

CMP205: Computer Graphics



Lecture 9: Graphics Pipeline

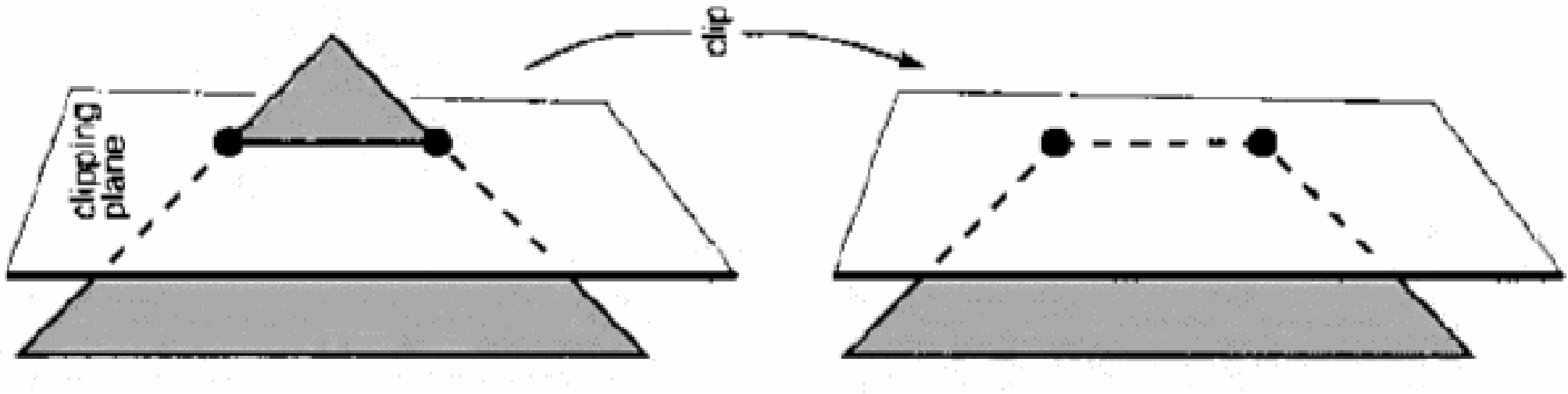
Mohamed Alaa El-Dien Aly
Computer Engineering Department
Cairo University
Fall 2012

