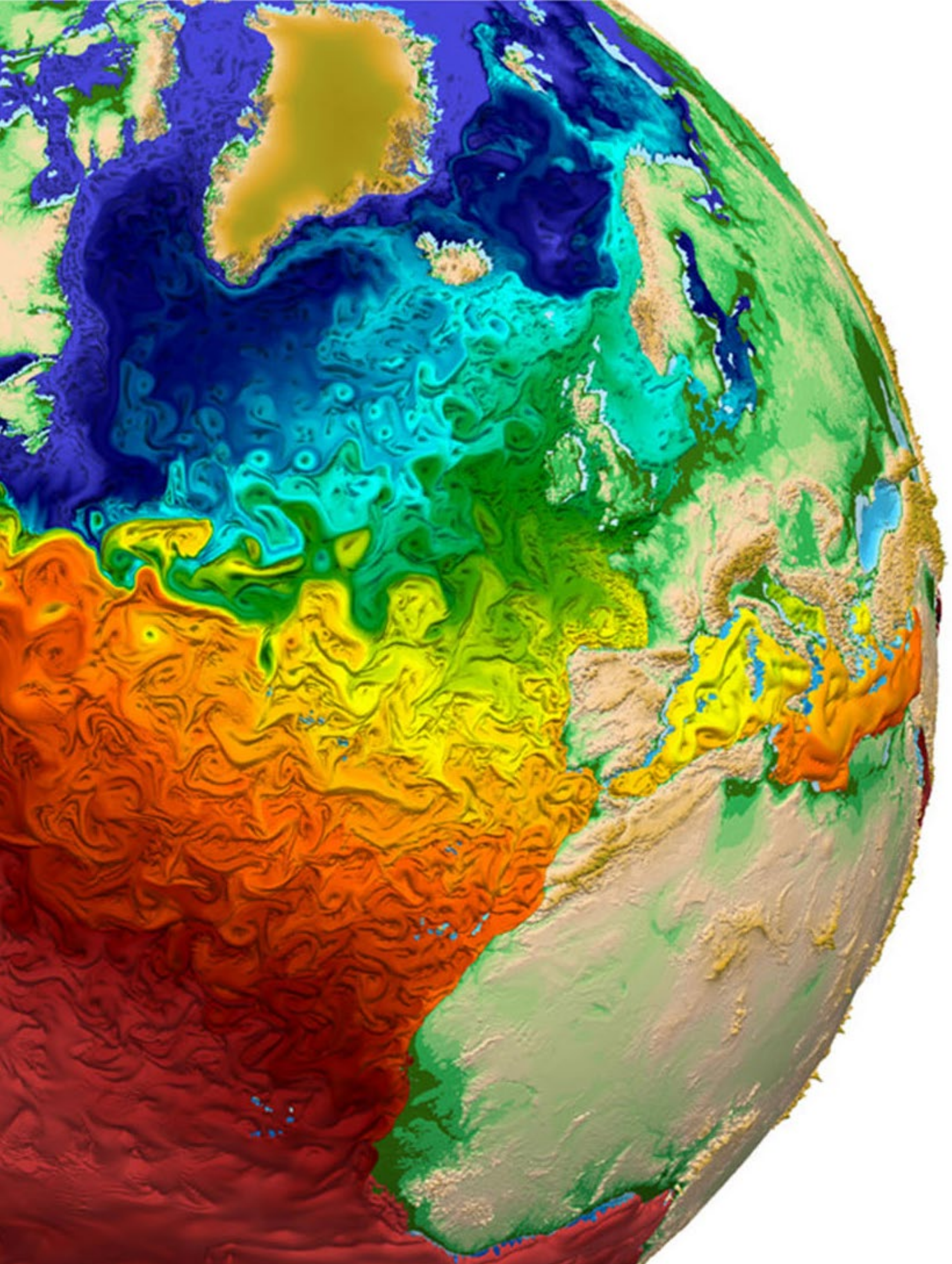
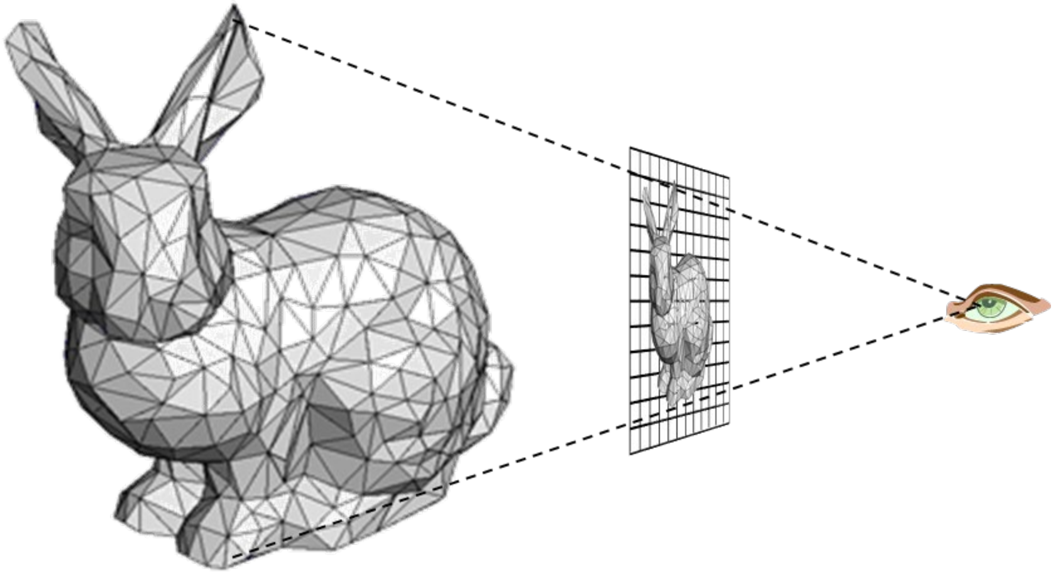


# 3D Graphics and Visualization

Professor Eric Shaffer

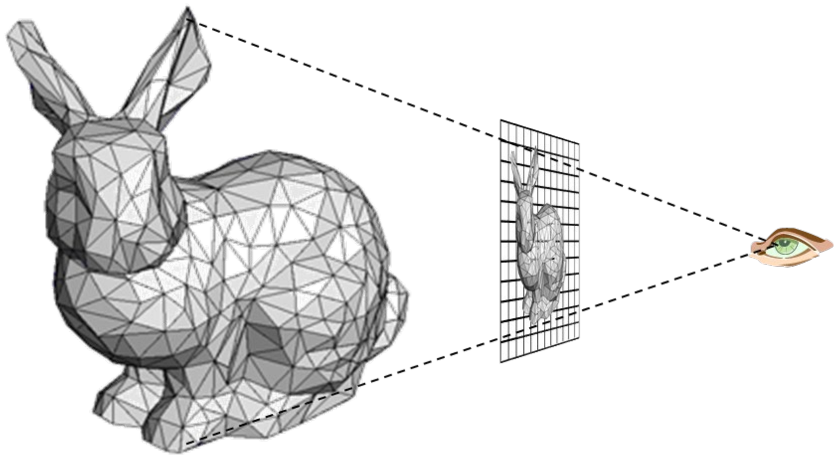
ILLINOIS

# Understanding Real-time Rendering



- 3D surfaces are modeled with triangle meshes
- Each triangle is projected to 2D
- Each triangle is rasterized into pixels
- Each pixel is shaded according to some model

