433-324 Graphics and Interaction
Surface rendering and shading

Adrian Pearce

Department of Computing and Information Systems
University of Melbourne

The University of Melbourne

Lecture outline

Introduction

Surface rendering and shading

Gouraud shading

Phong shading

How are polygonal surfaces rendered?

Aim: understand how to *use* illumination models to create realistic shading through surface rendering.

Reading:

- Foley Section 14.2 Shading models for polygons.
- Angel Section 5.5 Polygonal Shading
- Akenine-Möller Section 5.Polygonal Shading
- Akenine-Möeller 5.5.1 Imlementing the Shading Equation

Shading techniques

How do we put these formulas to use? For polyhedral objects, we can use *flat* shading or *full* shading.

- In flat shading, compute the shading at some representative point of each (visible) polygon face (may be a vertex, or the centroid of the face, assuming it lies within the face). Then use this computed value to fill the whole polygon.
- In full shading, do the complete shading calculation at each pixel as the polygon is displayed.

In flat shading,

- the same value is used to render the entire polygon,
- it misses out on shading variations.

It is therefore realistic only for distant viewer and distant light source.

If our polyhedron is really intended as an approximation to a curved object, then *either* flat shading *or* full shading are inappropriate.

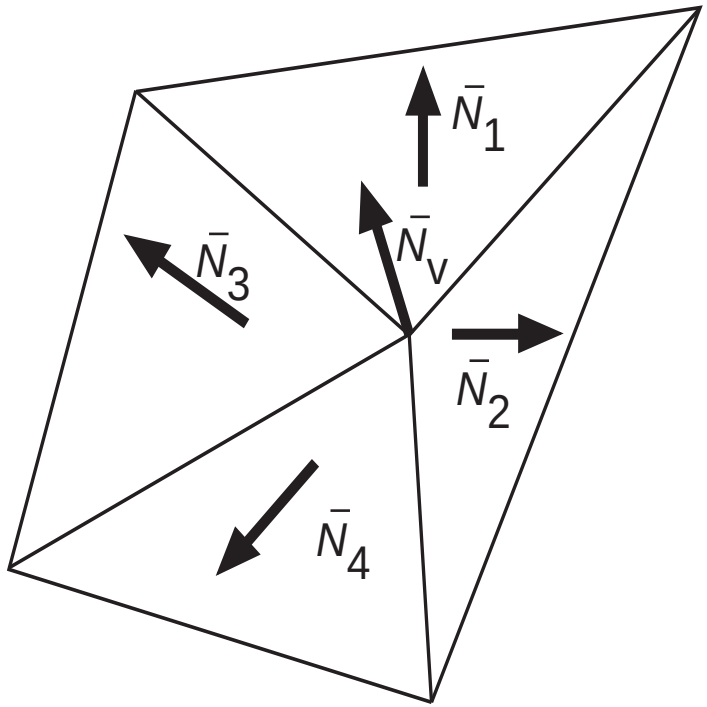
The discontinuities in surface normal where the face meet will lead to artefactual *discontinuities* in the shading.

There are two cures for this: *Gouraud* shading and *Phong* shading. In both, compute an "average" surface normal at each visible vertex, by averaging the normals of the faces that meet there.

- Because of the contrast enhancing caused by lateral inhibition in our visual systems, these false edges are strongly perceived and are therefore even more objectionable.

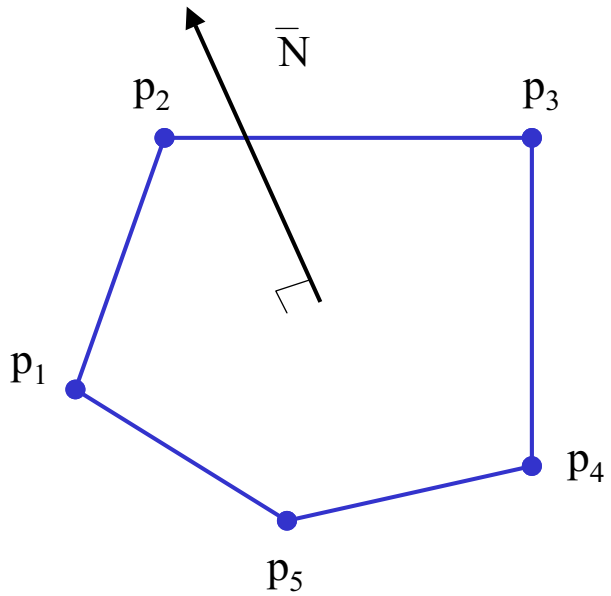
Gouraud shading

Normalised polygon surface normals averaged to obtain vertex normals in Gouraud shading (Foley Figure 14.18).



