5.2 Illumination and Shading Methods

5.2.1 Illumination Theory and Models

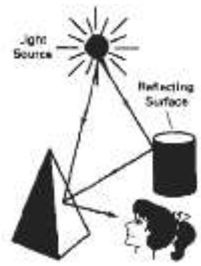
Realistic displays of a scene are obtained by **perspective projection** and applying **natural light effects** to the visible surfaces an illumination model (lighting model) and sometimes called shading model,
Shading Model is used to **calculate the intensity of light** that we should see at a given point on the surface of an object.
A surface rendering algorithm uses the intensity calculations from an illumination model to determine the light intensity calculation from an illumination model.

Light sources

Point source: tungsten filament bulb image can be seen

Distributed light source: fluorescent light

When light is incident on an opaque surface part reflected part
absorbed. Surface that are rough or grainy tend to scatter reflected light in all direction is called diffuse reflection



Light sources create highlights or bright spots called specular-reflection.

Ambient light is the background light present in a scene that comes from all directions. It's not from a direct light source — instead, it represents light that has bounced around so much (reflections, scattering) that it becomes uniform across the scene.

5.2.1.1 Ambient Light

Ambient light surface directly not exposed directly but visible if nearby objects are illuminated

Combination of light reflections from various surfaces to produce a uniform illumination called the ambient light or background light(no shadow's produced)

It has no spatial or direction characteristics and amount on each object is a constant for all surfaces and over all directions

5.2.1.2 Diffuse Reflection

It is the effect exhibited by **grainy surfaces** that disperse light rays in all directions

Ambient light is an approximation of global diffuse, light effects .

Diffuse reflections **are constant over each surface** in a scene and are **independent of**

viewing direction

k_d or diffuse reflection coefficient or diffuse reflectivity (0 to 1)

k_d is nearly 1 for highly reflective surface and k_d is 0 where light absorbs (black surfaces)

Diffuse reflection intensity at any point on the surface as

$$\mathbf{I_{ambDiff} = k_d \; . \; I_d}$$

Where $I_{ambDiff}$ is ambient light due to diffusion and Id is light due to diffusion assuming diffuse reflections from the surface are scattered with equal intensity in all directions independent of the

Viewing direction (called “ideal diffuse reflectors”) also called Lambertian reflectors and governed by Lambert’s Cosine Law.

If “angle of incidence” between incoming light direction and surface normal is 0

$$I_{LDiff} = k_d \,.\, I_L \cos \Theta$$

where I_{LDiff} is light due to diffusion

If N is unit normal vector to a surface and L is unit direction vector to the point light source then

$$I_{LDiff} = k_d \,.\, I_L (N \,.\, L)$$

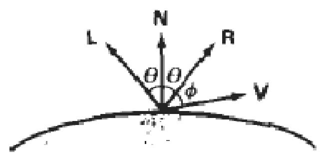
In addition many graphics packages introduce an ambient reflection coefficient k_d to modify ambient light intensity k_d then

$$I_{Diff(i.e.\ total)} = k_a \,.\, I_a + k_d \,.\, I_L (N \,.\, L)$$

5.2.1.3 Specular Reflection (Phong Model)

When we look at an illuminated shiny surface, such as polished metal, an apple etc we see a highlight or bright spot , at certain viewing direction this phenomenon is called “specular reflection” and is the result of total or near total reflection of the incident light in a concentrated region around the “specular reflection angle”.

This angle is equal to the angle of incidence.



N – unit normal surface vector
R – unit vector in the direction of ideal specular reflection
L – unit vector directed towards point light source
V – unit vector pointing to viewer from the surface position
ø – viewing angle relative to specular reflection direction R

For ideal reflector (perfect mirror) incident light is reflected only in the specular reflection direction i.e. V and R coincide (ø = 0).