Agenda

- Clipping
- Graphics Pipeline

Acknowledgment: Some slides adapted from Steve Marschner and Sayed Hemayed

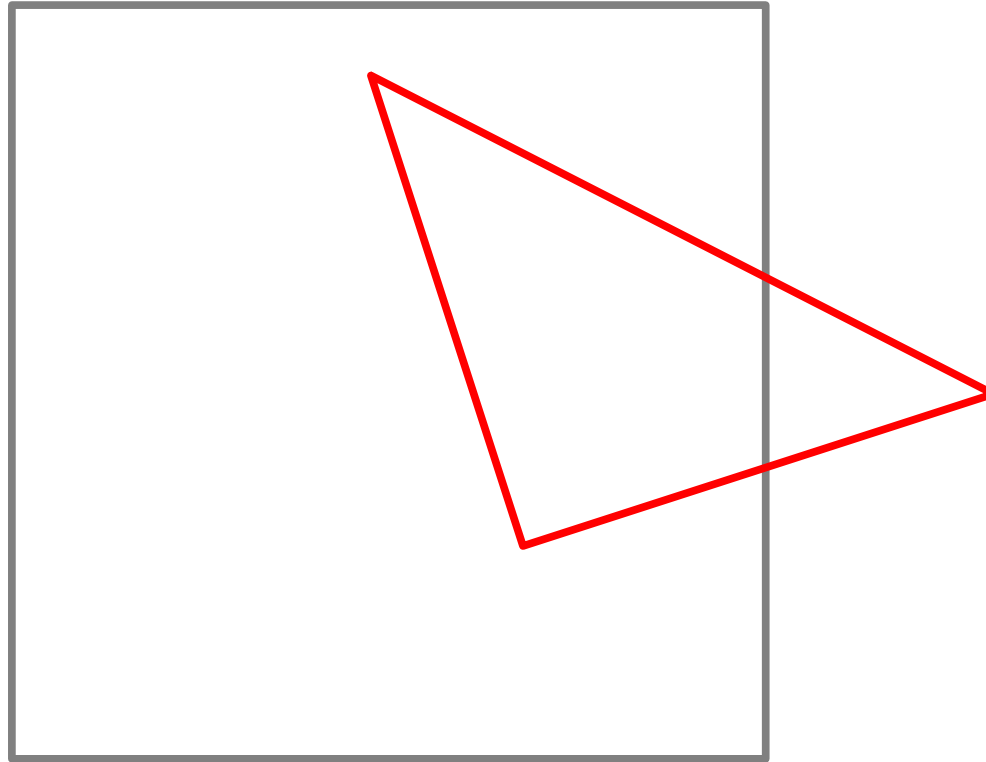
Clipping



A triangle is “clipped” if it cuts a plane

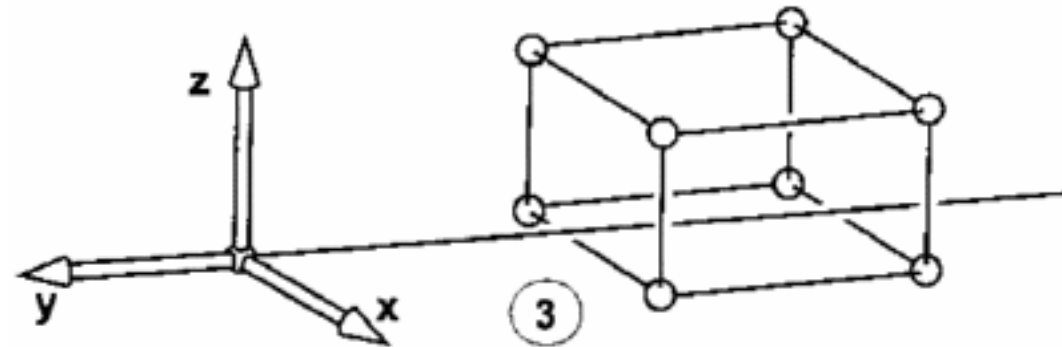
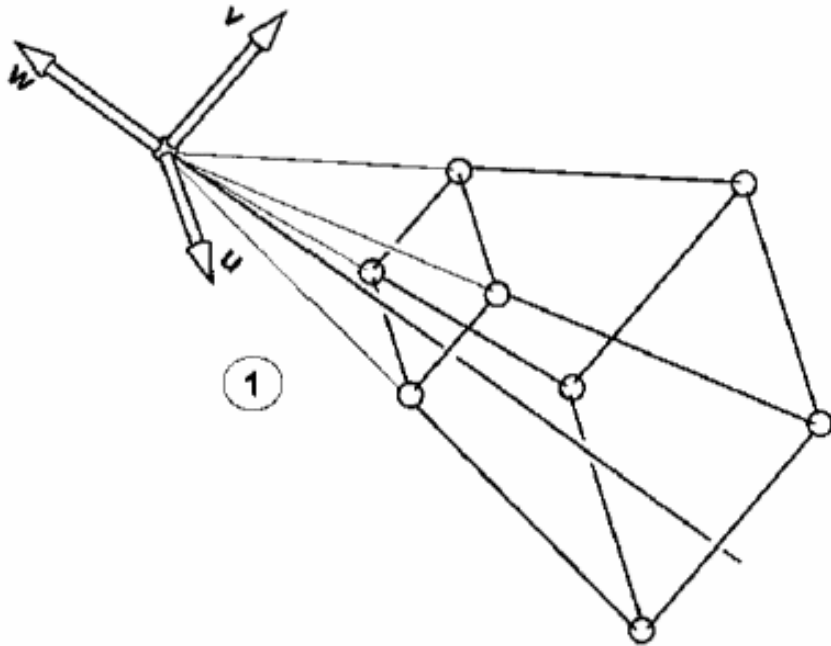
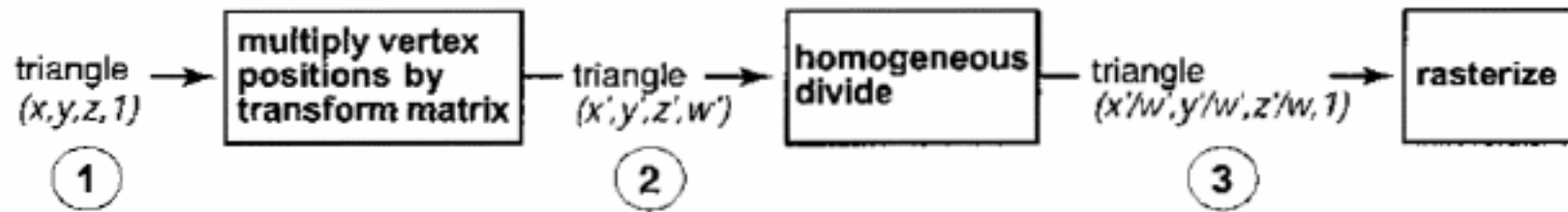
When is clipping required?

Clipping



Clipping required if a triangle cuts the view volume

Clipping



Clipping

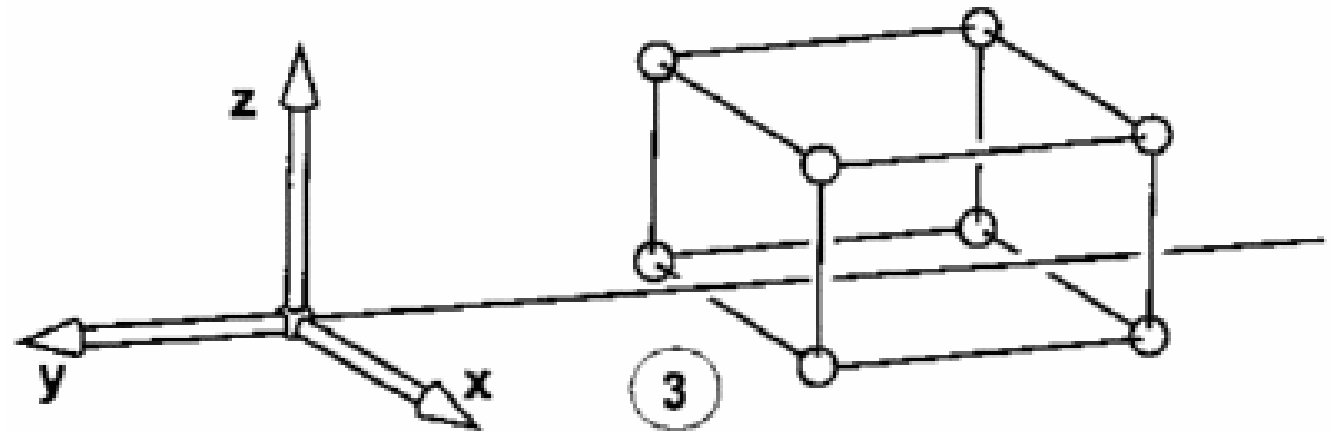
```
for each of the six planes
  if triangle entirely outside plane
    break // not visible
  else if triangle cuts plane
    clip triangle
    if quadrilateral left
      convert into two triangles
```

Where should clipping be done?