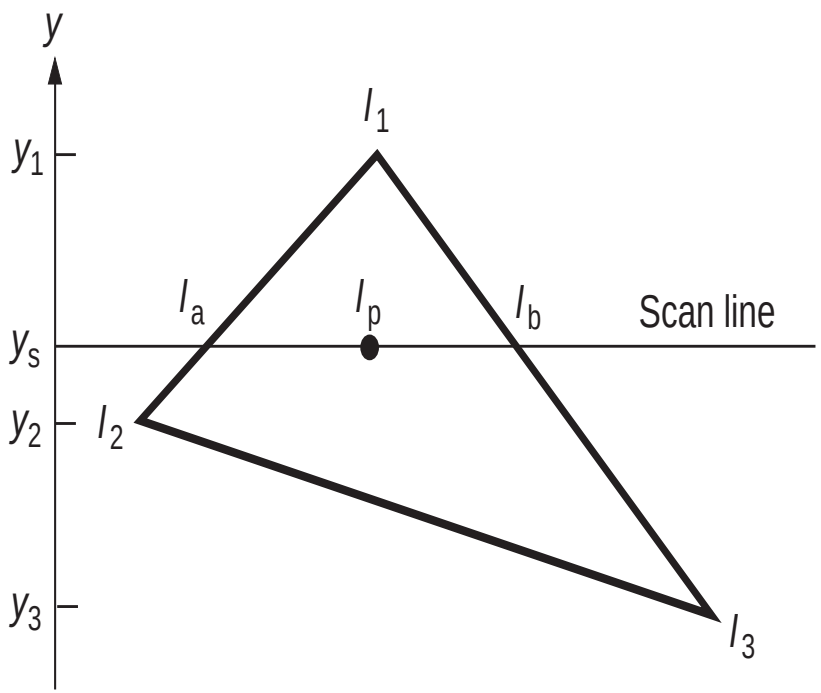
Calculating surface normals

Assuming all polygons store vertices in same *walking order*, the sign for surface normals should be correct and obtained simply calculating the cross products of the vectors $(\mathbf{p_2} - \mathbf{p_1})$ and $(\mathbf{p_3} - \mathbf{p_2})$.



$$\mathbf{N} = (\mathbf{p_2} - \mathbf{p_1}) \times (\mathbf{p_3} - \mathbf{p_2})$$

In Gouraud shading, compute the shading at each vertex. When filling a polygon in a scan-line fashion, interpolate the shading linearly along each edge, and then interpolate linearly along each scan-line segment (Foley Figure 14.19).



$$I_a = I_1 - (I_1 - I_2)\frac{y_1 - y_s}{y_1 - y_2}$$

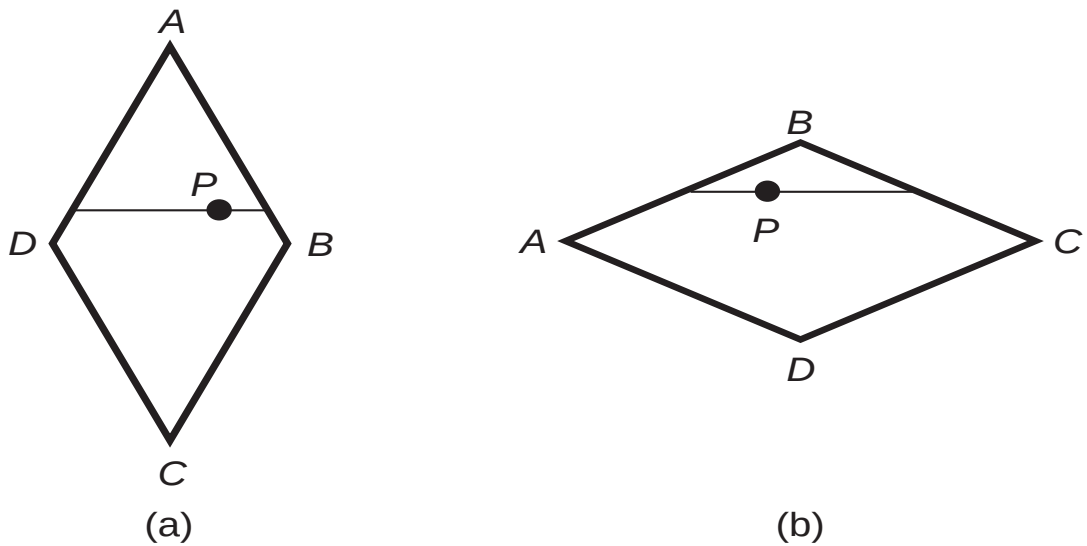
$$I_b = I_1 - (I_1 - I_3)\frac{y_1 - y_s}{y_1 - y_3}$$

