# Computer Graphics

## 1. Graphics Systems

I-Chen Lin

National Yang Ming Chiao Tung University

Textbook: E.Angel, D. Shreiner Interactive Computer Graphics, 6th Ed., Pearson

Ref: D.D. Hearn, M. P. Baker, W. Carithers, Computer Graphics with OpenGL, 4th Ed., Pearson

# Intended Learning Outcomes

- On completion of this chapter, a student will be able to:
  - Outline the preliminary concept of a **graphics system**.
  - List the key **breakthroughs** in the development of graphics systems.
  - Describe the concept of **3D projection**.
  - Recognize images produced by **perspective projection**.

# Computer Graphics

- Computer graphics deals with all aspects of creating images with a computer.
  - Hardware
  - Software
  - Applications

# Example

- Where did this image come from?
- What hardware/software did we need to produce it?

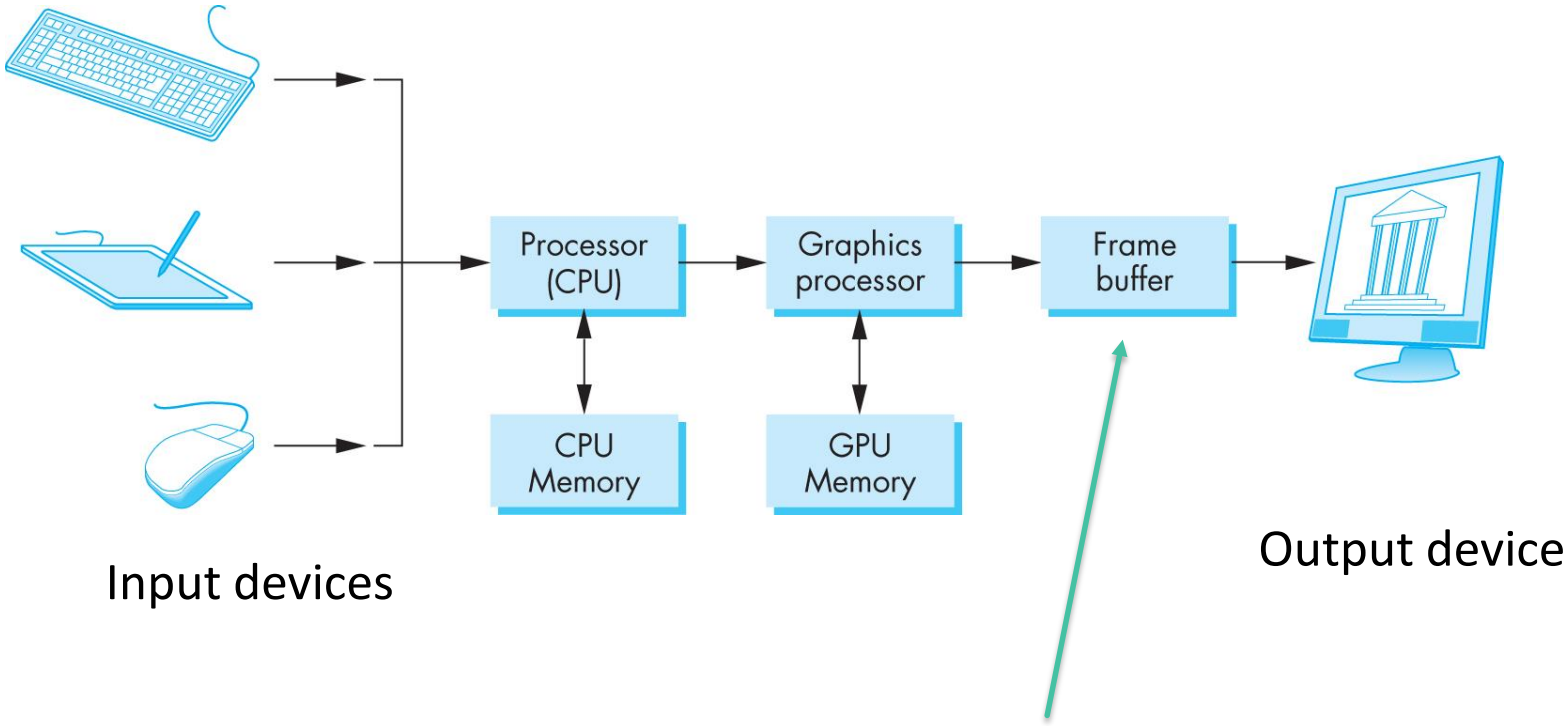
# Preliminary Answers

- **Application**: The spherical object is an artist's rendition for an animation to be shown in a domed environment (planetarium)

- **Software**: Maya for modeling and rendering but Maya is built on top of Graphics API.

- **Hardware**: PC with graphics cards for modeling and rendering

# Basic Graphics System

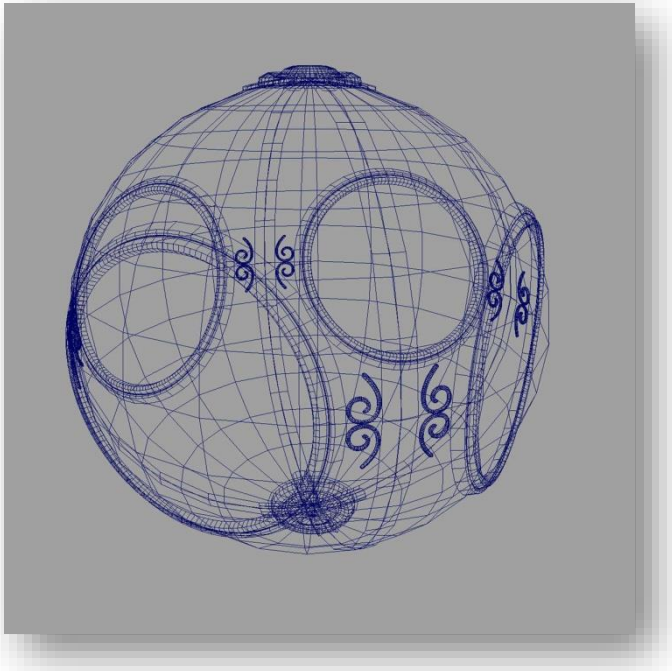


# Computer Graphics: 1950-1960

- Computer graphics goes back to the earliest days of computing
  - Strip charts
  - Pen plotters
  - Simple displays using A/D converters to go from computer to calligraphic CRT
- Cost of refresh for CRT is high
  - Computers slow, expensive, unreliable

