# Chapter -8 (14 Marks)

b

10. What is attenuation factor and Mach Band effect? Discuss the effect of Mach Band in Phong, Gourand and constant Shading. [2+2+3]
11. Define and explain the term ambient light, diffuse reflection and specular reflection with appropriate mathematical expressions. [2+2+3]

10. What do you mean by illumination model? Explain about specular reflection with appropriate mathematical expression. [2+5]
11. What are the different methods available for shading? Which one is more realistic and why? Explain with necessary derivation and algorithms. [1+6]

b

[2+6]

8. What is illumination model ? How light intensity of a point can be calculated? Also, discuss about the type of light source in intensity calculation. [2+6]
9. What is Phong Shading Model? Write down the algorithm for this shading model. Can we use this method to reduce Mach-Band effect? [2+4+2]

8. Find out the total intensity at the centroid of a triangle defined by A(2, 1, 1), B(0, 1, 1), C(0, 0, 1), when illuminated by a point light source of intensity $I_L = 0.6$ at (3, 2, 8) using Phong Illumination model. The viewer is at (4, 3, 8). Assume ambient intensity Ia = 0.1 and parameters: ka = 0.5, kd = 0.8, ks = 0.7, take x = 5. [centroid: (x1 + x2 + x3)/3, (y1 +y2 y3)/3, (z1 + z2 +z3)/3]. Explain briefly different ways of shading this triangle. [8+6]

[6+2]

8. What do understand by diffused and specular reflections and explain in detail how these terms are included in illumination model? [5+5]
9. Define the term illumination and rendering. Write down the steps for phong shading method. [2+6]

b

8. Derive the expression to calculate the total light intensity in a point. [8]
9. Compare and contrast between Gouraud and Phong shading model. [8]

8. Define the term Surface rendering with Illumination model. Derive an expression to calculate the intensity of Diffuse reflection with necessary equations and figures. How do you consider the distance to calculate the intensity for Specular and Diffuse Reflection? [2+5+3]
9. What is Phong shading method? Can we use this method to reduce Mach-Band effect? [6+2]

b

9. List down different types of object and explain how Phong illumination model is used to calculate intensity in for these objects along with mathematical expression. [8]
10. Explain in detail about Phong shading. [8]

***

9. Derive the expression to calculate the intensity of Specular Reflection in the presence of Point light source. Also write the expression for multiple light sources. How do you consider the distance to calculate the intensity for Specular Reflection? [8+4]
10. Write down an algorithm for intensity interpolation shading scheme. [7]

***

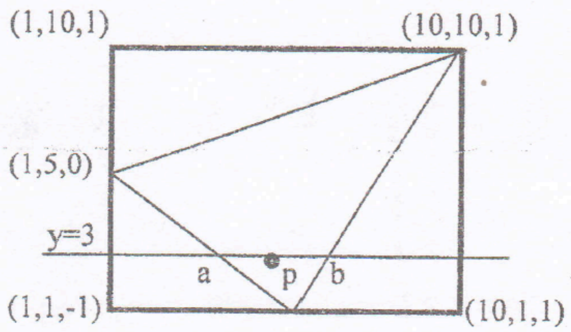
b

Compare the Gouraud shading with Phong shading. Develop the expression for Phong model considering the intensity attenuation for multiple point light sources with necessary figures. [6+8]

Develop a phong illumination model. Show how this model is used for rendering by deriving of expression for phong shading. [6+8]

b

8.




Find out intensity of light reflected from the midpoint P on scan line y = 3 in the above given figure using Gouraud shading model. Consider a single point light source located at positive infinity on Z-axis and assume vector to the eye as (1,1,1). Given d = 0, K = 1, $I_a = 1$, $I_L = 10$, Ks = 2, Ka = Kd=0.8 for use in a simple illumination model. [12]

b

9. Define and explain the term ambient light, diffuse reflection and specular reflection with appropriate mathematical expressions. [7]
10. Explain the method of Phong shading for polygon rendering. [7]

8. Explain about different types of lighting sources and how these light sources affect the illumination model? Explain about the intensity interpolation surface rendering technique by highlighting its pro and cons. Also give example about phong illuminations model. [3+5+6]

b

7. Explain the general illumination model. How this model is used for rendering by using gouroud shading. [7+7]

8. Define the terms: [10

i) Ambient light
ii) Lambert cosine law
iii) Diffuse reflection
iv) Specular reflection.

