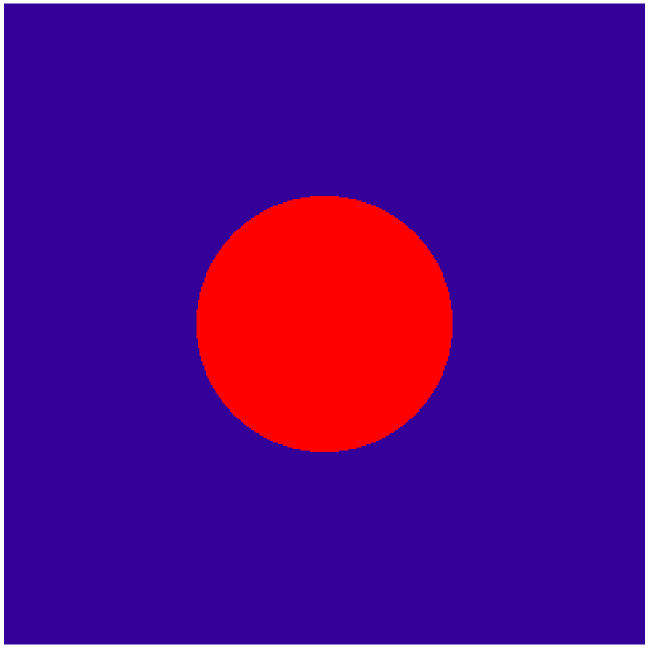
# Chapter 6:

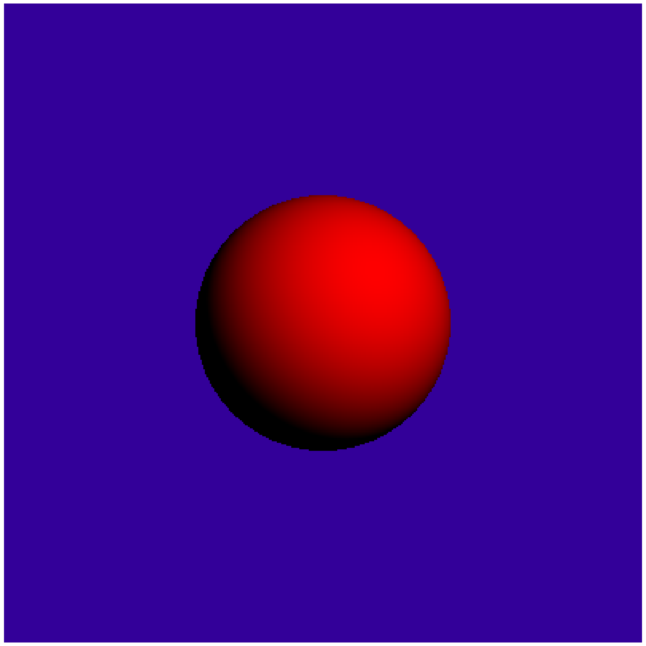
# Computer Graphics

Illumination Models &
Surface Rendering Methods

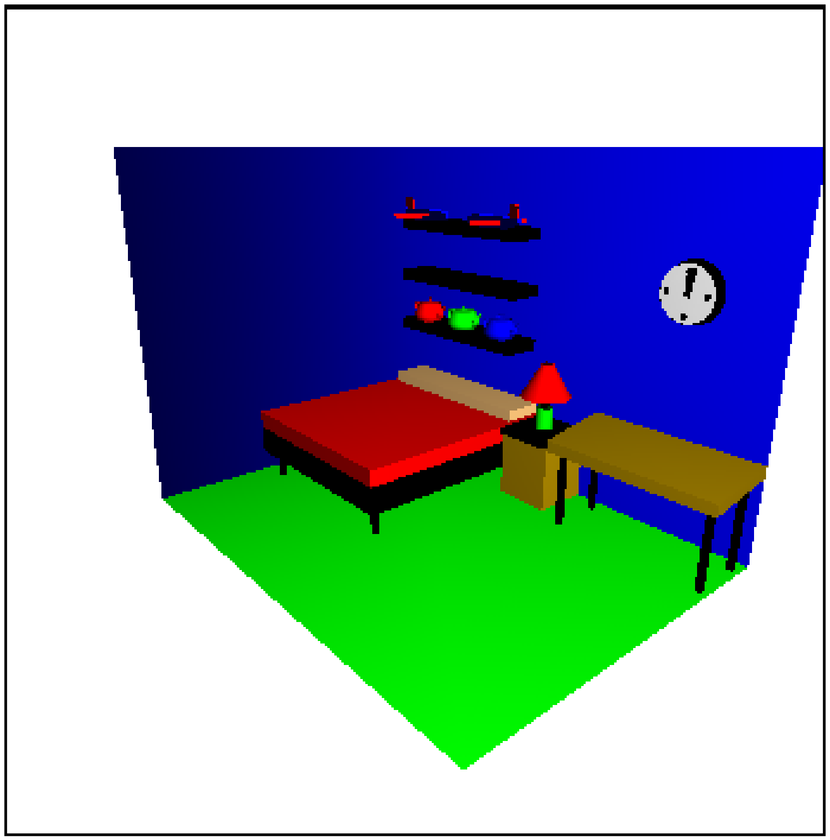
# Why Lighting?