Clipping

3. Clipping in the View Volume

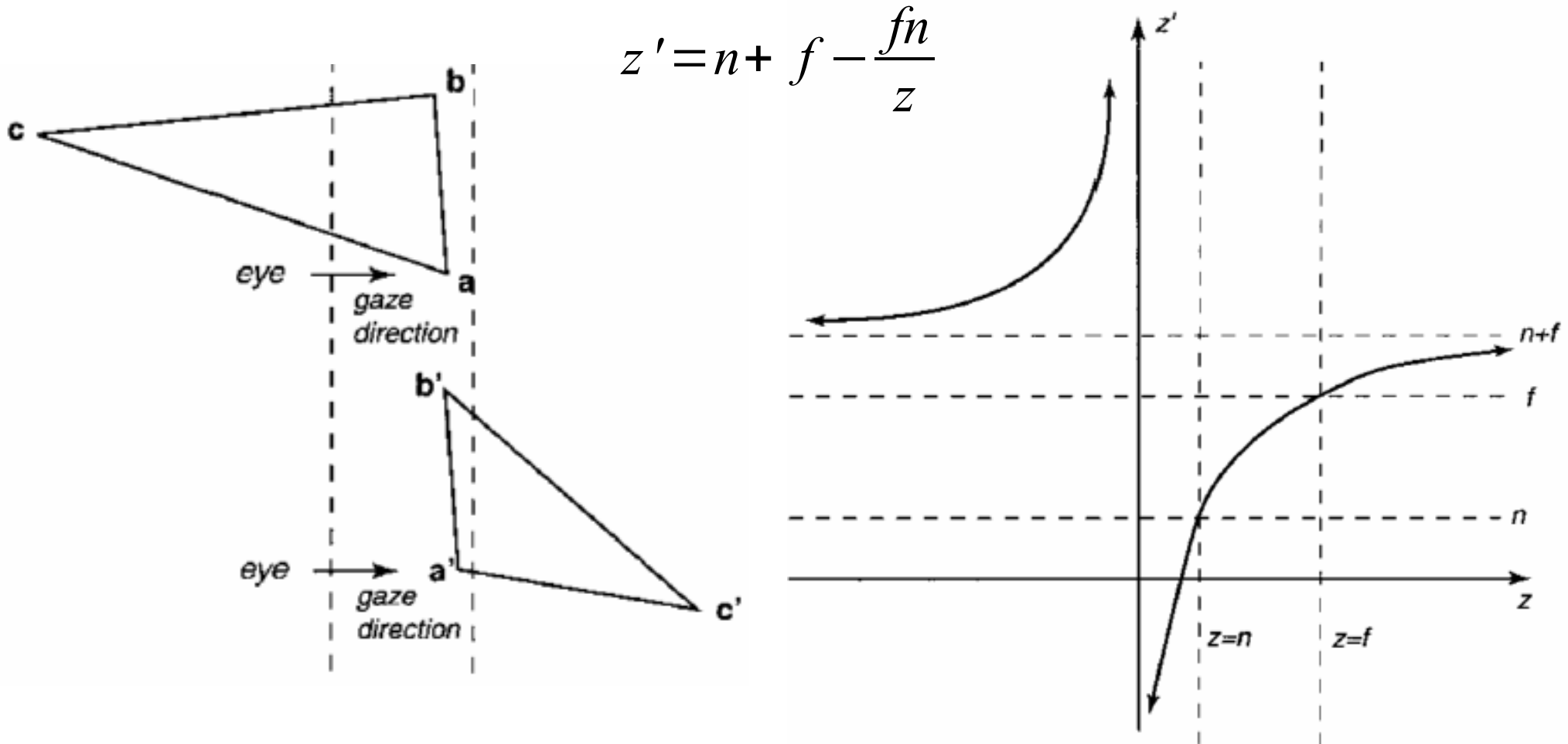
$$\begin{aligned} -x + l &= 0 \\ x - r &= 0 \\ -y + b &= 0 \\ y - t &= 0 \\ -z + n &= 0 \\ z - f &= 0 \end{aligned}$$



Problem?

Clipping

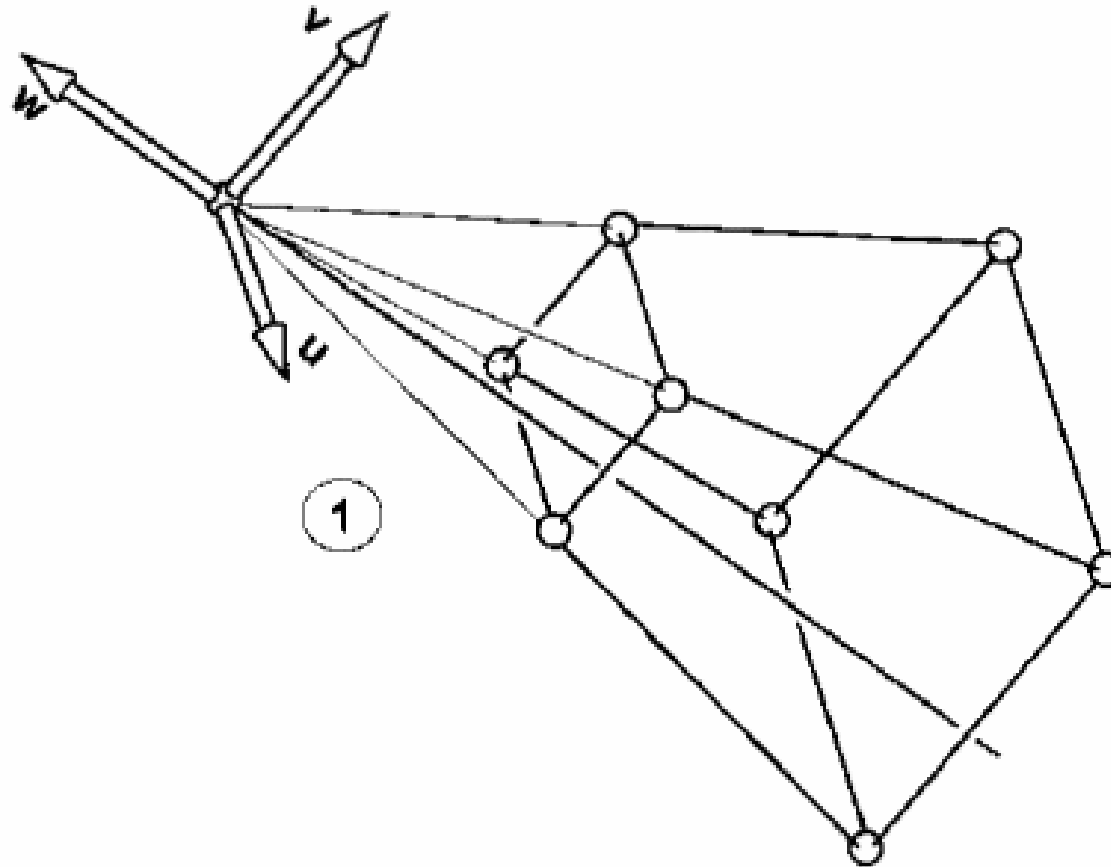
3. Clipping in the View Volume



Discontinuity at $z = 0$

Clipping

1. Clipping in the View Frustum



Compute plane equations for the view frustum

Problem? Planes equations more complicated than (3).