# What Questions to Ask?

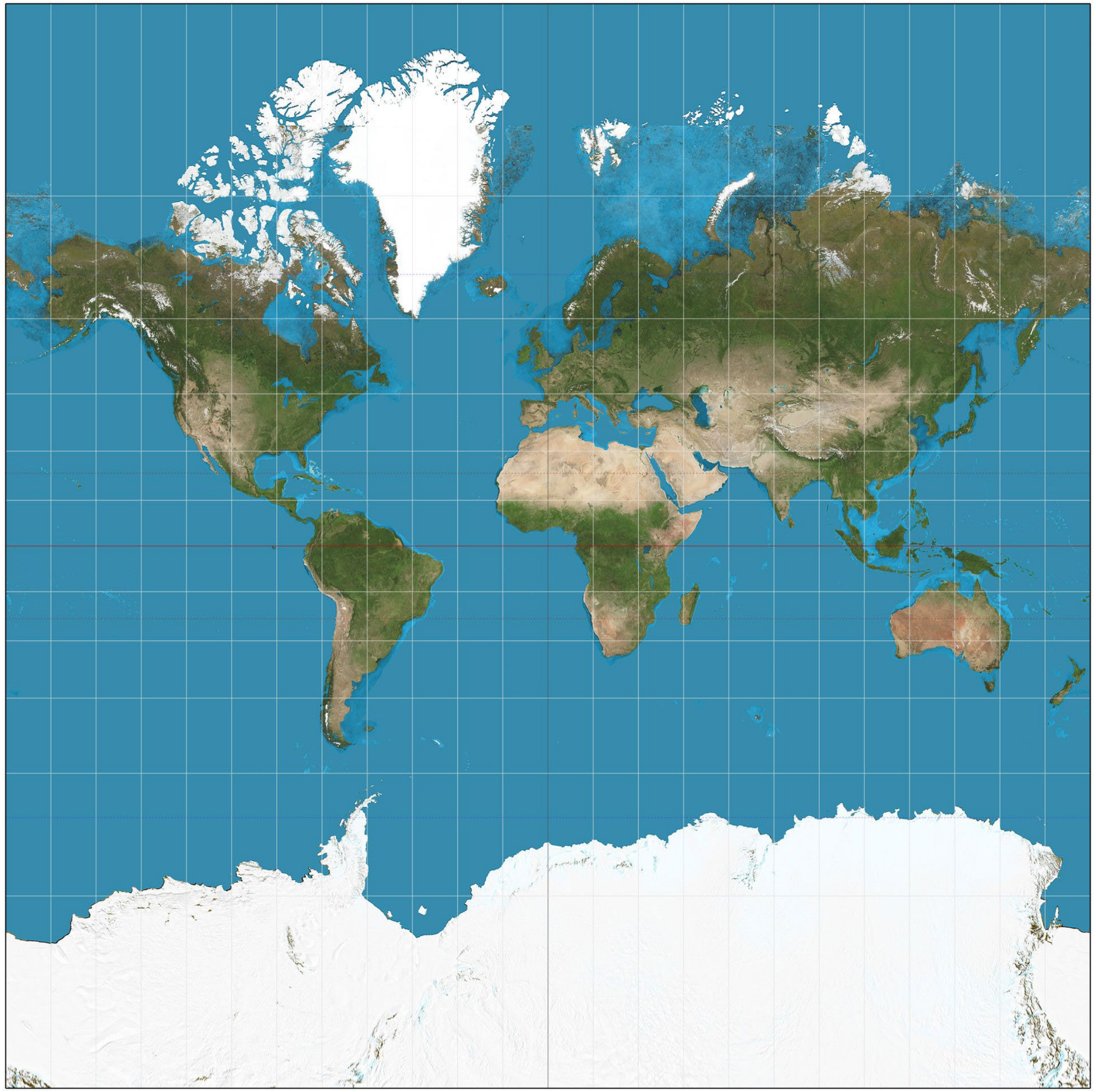


- What determines how performant an application is?
- What visual artifacts of the rendering process impact visualization?
  - Projection
  - Shading
  - Hidden surface removal

# Performance

- Complicated issue
- Number of triangles in surface model often the key factor
  - Fewer triangles = faster rendering

