Computer Graphics

(Viewing in 3D)

Lecture 006

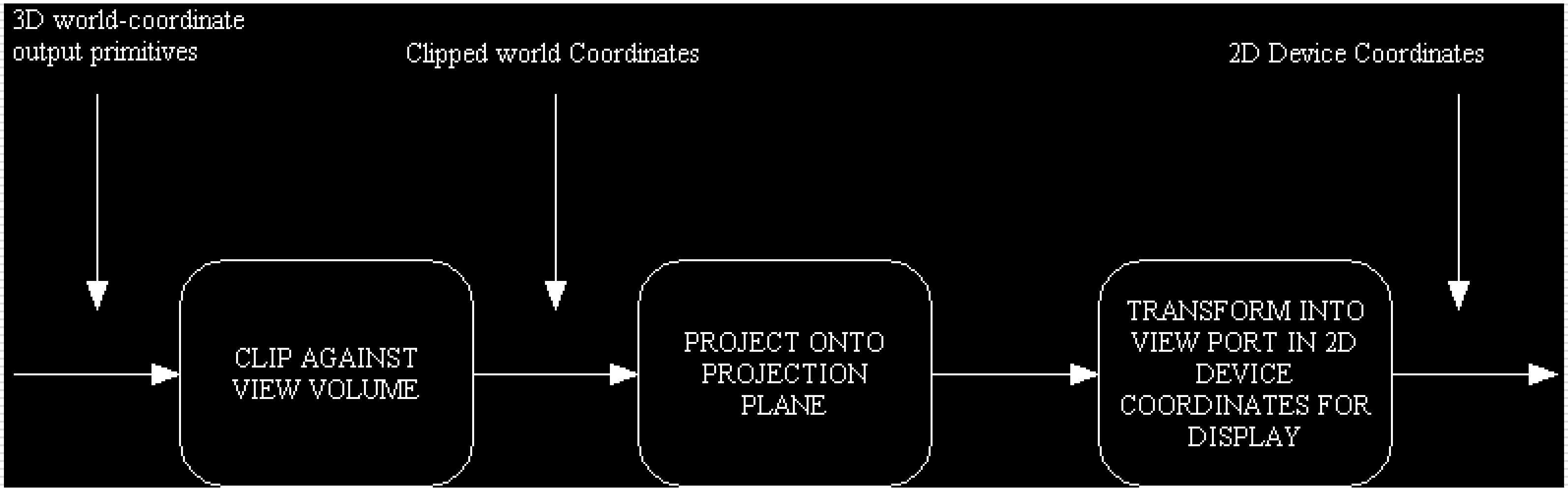
Projections (ref. Foley, van Dam)

- The 3D viewing process is inherently more complex than is the 2D viewing process. In 2D, we simply specify a window on the 2D world and a view port on the 2D view surface.
- The solution to the mismatch between 3D objects and 2D displays is accomplished by introducing *projections* (which transform 3D objects onto a 2D projection plane).
- In 3D viewing, we specify a *view volume* in the world, a projection onto a projection plane, and a view port on the view surface. Conceptually, objects in the 3D world are clipped against the 3D view volume and are then projected.
- The contents of the projection of the view volume onto the projection plane, called the *window*, are then transformed (mapped) into the view port for display.