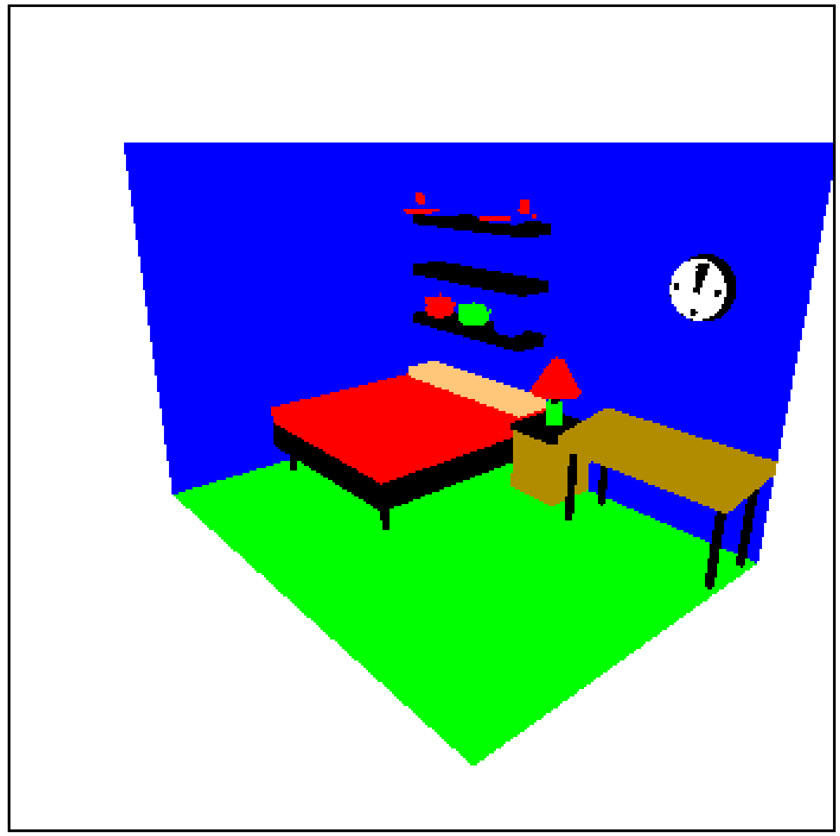
- If we don't have lighting effects nothing looks three dimensional!





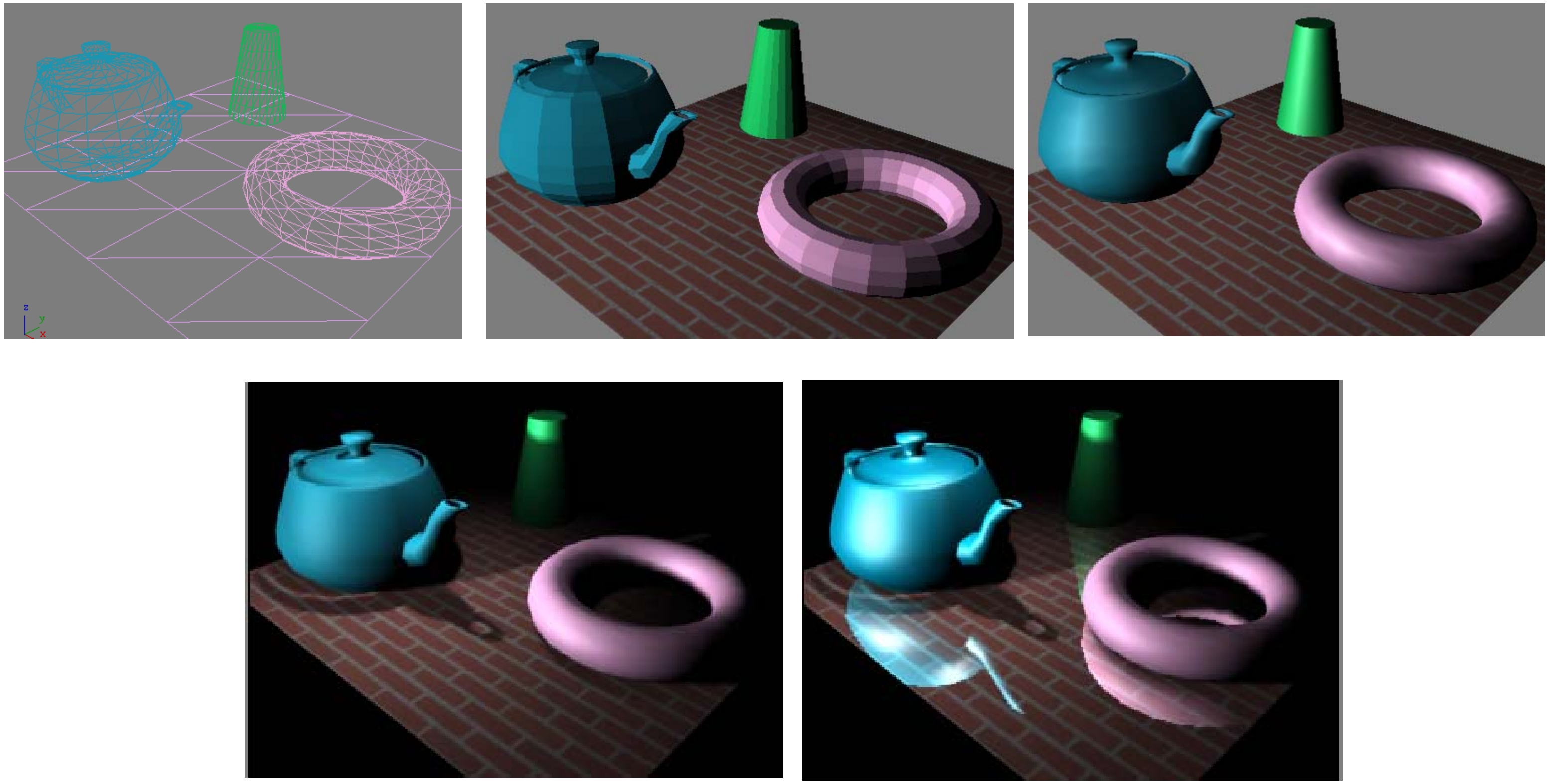
# Why Lighting? (cont...)