Clipping

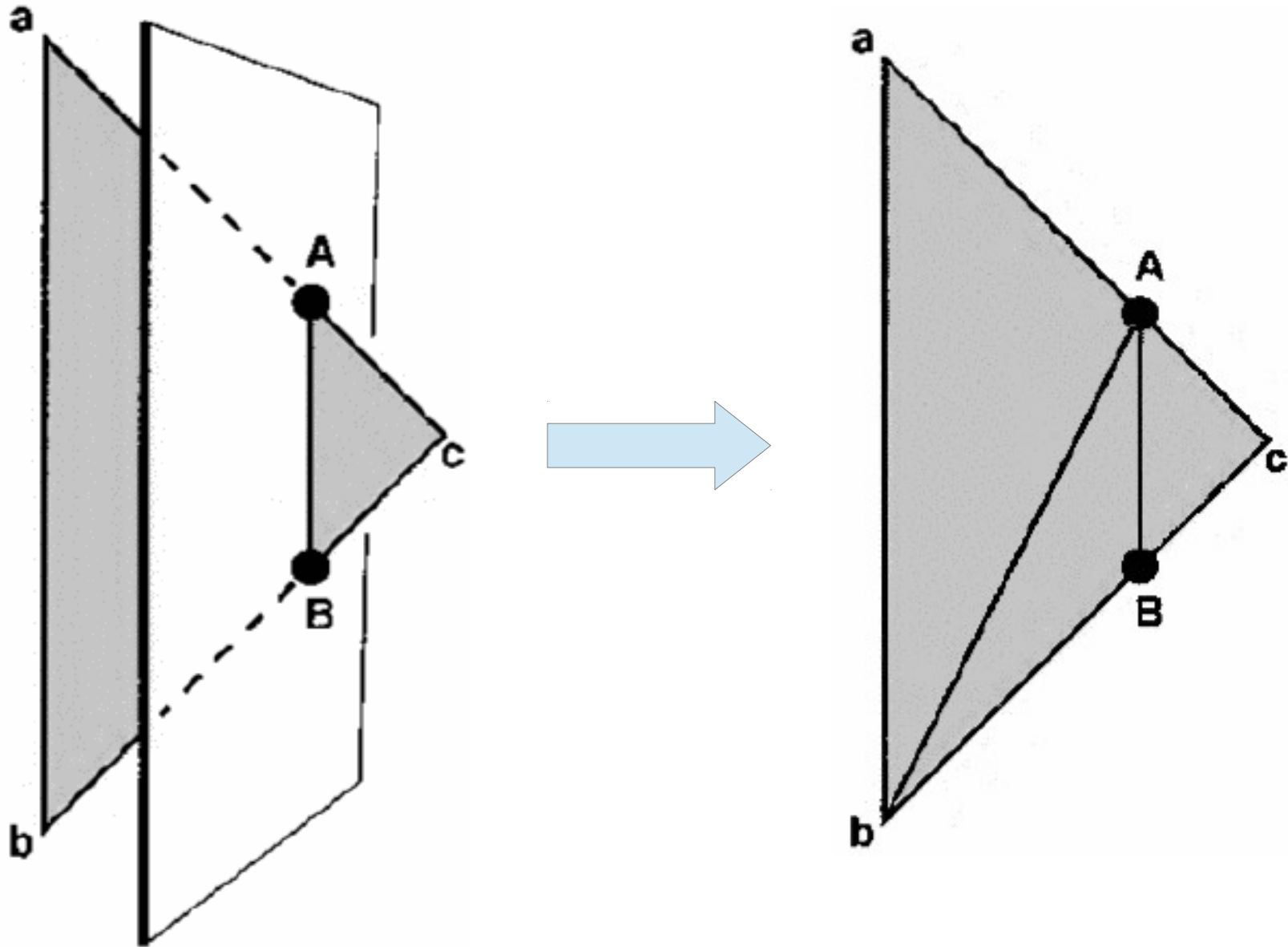
2. Clipping in Homogeneous Coordinates

Homogeneous Point in 4D: $\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$

Planes: $\begin{aligned} -x + lw &= 0 \\ x - rw &= 0 \\ -y + bw &= 0 \\ y - tw &= 0 \\ -z + nw &= 0 \\ z - fw &= 0 \end{aligned}$