# Computer Graphics: 1960-1970

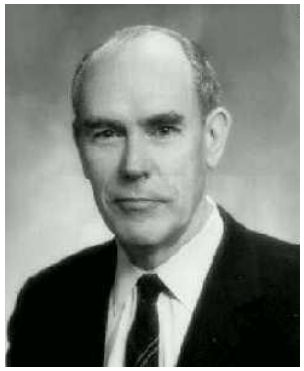
- *Wireframe* graphics
  - Draw only lines
- Sketchpad
- Display Processors
- Storage tube



wireframe representation of the spherical object

# Sketchpad

- **Ivan Sutherland**'s PhD thesis at MIT
  - Recognized the potential of man-machine interaction.
  - Sutherland also created many of the now common algorithms for computer graphics



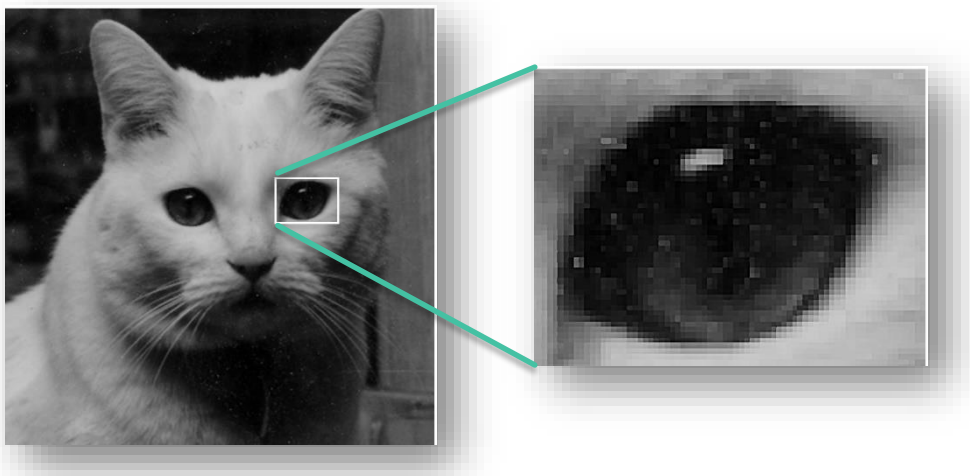
Ivan Sutherland,

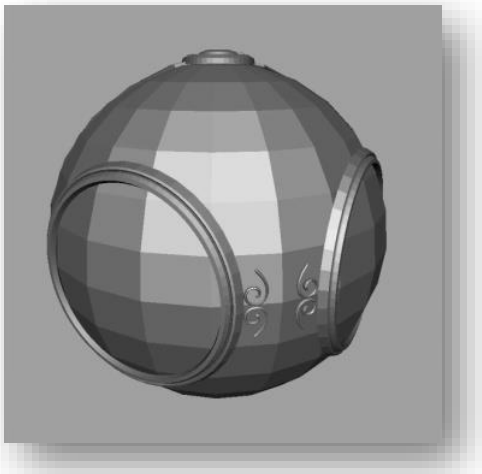
Turing Award winner, 1988

The console of the TX-2, Sketchpad Project

# Computer Graphics: 1970-1980

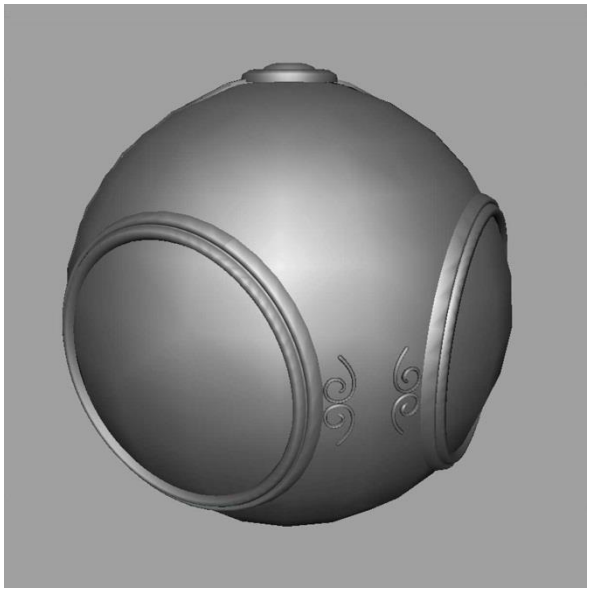
- Raster Graphics
  - Image produced as an array (the *raster*) of picture elements (*pixels*) in the *frame buffer*
  - Allows us to go from lines and wire frame images to filled polygons





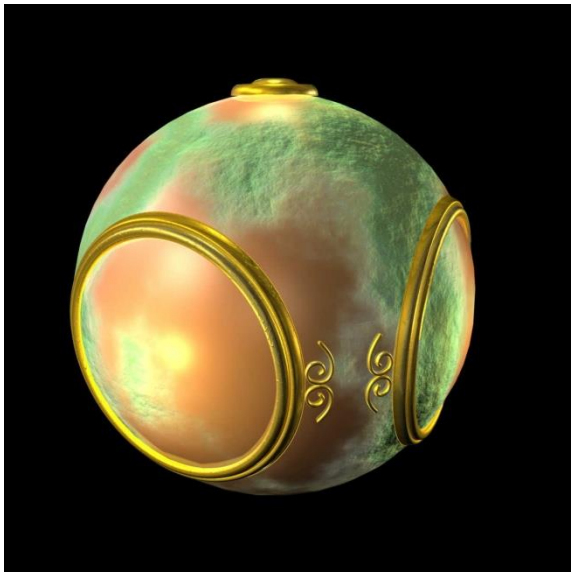
# Computer Graphics: 1980-1990

- Realism comes to computer graphics



smooth shading

environment mapping



bump mapping

Note: Gouraud shading published in 1971

# Computer Graphics: 1980-1990

- Special purpose hardware
  - Silicon Graphics geometry engine
    - VLSI implementation of graphics pipeline
- Industry-based standards
  - PHIGS
    - Programmer's Hierarchical Interactive Graphics System
  - RenderMan
- Networked graphics: X Window System
- Human-Computer Interface (HCI)

# Computer Graphics: 1990-2000

- OpenGL API
- Completely computer-generated feature-length movies (e.g. Toy Story) are successful.
- New hardware capabilities
  - Texture mapping
  - Blending
  - Stencil buffers, …