Also find equation for intensity of point by using Phong illumination model.

b

8. Under what condition(s) flat shading gives accurate rending? Mention the disadvantage of intensity interpolation technique and explain Phong shading with necessary mathematical calculation. Explain the diffuse reflection. [3+1+6+4]

8. Calculate the total intensity using phone secular reflection **model** by considering all type of light sources. [8]
9. Compare and Contract between Gouraud and Phong Shading **Mo**del. [8]

9. Explain the Gourad shading for polygon-rendering and compare it with phong shading. [8+2]

7. Discuss a constant intensity shading method. Mention the advantage of Phong shading over Gouraud shading. [7+3]

---

**Illumination** refers to the simulation of light in a scene. It's about how light interacts with objects, surfaces, and the environment to create realistic effects like shadows, reflections, and color variations

**Rendering** is the process of generating a final image or a sequence of images (for animations) from a 3D model.

## illumination model

In computer graphics, an **illumination model** (also known as a **shading model** or **lighting model**) is a mathematical framework used to simulate the way light interacts with surfaces to produce realistic images. It describes how light is scattered or absorbed when it hits an object's surface, influencing how the object appears to an observer. Illumination models help in determining the color and brightness of each pixel in a rendered image.

---

## Surface rendering algorithms

Surface rendering algorithms are techniques used in computer graphics to generate a visual representation of 3D surfaces in a 2D image. These algorithms take 3D models, which are often defined by their geometric shape, and compute how these surfaces should appear when viewed from a particular angle, with specific lighting, material properties, and other visual effects.

---

💡 Explain different types of light sources and how they effect the illumination model?

### Light Sources

Object that radiates energy are called light sources such as sun, lamp, bulb etc.

### Types of Light Sources

1. **Point Source**:
    - A **point light source** is a theoretical light source that emits light equally in all directions from a single point in space.

    **Effect on Illumination Model:**

    - **Sharp Shadows:** Point light sources create sharp, well-defined shadows because the light rays originate from a single point.
    - **Intensity Falloff:** The intensity of light decreases with distance from the source, typically following an inverse square law.

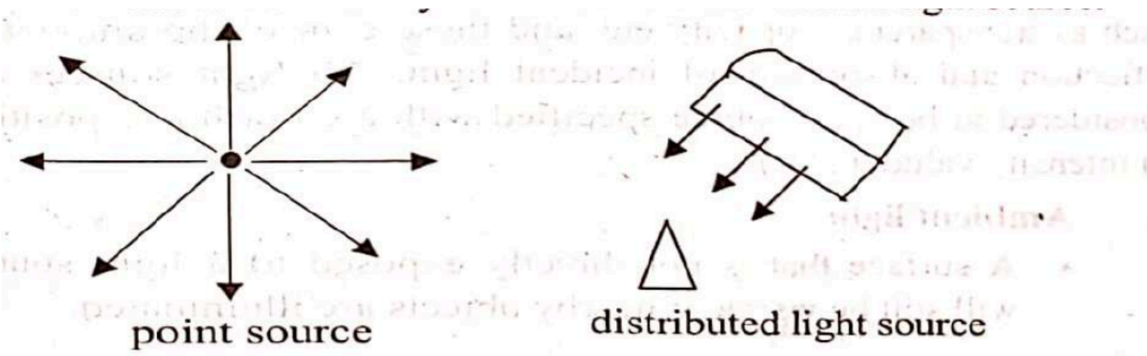
- **Highlights and Reflections:** Point lights produce small, bright highlights on shiny surfaces (specular reflection), depending on the surface's reflectivity and the viewer's position.

2. **Distributed Light Source**:

   - A **distributed light source** (also known as an **area light** or **extended light source**) emits light from a region of space rather than a single point. Examples include the sun (as seen from Earth), fluorescent lights, or a large LED panel.