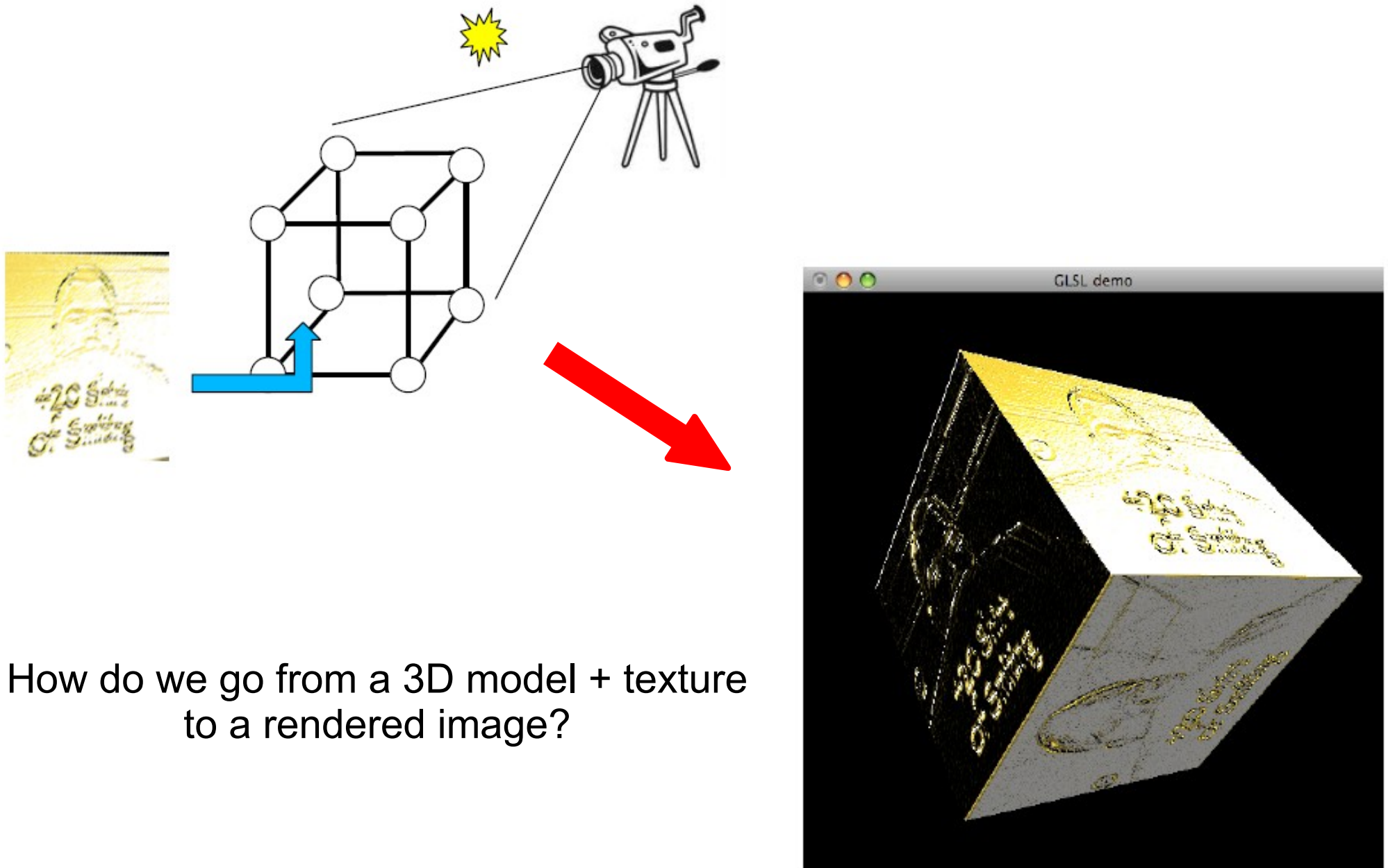
Easy equations and no discontinuities !

Clipping



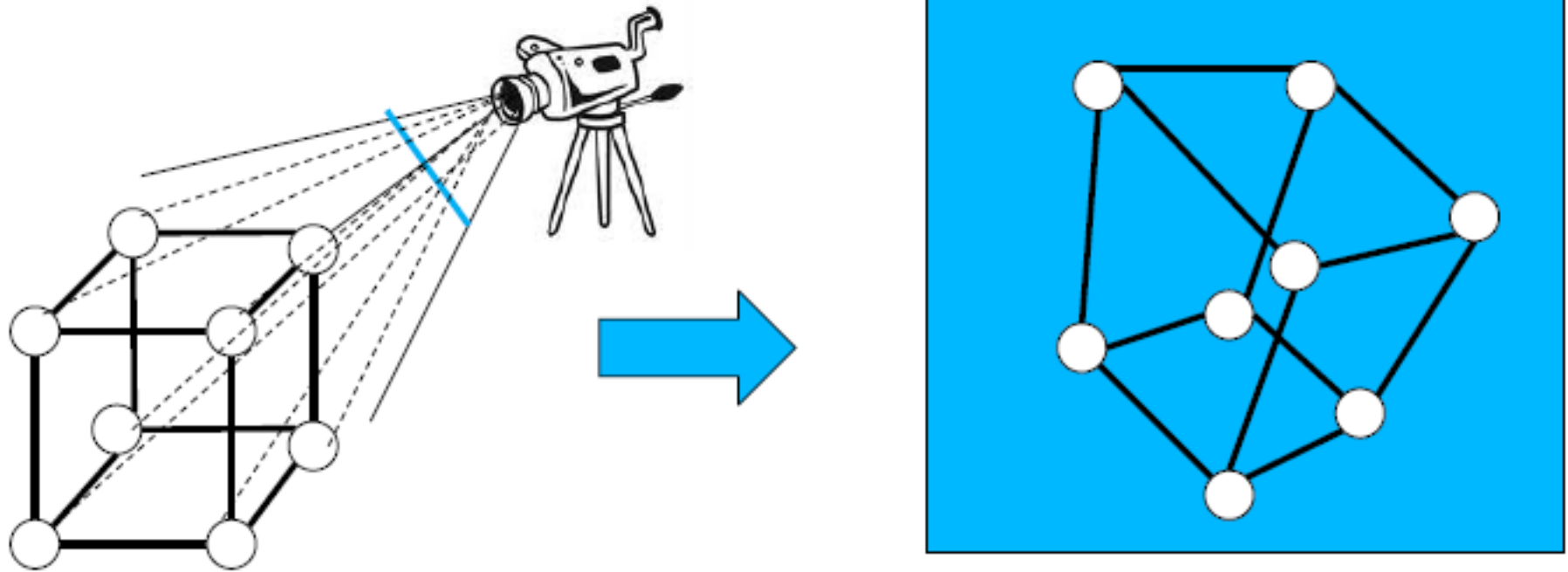
Find intersection points and clip triangle