# Computer Graphics: 2000-

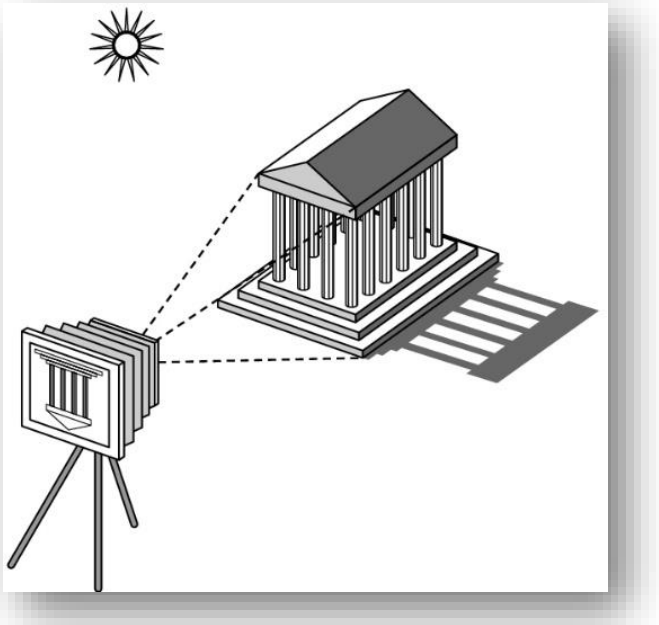
- Photorealism
- Graphics cards for PCs dominate the market
  - Nvidia, ATI (-> AMD)
  - GPU (Graphics processing unit)
- Game boxes and game players determine directions of the market
- Computer graphics routine in movie industry: Maya, Lightwave.
- *Programmable pipelines*

# Image Formation

- Fundamental imaging notions
- Physical basis for image formation
  - Light
  - Color
  - Perception
- Synthetic camera model
- Other models

# Elements of Image Formation

- Objects
- Viewer
- Light source(s)

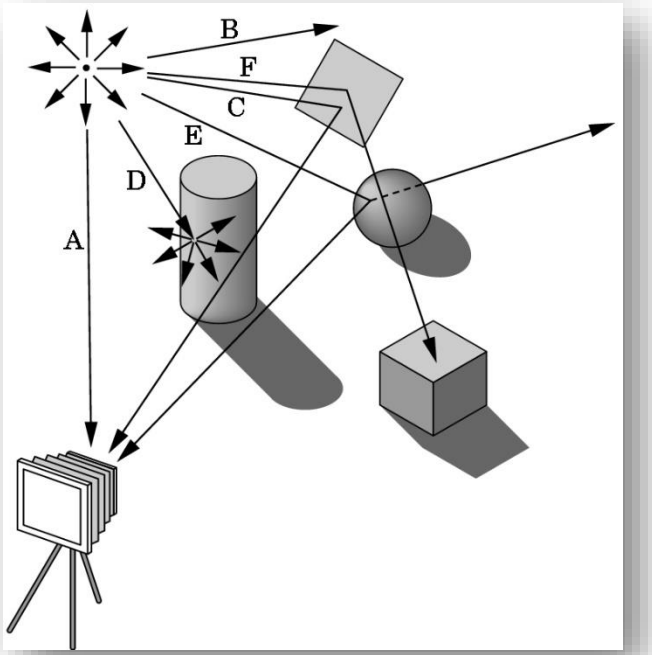


- Attributes that govern how light interacts with the materials in the scene
- Note the independence of the objects, the viewer, and the light source(s)

# Ray Tracing and Geometric Optics

- One way to form an image is to follow rays of light from a point source finding which rays enter the lens of the camera.

- However, each ray of light may have multiple interactions with objects before being absorbed or going to infinity.



# Light

- *Light* is the part of the electromagnetic spectrum that causes a reaction in our visual systems

- Generally these are wavelengths in the range of about 350-750 nm (nanometers)

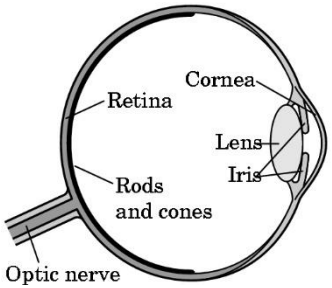
# Luminance and Color Images

- Luminance Image
  - Monochromatic
  - Values are gray levels
  - Analogous to working with black and white film or television

- Color Image
  - Has perceptional attributes of hue, saturation, and lightness
  - Do we have to match every frequency in visible spectrum?

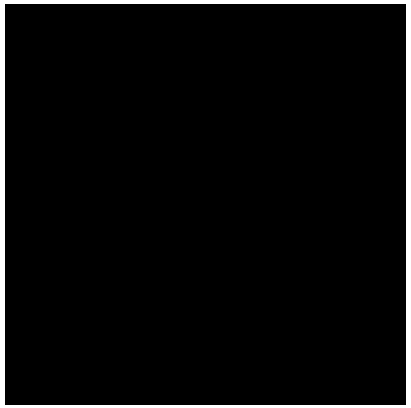
# Three-Color Theory

- Human visual system has two types of sensors
  - Rods: monochromatic, night vision
  - Cones
    - Color sensitive
    - Three types of cones
    - Only three values (the *tristimulus* values) are sent to the brain



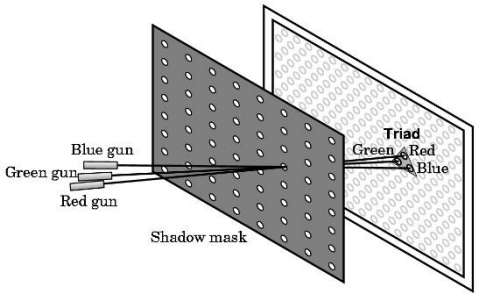

- Need only match these three values
  - Need only three *primary* colors

# Additive and Subtractive Color



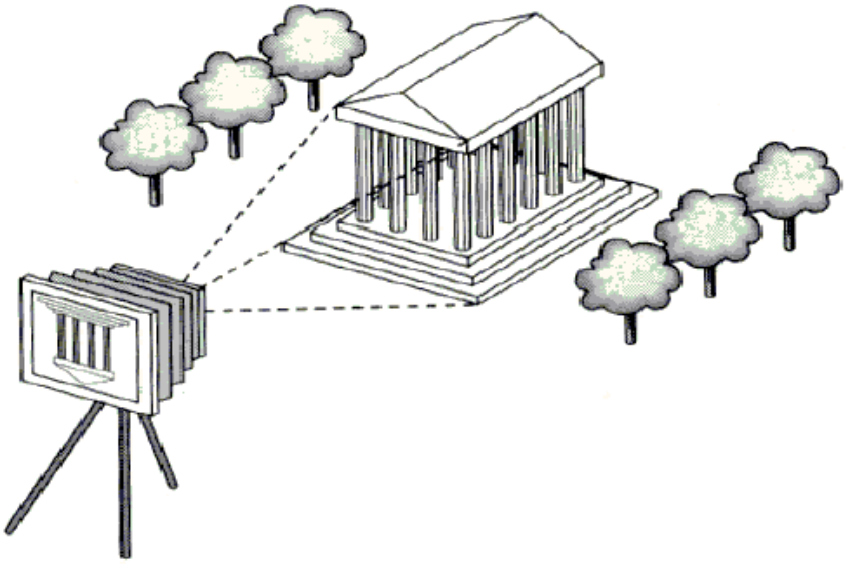
- Additive color
  - Form a color by adding amounts of three primaries
    - CRTs, LCD, projection systems, positive film
  - Primaries: Red (R), Green (G), Blue (B)