   **Effect on Illumination Model:**

   - **Soft Shadows:** produce soft, diffuse shadows with gradual transitions from light to dark (penumbra). This occurs because different points on the light source illuminate objects from slightly different angles.
   - **More Realistic Lighting:** can better simulate real-world lighting conditions, where light often comes from extended sources (like windows or fluorescent lights), creating a more natural and realistic look.
   - **Smooth Intensity Variation:** Because the light is distributed over an area, the falloff in intensity is smoother compared to a point source.

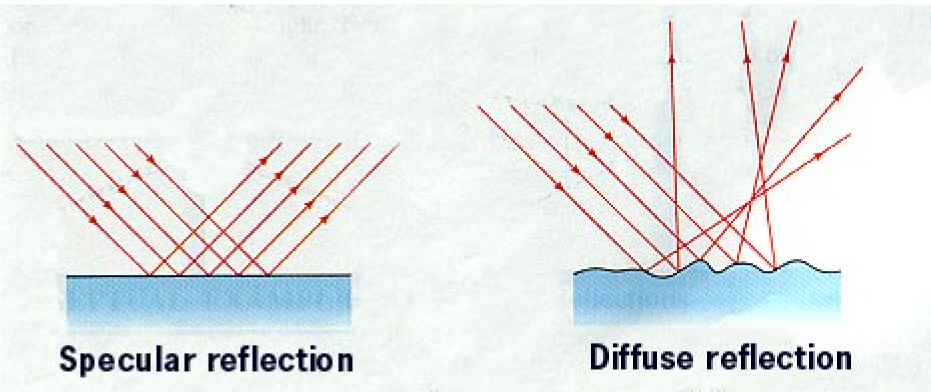



---

**Light reflection** is the process by which light rays bounce off a surface. When light hits an object, some of it is absorbed, and the rest is reflected. The nature of this reflection depends on the surface's characteristics and the light's angle of incidence.

## Types of Light Reflection

1. **Diffuse Reflection**:
    - **Diffuse reflection** occurs when light strikes a rough or matte surface, causing the light to scatter in many directions.
    - Because the surface is not perfectly smooth, the reflected rays are scattered in various directions, leading to a uniform brightness when viewed from any angle.
    - This type of reflection is common for everyday objects like paper, cloth, unpolished wood, or matte paint, giving them a soft, non-shiny appearance.
2. **Specular Reflection**:
    - **Specular reflection** occurs when light reflects off a smooth, shiny surface, such as a mirror, polished metal, or still water.
    - In this type of reflection, the light rays reflect in a single, consistent direction, following the law of reflection (the angle of incidence equals the angle of reflection).
    - Specular reflection results in clear, sharp images and bright highlights, known as **specular highlights**, that change position depending on the viewer's perspective.



## Basic Illumination Models

### 1. Ambient Light

**Ambient light** is the simplest form of lighting in an illumination model. It represents a constant light that is present everywhere in a scene, simulating indirect lighting that has bounced off multiple surfaces and is uniformly distributed.

- **Characteristics**:
  - It does not come from a specific direction.
  - It provides a base level of brightness to all objects in the scene, preventing them from appearing completely dark in shadowed areas.
  - Ambient light does not create shadows or highlights and affects all surfaces equally.

- **Expression for Ambient Light**:

  The intensity of ambient light $I_a$ on a surface is given by:

  $$I_a = k_a \cdot I_{ambient}$$

  where:

  - $k_a$ is the **ambient reflectivity** of the material (a value between 0 and 1), indicating how much ambient light the surface reflects.
  - $I_{ambient}$ is the **ambient light intensity** in the scene (a constant value).

### 2. Diffuse Reflection

**Diffuse reflection** models the way light is scattered uniformly in all directions when it strikes a rough or matte surface. This type of reflection depends on the angle of the incoming light relative to the surface normal.

- **Characteristics**:
  - It gives objects a matte appearance.

- The intensity of diffuse reflection is proportional to the cosine of the angle between the light source direction and the surface normal (Lambert's Cosine Law).