Graphics Pipeline



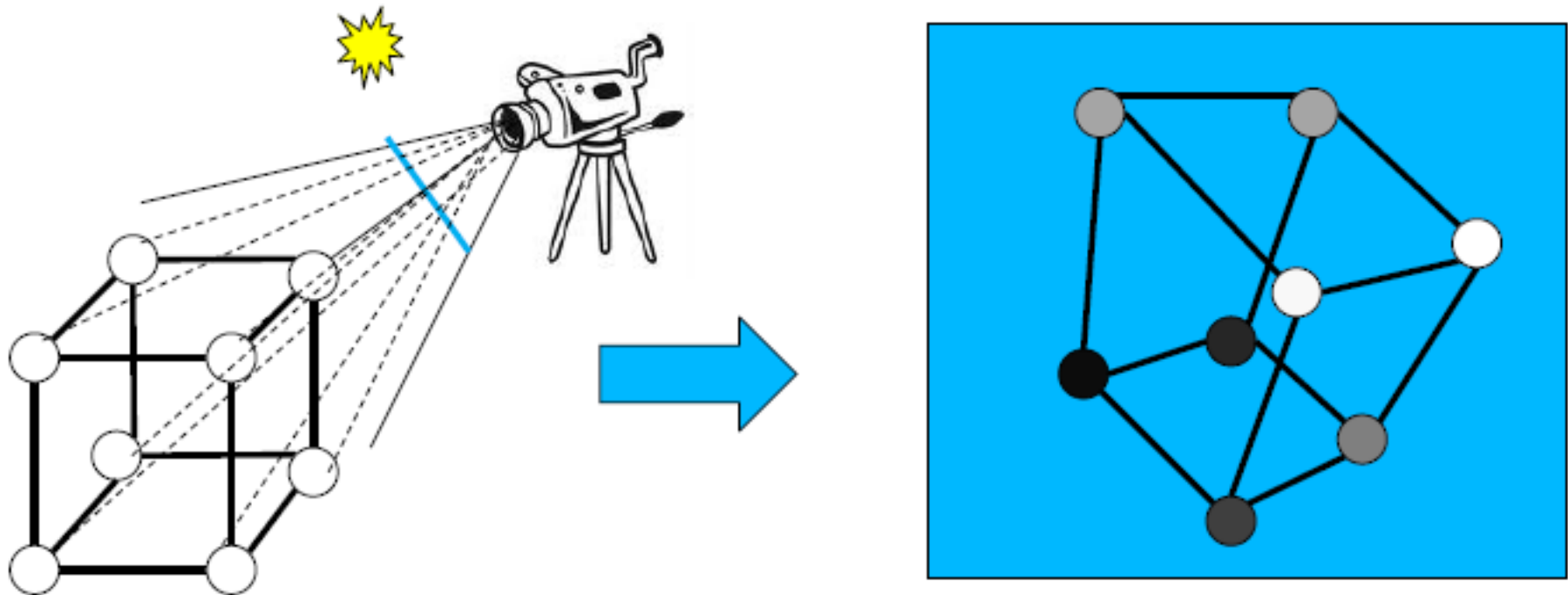
How do we go from a 3D model + texture to a rendered image?