Shadow Mask CRT

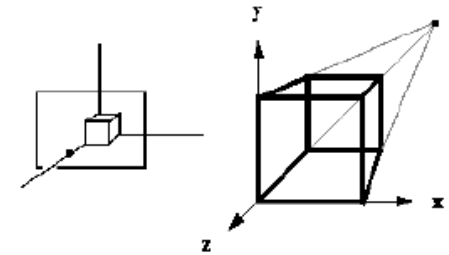



  - Subtractive color
  - Form a color by filtering white light with:
    - Cyan (C), Magenta (M), and Yellow (Y) filters
    - Printing, Negative film

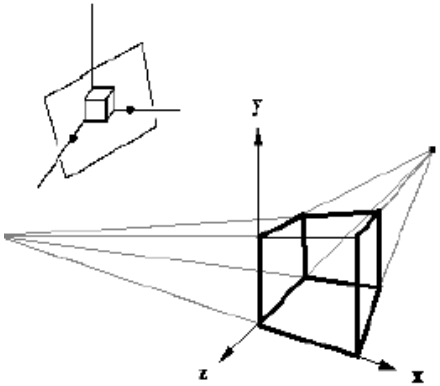
# Basic 3D Graphics



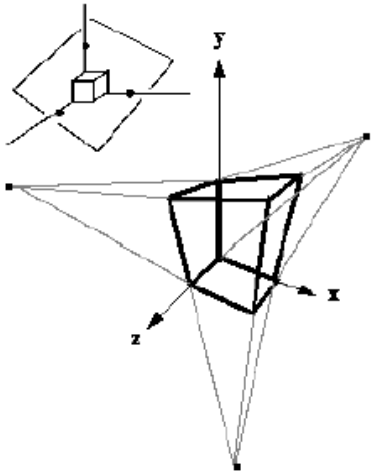
# Vanishing points



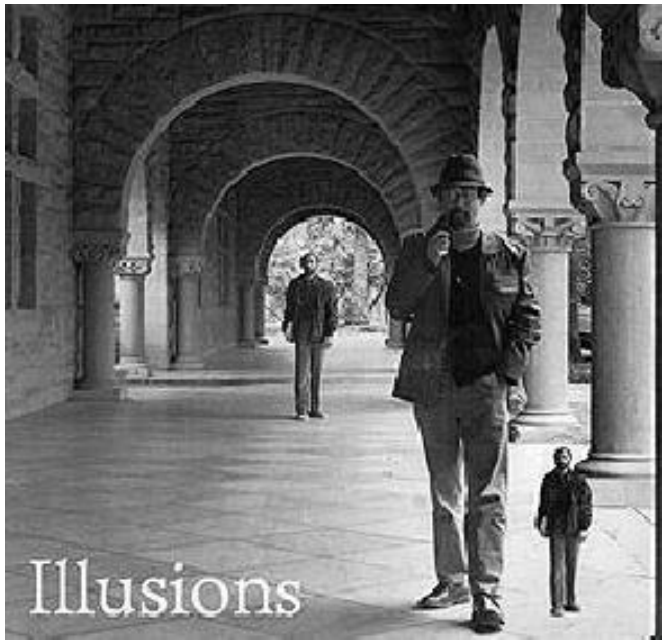

*One Point Perspective*
(z-axis vanishing point)



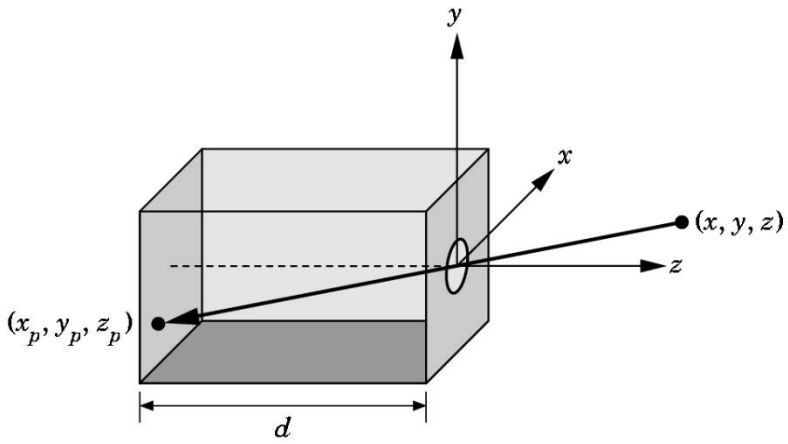

*Two Point Perspective*
z, and x-axis vanishing points




*Three Point Perspective*
(z, x, and y-axis
vanishing points)




# Pinhole Camera



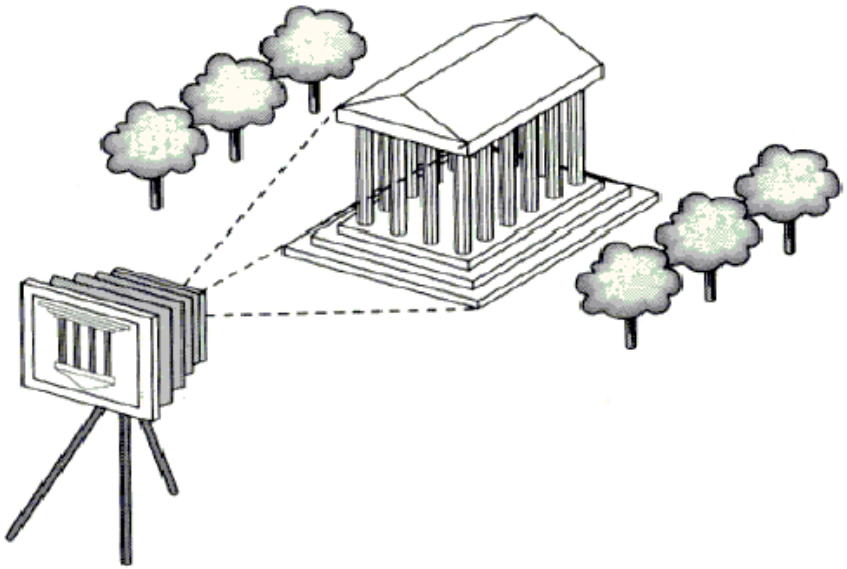

Use trigonometry to find projection of point at (x,y,z)