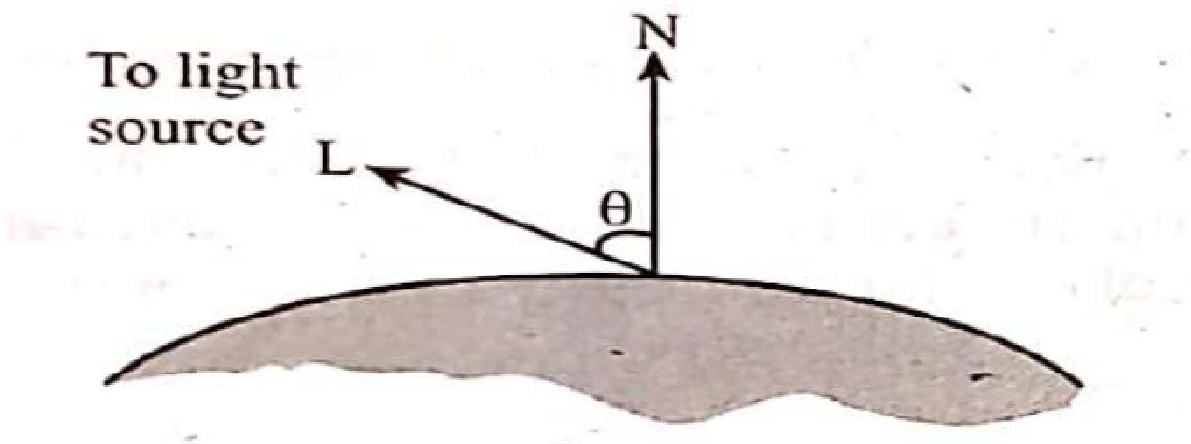
- **Expression for Diffuse Reflection**:

  The intensity of diffuse reflection $I_d$ on a surface is given by:

  $$I_d = k_d \cdot I_{light} \cdot (\vec{\mathbf{L}} \cdot \vec{\mathbf{N}})$$

  where:

  - $k_d$ is the **diffuse reflectivity** of the material (a value between 0 and 1), indicating how much diffuse light the surface reflects.
  - $I_{light}$ is the **intensity of the light source**.
  - $\vec{\mathbf{L}}$ is the **unit vector** in the direction from the surface point to the light source.
  - $\vec{\mathbf{N}}$ is the **unit normal vector** to the surface at the point of interest.
  - $(\vec{\mathbf{L}} \cdot \vec{\mathbf{N}})$ is the **dot product** of vectors $\vec{\mathbf{L}}$ and $\vec{\mathbf{N}}$, which equals $\cos(\theta)$, where $\theta$ is the angle between the light direction and the surface normal.




## 3. Specular Reflection (Phong Specular Reflection Model)

**Specular reflection** models the mirror-like reflection of light from smooth, shiny surfaces, resulting in highlights known as **specular highlights**. The Phong specular reflection model is a simple but effective way to simulate these highlights.

- **Characteristics**:

- It depends on the viewer's position.
- The intensity of the specular highlight is highest when the view direction aligns with the direction of the perfect reflection of the light source.

- **Phong Specular Reflection Model**:

  The Phong model introduces a shininess coefficient $n$ that controls the sharpness of the specular highlight. The intensity of specular reflection $I_s$ is given by:

  $$I_s = k_s \cdot I_{light} \cdot (\vec{\mathbf{R}} \cdot \vec{\mathbf{V}})^n$$

  where:

  - $k_s$ is the **specular reflectivity** of the material (a value between 0 and 1), indicating how much specular light the surface reflects.
  - $I_{light}$ is the **intensity of the light source**.
  - $\vec{\mathbf{R}}$ is the **reflection vector**, which is the direction that a perfectly reflected ray of light would take.
  - $\vec{\mathbf{V}}$ is the **unit vector** pointing towards the viewer or camera.
  - $(\vec{\mathbf{R}} \cdot \vec{\mathbf{V}})$ is the **dot product** of vectors $\vec{\mathbf{R}}$ and $\vec{\mathbf{V}}$.
  - $n$ is the **shininess exponent** (also called the Phong exponent), a material property that controls the size of the specular highlight; higher values lead to smaller and sharper highlights, indicating a shinier surface.