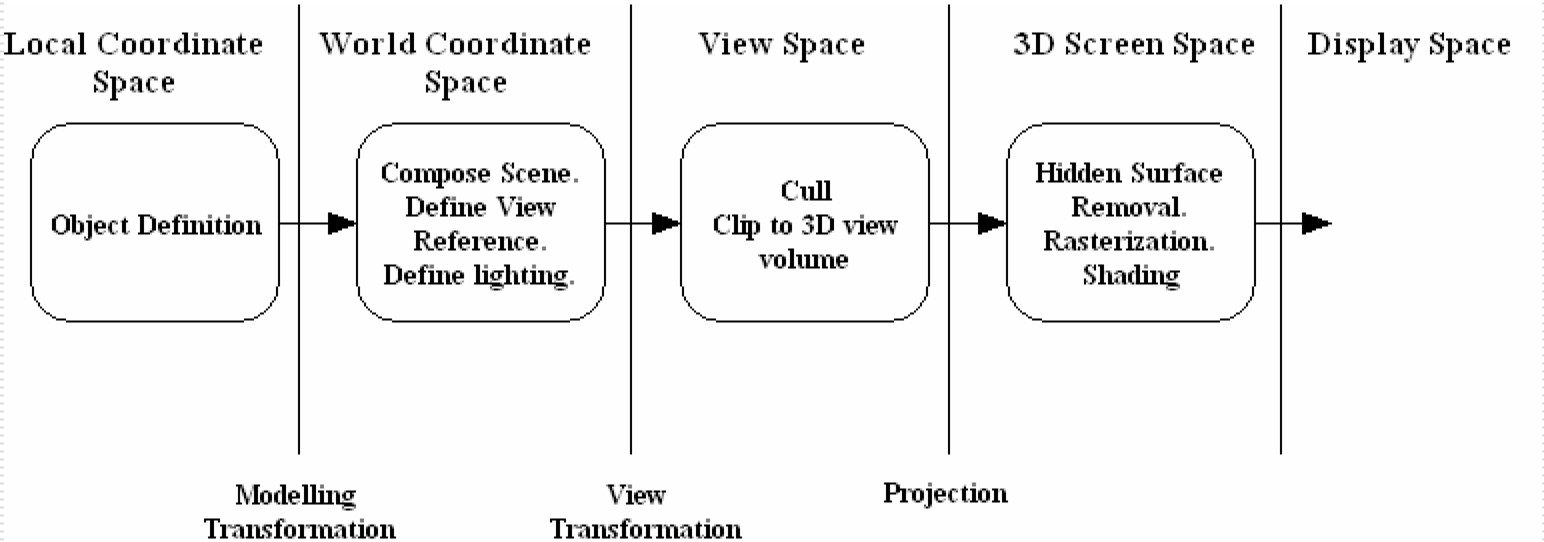
Conceptual model of the 3D viewing process

- Usually local coordinate systems (for individual models) are first transformed into the world coordinate system.

- The following is the conceptual model of the 3D viewing process. Note that operations like visible-surface determination and shading are not being shown here.



The graphics pipeline (Geometric Operations)



Projections (Perspective and Parallel)

- In general, projections transform points in a coordinate system of dimension n into points in a coordinate system of dimension less than n.
- The projection of a 3D object is defined by straight projection rays (called *projectors*) emanating from a centre of projection, passing through each point of the object, and intersecting a *projection plane* to form the projection.
- In CG we deal with planar geometric projections because the projection is onto a plane rather than some curved surface and uses straight rather than curved projectors.
- We will consider *perspective* and *parallel* projections.
- The distinction lies in the relation of the center of projection to the projection plane. If the distance from the one to the other is finite then the projection is perspective, otherwise, it is parallel.

An arbitrary 3D view (Definition)

- The viewing-reference coordinate system (VRC) is defined in terms of a view reference point (VRP) which represents the origin, a view-plane normal vector (VPN) and a view-up vector (VUP).

- This is analogous to placing a camera (or an eye) at the VRP pointing in the direction given by the VPN. The camera is then rotated around the perpendicular to the plane that passes through it (the camera) so that the VUP is vertical.

