

# Computer Graphics

## Shading

(Surface Rendering Methods)

Prepared by  
**Dr. Md. Manjur Ahmed**  
Faculty of Computer Systems and Software  
Engineering  
[manjur@ump.edu.my](mailto:manjur@ump.edu.my)

# Chapter Description



- **Aims**
  - Basic of Computer Graphics.
- **Expected Outcomes**
  - Understand the basic concept of computer graphics. (CO1: Knowledge)
  - Ability to use the computer graphics technology. (CO1: Knowledge)
- **References**
  - Computer Graphics by Zhigang Xiang, Schaum's Outlines.
  - Donald Hearn & M. Pauline Baker, Computer Graphics with OpenGL, 4th Edition, Boston : Addison Wesley, 2011.

# Shading

- **Shading Model:** How light interacts with materials
- **Surface-rendering:** Intensity calculations for all projected pixel positions using an illumination model
  - consider the various (or all) surfaces in a scene.

# Shading

We will discuss shading models:

- Flat
- Gouraud
- Phong

# Polygon Shading

- Consider for each polygon (flat shading)
- For each vertex (Gouraud Shading)
- For each pixel (Phong Shading)

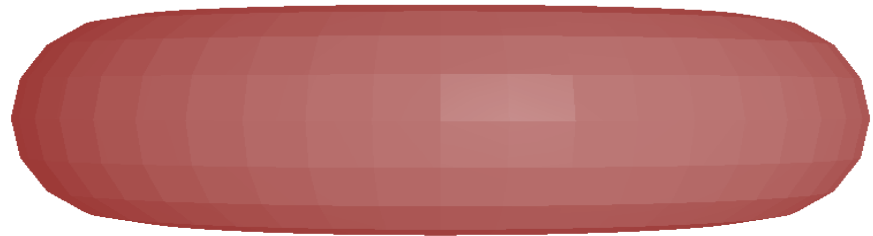
# Brute-Force Shading

- Calculate the surface normal at each visible point
- applying the desired illumination model at that visible point.
- Limitation: **highly expensive.**

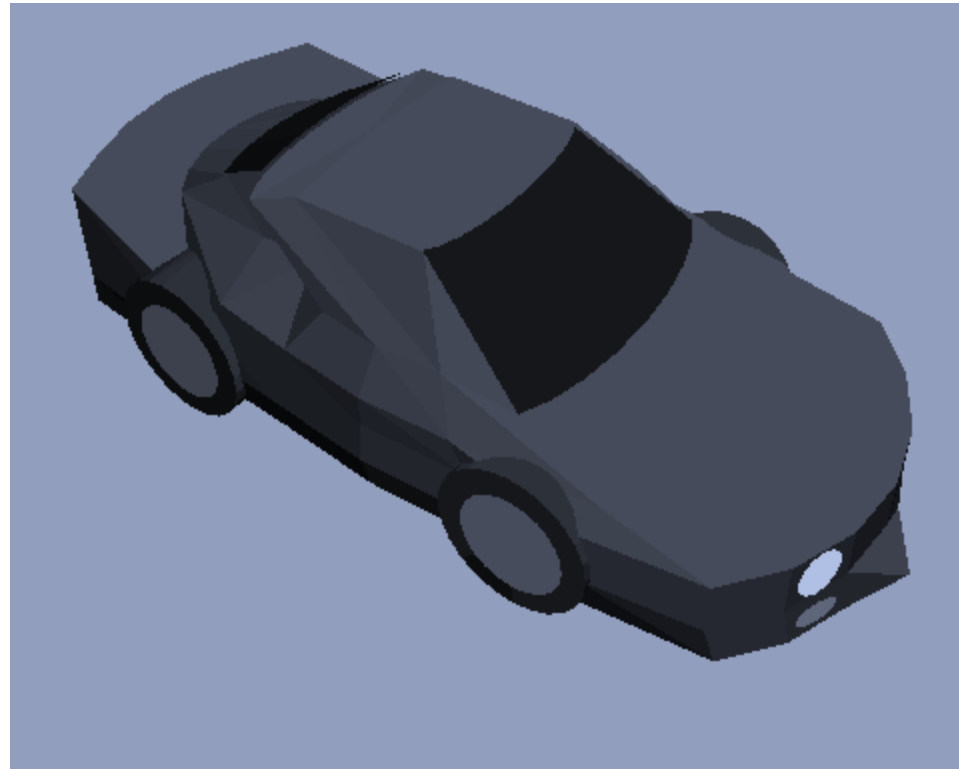


# Flat Surface Rendering

- Same color is assigned to entire polygon i.e. in all surface positions .... Consider one center point only
- Illumination at a center point on the surface (of a polygon) is calculated
- This illumination is considered for the entire surface
- Advantages: Surface rendering is extremely fast,
- Limitation: Can be unrealistic
  - Faceting occurs because of Mach Banding effect.