$$I_p = I_b - (I_b - I_a)\frac{x_b - x_p}{x_b - x_a}$$

- Gouraud shading is also called intensity interpolated shading.

Orientation dependence

The results of interpolated-shading is not independent of the projected polygons position (Foley Figure 14.22).



Perspective distortion

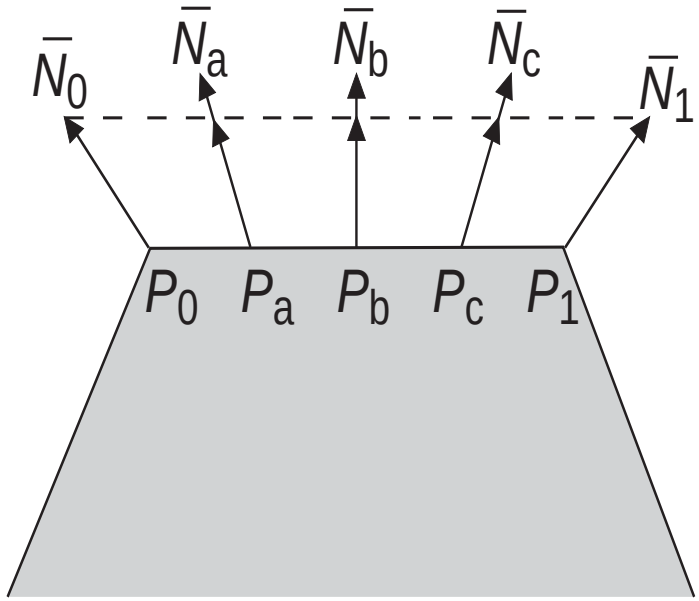
Imagine you have a polygon with vertex 1 more distant than vertex 2.

In this case you get perspective distortion, since perspective foreshortening means that the difference from one scan line to another increases in the direction of the further coordinate.

Under these conditions you can get polygonal silhouettes and edges can look weird.