# Introduction

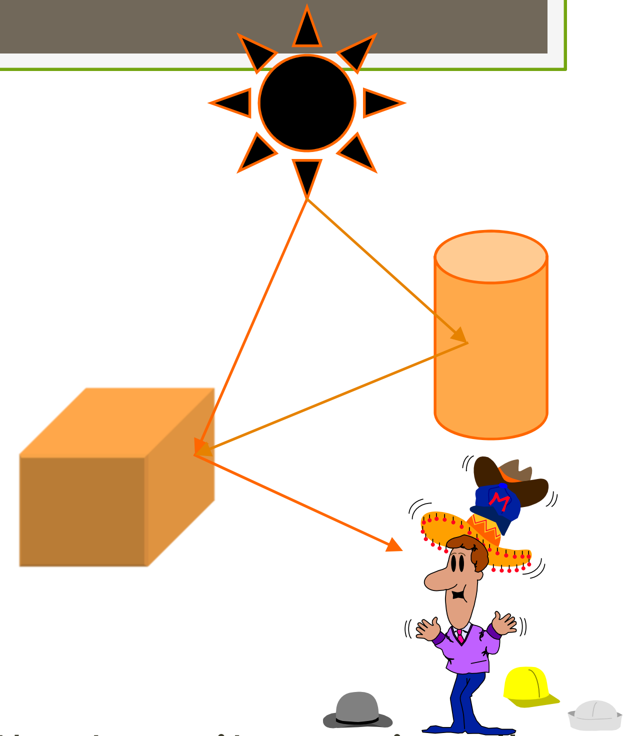
- Realistic displays of a scene
  - Perspective projections of objects
  - Applying lighting effects



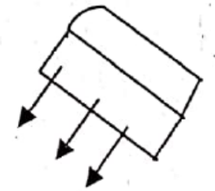
# Introduction (Contd.)

- Illumination model
  - Lighting model or Shading model
  - Calculate the intensity of light for a given point on the surface of an object
- Surface-Rendering algorithm
  - Use the intensity of a given point to determine the light intensity for all projected pixel position in a polygon

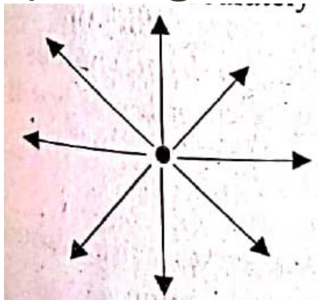
# Light Sources



- Types of light source
  - light source (direct)
  - light reflector (indirect)
- Two light emitter models
  - Point light source: The source that emit rays in all direction (e.g. Bulb)
  - Distributed (Area) light source: rays originate from the finite area (e.g. tubelight)



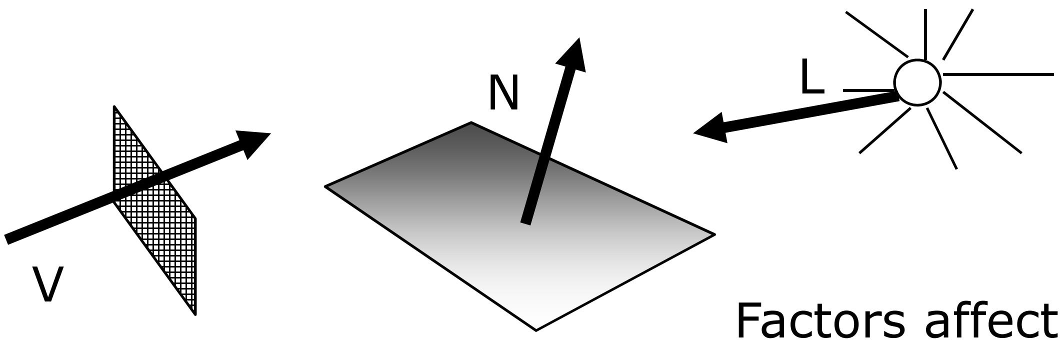
distributed light source



point source

# Illumination Models

- Concerning methods for calculating light intensity
  - Also called *Lighting* Models
  - An approximation for physical optical laws

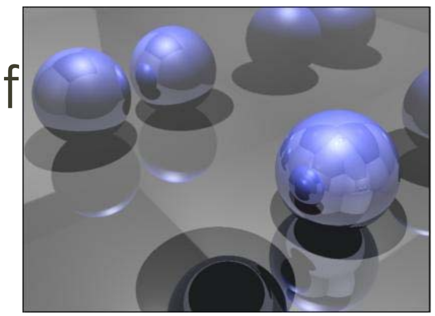


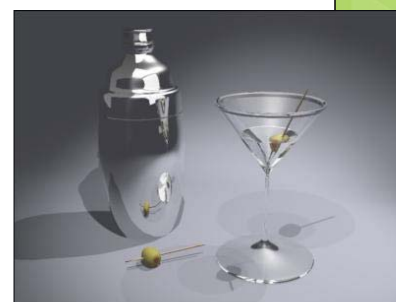

Factors affecting illumination models
- Position
- Orientation
- Material
- Light source
- Viewer

# Types of illumination models

- Local Illumination Models
  - Only considering the interchanges of the light sources

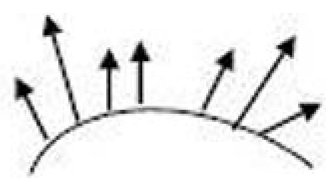

- Global Illumination Models
  - Concerning the interchange of light between all surfaces





