the normal vector to the surface with one or two other vectors (usually L and H), and compute some function of the resulting values.

The following table shows the actual execution times, in seconds, of the program used to generate the pictures of figure 4, executing on a PDP-11/70. The first column shows times obtained when using Lambert's law shading; the second column shows times for Phong's rule (equation 2, above). The row labelled "Slow Phong" is for Phong's interpolation scheme, not using the finite difference speedup mentioned above; "Fast Phong" is for Phong"s algorithm with finite differencing, "Eqn. 6" is for the new method derived from equation 6, and "Gouraud" is for Gouraud shading.

	Lambert	Eqn. 2
Slow Phong	81.7	177.2
Fast Phong	46.7	146.8
Eqn. 6	31.0	125.8
Gouraud	10.0	10.2

7. Conclusions

We have noted several difficulties with two important and well-known smooth-shading algorithms, particularly the problems of rotation variance and Mach bands. In the course of the investigation, we have devised a new method of implementing

Phong's normal interpolation scheme which, by measurement, is, in the best case, about 35% faster than the best previously known algorithms.

Among the research problems which remain outstanding are finding an acceptably efficient rotation invariant interpolation method which works in the general case, and satisfactorily eliminating the Mach bands which perpetually turn up in smooth-shaded polyhedral images.

Acknowledgements

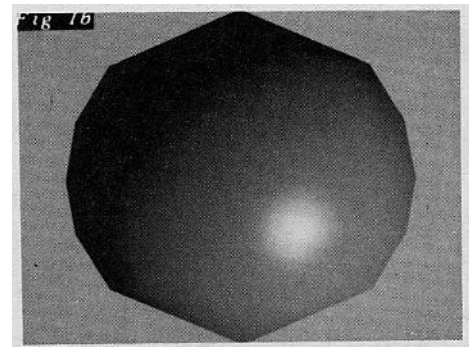
The content of this paper owes much to the stimulus of conversations with Ed Catmull, Jim Clark and Lance Williams. One of the reviewers provided a number of important suggestions and corrections which I have gratefully adopted.

Bibliography

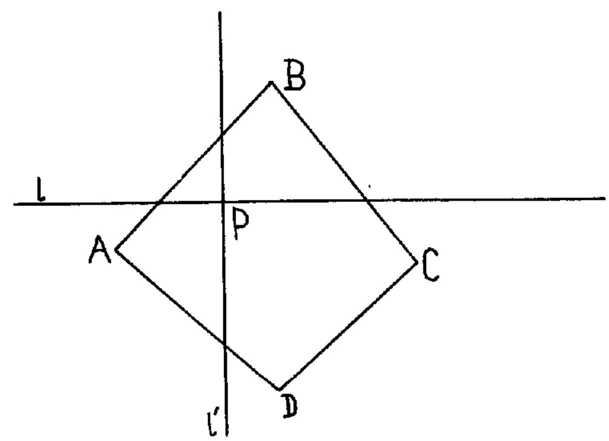
[Blinn 77] Models of Light Reflection For Computer Synthesized Pictures, by James F. Blinn, SIGGRAPH '77 Proceedings pp. 192-198, July 1977.

[Gouraud 71] Computer Display of Curved Surfaces, by Henri Gouraud, Department of Computer Science, University of Utah, UTEC-CSc-71-113, June 1971.

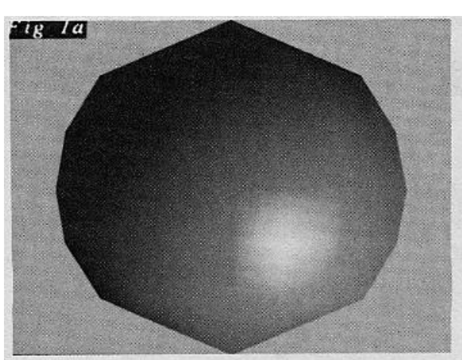
[Phong 73] Illumination for computer-generated images, by Bui Tuong-Phong, Department of Computer Science, University of Utah, UTEC-CSc-73-129, July 1973.