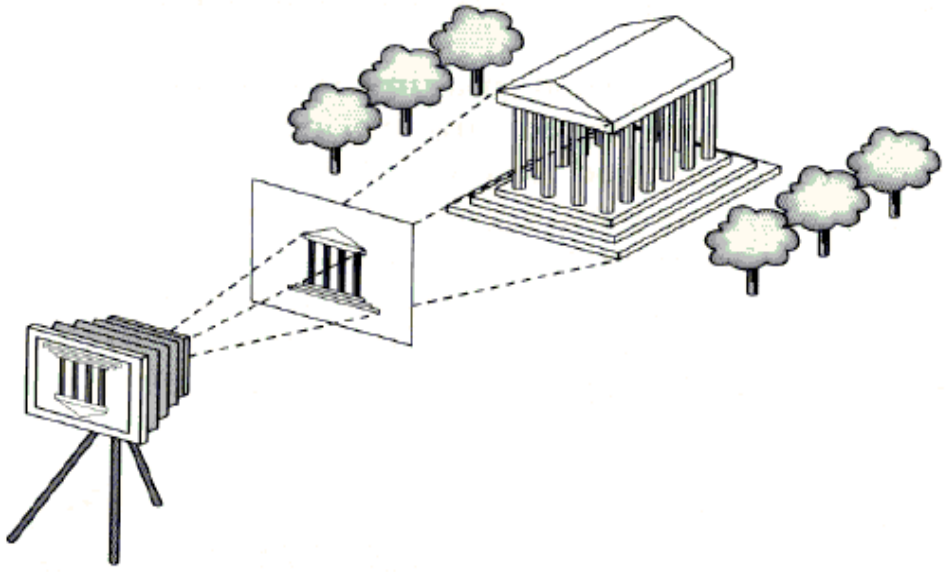
$$x_p = -x/(z/d) \qquad y_p = -y/(z/d) \qquad z_p = d$$

These are equations of simple perspective

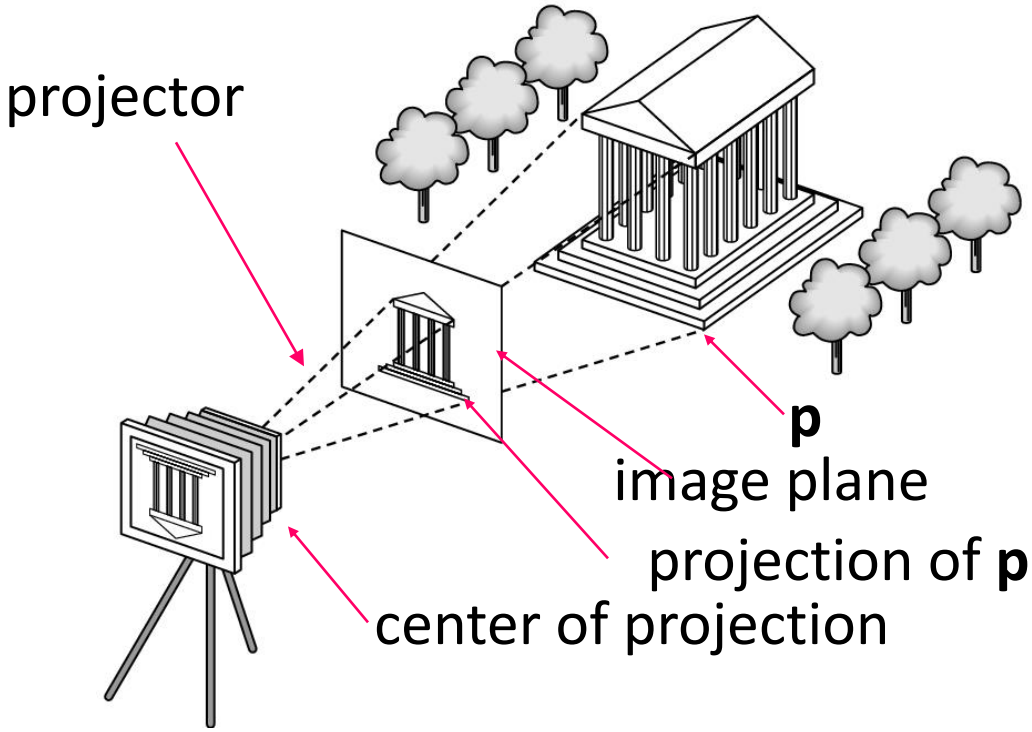
# Perspective projection

- Taking  photographing as an example.