- **Calculating the Reflection Vector $\mathbf{R}$**:

  The reflection vector $\vec{\mathbf{R}}$ can be calculated using the surface normal $\vec{\mathbf{N}}$ and the light direction vector $\vec{\mathbf{L}}$ with the formula:

  $$\vec{\mathbf{R}} = 2(\vec{\mathbf{N}} \cdot \vec{\mathbf{L}})\vec{\mathbf{N}} - \vec{\mathbf{L}}$$

  where $(\vec{\mathbf{N}} \cdot \vec{\mathbf{L}})$ is the dot product between the surface normal $\vec{\mathbf{N}}$ and the light direction $\vec{\mathbf{L}}$.

## Lambert's Cosine Law

**Lambert's Cosine Law** states that the intensity of light reflected from a diffusely reflecting surface is directly proportional to the cosine of the angle (θ) between the incident light direction and the surface normal.

**Expression of Lambert's Cosine Law**

Mathematically, Lambert's Cosine Law is expressed as:

$$I = I_0 \cdot \cos(\theta)$$

where:

- $I$ is the **reflected light intensity** perceived by the observer.
- $I_0$ is the **incident light intensity** on the surface.
- $\theta$ is the **angle** between the incident light direction and the surface normal.

---

8. Derive the expression to calculate the total light intensity in a point. [8]

The **total intensity** $I$ of light at a point on a surface is the sum of the ambient, diffuse, and specular components:

$$I = I_a + I_d + I_s$$

Substituting the expressions for each component, we get:

$$I = k_a \cdot I_{\text{ambient}} + k_d \cdot I_{\text{light}} \cdot (\vec{\mathbf{L}} \cdot \vec{\mathbf{N}}) + k_s \cdot I_{\text{light}} \cdot (\vec{\mathbf{R}} \cdot \vec{\mathbf{V}})^n$$

## Summary of the Expression

The total intensity $I$ of light at a single point on a surface is given by:

$$I = k_a \cdot I_{\text{ambient}} + k_d \cdot I_{\text{light}} \cdot \cos(\theta) + k_s \cdot I_{\text{light}} \cdot (\cos(\alpha))^n$$

where:

- $\cos(\theta) = (\vec{\mathbf{L}} \cdot \vec{\mathbf{N}})$ is the angle between the light direction and the surface normal.
- $\cos(\alpha) = (\vec{\mathbf{R}} \cdot \vec{\mathbf{V}})$ is the angle between the reflected light direction and the viewer's direction.

---

**Shading** refers to the process of determining the color and brightness of a surface in a 3D scene to create the appearance of depth, texture, and lighting.

Shading plays a crucial role in making a 3D object look more realistic by simulating how light interacts with surfaces.

