1 Administrivia

Announcements

Change to course description. Additional prerequisite: must know what a “pixel” is.

Assignment

Read 2.1–4.

From Last Time

Pointers and memory allocation in C. How’s that project coming along?

Outline

1. CG Applications.


3. Objects and Images.

5. Pinhole and Synthetic Cameras.


**Coming Up**

OpenGL introduction lab.

## 2 CG Applications

   
   (a) Not real-time: can use model-render paradigm.

   (b) *Lots* of off-line compute cycles. Good physics.

   (c) High quality results.

2. Games: Quake, etc.

   (a) Real-time, interactive.

   (b) Lots of on-line compute cycles.

   (c) Procedural physics: fast, not too accurate.

   Real physics: lots of compute cycles (slow), accurate.

3. Simulation: Well, a surgical simulation is “like” a game.

## 3 Graphics System Components

Frame buffer attributes:
1. Unit: picture element (pixel).

2. Discretization process (rasterization): geometry info to raster (array or line of pixels).

3. Depth: 1, 8, 16, 24 bits.

4. Resolution: $640 \times 480$, $800 \times 600$, $1024 \times 768$, etc.

   Aspect ratio.

4 Objects and Images

1. In any visualization process, painting, photography, etc., there are two key elements: object and viewer.

   Object exists independently.

   Its image is dependent upon view and other things such as light.

2. In CG, an image is composed of “polys” — usually triangles.

5 Vision

1. Human vision: rods (night) and cones (day).

   Visual acuity: resolution.

   Three types of cones, each most sensitive to a particular light frequency (blue, green, yellow).

   Response is non-linear for one type of cone and non-uniform between cones.

2. CG uses three color system of linear combinations of monochromatic red, green, blue.

6 Pinhole and Synthetic Cameras

Consider the project of an object of height $h$ onto the focal plane of a pinhole camera:
1. What's the height of the image \((h')\)?

2. What's the angle of the field of view?

3. Depth of field.

4. CG’s synthetic camera adds a clipping rectangle to this basic set-up.

5. Independence of objects and camera.

7 Graphics Pipeline

1. 3-D points represented by four element vectors.

2. Transformations applied by multiplications by series of \(4 \times 4\) matrices.

3. Stages:

   (a) Transformer: rotate, shift, scale. Convert world coordinates to window coordinates.

   (b) Clipper.

   (c) Projector: 3-D to 2-D. Ortho, perspective views.

   (d) Rasterizer.