# Reflected Light

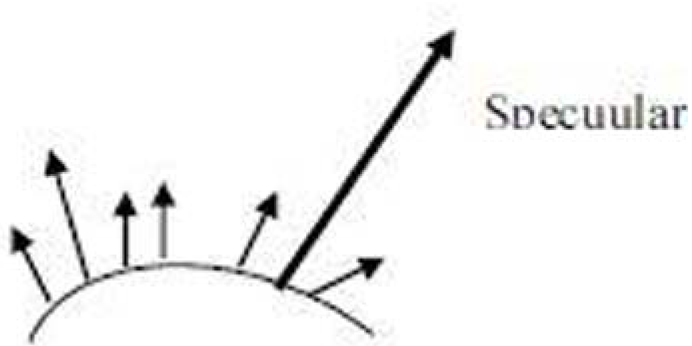
- When light is incident on a surface, some part of it is reflected, some part is absorbed and some part is transmitted through the material.
- Diffuse reflection
  - Rough or graining surface tend to scatter the reflected in all direction
  - With diffuse reflection, the surface appears equally bright from all viewing direction

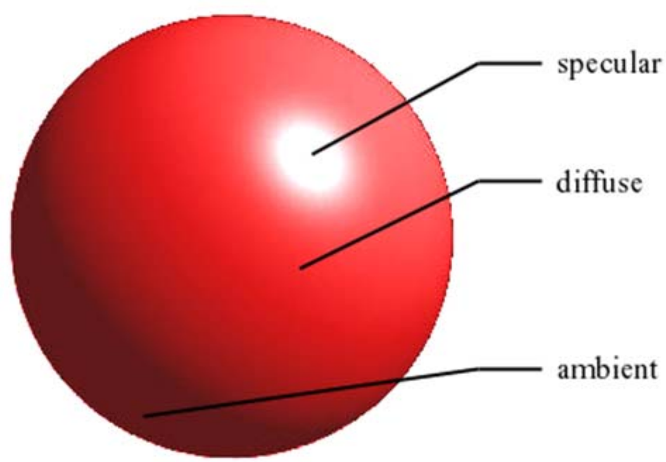


Diffuse reflection

# Reflected Light

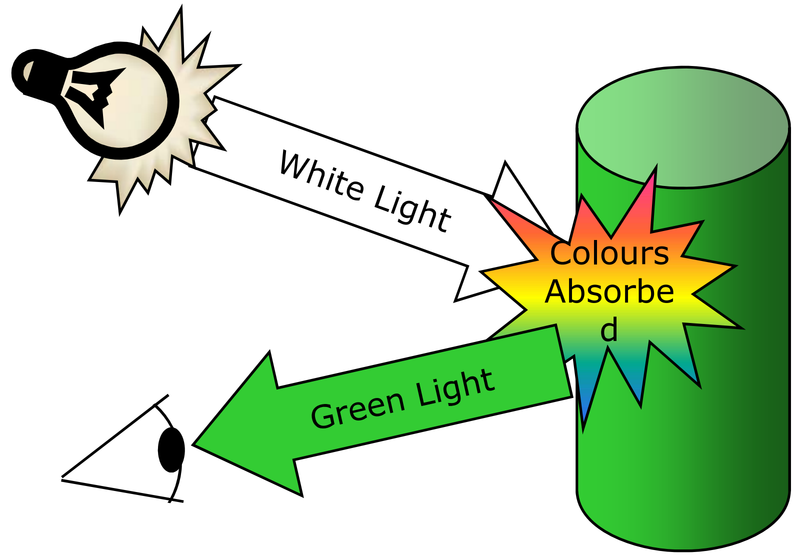
- Specular reflection
  - In addition to diffuse reflection, light sources create highlights or bright spots.
  - This is more pronounced on shiny surfaces.



# Reflected Light

- The colours that we perceive are determined by the nature of the light reflected from an object
- For example, if white light is shone onto a green object most wavelengths are absorbed, while green light is reflected from the object

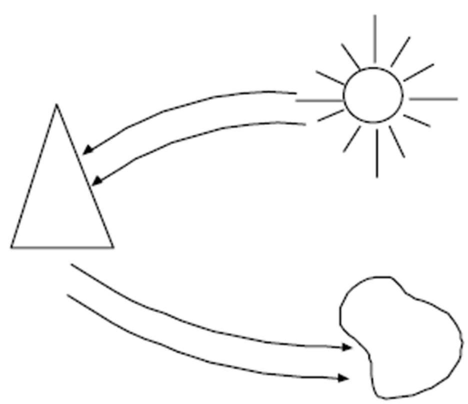


# Background

- Illumination model (Shading Model) is used to Calculate intensity of light that we see at a given point on the surface of an object.
- Intensity calculations are based on the optical properties of surfaces, the background lighting conditions and the light source.
- Surface Rendering algorithm uses the intensity calculation from an illumination model to determine the light intensity of all pixel positions for various surfaces in the scene
- Photorealism in computer graphics involves two elements:
  - Accurate graphical representations of object
  - Good physical descriptions of lighting effect in scene
- Illumination model: derived from physical laws that describe surface light intensities

# Basic Illumination Model

- Ambient Light → Background Light
  - Object not exposed directly to a light source but still is visible if nearby objects are illuminated
  - It is non directional light source that is the product of multiple reflections from surrounding environment.
  - Simple way to model the combination of light reflections from various surfaces to produce a uniform illumination call ambient light or background light



# Basic Illumination Model

- Ambient Light → Background Light
  - Has no spatial or directional characteristics
  - Amount of ambient light incident on each object is a constant for all surfaces and over all directions but the intensity of reflected light depends upon optical properties of the surface
  - Ambient light produces flat shading → not desirable in general, so scenes are illuminated with other light source together with ambient light
  - If $I_a$ is amount of ambient light incident on any surface, the ambient light reflection is given by,

