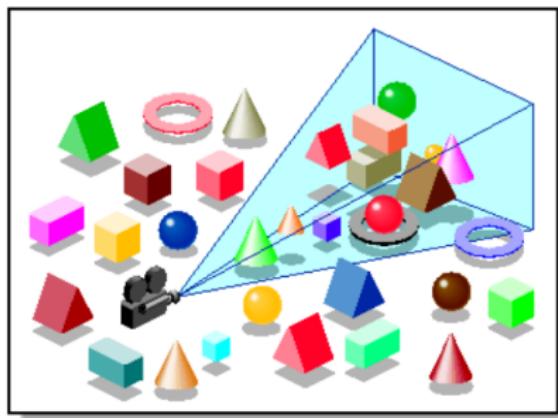


CS 461 - Computer Graphics

Culling & Hidden Surface Removal

Concept

- ▶ World → view
- ▶ A typical synthetic world is composed of a very large number of primitives, but the portion of these primitives that are relevant to the rendering of any single frame is very small

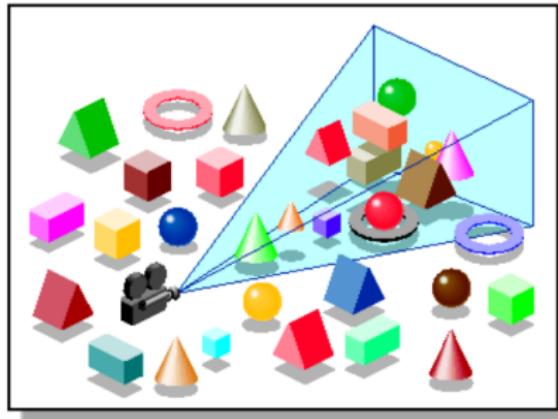


Naive solution

for each pixel

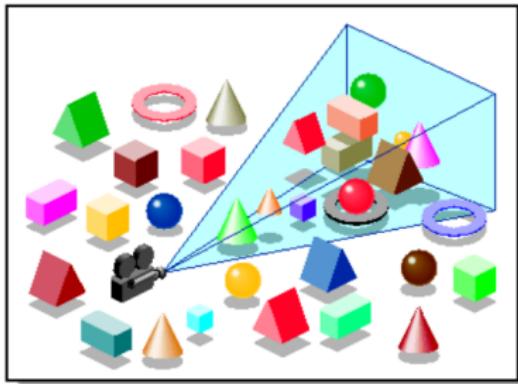
find closest primitive

render pixel with color of closest primitive



Idea

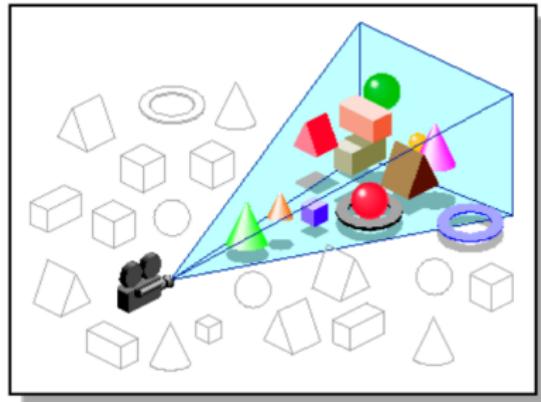
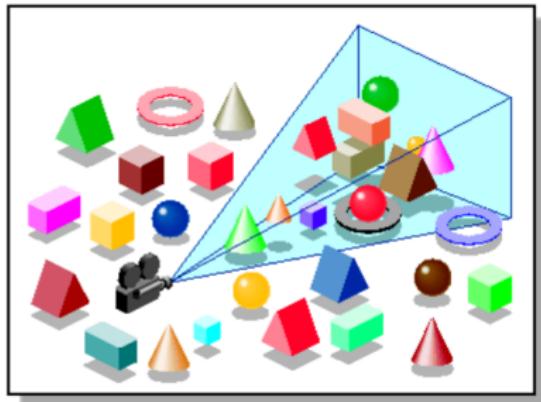
- ▶ Remove primitives that are not relevant to the rendering - Culling
- ▶ Various reasons:
 - ▶ they are outside the field of view (frustum culling)
 - ▶ they are occluded by other objects (occlusion culling)
 - ▶ they are occluded by front-facing primitives of the same object (back-face culling)
- ▶ Cost!!!
- ▶ Hidden Surface Elimination