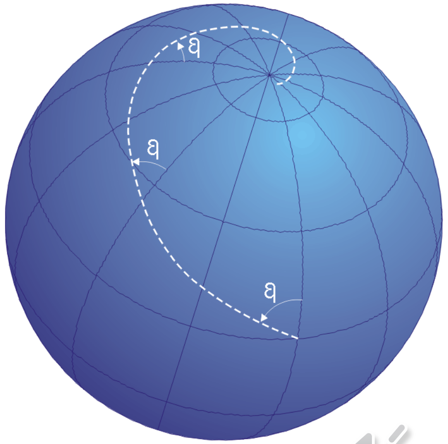
# Projections and Distortion



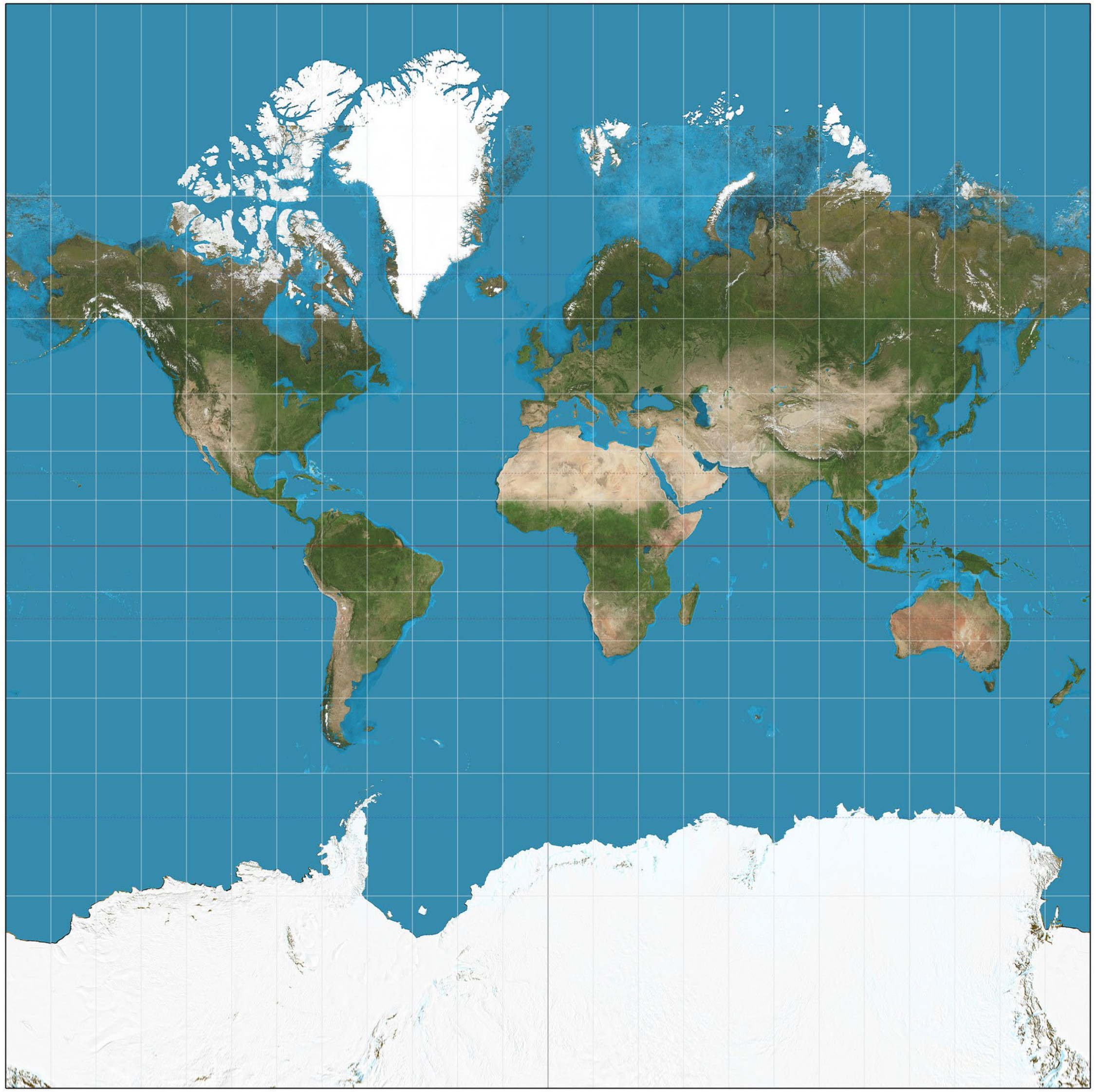
## Mercator Projection

- Developed by Gerardus Mercator in 1569
- Lines of constant bearing are straight lines on map.
- Revolutionary for naval navigation
  - Easily plot a course of constant bearing between 2 locations
  - Just maintain a heading using a compass
  - Avoids repeated course corrections to new heading

A constant bearing means maintaining a constant angle between the direction of navigation and true north. A ***rhumb line*** is a course of constant bearing…it will cross lines of longitude at the same angle and spiral into the pole



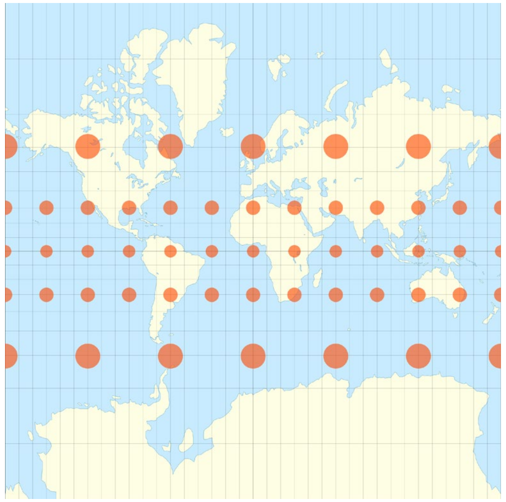
# Mercator Projection



Excellent for the task of navigation

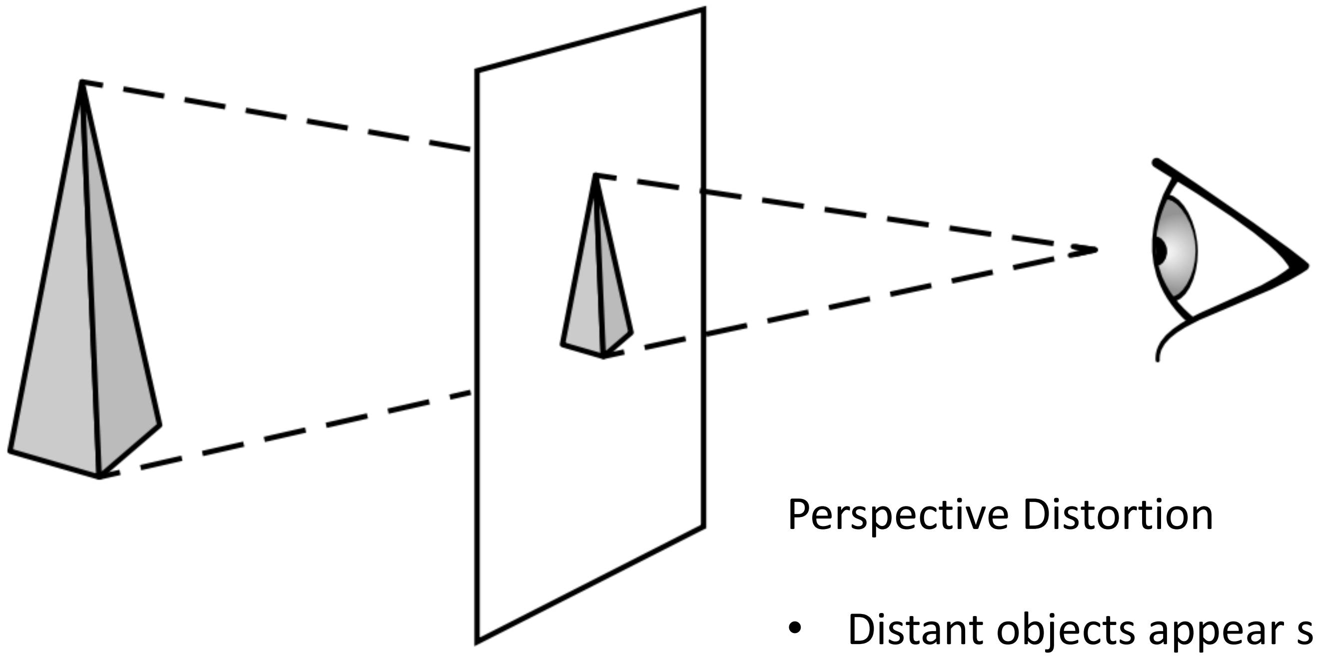
Poor for the task of comparing relative areas