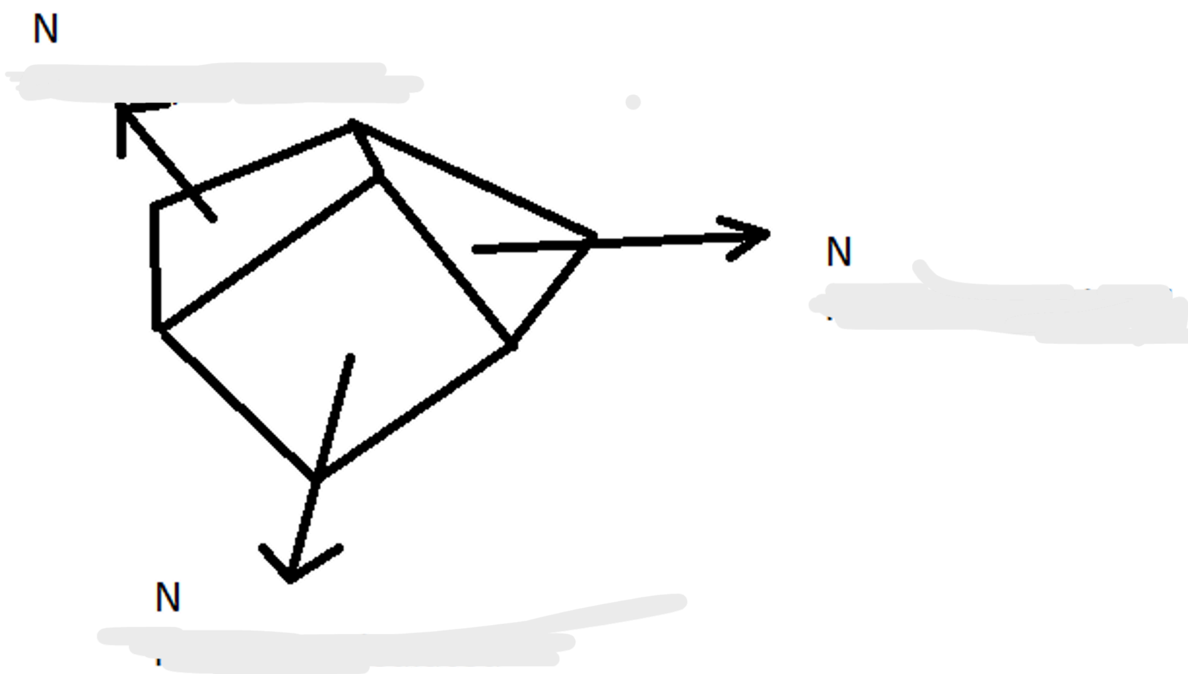
## Surface Rendering Methods

1. Constant Shading
2. Giraud Shading
3. Phong Shading
4. Fast Phong Shading

---

## Constant Shading

**Constant shading**, also known as **flat shading**, is a simple shading technique used in computer graphics to render 3D objects. This technique involves applying a single color to each polygon of a 3D model, making the entire surface of each polygon appear uniform in color and brightness.




## How Constant Shading Works

1. **Single Normal Per Polygon**: In constant shading, a single normal vector is calculated for each polygon (usually a triangle or quadrilateral). This normal is typically computed by taking the cross product of two edges of the polygon, resulting in a vector perpendicular to the polygon's surface.
2. **Single Light Calculation**: The shading (color and brightness) of the entire polygon is determined by computing the lighting model only once using the polygon's normal vector. The lighting calculation considers factors such as the light source direction, intensity, and the material properties of the surface.
3. **Uniform Appearance**: Since the shading is calculated once per polygon, every point on the polygon's surface is rendered with the same color and brightness. This gives the polygon a flat, uniform appearance.

## Advantages of Constant Shading

- **Speed and Simplicity**: Constant shading is computationally inexpensive and easy to implement because it requires minimal calculations per polygon.
- **Ideal for Simple Objects**: It is well-suited for rendering simple objects where high detail or smooth transitions in shading are not necessary.
- **Useful for Wireframe Views**: Often used in wireframe views or when the emphasis is on the shape and structure of the model rather than its surface details.

## Disadvantages of Constant Shading

- **Lack of Realism**
- **Visible Polygon Edges**: The uniform color per polygon can cause the edges between adjacent polygons to be quite noticeable.

---

## Gouraud Shading (Intensity Interpolation Method)

Gouraud shading is a technique used in computer graphics to simulate the smooth transition of colors across a surface. It's named after the French computer scientist Henri Gouraud, who introduced it in 1971. Here's how it works:

1. **Vertices Coloring**: The color of each vertex of a polygon is calculated based on its normal vector and the light sources in the scene. This involves computing the light intensity at each vertex.
2. **Interpolating Colors**: Once the colors of the vertices are determined, Gouraud shading interpolates these colors across the surface of the polygon. This interpolation happens across the entire surface of the polygon, giving the illusion of smooth shading.
3. **Rendering**: The interpolated colors are then used to render the polygon, creating a smooth gradient effect across its surface.

Gouraud shading is efficient because it reduces the computational load by limiting the shading calculations to the vertices and then interpolating the results. However, it can sometimes produce artifacts, especially in cases of high contrast or sharp changes in lighting, because it doesn’t account for variations within the polygon itself.

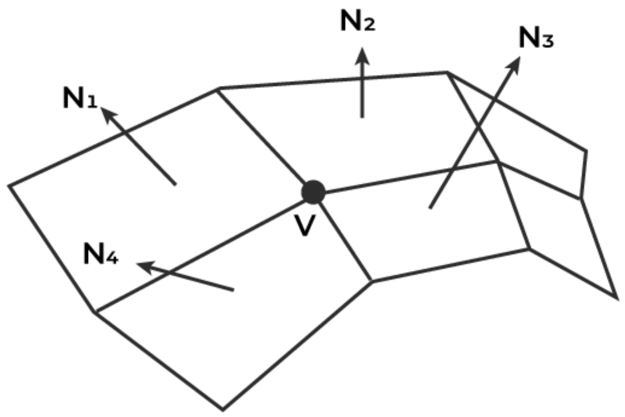
For more advanced shading and to handle such artifacts, techniques like Phong shading, which interpolates normals rather than colors, can be used.