Frustum culling

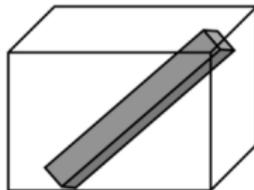
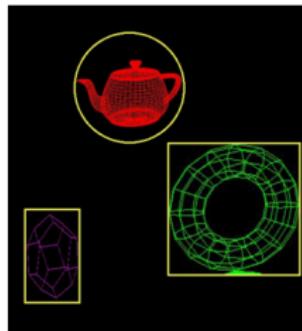
- ▶ Why frustum?
- ▶ Perspective & Orthographic projections
- ▶ Why to remove?
- ▶ How to remove?
 - ▶ Point inside-outside test
 - ▶ Cohen-Sutherland line clipping - 6 bits
 - ▶ Sutherland-Hodgman polygon clipping - for each plane

Frustum culling



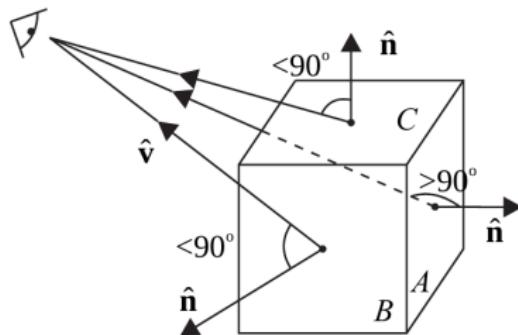
Frustum culling

- ▶ Bounding box



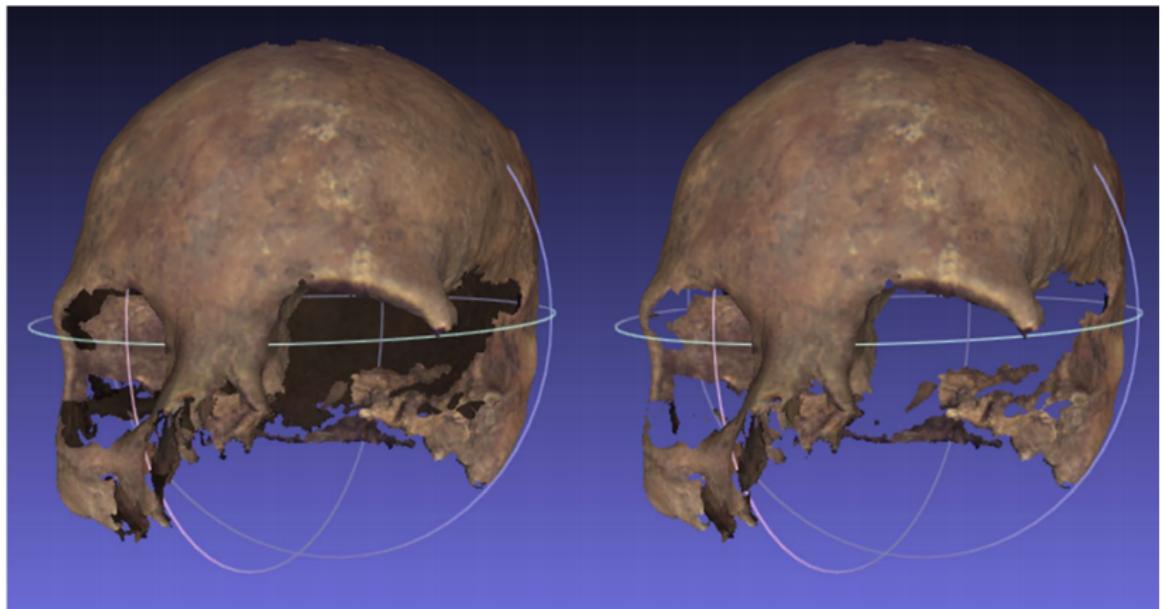
Back-face culling

- If models are constructed in such a way that the back sides of polygons are never visible, then we can cull polygons showing their back-faces to the viewer