    $I = K_a * I_a$

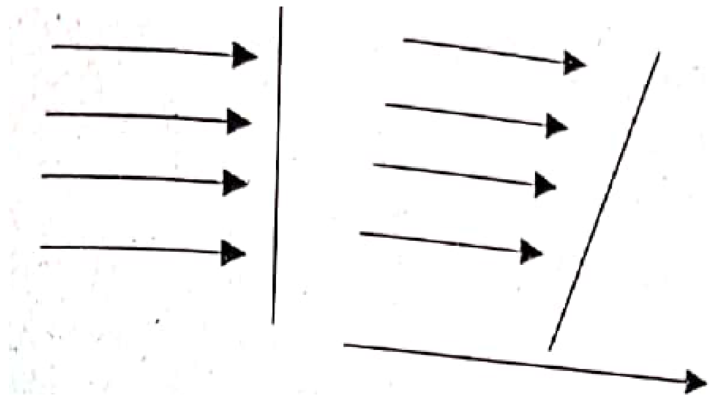
    Where, $K_a$ = ambient reflectivity or ambient reflection coffiecient which ranges from 0 to 1. (depends on material property)

# Basic Illumination Model

- Diffuse Reflection
  - Diffuse reflection are constant over each surface in a scene independent of viewing direction → *ideal diffuse reflectors*
  - Amount of incident light that is diffusively reflected is defined with a surface parameter $K_d$ called diffuse-reflection coefficient (diffuse reflectivity)
  - Value of $K_d$ is between 0 to 1. For highly reflecting surface $K_d$ → 1 and for dull surface $K_d$ → 0
  - Diffuse reflection intensity when scene illuminated only with ambient light:

$$I_{ambdiff} = K_a I_a$$

  - Illuminating object by a point light source, whose rays enumerate uniformly in all directions from a single point. The object's brightness varies form one part to another, depending on the direction and distance to the light source.
  - Surface perpendicular to the direction of incident light appears brighter than one with oblique angle to the direction of the incoming light.



# Basic Illumination Model

- Diffuse Reflection
  - Follow lambert's cosine law → radiant energy from a small surface area *dA* is proportional to the cosine of angle *θ* between surface normal and incident light direction
  - If θ is angle of incidence between incoming light direction and the surface normal then the intensity in projected area of a surface path perpendicular to light direction is proportional to cosθ.
  - If the incoming light from the source is perpendicular to the surface at particular point, that point is fully illuminated
  - As the angle of illumination moves away from the surface normal, the brightness of the point drops off.
  - If $I_1$ is the intensity of the point light source, then the diffuse reflection equation for a point on the surface is

$$I_{1diff} = K_d I_1 \cos\theta$$

  - A surface is illuminated by a point source only if the angle of incidence is in the range of 0 degree to 90 degree for which cosθ is in the range of 0 to 1. When cosθ is negative, the light source is behind the surface.

# Basic Illumination Model

- Diffuse Reflection
  - If $\vec{N}$ is unit normal vector to a surface and $\vec{L}$ is direction vector to the point source from a position on the surface then,

$$\vec{N}.\vec{L} = \cos\theta$$

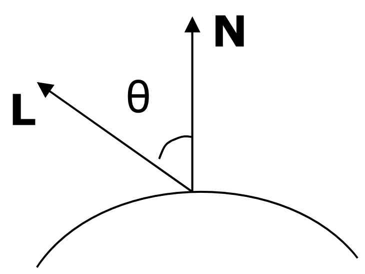
So,

$$I_{1diff} = K_d I_1 (\vec{N}.\vec{L})$$

  - We can combine the ambient and point source intensity calculation to obtain an expression for total diffuse reflection

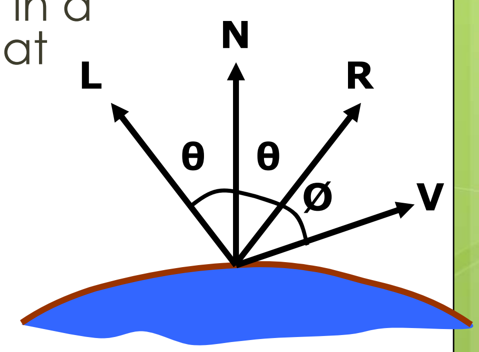
$$I_{diff} = K_a I_a + K_d I_1 (\vec{N}.\vec{L})$$

where both Ka and Kd depends upon surface material properties and are assigned value in the range from 0 to 1



# Specular Reflection

- Bright spot seen at an illuminated shiny surface when viewed at certain direction
  - Polished metal surface, person's forehead, apple etc. exhibit specular reflection
- In fact an image of light source
- Result of total or near total reflection of incident light in a concentrated region around the specular reflection at an angle θ
- Fig:
  - **L** → unit vector pointing to light source
  - **N** → unit surface normal vector
  - **R** → unit vector in direction of specular reflection
  - **V** → unit vector pointing viewer




- Ideal reflector exhibit specular reflection in the direction of R only (i.e Ø=0) but for non-ideal case specular reflection is seen over finite range of viewing positions
- Phong model, developed by Phong Bui-Tuong, is used to calculate the specular reflected range.