- Africa is actually 14 times larger than Greenland
- Areas farther from the equator appear larger
- Circles on the indicatrix show relative distortion



ILLINOIS

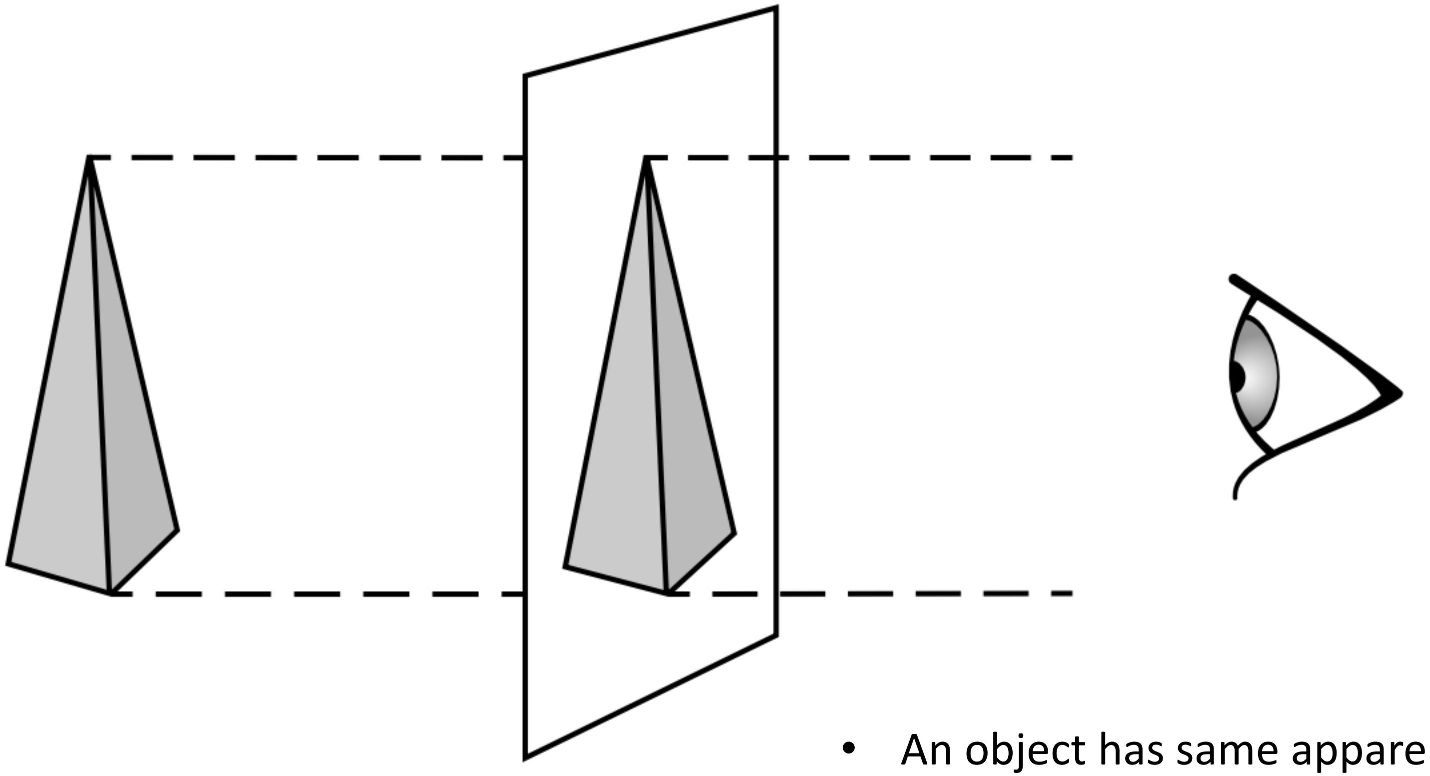
# Perspective Projection



Perspective Distortion

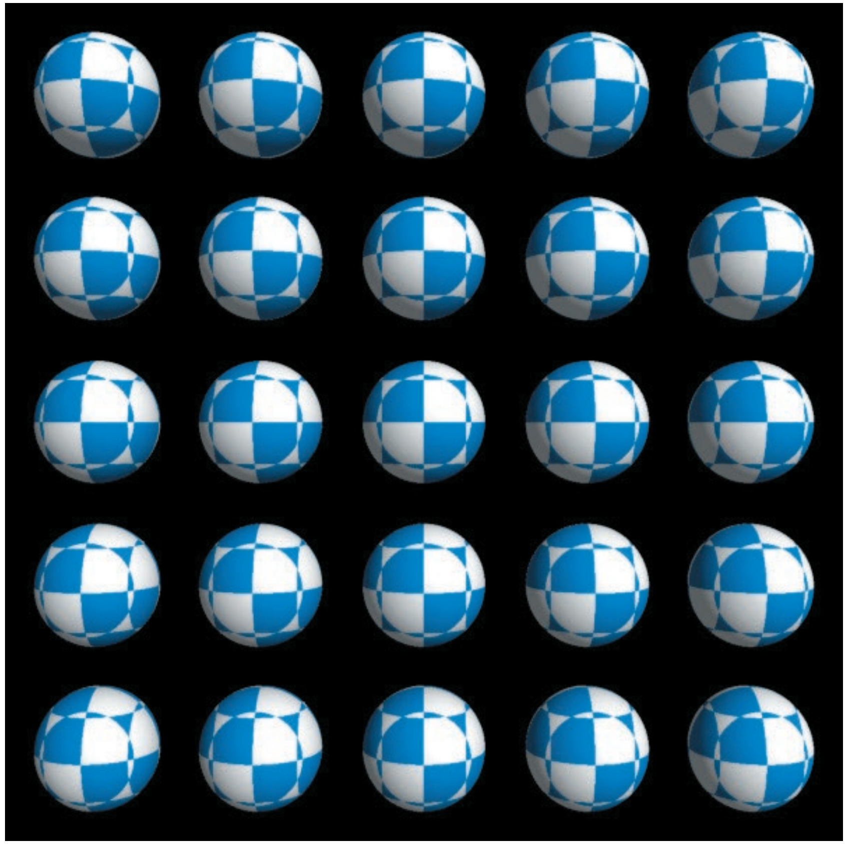
- Distant objects appear smaller than the same object close up
- Objects close to the image plane and away from CoP will look elongated
- CoP = center of projection