- $\hat{v} \cdot \hat{n} < 0$

Back-face culling

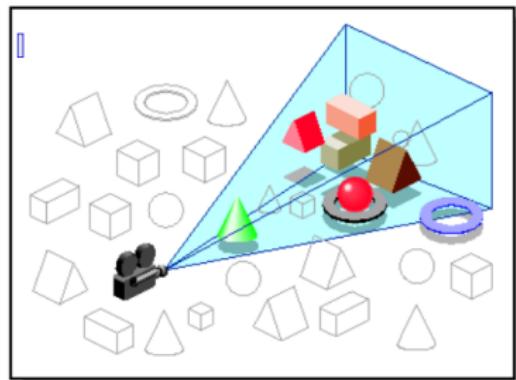
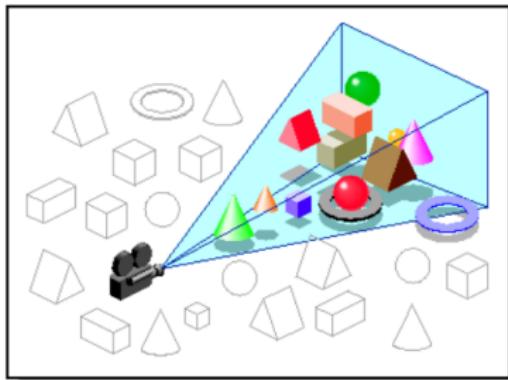


Back-face culling

- ▶ Conditions:
 - ▶ Solid-model
 - ▶ Manifold
 - ▶ No boundary
- ▶ Can cull front or back faces
- ▶ Can sometimes double performance (cube example)

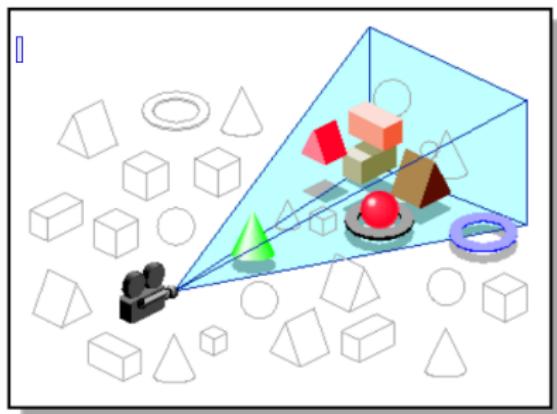
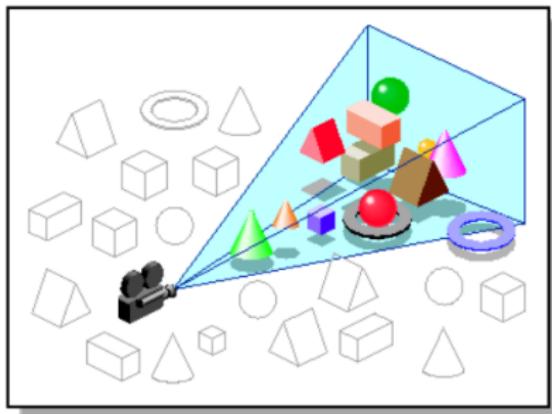
Occlusion culling

- ▶ A very small portion of the primitives in a large scene are visible for a given set of viewing parameters
- ▶ The rest are hidden by other primitives nearer to the observer
- ▶ Occlusion culling aims at efficiently discarding a large number of primitives before computationally expensive HSE algorithms are applied



Occlusion culling

- An occluder is a primitive, or a combination of primitives, that occludes a large number of other primitives, the occludees, with respect to a certain viewpoint



Portal culling

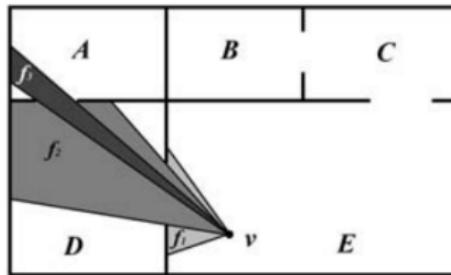
- ▶ Indoor static object (VR)
- ▶ 3D scene can be broken into areas called cells, which are linked together by portals
- ▶ A portal is a window in 3D space that allows objects in one cell to be seen from another cell



- ▶ Intuition or Idea!!!