# Phong Model

- Intensity of specular reflection: proportional to

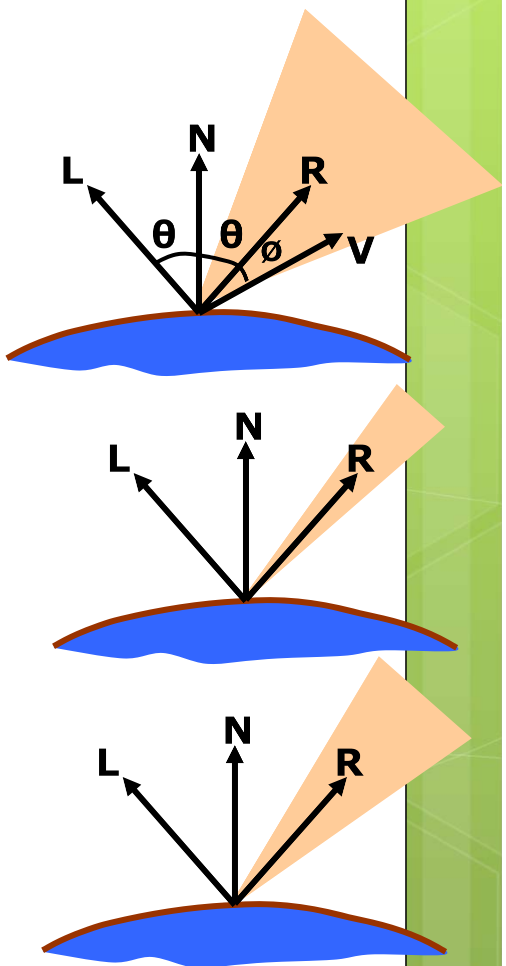
$$\cos^{n_s} \phi$$

  - $n_s$ → **specular reflection parameter** (depends on surface)
  - Ø ranges from 0 to $90^0$ (i.e cos Ø varies from 0 to 1)
- A very shinny surface is modeled with a large value for $n_s$ and duller surface is assigned smaller values. For perfect reflector $n_s$ is infinite
- Phong model calculate the specular reflection light intensity using **specular-reflection coefficient**, *w(θ)* for each surface

$$I_{spec} = w(\theta) I_l \cos^{n_s} \phi$$

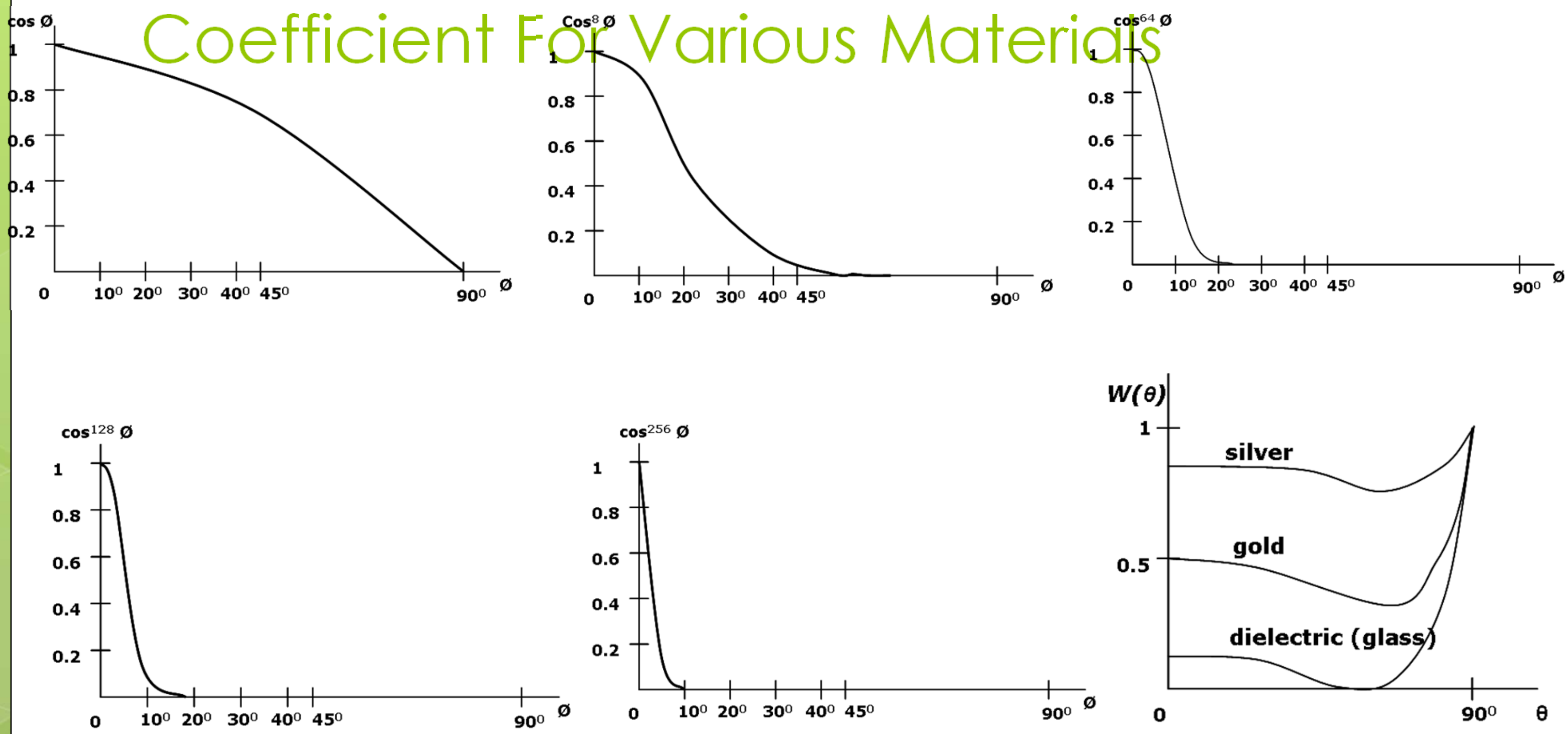
where $I_1$ = incident light intensity

- At $\theta = 90^0$, $w(\theta) = 1$ → all incident light is reflected



$$\cos^{n_s} \phi$$

## Plot For and Specular Reflection Coefficient For Various Materials

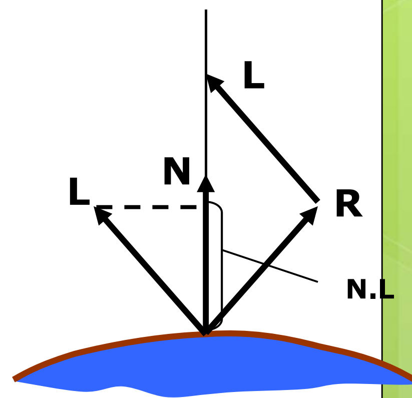


# Phong Model (contd...)

- Simplified form: assume *w(θ) =* $k_s$ *= constant*

$$I_{spec} = k_s I_l (\vec{V}.\vec{R})^{n_s}$$

- If we add ambient light and diffuse reflection component then total intensity is given as:




$$I = I_{diff} + I_{spec}$$
$$= k_a I_a + k_d I_l (\vec{N}.\vec{L}) + k_s I_l (\vec{V}.\vec{R})^{n_s}$$