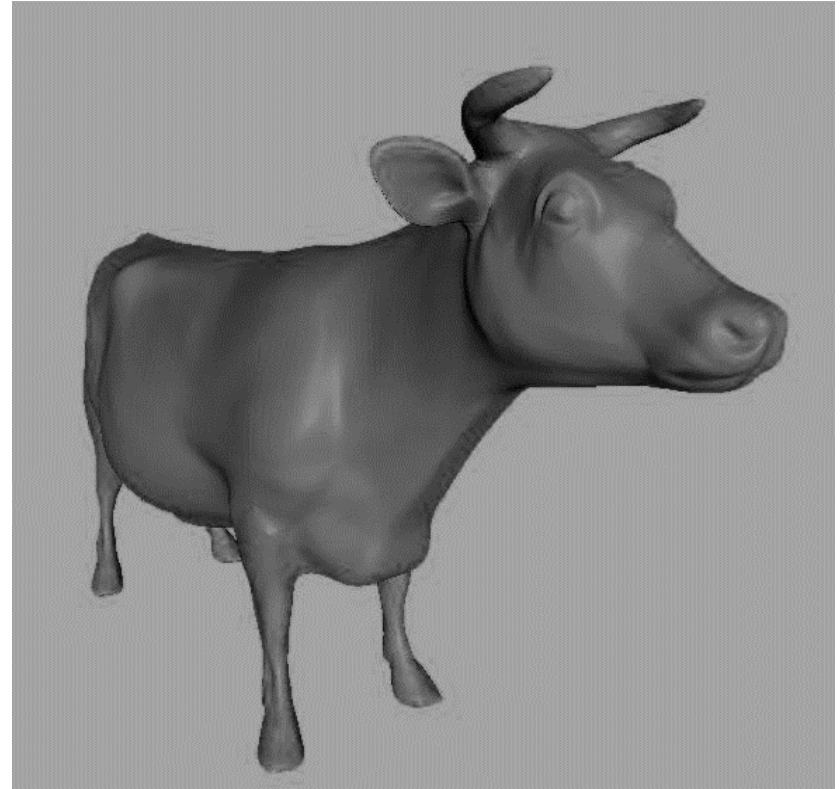
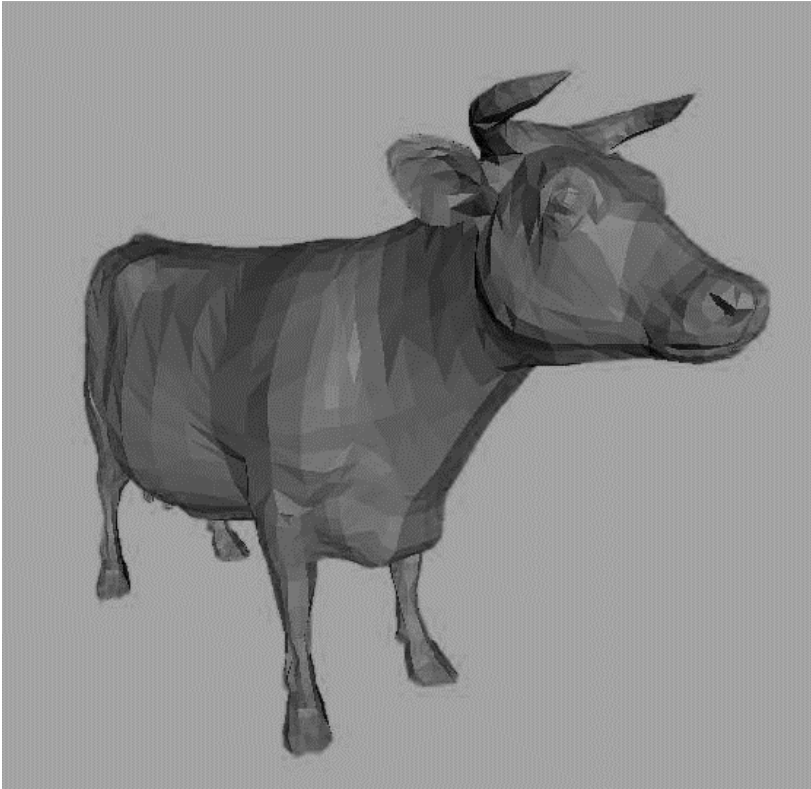


# Flat Shading





# Overcoming Limitations of Flat Shading



- Add lots and lots of polygons ---- make the surface more smooth to calculate the intensity of pixels.