# Orthographic Projection

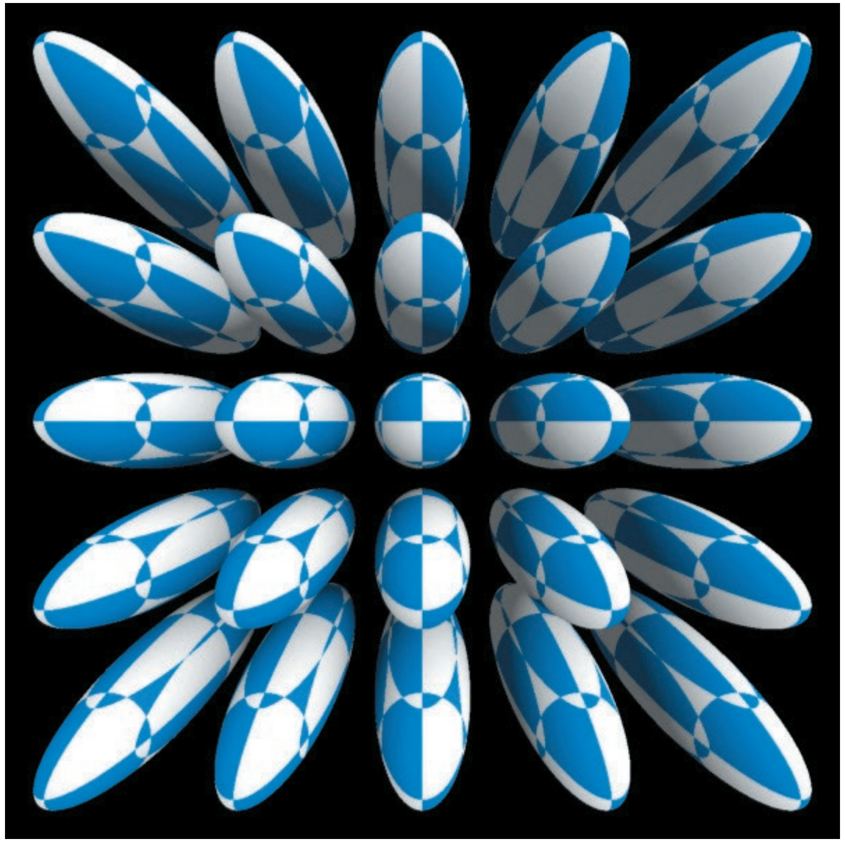


- An object has same apparent size regardless of distance from the eye.
- Foreshortening can still occur if object is angled away from eye

# Projections and Distortion



Orthographic Projection

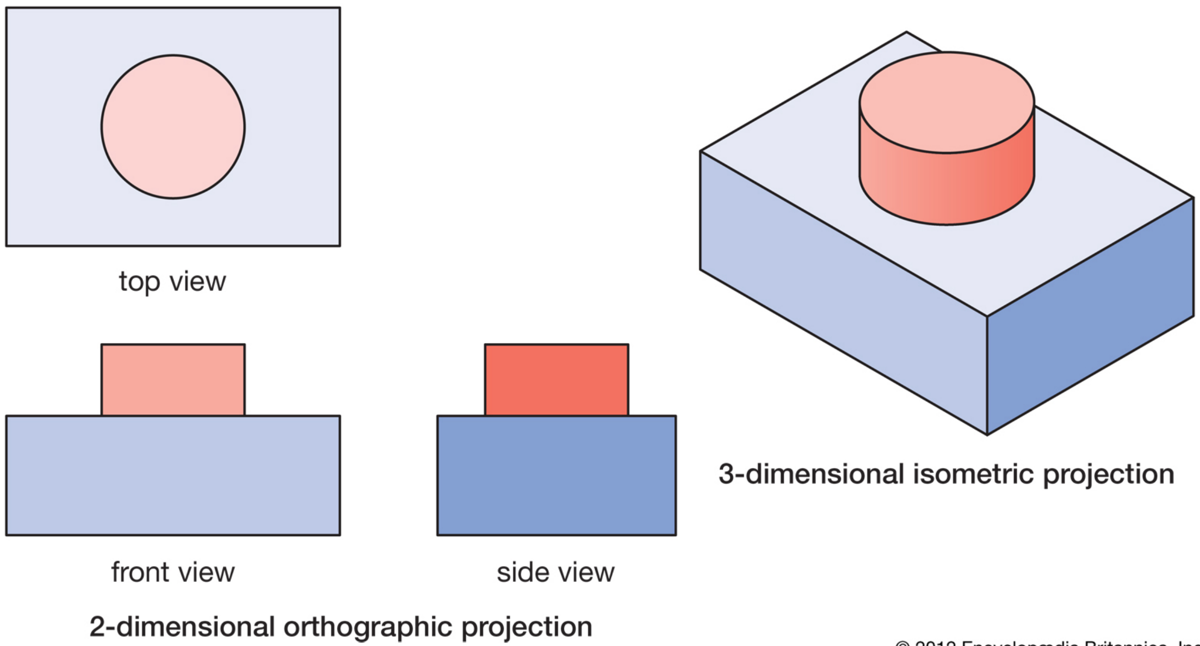


Perspective Projection

# Orthographic Projection for Engineering

If comparing lengths is important for application, consider orthographic

**Orthographic and isometric projections of an object**



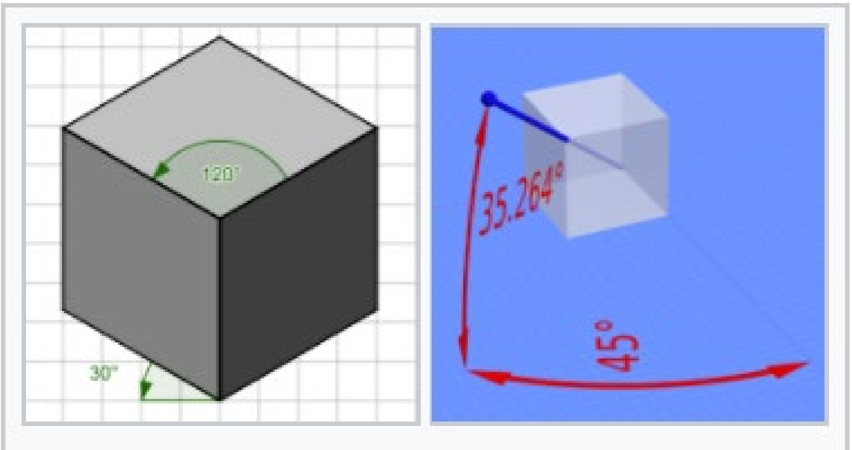
# Isometric Projection

Isometric projections are commonly used in technical drawings and some computer game graphics.

In an isometric projection the three axes appear 120° from each other and are equally foreshortened.

It can be achieved by:

1. rotating an object 45° around the vertical axis (Y)
2. rotating ~35.3° () through the horizontal axis (X)
3. projecting orthographically onto XY plane




There are actually 8 different orientations that could be used to achieve an isometric projection.