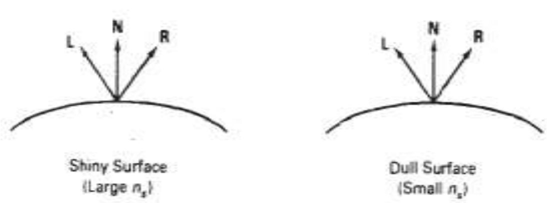
Shiny surfaces have narrow ø and dull surfaces have wider ø

An empirical model for calculating specular reflection range was developed by Phong Bui Tuong called “Phong specular reflection” model and it sets the intensity of specular reflection directly proportional to cosnsø ø → 0 to 90

Specular reflection parameter n_s is determined by type of surface

Very shiny surface has large n_s value

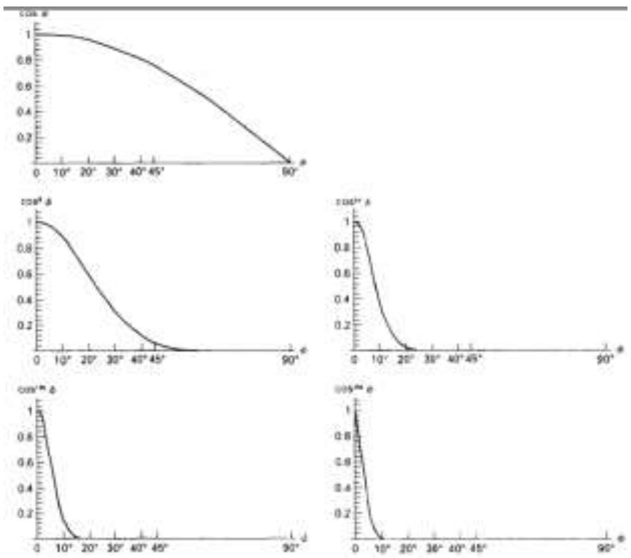
Very dull surface has smaller n_s value (down to 1)



Rough surface, e.g. chalk has n_s = 1

Intensity of specular reflection depends on material properties of surface, 0,other factors such as polarization , color of incident light.

For monochromatic specular reflections intensity variations can be approximated by SR coefficient W(ϴ)



W(ϴ) tends to increase as ϴ increases, at ϴ = 90 W(ϴ) = 1 and all incident light is reflected.

Fresnal's law of reflection describes specular reflection intensity with ϴ and suing W(ϴ), Phong specular reflection model as

$$I_{spec} = W(\Theta)\ I_L\ \text{Cosns}\ ø$$

where I_L is intensity of light source

ø is viewing angle relative to the specular reflection direction R.

So transparent materials like glass exhibit specular reflection as 0 approaches 90. At 0 = 0 about 4 percent of the incident light on a glass surface is reflected.

Now also $I_{spec} = W(ø)\ I_L\ (V . R)$ as $V . R = \text{Cosns}\ ø$

R can be calculated in term of N and L

$R + L = (2N . l)N$ or $R = (2N . l)N – L$

Simplified Phong model is obtained by halfway vector H between L and V to calculate the range of Specular reflections.

Replacing V . R in equation with N . H → Cos ø replaced by Cos α Half way vector H

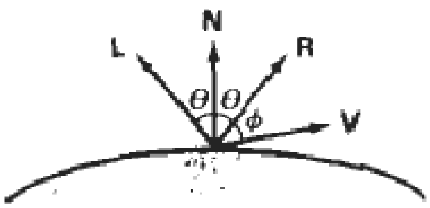
$$H = \frac{L+V}{|L + V|}$$

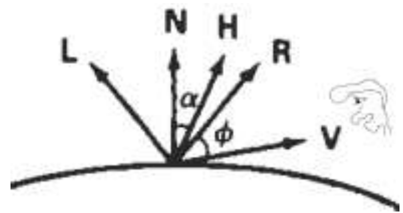
We add the light's direction vector and view vector together and normalize the result.

Halfway vector that is a unit vector exactly halfway between the view direction and the light direction

Instead of using the angle between the reflection vector and the viewing vector, the angle between the halfway vector and the surface normal vector is used

Specular highlights will be seen stronger when the angle between the halfway vector and the surface normal vector is minimum





If both viewer and light source are sufficiently far from surface both V and L are constant over the surfaces. Hence H is constant.