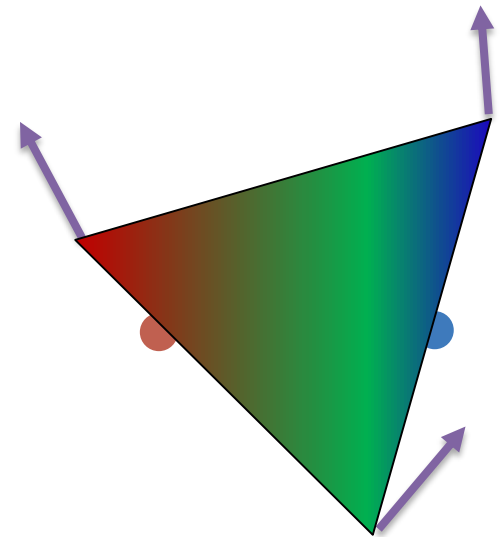
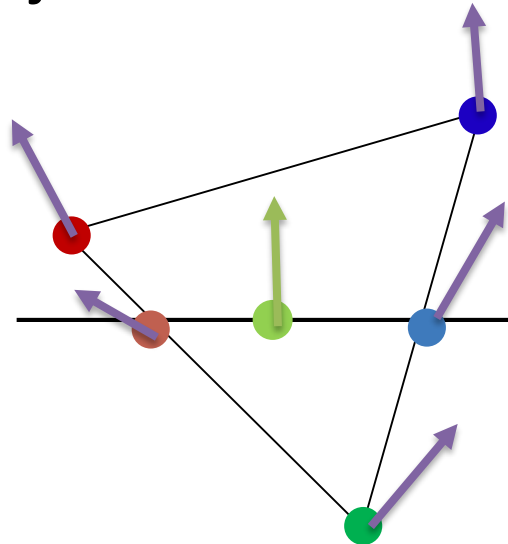
# Gouraud Surface Rendering

- **intensity-interpolation surface rendering**
- For each vertex, Intensity levels are calculated.  
After that, interpolated across the surface.

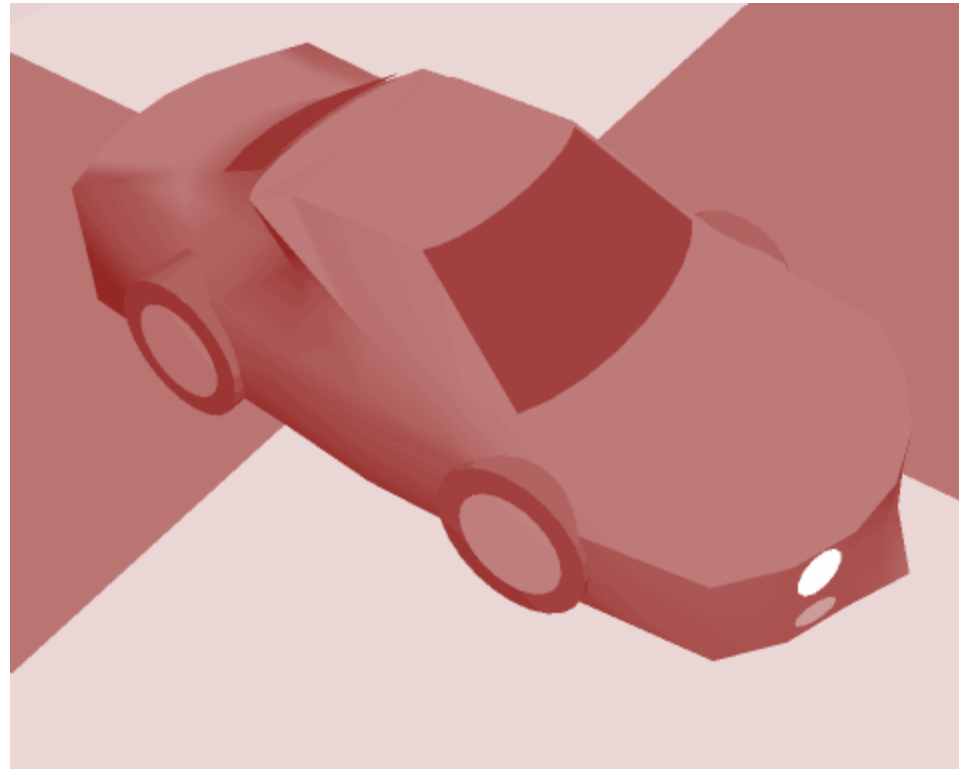


# Gouraud Surface Rendering (cont...)

- Algorithm: To render a polygon
  1. For **each vertex** of the polygon, determine the **average unit normal** vector
  2. For **each polygon vertex**, calculate the **light intensity** at these vertex based on an illumination model
  3. Then **Linearly interpolate** the vertex intensities over the projected area of the polygon