**Pros of Gouraud Shading:**

1. Efficient and fast computation.
2. Smooth color transitions across surfaces.
3. Reduces the need for complex per-pixel calculations.
4. Suitable for real-time rendering applications.

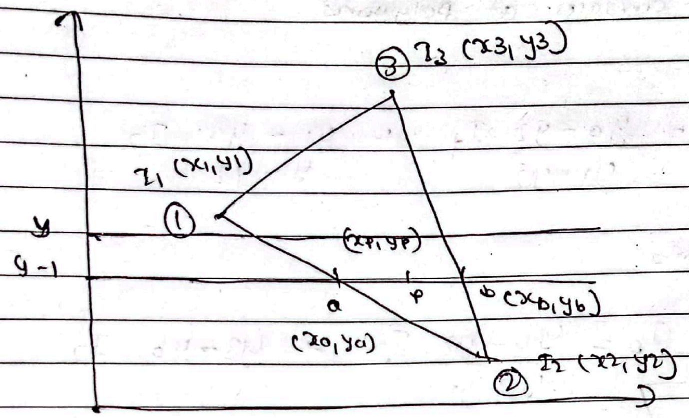
**Cons of Gouraud Shading:**

1. Can produce visible shading artifacts, especially on curved surfaces.
2. Less accurate for capturing intricate lighting details.
3. Doesn't handle specular highlights well.
4. Interpolated colors might not accurately represent the actual surface lighting.




The Normal Vector at Vertex V is Calculated as the Average of the Surface Normal for each Polygon Sharing that Vertex

Algorithm:




(1) Determine the average unit normal vector at each polygon vertex

$$N_v = \frac{\sum_{i=1}^{n} N_i}{\left| \sum_{i=1}^{n} N_i \right|}$$

$n \to$ no of surfaces of polygon sharing that vertex

(2) By illumination we get the intensity at each vertex

$$I = K_a I_a + k_d I_l (\vec{N} \cdot \vec{L}) + k_s I_l (\vec{V} \cdot \vec{R})^n$$

③ Linearly interpolate the vertex intensities over the ~~surt~~ surfaces at polygons

$$I_a = \frac{y_a - y_2}{y_1 - y_2} I_1 + \frac{y_1 - y_a}{y_1 - y_2} I_2$$

Similarly

$$I_b = \frac{y_b - y_2}{y_3 - y_2} I_3 + \frac{y_3 - y_b}{y_3 - y_2} I_2$$

Then at P :

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

And ~~after~~ along the edge we can make incremental calculations for intensity values.

Similar calculations are used to obtain intensities at successive horizontal pixel positions along each scan line.

## Phong Shading

Phong shading is a technique used in computer graphics to simulate the way light interacts with surfaces, providing a more realistic appearance than simpler shading models. It's named after Bui Tuong Phong, who introduced it in the 1970s. Here's a brief overview:

1. **Interpolation of Normals**: Phong shading involves interpolating surface normals across the polygons of a 3D model. Unlike flat shading, where a single normal is used per polygon, Phong shading interpolates normals across the surface of the polygon to give a smoother appearance.
2. **Lighting Model**: It uses the Phong reflection model to compute the color of each pixel. This model combines three components:
   - **Ambient Reflection**
   - **Diffuse Reflection**
   - **Specular Reflection**
3. **Per-Pixel Computation**: Phong shading performs calculations per pixel rather than per vertex, resulting in smoother and more detailed lighting effects. This is achieved by interpolating normals across the surface and computing the lighting model at each pixel.

Overall, Phong shading helps create more visually appealing and realistic images by producing smooth gradients and specular highlights.

**Pros:**

1. **Smooth shading**: Provides realistic, smooth surface appearance.
2. **Detail**: Enhances surface details and highlights.
3. **Interpolation**: Improves visual quality by interpolating normals.
4. **Flexibility**: Adaptable to various lighting conditions.