Portal culling

- ▶ Concepts of Stab tree, Potential Visible Set (PVS) Matrix, BSP tree



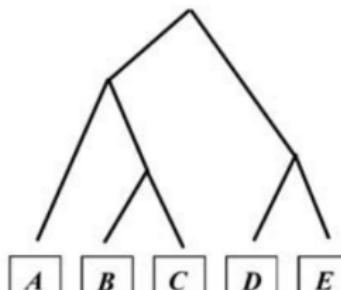
(a)



(b)

| | | | | | |
|---|---|---|---|---|---|
| | A | B | C | D | E |
| A | 1 | | | 1 | 1 |
| B | | 1 | 1 | | 1 |
| C | | 1 | 1 | 1 | 1 |
| D | 1 | | 1 | 1 | 1 |
| E | 1 | 1 | 1 | 1 | 1 |

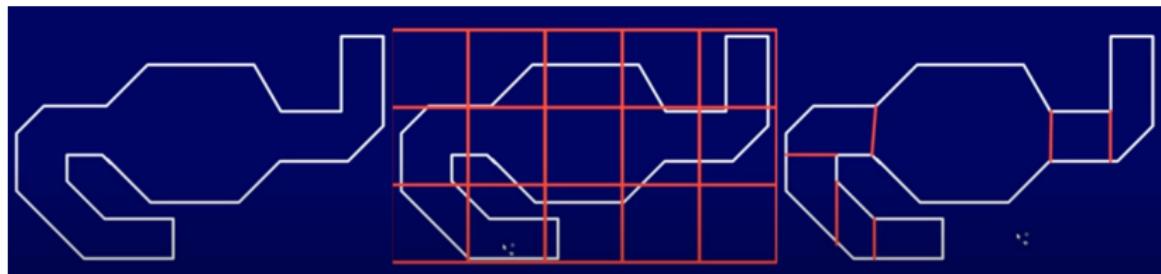
(c)



(d)

Portal culling

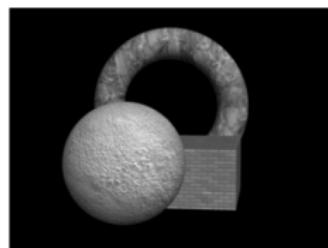
- ▶ Algorithm
- ▶ Convex subdivision



- ▶ General case - No cells/portals
- ▶ Simple solution - Planar occluder - ranking

Depth (Z)-buffer Algorithm

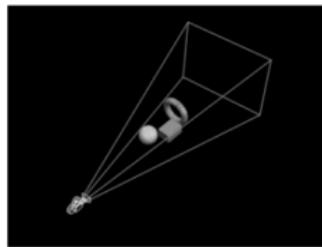
- ▶ The Z-buffer is a classic image space HSE algorithm
- ▶ Maintain a two-dimensional memory of depth values



(a)