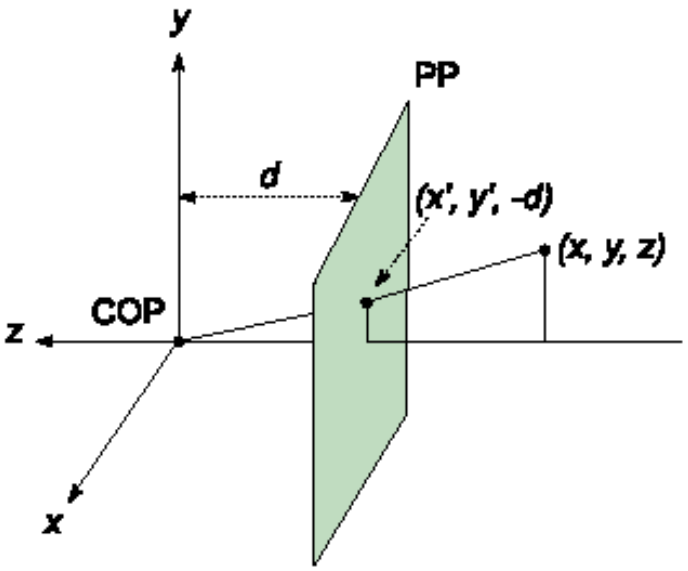


# Synthetic Camera Model

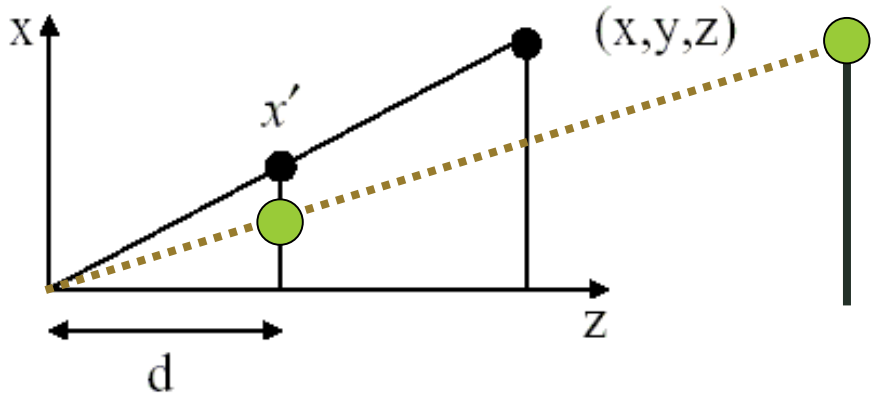


# Perspective projection (cont.)

- Projection

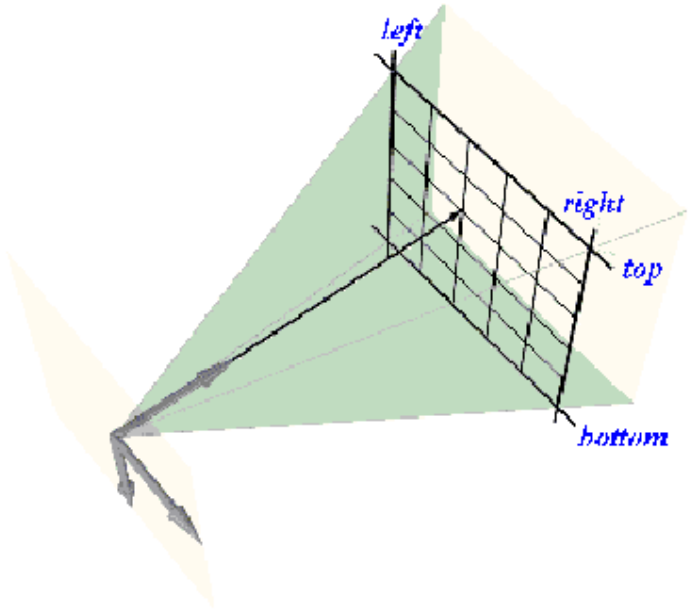


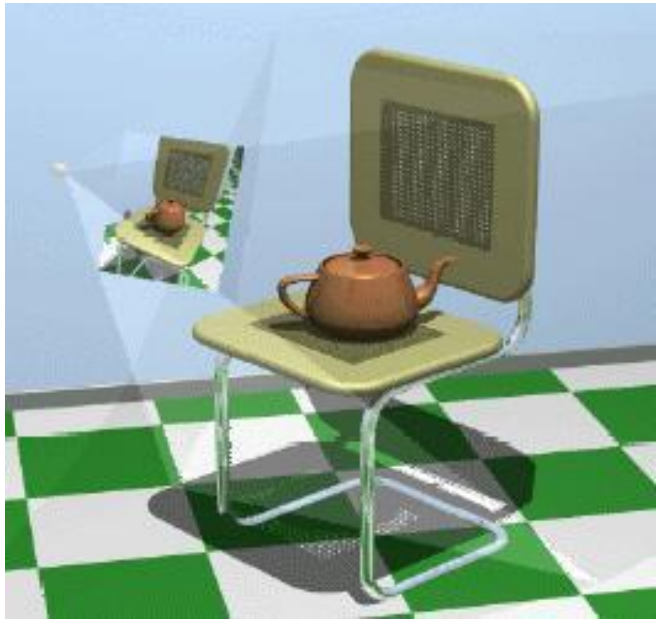

Using similar triangles gives:



# Perspective projection (cont.)

- Let pupils as the pinhole and a screen as the film.



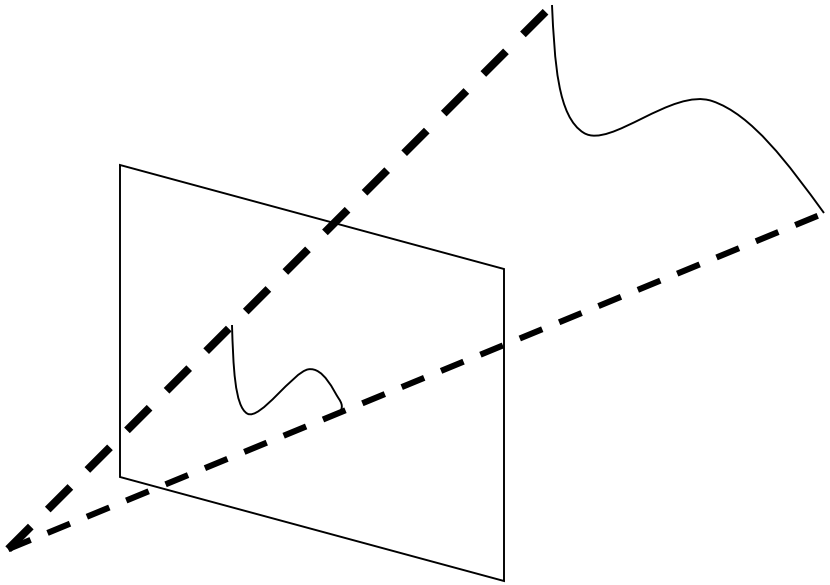



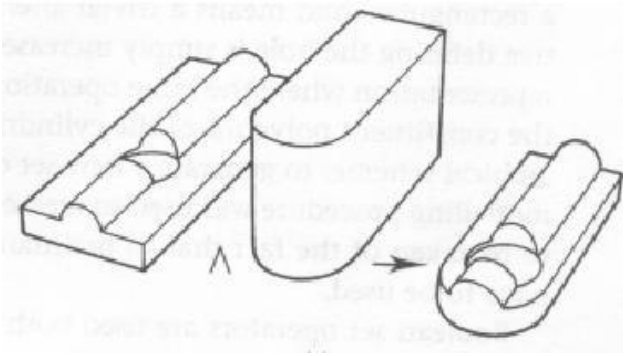
# Generating perspective views

- From the continuous world to a digital one.
- Representing by surfaces?