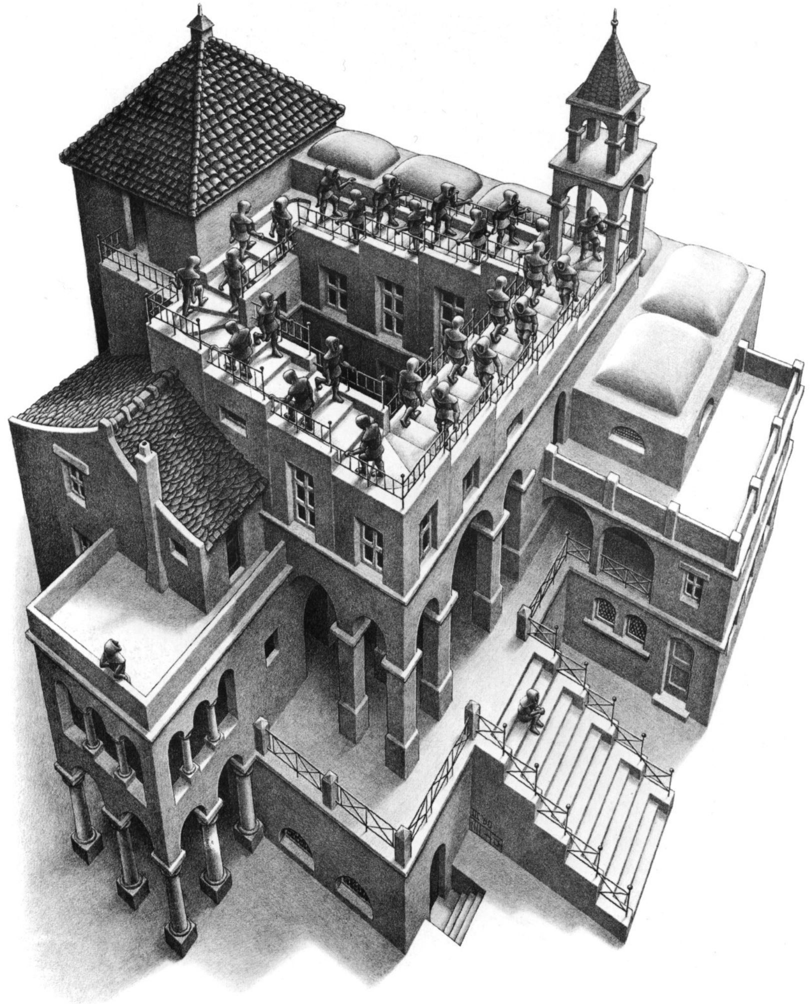
Gives the impression of 3D but relative lengths are preserved.

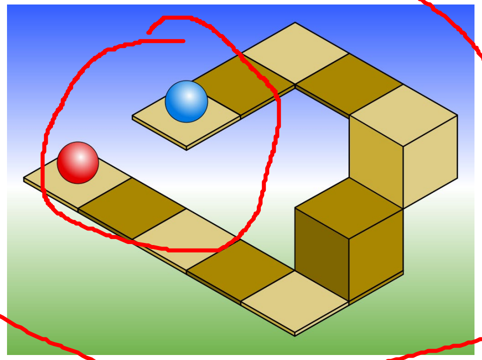
# Isometric Projection in Art

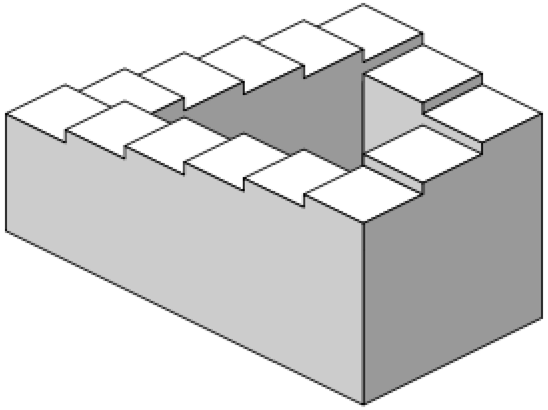
Used to generate 3D perspective in Chinese and Japanese art

The "Qing Court Version" of *Along the River During the Qingming Festival* (清院本清明上河圖)
an 18th-century remake by Chen Mei, Sun Hu, Jin Kun, Dai Hong, and Cheng Zhidao

# Isometric Projection in Art



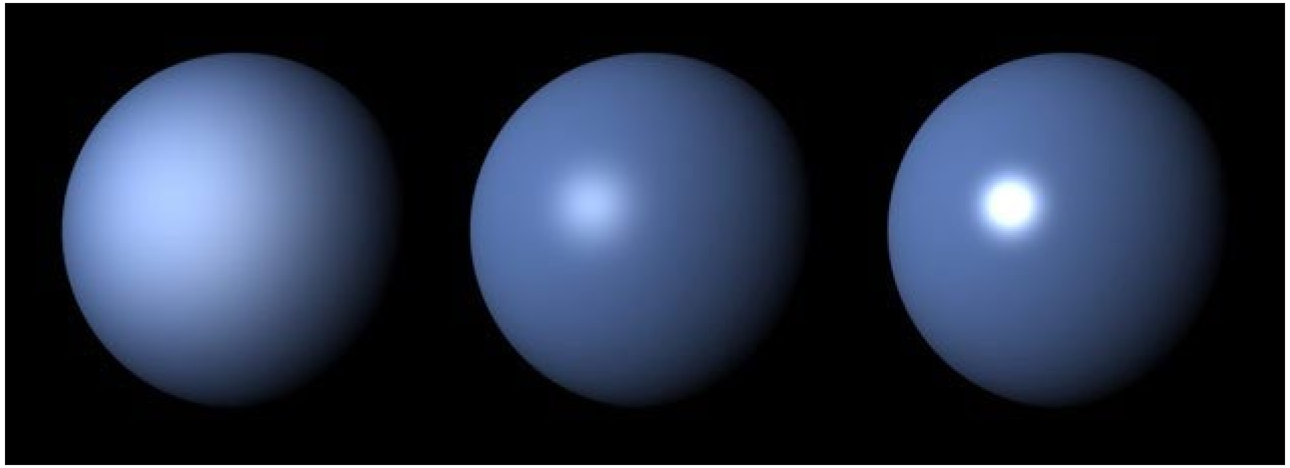




It can easily generate optical illusions

Since objects different depths project to the same size you cannot judge distance effectively