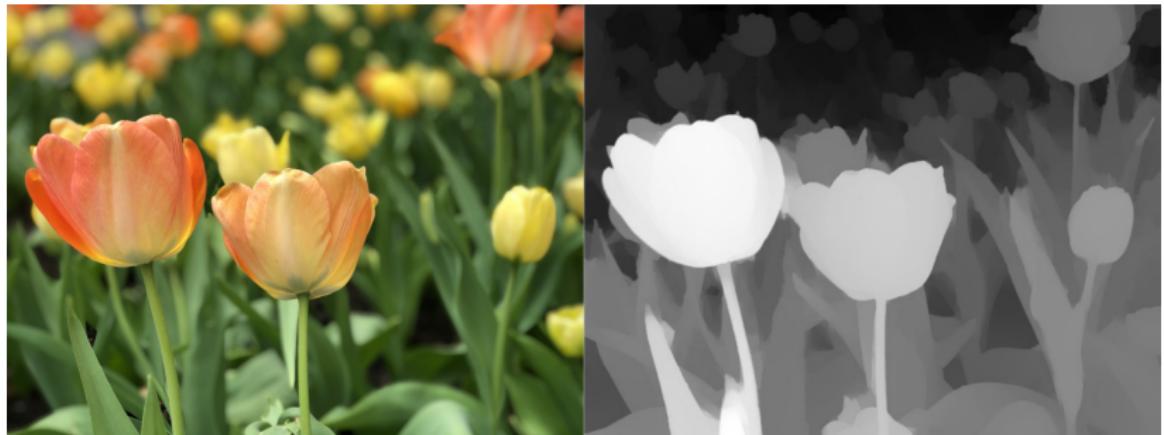


(b)



(c)

Depth (Z)- buffer



Depth (Z)- buffer

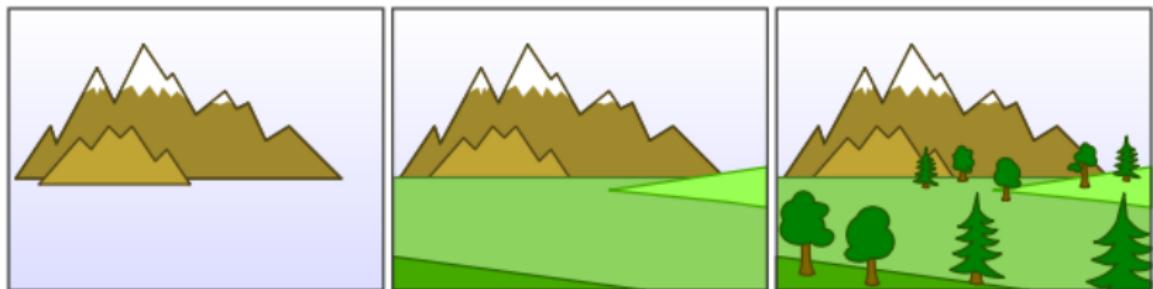


Depth (Z)- buffer



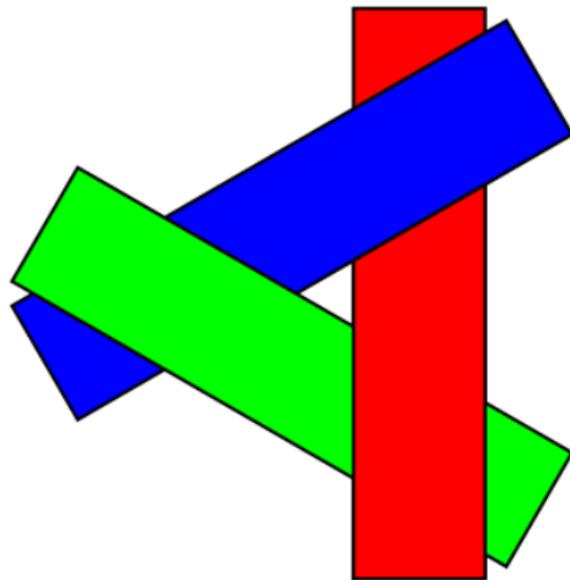
Painter's Algorithm

- ▶ This algorithm sorts polygons according to their distance from the observer and displays them in reverse order
- ▶ The way painter works



Painter's Algorithm

- ▶ Overlaps in z arise when the z extents of polygons overlap
- ▶ Sub-divide



Other concepts - quick overview

- ▶ Binary Space Partitioning
- ▶ Warnock's algorithm
- ▶ **Ray-tracing**
- ▶ Scan-line algorithm...

Next class

- ▶ October 7th Wednesday - 9 to 10
- ▶ Topic: Colors