# Represented by primitives

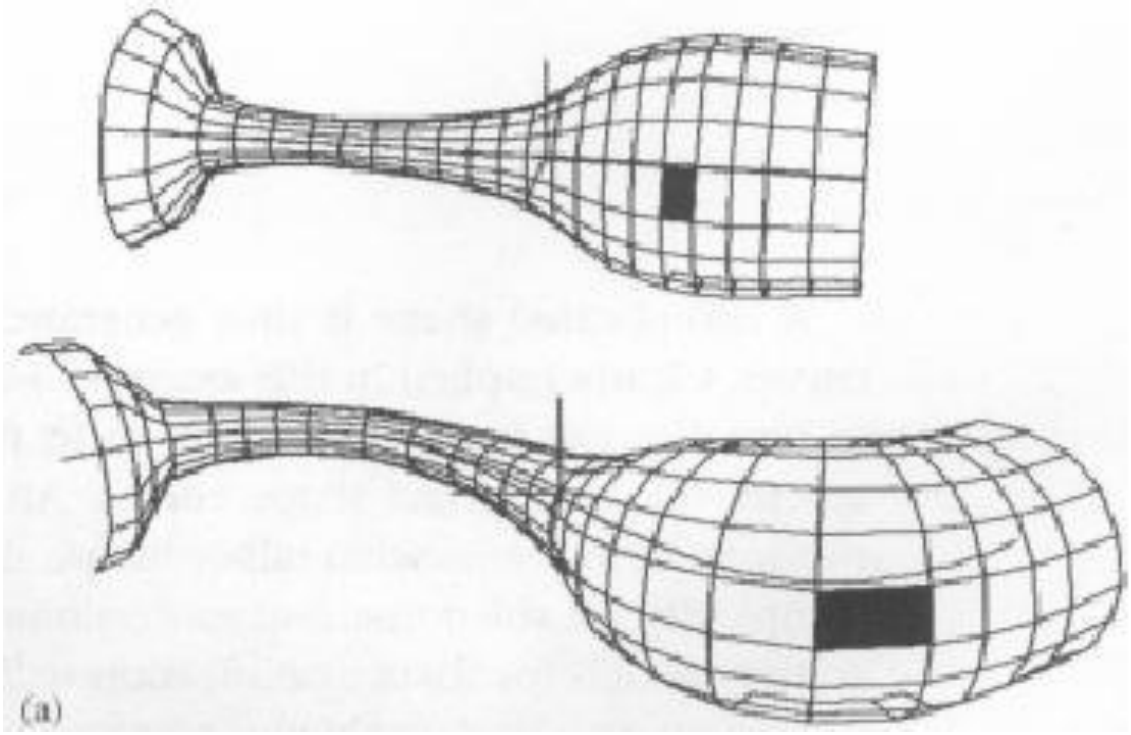
- Curves and surfaces are **inefficient** to render directly.





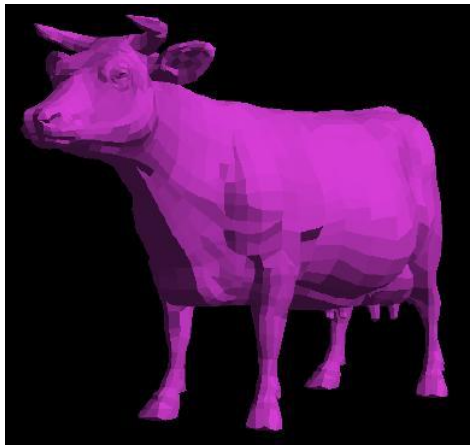
# Represented by primitives (cont.)

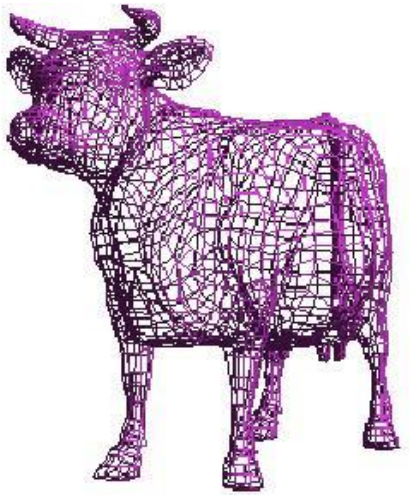
- We use primitives such as polygons instead.