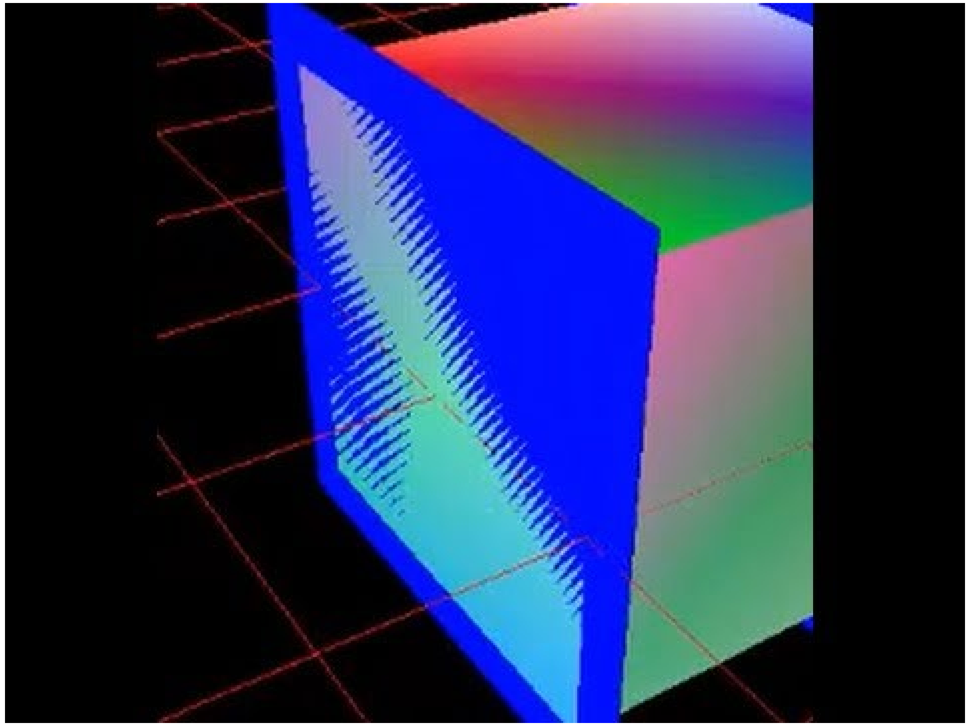
# Shading and Visualization



Be aware of the impact of shading on visualization

- Important 3D visual cue
  - Especially diffuse shading in Blinn-Phong
- Was non-white light used?
  - Can change the rendered color of the surface
- Too much ambient light can wash away details
  - Too little can leave structures too dark

# Hidden Surface Removal and Z-Fighting



Can occur when 2 surfaces are co-planar or close to co-planar

The “Z” refers to depth…distance from the camera

The rendering engine inconsistently determines which surface is closest

Why?

# Hidden Surface Removal and Z-Fighting

Each fragment has a z-value (positive depth from camera)

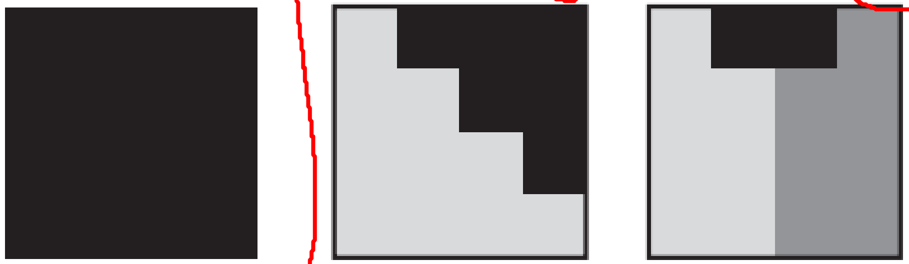
Hidden surface removal compares the z-values of fragments at same screen location

Fragment with least z-value is retained

| | | | |
|---|---|---|---|
| ∞ | ∞ | ∞ | ∞ |
| ∞ | ∞ | ∞ | ∞ |
| ∞ | ∞ | ∞ | ∞ |
| ∞ | ∞ | ∞ | ∞ |

| | | | |
|---|---|---|---|
| 1 | ∞ | ∞ | ∞ |
| 1 | 3 | ∞ | ∞ |
| 1 | 3 | 5 | ∞ |
| 1 | 3 | 5 | 7 |

| | | | |
|---|---|---|---|
| 1 | ∞ | ∞ | 1 |
| 1 | 3 | 3 | 1 |
| 1 | 3 | 3 | 1 |
| 1 | 3 | 3 | 1 |

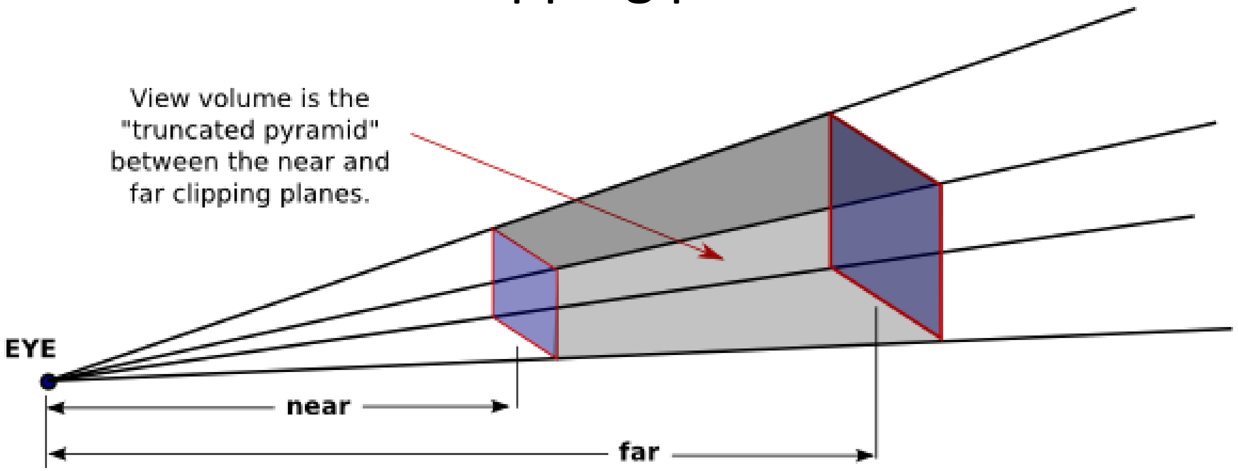


# Hidden Surface Removal and Z-Fighting

Depths from the camera lie in the range [***n***, ***f***]

***n*** is the positive distances to the near clipping plane

***f*** is the positive distance to the far clipping plane




To simplify things, assume depths are positive integers {0,1,…B-1}

Map ***n*** to 0 and ***f*** to B-1 →each integer in our range corresponds to a bucket of depth $\Delta z = \frac{f-n}{B}$