- For multiple lig

$$I = k_a I_a + \sum_{i=1}^{n} I_{li} \left[ k_d (\vec{N}.\vec{L}_i) + k_s (\vec{V}.\vec{R}_i)^{n_s} \right]$$

# Intensity Attenuation

- For realistic lighting effects, we should take account the intensity attenuation.
- The intensity of radiant energy at a point *d* distance far from source is attenuated by $1/d^2$ *where d is the distance that the light has traveled*
- Using merely $1/d^2$ as attenuation factor for our simple single point light source model, too much intensity variation is produced when *d* is small and a little variation when *d* is large
- So, realistic image can't be produced by just considering $1/d^2$.
- Graphical packages have compensated the problem by using inverse linear quadratic function of d for intensity attenuation as:

$$f(d) = \frac{1}{a_0 + a_1 d + a_2 d^2}$$

where d is the distance of light source. a0, a1 and a2 are called "constant attenuation", "linear attenuation" and "quadratic attenuation". These are properties of light

- The Phong illumination model considering attenuation is:

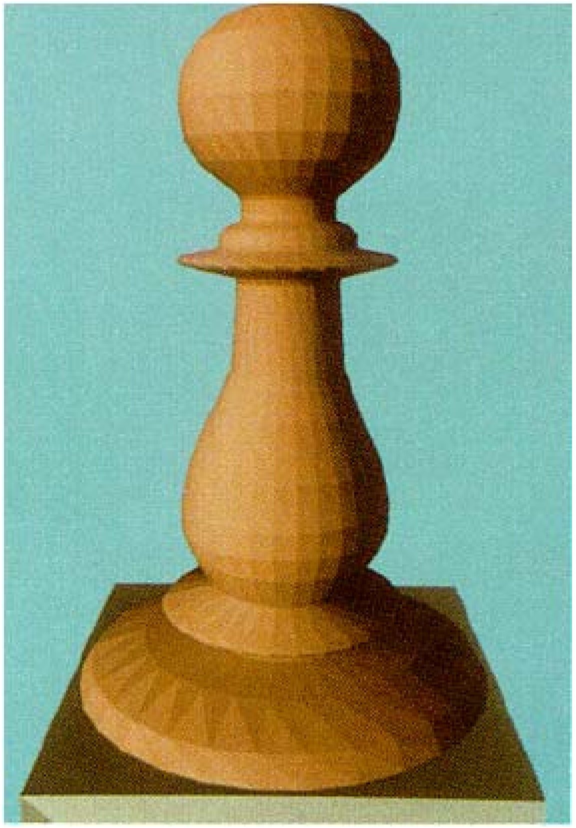
$$I = k_a I_a + \sum_{i=1}^{n} f(d_i) I_{li} \left[ k_d (\vec{N}.\vec{L}_i) + k_s (\vec{V}.\vec{R}_i)^{n_s} \right]$$

# Polygon Rendering Methods

- Illumination model is applied to fill the interior of polygons
- Curved surfaces are approximated with polygon meshes
  - But polyhedra that are not curved surfaces are also modeled with polygon meshes
- Two ways of polygon surface rendering
  - Single intensity for all points in a polygon
  - Interpolation of intensities for each point in a polygon
- Methods:
  - Constant Intensity Shading
  - Gouraud Shading
  - Phong Shading

# Constant Intensity Shading

- Flat shading
  - Each polygon shaded with single intensity calculated for the polygon
- Useful for displaying general appearance of a curved surface
- Assumption made for Accurate rendering conditions:
  - Object is a polyhedron and not an curved surface approximation
  - All light sources should be sufficiently far from the surface (i.e **N.L** and attenuation function are constant over the polygon surfaces)
  - Viewing position is sufficiently far (i.e **V.R** is constant over the surface)
- Drawback: intensity discontinuity at the edges of polygons



Flat rendering

Gouraud rendering

Flat rendering

Gouraud rendering

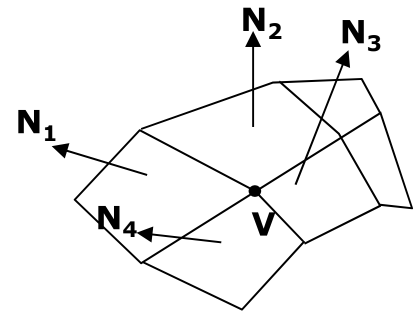
# Gouraud Shading

- Calculation Steps:
  - Determine the average unit normal vector at each polygon vertex
  - Calculate each of the vertex intensities by applying an illumination model
  - Linearly interpolate the vertex intensities over the polygon surface
- Intensity discontinuity at the edges of polygons is eliminated
- Drawback:
  - **Mach bands**: bright and dark intensity
    linear interpolation of intensities
  - Could miss specular reflection
    - Could be reduced by dividing the surfac
      of polygons or by using other methods, such as *Phong shading*

# Gouraud Shading (contd...)

- Average Unit Normal: Obtained by averaging the surface normals of all polygons sharing the vertex

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

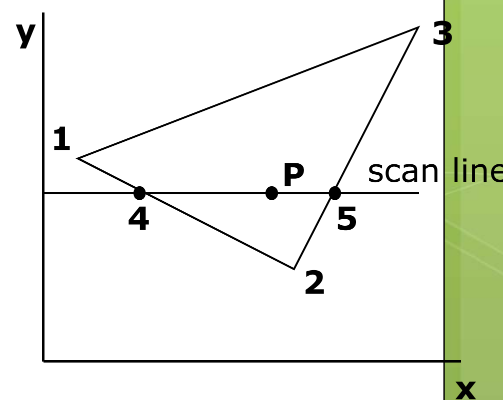



- Intensity interpolation:
  - Along the polygon edges are obtained by interpolating intensities at the edge ends