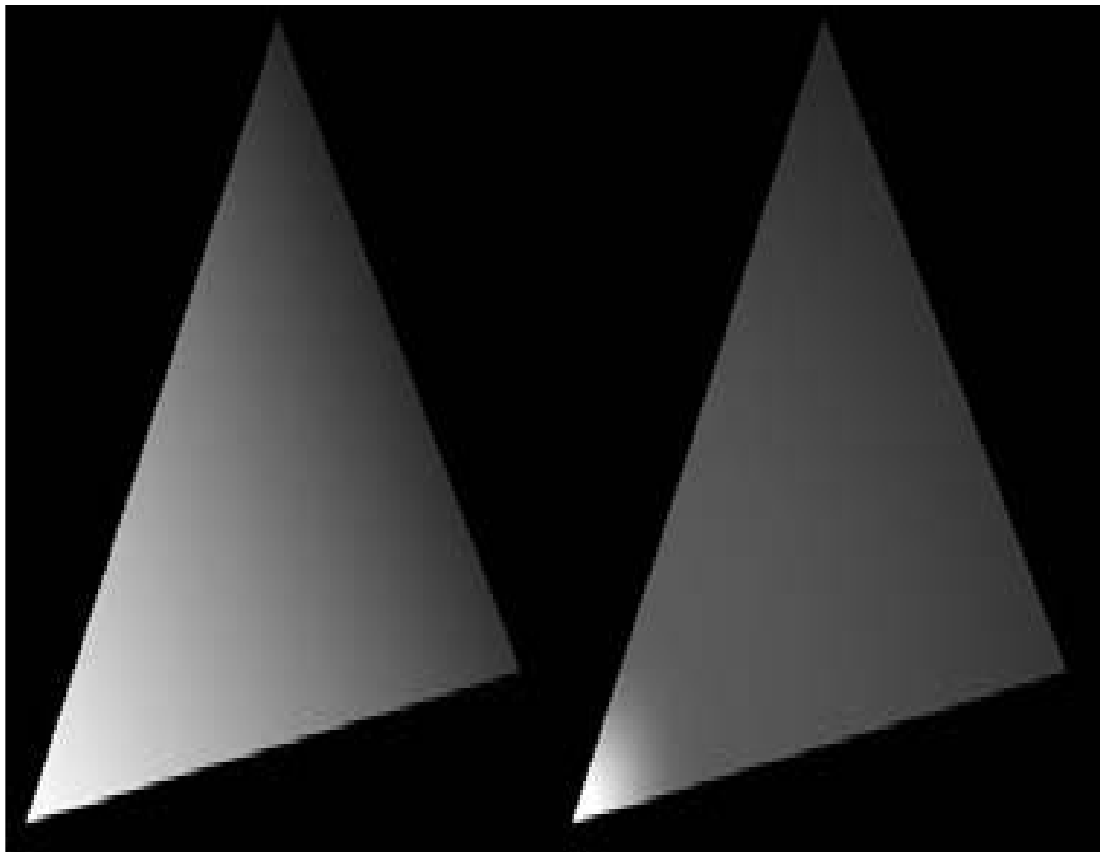
Phong shading

Phong shading is similar to Gouraud, except interpolate the surface normals, and do full shading calculation at every pixel using these interpolated normals (Foley Figure 14.20).