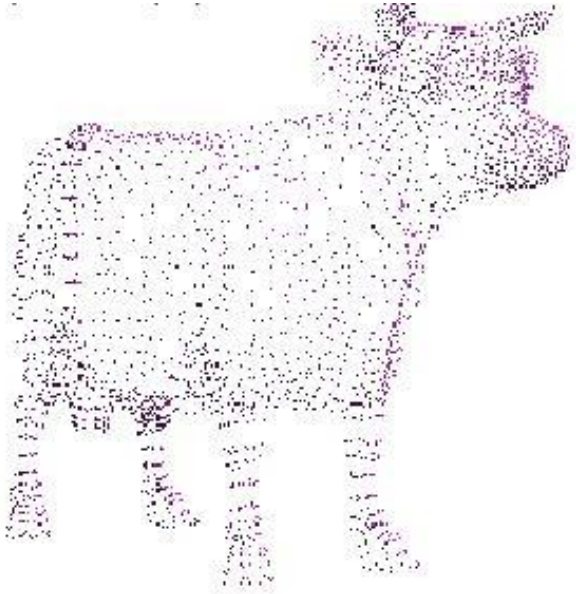


# Represented by primitives

- Polygons

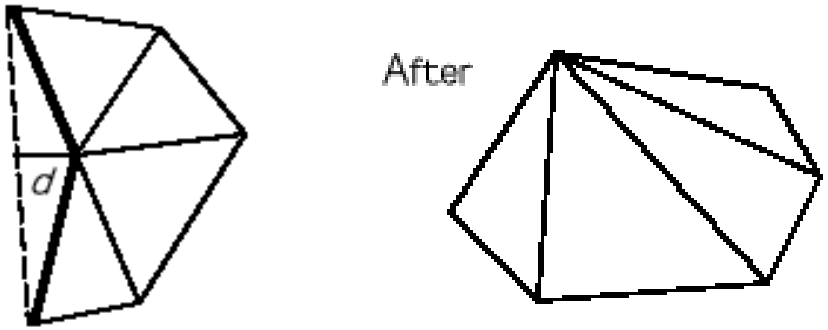




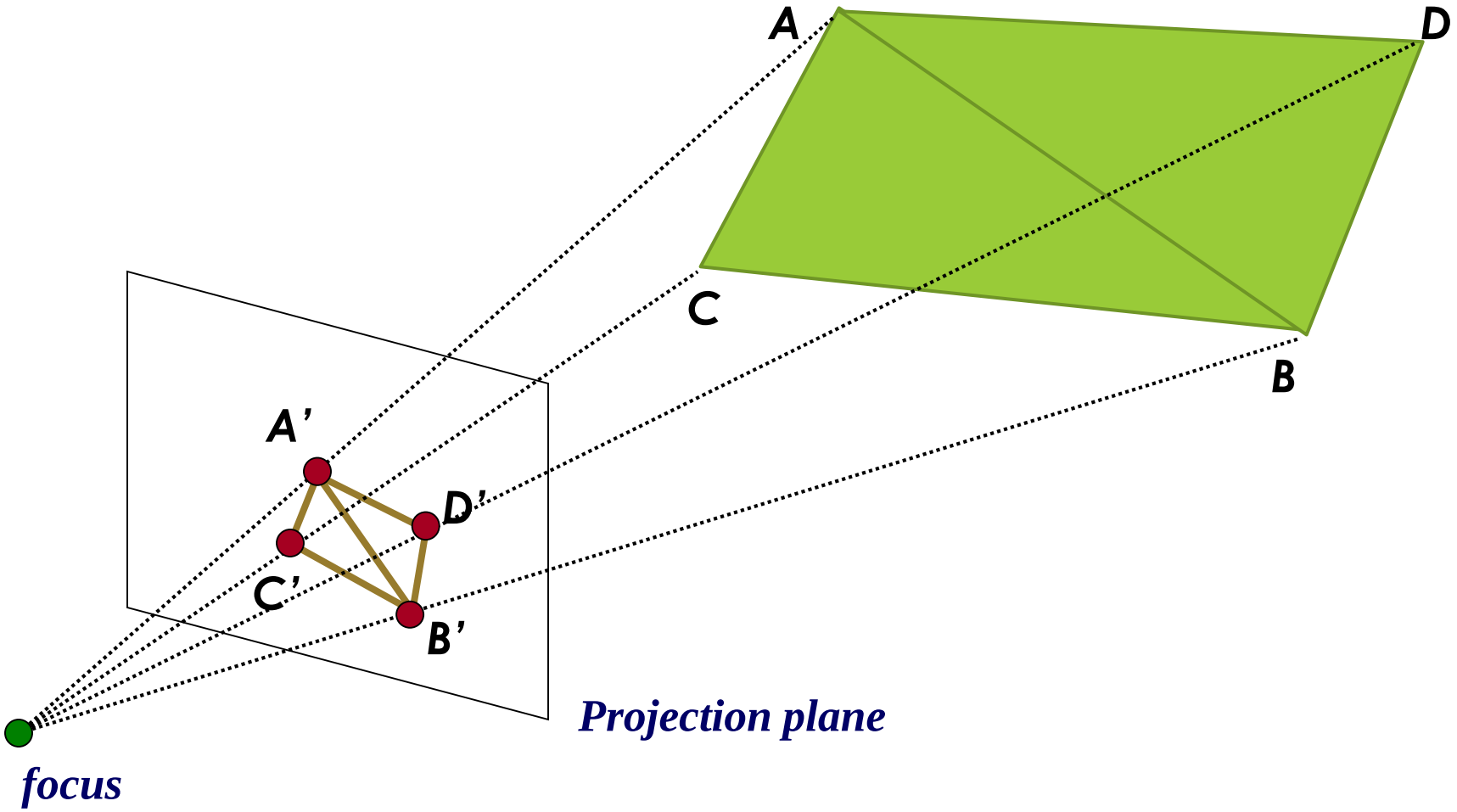


# Represented by primitives (cont.)

- A triangle is usually the most basic primitive.
- Polygons -> triangles.

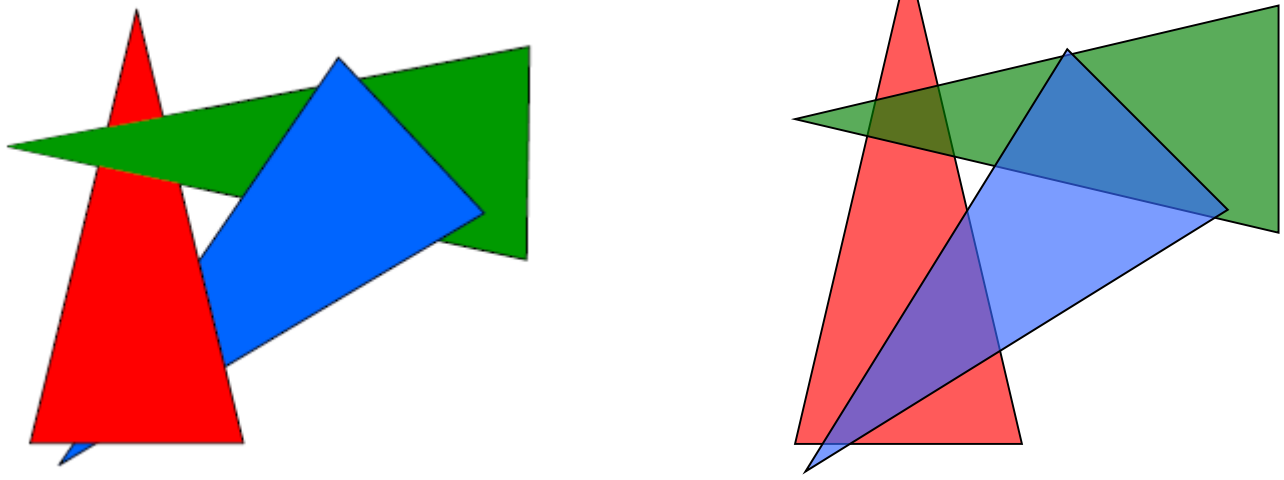


# Projection of triangles



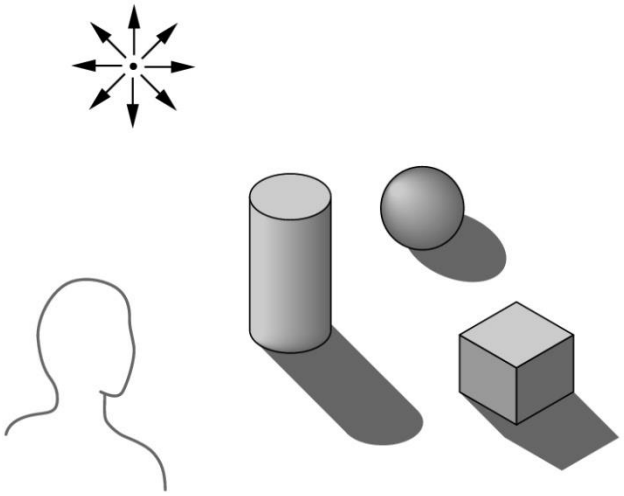
# Visibility

- If we draw triangles directly, our screen will be a “mess”.
- Remove hidden surfaces.



# Global vs Local Lighting

- Cannot compute color or shade of each object independently
  - Some objects are blocked from light
  - Light can reflect from object to object
  - Some objects might be translucent



# A realistic 3D view

- Delicate 3D models.
- Perspective.
- Hidden surface removal.
- Shading (lighting & reflection).
- Shadow.
- Detailed textures and normals

# Appendix: What's a so-called "3D" movie?

- "3D movies" or "Stereo movies" often refer to movies that can provide binocular cues.