**Cons:**

1. **Performance**: Computationally intensive and may impact rendering speed.
2. **Complexity**: More complex than simpler shading techniques.
3. **Overhead**: Requires additional memory for storing normals.
4. **Artifacts**: May produce artifacts if normal interpolation is not handled well.

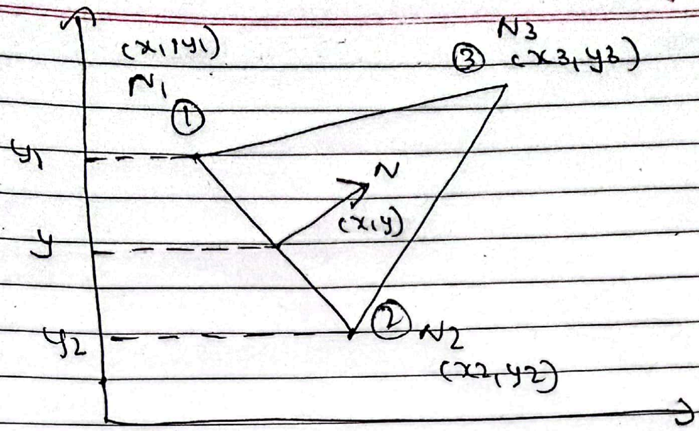
## Alogirthm

1. Determine the average unit normal vector at each polygon vertex

$$N_v = \frac{\sum_{i=1}^{n} N_i}{\left| \sum_{i=1}^{n} N_i \right|}$$

n → no of surfaces of polygon sharing that vertex

2. Linearly interpolate the vertex normals over the surface of polygon.




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

→ Applying the illumination model along each scan ~~we~~ to determine projected pixel intensities at surface points.

What is attenuation factor and Mach Band effect? Discuss the effect of Mach Band in Phong, Gourand and constant Shading. [2+2+3]

## Attenuation Factor

**Attenuation factor** generally refers to the reduction in intensity of a signal, light, or other forms of energy as it travels through a medium.

$$\text{Attenuation} = \frac{1}{a_0 + a_1 d + a_2 d^2}$$

where $d$ is the distance from the light source, and $a_0$, $a_1$, and $a_2$ are coefficients that determine how light intensity decreases.

## Mach Band Effect

The "match band effect" in computer graphics usually refers to the appearance of visible bands or color discontinuities on surfaces in a rendered image. This effect can be problematic in various shading techniques, leading to a less smooth or realistic appearance.

## Effects on Shading Techniques:

1. **Constant Shading (Flat Shading):**
    - This technique is the most susceptible to match band effects because the entire polygon is shaded with a single color. As a result, any discontinuity in color or lighting between adjacent polygons can be very noticeable, creating a banded appearance along polygon edges.
2. **Gouraud Shading:**
    - Gouraud shading can reduce the appearance of bands compared to constant shading because it interpolates colors across the surface. However, if the polygons are large or if there are significant differences in lighting or colors between adjacent polygons, banding can still occur, particularly at the edges where the interpolation might not be smooth.
3. **Phong Shading:**
    - Phong shading generally minimizes the match band effect compared to constant and Gouraud shading. The interpolation of normals and subsequent computation of pixel colors provides a smoother gradient of

colors and lighting, making the transition between different shaded areas more gradual and less prone to visible bands.

---

| Feature | Gouraud Shading | Phong Shading |
|---|---|---|
| **Method** | Per-vertex shading | Per-pixel shading |
| **Interpolation** | Interpolates color values across the surface | Interpolates normals across the surface |
| **Calculation Location** | Calculations are done at vertices and then interpolated across the face | Calculations are done for each pixel based on interpolated normals |
| **Quality** | May result in less accurate lighting effects due to interpolation of colors | Typically provides more accurate and smoother lighting effects |
| **Performance** | Generally faster because calculations are fewer (vertex-based) | Computationally more expensive due to per-pixel calculations |
| **Specular Highlights** | Less accurate, can miss sharp specular highlights | More accurate, can capture detailed specular highlights |
| **Usage** | Suitable for real-time applications where performance is crucial | Used in applications where visual accuracy is important and performance is less critical |
| | Introduced by henry gouraud | Introduced by Phong Bui Tuong |