

'An Improved Illumination Model for Shaded Display'

Colby Rush

April 25, 2014

1 Overview

This paper describes current methods used to render a two-dimensional image in a three-dimensional scene, specifically the methods used in global illumination and light intensity modelling. It then proposes a new, improved algorithm to model this information that yields a rendering time decrease of about three-fold, while still having significant room for improvement.

2 Key Points

The first key point of the paper lies in it's description of previous modelling algorithms, such as Phong's, which have their own advantages, but lack true realism. The main missing element from most realistic rendering lies in both specular reflection, universal lighting, and the simulation of shadows.

The second key point lies in the description of the proposed improved model. Provided by ray optics, it calculates a "light intensity" for each ray from the surface of the shape to the viewer, taking the intended surface material, reflection, and direction into account to come up with the intended light intensity. The main component in determining surface material, which is the main component in determining how the light interactions with the drawn object, lies the in the variance of refraction variable. In effect, adjusting the refraction can result in a totally different lighting scheme for the overall picture.

The third key point goes over the difference between modelling with a single object and multiple ones. While rendering is relatively simple with one object, when more are introduced, light will reflect off of multiple objects before reaching the view, almost exponentially increasing the amount

of calculations that must be made. This keeps going through the "tree" of calculations that must be made until either completed or truncated due to lack of space, calculating the light intensity at each node.

The fourth key point involves the determining of visible surfaces, by which the method does by tracing from the viewer to the object. It then calculates the light intensity, and keeps going, repeating until no other object is touched, using a sphere as the bound for ray duration.

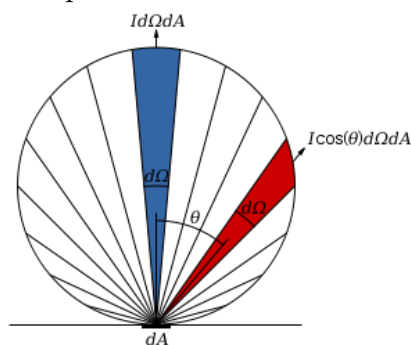
The fifth and final key point describes how the method handles anti-aliasing, by which it simply keeps diving each section into smaller sections until all four vertex points are a reasonably close average of light intensity, or it runs out of resolution. At the same time, this also ensures each object, no matter how small, receives intersection from a ray by giving those sections their own boundary sphere to allow the ray to know there is an object somewhere within it.

3 Terms and Equations

The paper was full of terms and equations I had never heard of before, starting right from the beginning with "illumination model", which is described later in the paper. I will be putting each new term/equation on its own paragraph from here on out.

Lambert's cosine law (also known as cosine emission law) is the law that states that the radiant intensity or luminous intensity observed from an ideal diffusely reflecting surface or ideal diffuse radiator is directly proportional to the cosine of the angle θ between the observer's line of sight and the surface normal.

Figure 1: Visual representation of Lambert's cosine law (1)



Bui-Tuong Phong shading is an interpolation technique for surface shading, also known as normal-vector interpolation shading. It specifically inter-

polates surface normals across rasterized polygons and computes pixel colors based on the interpolated normals and a reflection model. (2)

The intensity equation from Phong's model, better described by the paper itself:

Figure 2: Intensity from Phong's model

$$I = I_a + k_d \sum_{j=1}^{j=ls} (\vec{N} \cdot \vec{L}_j) + k_s \sum_{j=1}^{j=ls} (\vec{N} \cdot \vec{L}'_j)^n, \quad (1)$$

where

I = the reflected intensity,
 I_a = reflection due to ambient light,
 k_d = diffuse reflection constant,
 \vec{N} = unit surface normal,
 \vec{L}_j = the vector in the direction of the j th light source,
 k_s = the specular reflection coefficient,
 \vec{L}'_j = the vector in the direction halfway between the viewer and the j th light source,
 n = an exponent that depends on the glossiness of the surface.

Blinn and Newell's method of lighting techniques is a paper in itself, but boiled down is basically a less sophisticated and realistic method for what this paper is outlining. (3)

See paper's 17th reference for Kay. Otherwise, same as comment on Blinn and Newell.

The actual method, described better in the paper:

Figure 3: Paper's proposed model

$$I = I_a + k_d \sum_{j=1}^{j=ls} (\vec{N} \cdot \vec{L}_j) + k_s S + k_t T, \quad (2)$$

where

S = the intensity of light incident from the \vec{R} direction,
 k_t = the transmission coefficient,
 T = the intensity of light from the \vec{P} direction.

Fresnel's reflection law is similarly described adequately in the actual paper: the coefficients should vary as a function of incidence angle in a manner that depends on the material's surface properties.

Bounding volumes are also described in the key points, but I'd never heard of them until now.

4 Conclusion

This paper shed a lot of light (heh) on how complicated lighting actually is in comparison to actually just drawing the shapes within OpenGL. In fact, you could argue that the drawing is the easy part; it's making everything reflect light realistically that is so difficult. So many different techniques are used, and are probably going to be developed, that it's easy to get lost and confused in how to go about doing it. However, this paper presented an algorithm that seemed, at least on the surface, relatively straightforward. All that really seemed needed to be adjusted is the refraction index, to set the "surface material" of each shape.

But I also learned some of the mathematical and mechanical foundations behind each model, how it goes about actually drawing and handling each "ray" of light, how it traces and uses each shape it encounters to calculate the light intensity by the time it finally reaches the viewer.

In conclusion, this paper, which seemingly scary, was in reality a decent crash course in the theory behind lighting systems in graphics processing, while outlining a good technique to use as an underlying graphics lighting system.

5 Bibliography

- (1) en.wikipedia.org/wiki/File:Lambert_Cosine_Law_1.svg
- (2) en.wikipedia.org/wiki/Phong_shading
- (3) ddm.ace.ed.ac.uk/lectures/DDM/Intro_Digital_Media/Lecture3/slides/p542-blinn.pdf