Graphics Pipeline



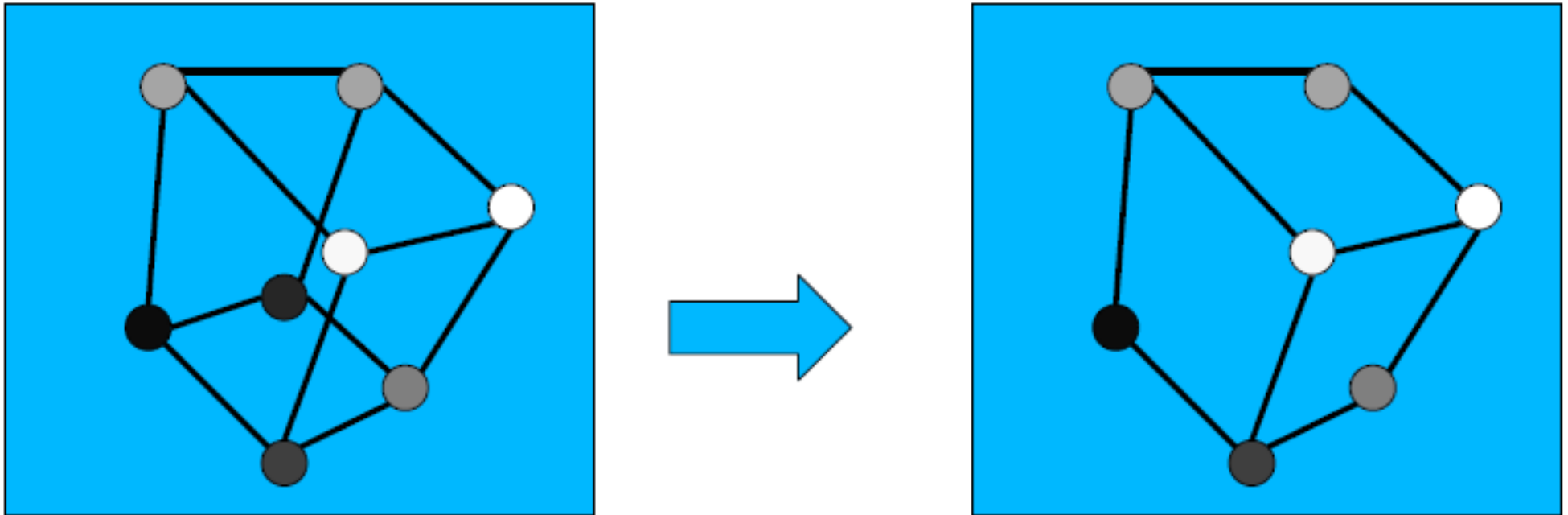
Viewing transformations on vertices

Graphics Pipeline



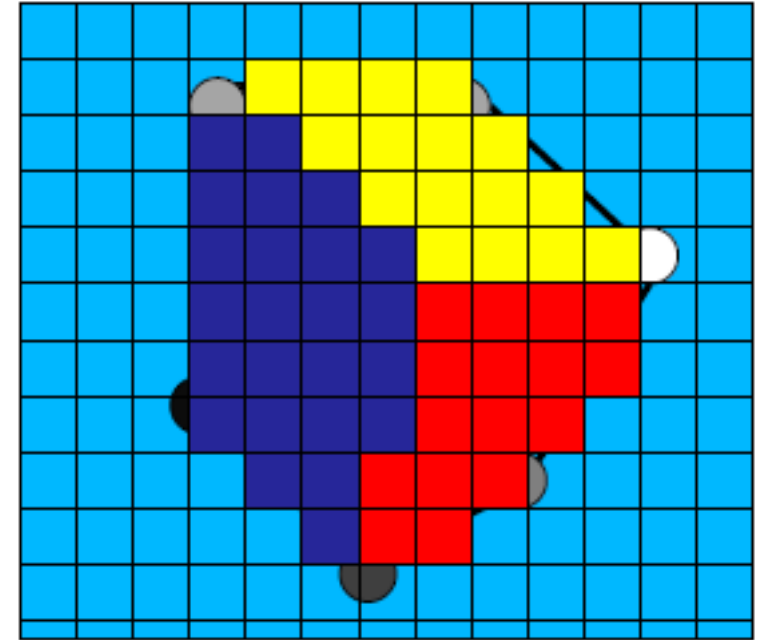
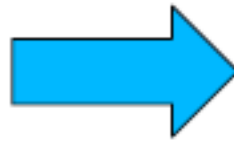
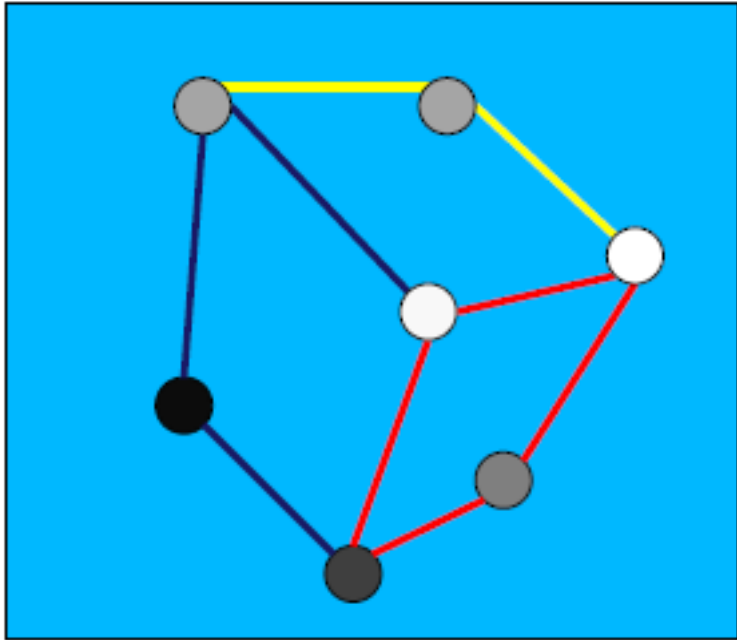
Lighting/Shading for vertices

Graphics Pipeline



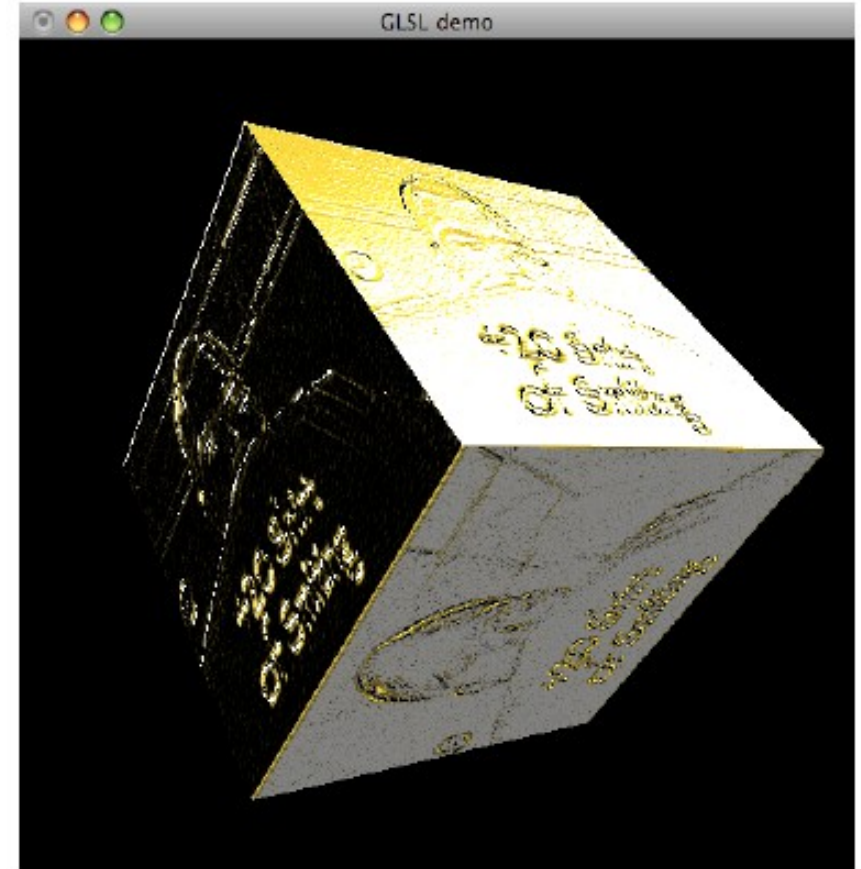
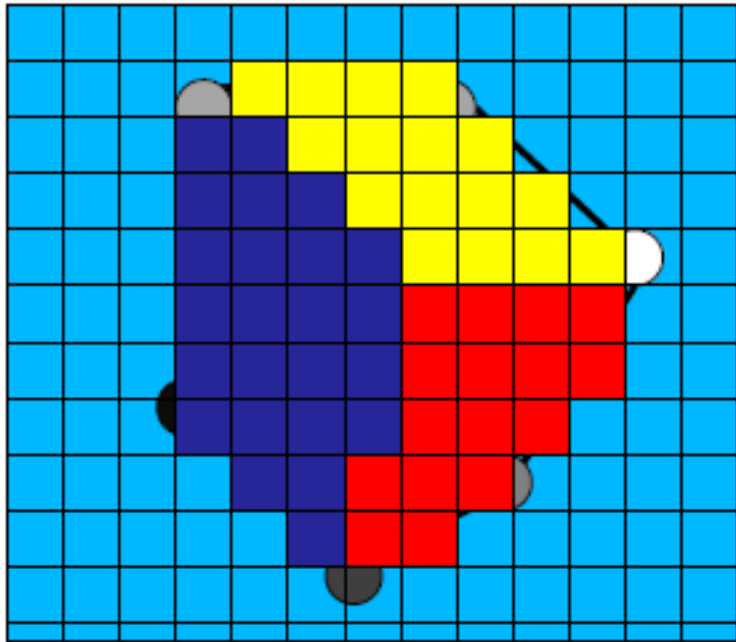
Clipping and Hidden Surface Removal

