

Department of Computer Science and Engineering
CSP: 316 Computer Graphics Lab

5. Hidden Surface removal: Aim is to understand the need of hidden surface removal or realistic viewing.

Implement back face removal technique on a unit cube (every surface should have a different colour). Show the change in visible surfaces by changing the viewing position.

Implement two of the polygon visible surface algorithms studied in the class such as Z-Buffer

algorithm, scan line algorithm, painter's algorithm, BSP tree algorithm or area subdivision algorithm.

Implement a simple ray tracer for spheres and polygons. Use OpenGL features for showing the

rendering effect.

Assessment:

You should be able to answer the following questions.

What is object space and image space?

How many numbers of surfaces Z- buffer algorithm can handle?

Do we need to sort (or input in order) surfaces for a Z-buffer algorithm?

Can we handle concave surfaces with back face removal algorithm?

Which hidden surface removal algorithm should be preferred if the number of surfaces is large?

Which hidden surface removal algorithm should be preferred if the number of overlapping surfaces is large?

How can the algorithms discussed be adapted to work with polygons containing holes?

How might you modify the BSP-tree algorithm to accept objects other than polygons?