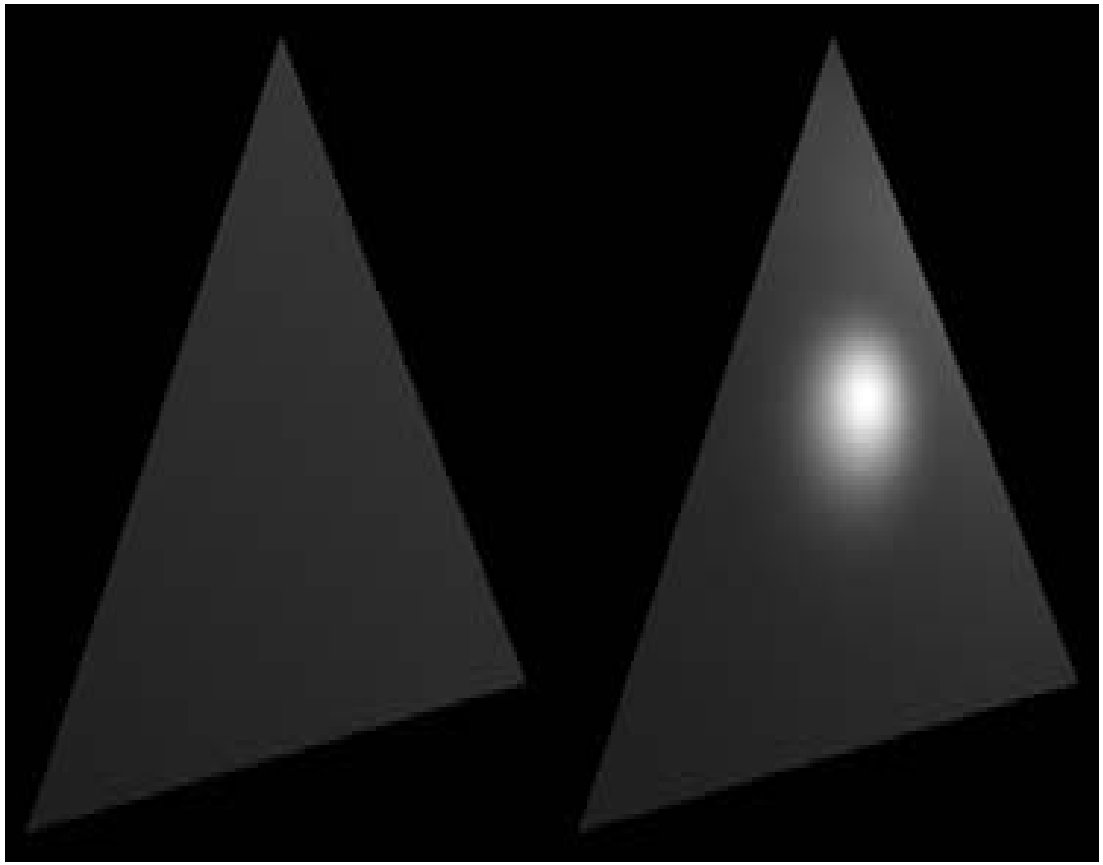


Normal vector interpolation used in Phong shading. What is the computational cost of Phong versus Gouraud?

Gouraud shading (left) Phong shading (right) with highlight falling at left vertex (Foley Figure 14.21).



Highlight falling in centre of polygon face (Foley Figure 14.21).

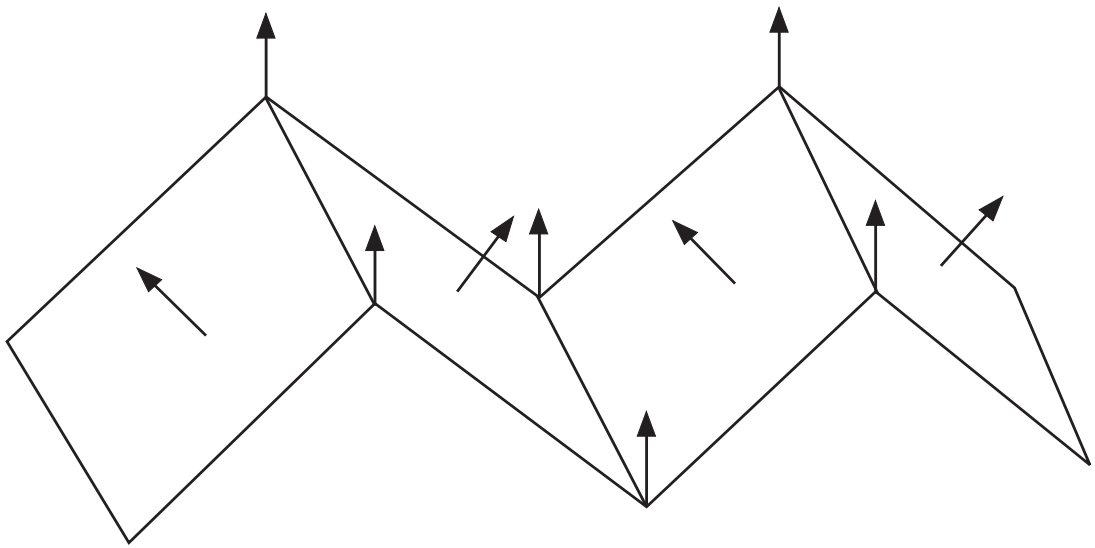


Gouraud shading with specular reflection

Intensity interpolation (Gouraud shading) *can* be used with *specular* reflection but get less pronounced highlights (according to specular reflection coefficient).

Unrepresentative vertex normals

Averaged vertex normals may not represent surface geometry adequately (Foley Figure 14.24).



If interpolation is used, this example *should* appear like corrugated iron!

- In this example, there will be little or no variation in the shade, if the light source is distant.
- This is because we *interpolate* between the normals, therefore the *actual* surface normal information inbetween is lost!
- Particularly true for objects represented by with lots of small polygons.

Summary

- Polygons can be rendered to produce curved surfaces by calculating surface normals at vertices and performing either
  - interpolation over intensity (Gouraud shading), or
  - interpolation over surfaces normals (Phong shading).
- Phong shading requires calculation of the illumination model at each *pixel*, whereas
- Gouraud shading only requires calculating the illumination model at each *vertex*.