Mach Bands

• Adjacent solid gray quads in increasing brightness

• Intensity on the retina

• Intensity perceived
Color Interpolation

- Vertex shader computes positions of vertices (in viewport coordinates)
- Vertex shader also computes colors of vertices (result of vertex lighting)
- Scan converter fills triangle with fragment positions
- Scan converter interpolates vertex colors across fragments
Gouraud Shading

- Vertex shader computes color based on vertex data
  - material color
  - vertex position
  - vertex normal
- Using the same “face” surface normal for all three triangle vertices yields faceted shading
- Using independent vertex normal yields smooth (Gouraud) shading
• Face normal of triangle
\[ n = (x_1 - x_0) \times (x_2 - x_0) \]
  – Needs to be normalized

• Per vertex normal
  – Sum of normals of triangle faces that share the vertex
  – Needs to be normalized
Transforming Normals

- First order neighborhood of a point on a surface described by a tangent plane
- Plane equation: \( Ax + By + Cz + D = 0 \)
- Plane normal: \( \begin{pmatrix} A \\ B \\ C \end{pmatrix} \)
Transforming Normals

- Plane equation: \( \mathbf{n} \cdot \mathbf{x} = 0 \)
- Let \( M \) be an affine transformation
- Transformed geometry \( \mathbf{x}' = M \mathbf{x} \)
- New normal \( \mathbf{n}' \) such that \( \mathbf{n}' \cdot \mathbf{x}' = 0 \)

\[
\mathbf{n}' \cdot M \mathbf{x} = 0
\]

\(
\mathbf{n} \cdot \mathbf{x} = 0
\)

\[
\mathbf{n}' = \mathbf{n} \mathbf{M}^{-1}
\]

(not really, but at least their parallel)

\[
\mathbf{n}' = \mathbf{n} \mathbf{M}^{-1}
\]

- \( \mathbf{n}' \) needs to be normalized

\[
\mathbf{n}' = (M^{-1})^T \mathbf{n}
\]

- \( (M^{-1})^T \) is called the Normal Matrix
Vertex Pipeline

Model Coords → Model Xform → World Coords → Viewing Xform → Viewing Coords → Perspective Distortion

Homogeneous Divide → Still Clip Coords. → Clipping → Clip Coords.

Window Coordinates → Window to Viewport → Viewport Coordinates