For non Planar surfaces N.H requires less computation than V.R

If V is coplanar with l and R(also N) then $\alpha = \varnothing / 2$

If V,L,N are non coplanar then $\alpha > \varnothing / 2$

Combined diffuse and specular reflections with multiple light sources

For single point light source

$$I = I_{diffuse} + I_{spec} = K_a I_a + K_d I_L (N . L) + K_s I_L (N . H)^{n_s}$$

For multiple light sources

$$I = K_a I_a + \sum_{i=1}^{n} I_{Li} [(K_d (N . L_i) + K_s (N . H_i)^{ns}]$$

5.2.2 Polygon Surface Shading Methods

Application of an illumination model to the rendering of the standard graphics objects those formed with polygon surfaces

The objects are usually polygon mesh approximation of curved surface objects but they may also be polyhedra that are not curved surface approximations

Scan line algorithms typically apply a lighting model to obtain polygon surface rendering in one or two ways each polygon can be rendered with a single intensity or the intensity can be obtained at each point of the surface using an interpolation scheme

5.2.2.1 Constant Shading (Flat Shading)

Fast and **simple** method for rendering of an object with polygon surfaces in CIS also called **flat shading**

Single intensity is calculated for each polygon and useful for quickly displaying the general appearance of curved surface

This method is accurate if

a. the object is a polyhedron and is not an approximation of an object with a curved surface
b. all light sources illuminating the object are sufficiently far from the surface
c. the viewing position is sufficiently far from the surface so that V.R is constant over the surface

Even if all conditions are not true , we can still reasonable approximate surface lighting effects using small polygon facets with flat shading and calculate the intensity for each facet at the center of the polygon.

Drawbacks:

Sharp **intensity discontinuation** is seen along the line joining two polygons

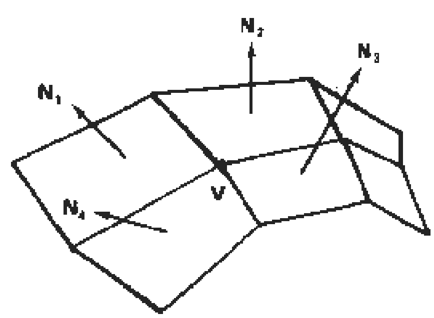
5.2.2.2 Gouraud Shading

This **intensity interpolation scheme** developed by Gouraud renders a polygon surface by **linearly interpolating intensity values across the surface**

Intensity values for each polygon are matched with the values of the adjacent polygon along the common edge thus **eliminating the intensity discontinuities occur in "flat shading"**

Calculation for each polygon surfaces

a. Determine the **average unit normal vector** at each polygon vertex.
b. Apply an **illumination model to each vertex** to calculate the vertex intensity
c. Linearly **interpolate the vertex intensities over the surface** of the polygon



N_1 normal to ABCD plane , N_2 normal to CDEF plane and so on .

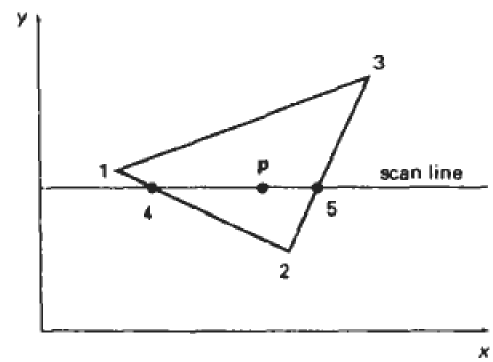
For any vertex position V normal unit vector

$$N_v = \frac{\sum_{k=1}^{n} N_k}{\left|\sum_{k=1}^{n} N_k\right|}$$

once N_v is known intensity at vertices can be obtained from lighting model

Average unit normal vector at a vertex are computed by taking the average of the normals of the neighboring faces. The resulting normal is a unit normal, the averaging is done by summing all the neighboring normals, and dividing it by the magnitude of this sum.

Next step: Interpolating intensities along polygon edges
fast method to find intensity at 4 using 1 and 2 using only vertical displacement



$$I_4 = \frac{y_4 - y_2}{y_1 - y_2} \cdot I_1 + \frac{y_1 - y_4}{y_1 - y_2} \cdot I_2$$

Similar process for I_5, using 3 and 2

For interior point p interpolated from the bounding intensities at point 4 & 5

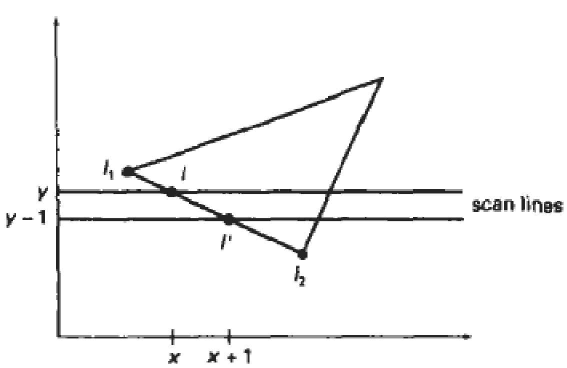
$$I_p = \frac{x_5 - x_p}{x_5 - x_4} I_4 + \frac{x_p - x_4}{x_5 - x_4} I_5$$