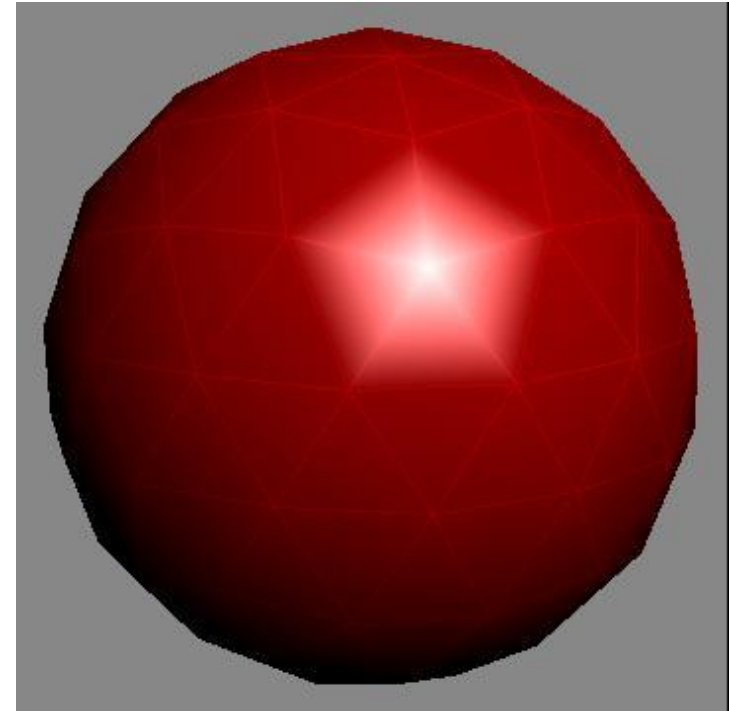


# Gouraud Shading Example



# Gouraud Shading

- Limitation:
  - Highlights on the interior of the polygon
- Linear interpolation still gives Mach banding



# Phong Surface Rendering

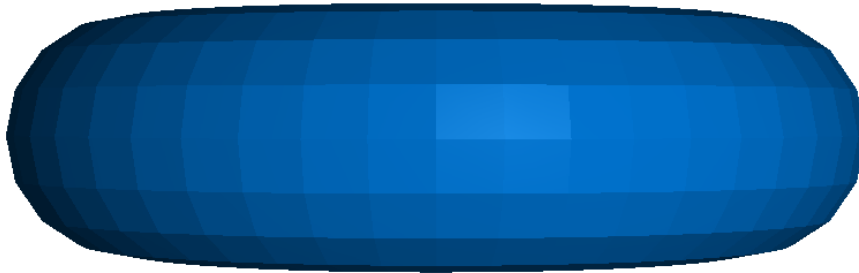
- Rendering a polygon , developed by Phong Bui Tuong
- Interpolates normal vectors instead of intensity values

# Phong Surface Rendering (cont...)

- Algorithm:

1. Determine the average unit normal vector at each vertex of the polygon
2. Linearly interpolate the vertex normals over the projected area of the polygon
3. Apply an illumination model at positions along scan lines to calculate pixel intensities using the interpolated normal vectors

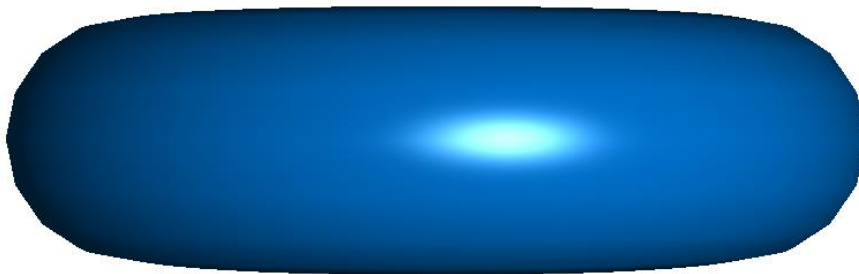
# Compare: Shading



*Flat*



*Gouraud*



*Phong*