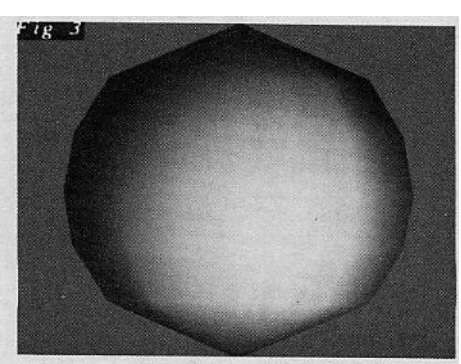
1b Specular Phong shaded sphere.



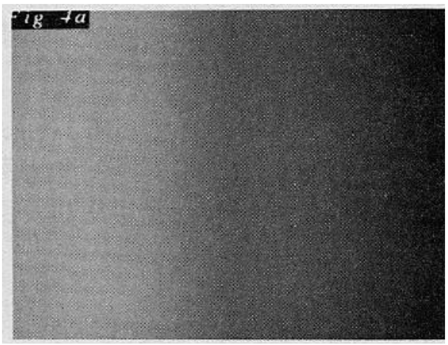
2 A Polygon with scanlines of two views from different angles.



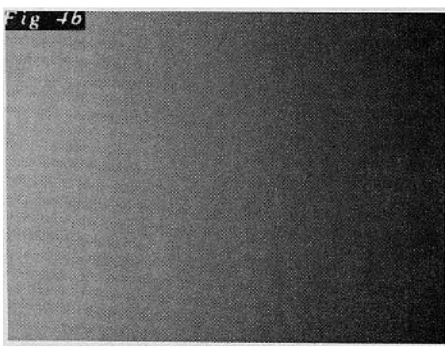
1a Specular Gouraud shaded sphere.



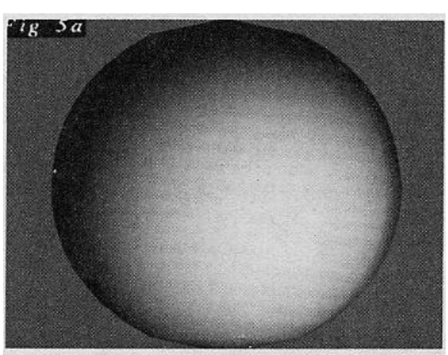
3 Gouraud shaded sphere, showing promenant Mach bands.



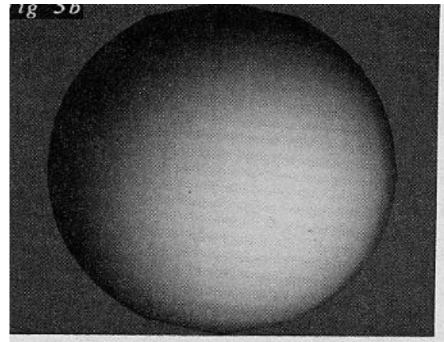
4a Phong shaded surface showing a Mach band.



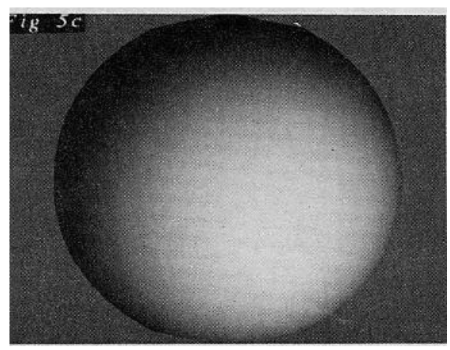
4b The same surface Gouraud shaded. Note the absence of a Mach band.



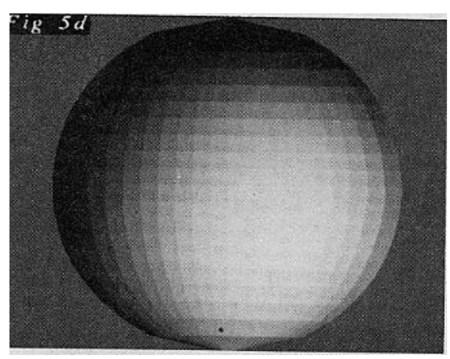
5a A Phong shaded sphere, displaying a Mach band near the right edge.



5b The same sphere, Gouraud shaded.



5c The same sphere, shaded using exact normals.



5d The same sphere, a faceted version using the uninterpolated normals of the approximating polygons.