Graphics Pipeline



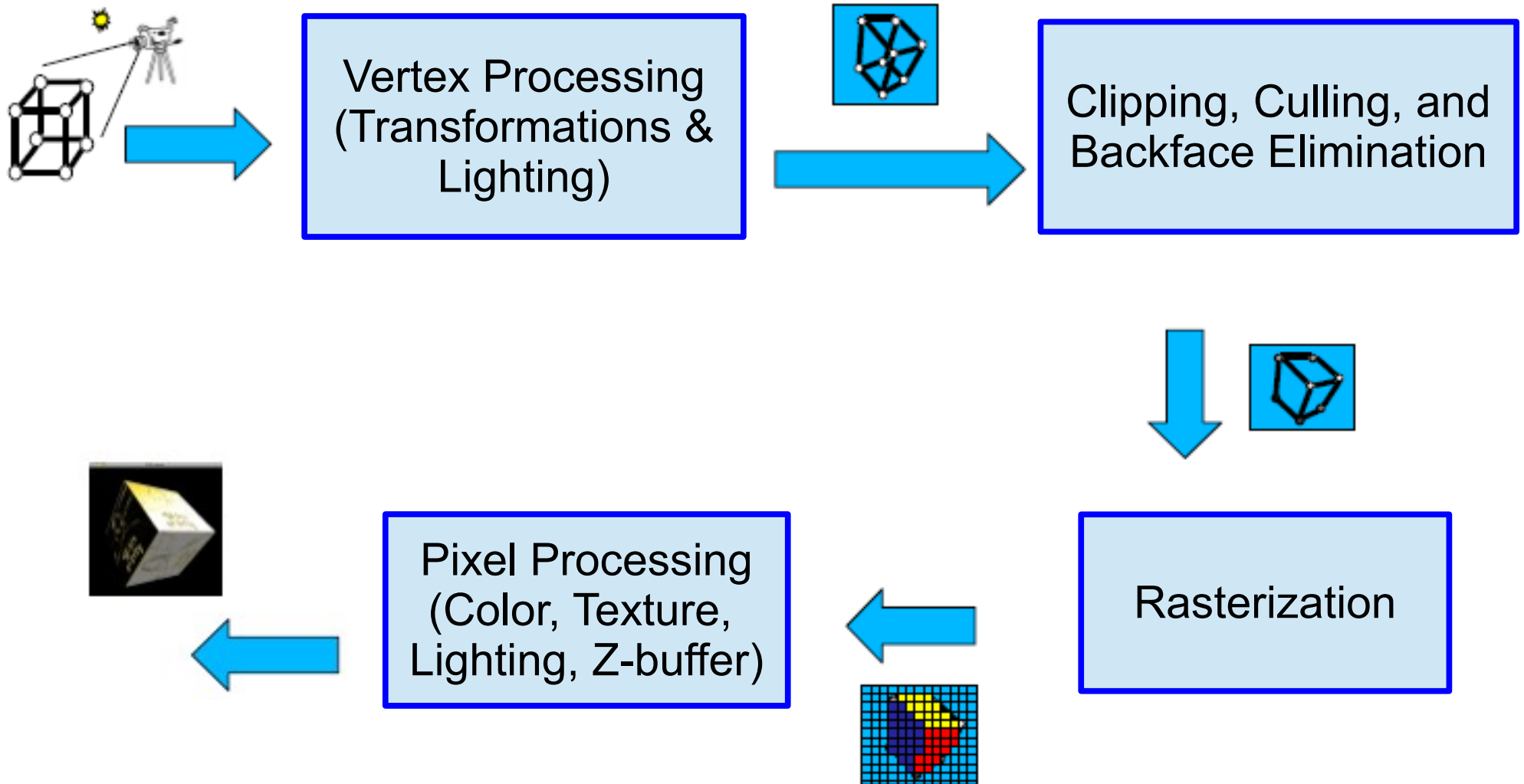
Rasterization

Graphics Pipeline



Pixel Processing & Texture Mapping

Graphics Pipeline



Graphics Pipeline

```
for each triangle in the model
  if triangle is facing away
    break

// Vertex Processing
Transform the vertices
Clip the triangle and homogenize
Compute lighting/shading for vertices

// Rasterization
for each pixel in the bounding box
  Compute barycentric coordinates
  if pixel inside triangle

    // Pixel Processing
    Compute color/shading
    Compute texture coordinates
    Compute z-value (depth)
    if depth < z-buffer
      Write pixel color and update Z-buffer
```

Recap

- Clipping
- Graphics Pipeline