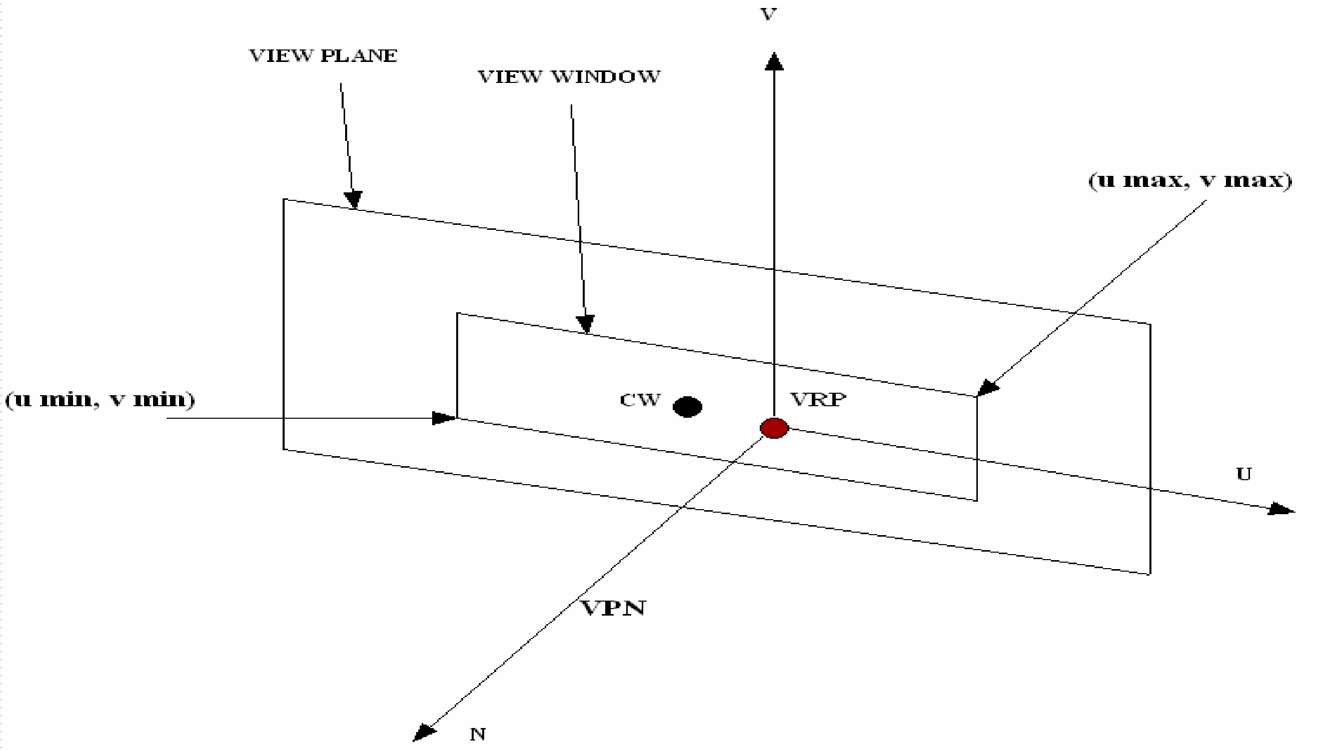
- The projection plane (view plane), is defined by a point on the plane (the VRP) and a normal to the plane (VPN). The view plane may be anywhere with respect to the world objects to be projected: it may be in front of, cut through, or be behind the objects.

- Given a view plane, a window on the view plane is needed (its contents are mapped into the view port).

- One axis of the VRC is the VPN; this axis is called the n-axis. The v-axis is defined such that the projection of VUP parallel to VPN onto the view plane is coincident with the v-axis. The u-axis is defined such that u,v, and n form a right handed-coordinate system.

An arbitrary 3D view (Diagram)

CW is the centre of the window.



Clipping in 3D

- The part of our world which is to be captured by our 2D image is determined by the projection window, the front and back clipping planes and the centre of projection. Together these define a *view volume* corresponding to that portion of the world to be displayed.

- For parallel projection this view volume has the shape of a parallelepiped and the shape of a frustrum (pyramid) for the case of perspective projections.

- Objects lying outside this volume must be clipped.

- Clipping usually occurs before the projection transformation.

- For perspective projections the clipping and subsequent projection is made much simpler if the view volume is normalized, or transformed into a regular parallelepiped view volume. The normalized view volume is defined by the six planes $x = -1$, $x=1$, $y=-1$, $y=1$, $z=0$, $z=-1$.

3D version of Cohen-Sutherland clipping algorithm

- This is simply a generalisation of the 2D version, using a 6-bit code instead of a 4-bit one, to enable clipping against the sides of a regular parallelepiped. Note that lines are clipped against planes instead of lines.

  - Bit 1 – point is above view volume $y > 1$
  - Bit 2 – point is below view volume $y < -1$
  - Bit 3 – point is right of view volume $x > 1$
  - Bit 4 – point is left of view volume $x < -1$
  - Bit 5 – point is behind view volume $z < -1$
  - Bit 6 – point is in front of view volume $z > 0$

- As in 2D a line is trivially accepted if both endpoints have a code of all zeros, and is trivially rejected if the bit-by-bit logical *and* of the codes is not all zeros. Otherwise, the process of line subdivision begins.

Culling or back-face Elimination

- Culling is an operation that compares the orientation of complete polygons with the view point or centre of projection and removes those polygons that cannot be seen.
- Culling eliminates all those surfaces whose normals point away from the COP, in the knowledge that such surfaces would be completely hidden by other front-facing surfaces of the same polyhedron.
- On average, half of the polygons in a polyhedron are back-facing and the advantage of this process is that a simple test removes these polygons from consideration by a more expensive hidden surface removal algorithm.
- The test for visibility is straightforward and is best carried out in view space.
  - Visibility := Np . N > 0 where Np is the polygon normal and N is the 'line of sight' vector.