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

$$I_5 = \frac{y_5 - y_2}{y_3 - y_2} I_3 + \frac{y_3 - y_5}{y_3 - y_2} I_2$$

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

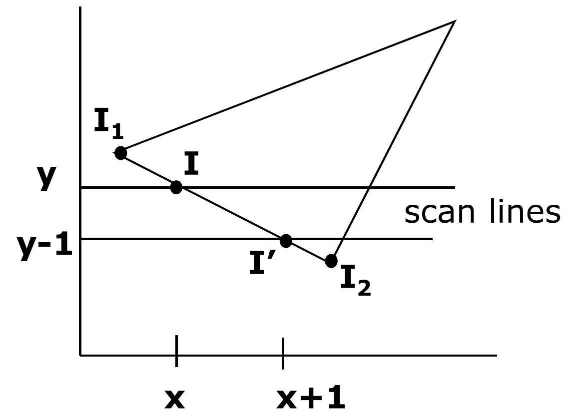


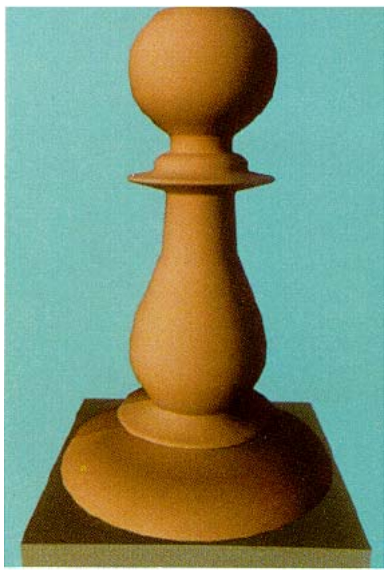
# Gouraud Shading (contd...)

Recursive Calculation along the scan line ??

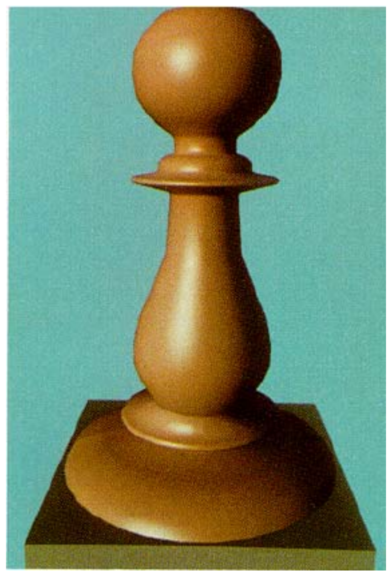
Recursive calculation along the edge

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



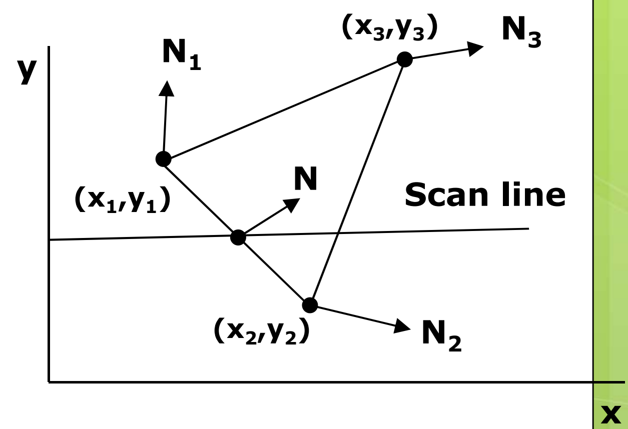
Gouraud rendering



Phong rendering

# Phong Shading

- More accurate method for rendering
- Fundamental: *Interpolate normal vectors and apply illumination model to each surface point*
- Calculation steps:
  - Determine average unit normal vectors at each polygon vertex
  - Linearly interpolate the vertex normals over the surface of the polygon
  - Apply an illumination model along each scan line to calculate projected pixel intensities for the surface points
- **Trade-off**: requires considerably more calculations




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

# Fast Phong Shading

- Surface rendering with Phong shading can be speeded up by using approximation in approximation in illumination model calculations of normal vector
- Fast Phong shading approximates the intensity calculations using Taylor series expansion and triangular surface patches.
- Since Phong shading interpolates normal vectors from vertex normals , we can express the surface normal N at any point (x,y) over a triangle as

$$N = Ax + By + C$$

Where A,B,C are determined from the three vertex equations.

$N_k = Ax_k + By_k + C$, k = 1,2,3 for (xk , yk ) vertex.

# Fast Phong Shading

- Omitting the reflectivity and attenuation parameters

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

- Using Taylor series expansion, we can write above expression as

$$I_{diff}(x,y) = T_5x^2+T_4xy+T_3y^2+T_2x+T_1y+T_0$$

- Using forward differences, we can evaluate above equation with only two additions for each pixel position (x, y) once the initial forward difference parameters have been evaluated.
- Although fast Phong shading reduces the Phong Shading calucations, it still takes approximately twice as long to render a surface with fast Phong shading as it does with Gouraud shading
- Normal Phong shading using forward differences takes about six to seven times longer than Gouraud shading

depth value calculation method

What are the differences between Constant shading, Gouraud shading, and Phong shading models? Using Gouraud shading technique to find the intensity at point (5,3) for the object with vertices A (0,0), B (5,0) C (4,2), D (1,2). The average intensities at the four vertices are $i_a = 5$, $i_b = 8$, $i_c = 9$, $i_d = 4$.