# Hidden Surface Removal and Z-Fighting

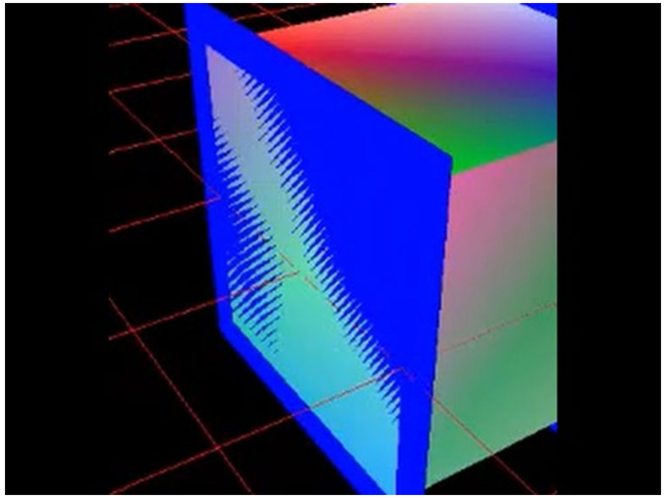
To simplify things, assume stored depths are positive integers {0,1,…B-1}

Map ***n*** to 0 and ***f*** to B-1 →each integer in our range corresponds to a bucket of depth $\Delta z = \frac{f-n}{B}$

If you render a scene in which surfaces have a separation of 1 m, if $\Delta z < 1$ then there should be no z-fighting

If the separation is less than the bucket depth…you can have z-fighting

- Cannot determine which surface is closest
- Rounding errors may switch which surface is chosen as closest in different parts of the scene



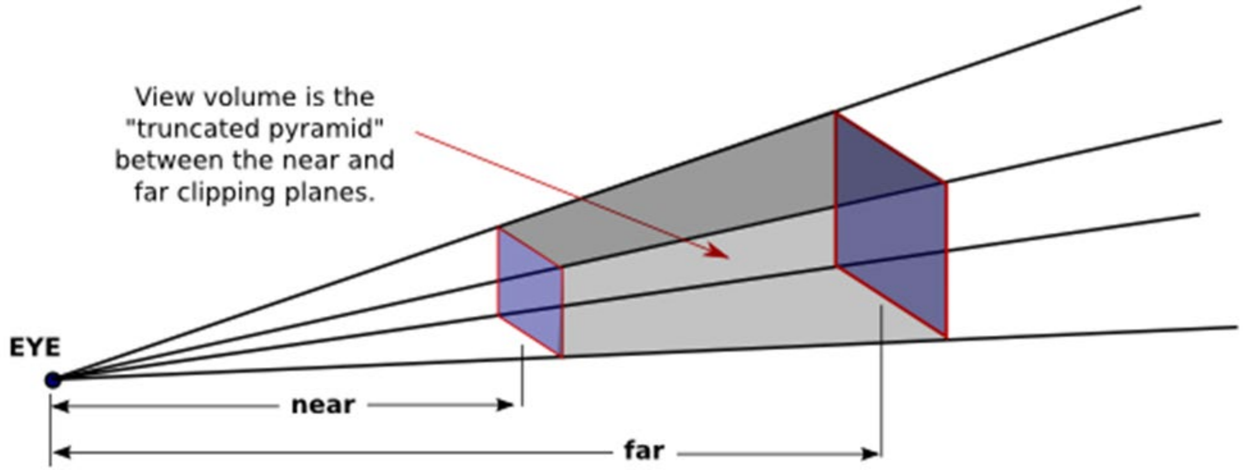
ILLINOIS

# Hidden Surface Removal and Z-Fighting

## Some fixes for z-fighting

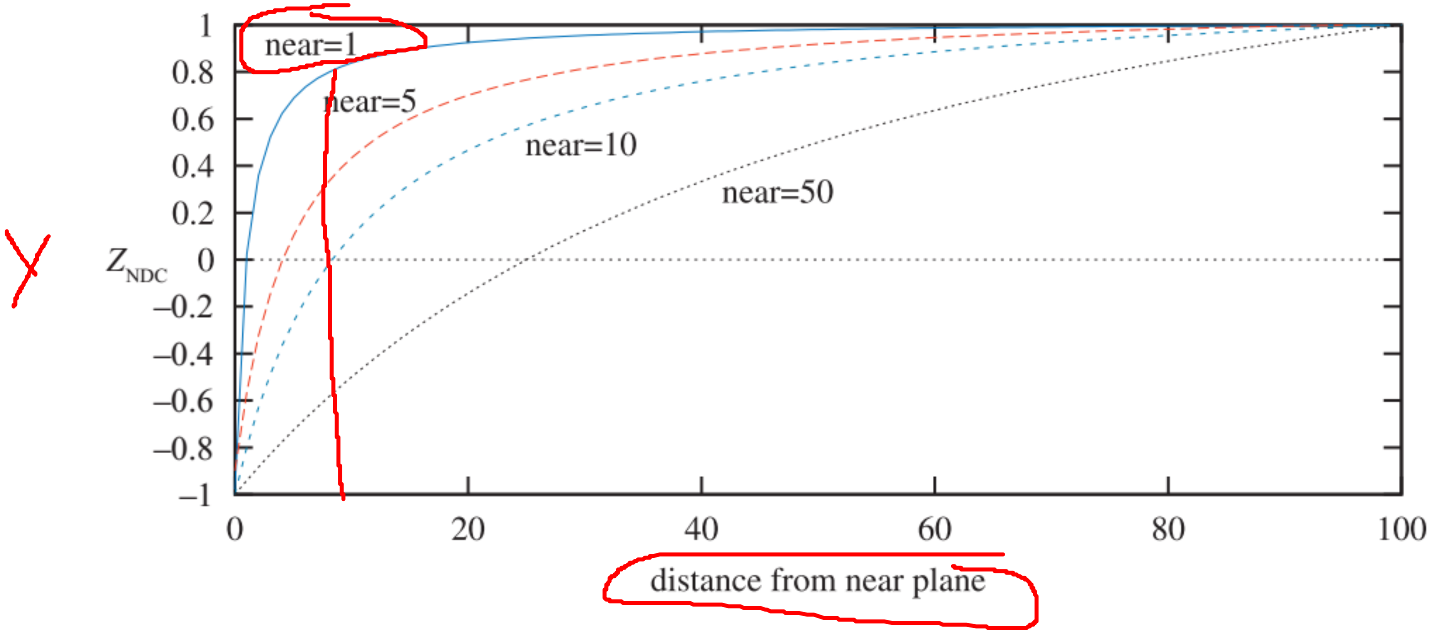
- Move the near and far planes closer together
- Move surfaces apart

$$\Delta z = \frac{f - n}{B}$$



# Hidden Surface Removal and Z-Fighting

In actuality, bucket sizes will vary by depth due to perspective projection




Here, ***f-n*** = 100 and each distance in the range is mapped into [-1,1]

- Cannot choose ***n***=0 as that results in an infinitely large bucket
- Larger bins at greater depths
  - Ability to do hidden surface removal degrades with distance