Easier than this is incremental calculations for successive edge intensity values

$$I = \frac{y - y_2}{y_1 - y_2} I_1 + \frac{y_1 - y}{y_1 - y_2} I_2$$

for next scan line y - 1

$$I' = I + \frac{I_2 - I_1}{y_1 - y_2}$$



Similar calculation to obtain intensities at successive horizontal pixel positions along each scan line

For color, intensity of each color component is calculated

Gouraud shading can be combined with a hidden surface algorithm to fill in the visible polygon

Advantages:

Removes **discontinuities of intensity at the edge** compared to constant shading model

Disadvantages:

Highlights on the surface are sometimes displayed with anomalous shapes and linear intensity Interpolation can cause bright or dark intensity streak s called **Mach Bands** to appear on the surfaces.

Mach bands can be reduced by dividing the surface into a greater number of polygon faces or Phong shading(requires more calculation)

5.2.2.3 Phong Shading

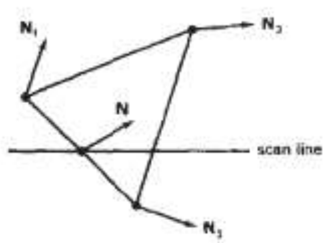
More accurate method for rendering a polygon surface is to interpolate normal vector and then apply the illumination model to each surface point called "Phong Shading" or "**Normal Vector Interpolation** Shading".

It displays more realistic highlights on a surface and greatly reduces Mach band effect.
Steps

a. Determine the **average unit vector** normal at each polygon vertex
b. Linearly **interpolate the vertex normals** over polygon surface
c. Apply an **illumination model** along each scan line to calculate projected pixel intensities for the surface points

N can be obtained by vertically interpolating between edge end point normals (N_1 and N_2)

$$\mathbf{N} = \frac{y - y_2}{y_1 - y_2} \cdot \mathbf{N_1} + \frac{y_1 - y}{y_1 - y_2} \cdot \mathbf{N_2}$$



Incremental methods are used to evaluate normals between scan lines and along each individual scan line (as in Gouraud) at each pixel position along a scan line the illumination model is applied to determine the surface intensity at that point

Advantage:
It produces **accurate results** than the direct interpolation

Disadvantage:
It requires considerable **more calculations,** making the approach slower in rendering

5.2.2.4 Fast Phong Shading

FPS approximates the intensity calculations **using a Taylor series** expansion and **triangular surface patches**

Surface normal at any point (x,y) over a triangle is $N = Ax + By + C$

A,B,C are determined from three vertex equations $N_k = Ax_k + By_k + C$ …. $k = 1,2,3$ (x_k, y_k vertex position)
Omitting reflexivity and attenuation parameters

$$I_{diff}(x,y) = \frac{L \cdot N}{|L| \cdot |N|} \qquad \frac{L \cdot (Ax + By + C)}{|L| \cdot |Ax + By + C|} \qquad \frac{(L \cdot A)x + (L \cdot B)y + L \cdot C}{|L|\ |Ax + By + C|}$$

We can write

$$I_{diff}(x,y) = \frac{ax + by + c}{(dx^2 + exy + fy^2 + gx + hy + i)^{1/2}} \qquad \text{……………………..(i)}$$

Where a,b,c,d are used to represent the various dot products eg $a = \dfrac{L \cdot A}{|L|}$

Finally denominator in eq(i) can be expressed as Taylor series expansion and retain terms up to second degree in x and y. This yields

$$I_{diff}(x,y) = T_5x^2 + T_4xy + T_3y^2 + T_2x + T_1y + T_0 \qquad \text{……………………..(ii)}$$

Where each T_k is a function of parameters a ,b ,c and so forth

Using forward difference we can evaluate (ii) with only two additions for each pixel position (x,y) once the initial forward different parameter have been evaluated

FPS is two times slower than Gouraoud shading, Normal Phong shading is 7 times slower than Gouraud

FPS can be extended to include specular reflections, FPS algorithms can be generalized to include polygons other than triangles and infinite viewing positions.