CS4620/5620: Lecture 6

Ray Tracing Basics

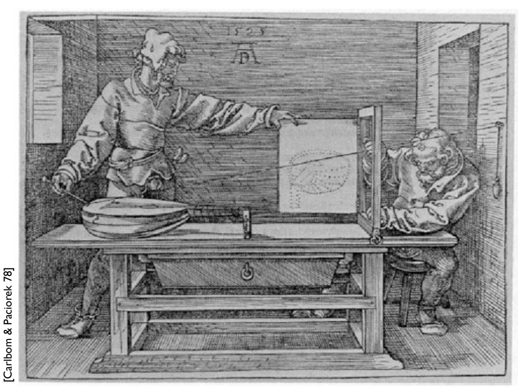
Announcements

- HW 1 is out
  - Due Sep 16, 9am, online on CMS
  - Work alone
  - Write and scan, or type, or solve using Matlab, but then write out what you did in full detail
- PA1 will be out this Friday
- Check schedule online for plan for semester
  - PA, HW, PPA
- Perspective

Ray generation vs. projection

- Viewing in ray tracing
  - start with image point
  - compute 3D point that projects to that point using ray
  - do this using geometry
- Viewing by projection
  - start with 3D point
  - compute image point that it projects to
  - do this using transforms
- Inverse processes

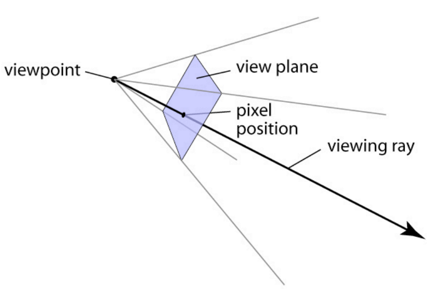
Plane projection in drawing



[Carlbom & Paciorek 78]

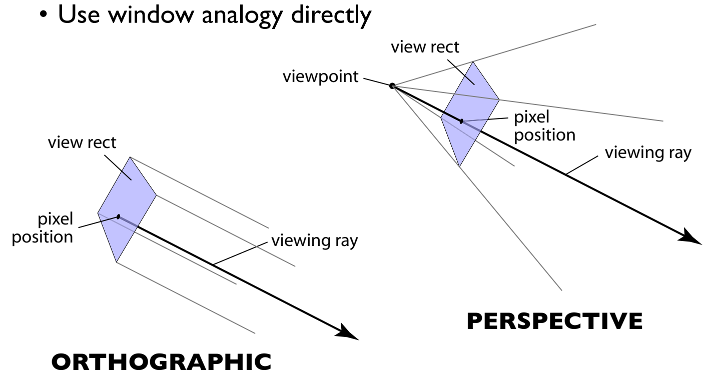
Generating eye rays

- Use window analogy directly



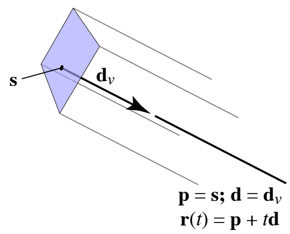
Generating eye rays

- Use window analogy directly



Generating eye rays—orthographic

- Just need to compute the view plane point **s**:

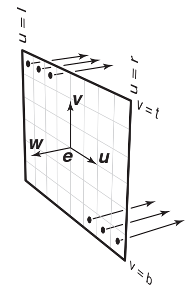


  – but where exactly is the view rectangle?

Generating eye rays—orthographic

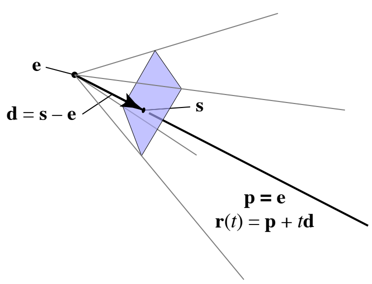
- Positioning the view rectangle
  – establish three vectors to be *camera basis:* **u**, **v**, **w**
  – view rectangle is in **u**–**v** plane, specified by l, r, t, b
  – now ray generation is easy:

$$\mathbf{s} = \mathbf{e} + u\mathbf{u} + v\mathbf{v}$$
$$\mathbf{p} = \mathbf{s};\ \mathbf{d} = -\mathbf{w}$$
$$\mathbf{r}(t) = \mathbf{p} + t\mathbf{d}$$



Generating eye rays—perspective

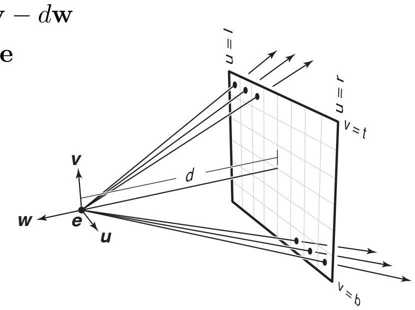
- View rectangle needs to be away from viewpoint
- Distance is important: "focal length" of camera
  – still use camera frame but position view rect away from viewpoint
  – ray origin always **e**
  – ray direction now controlled by **s**



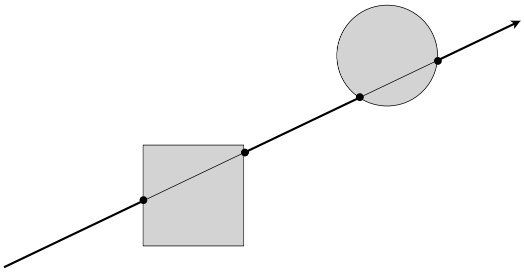
Generating eye rays—perspective

- Compute **s** in the same way; just subtract *d***w**
  – coordinates of **s** are (u, v, $-d$)

$$\mathbf{s} = \mathbf{e} + u\mathbf{u} + v\mathbf{v} - d\mathbf{w}$$
$$\mathbf{p} = \mathbf{e};\ \mathbf{d} = \mathbf{s} - \mathbf{e}$$
$$\mathbf{r}(t) = \mathbf{p} + t\mathbf{d}$$

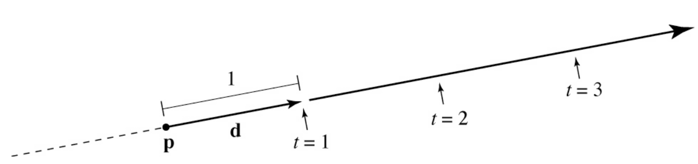


Ray intersection



Ray: a half line

- Standard representation: point **p** and direction **d**
  $\mathbf{r}(t) = \mathbf{p} + t\mathbf{d}$
  – this is a *parametric equation* for the line
  – lets us directly generate the points on the line
  – if we restrict to $t > 0$ then we have a ray
  – note replacing **d** with a**d** doesn't change ray ($a > 0$)



Ray-sphere intersection: algebraic

- Condition 1: point is on ray

  $$\mathbf{r}(t) = \mathbf{p} + t\mathbf{d}$$

- Condition 2: point is on sphere
  - assume unit sphere; see Shirley or notes for general

  $$\|\mathbf{x}\| = 1 \Leftrightarrow \|\mathbf{x}\|^2 = 1$$
  $$f(\mathbf{x}) = \mathbf{x} \cdot \mathbf{x} - 1 = 0$$

- Substitute:

  $$(\mathbf{p} + t\mathbf{d}) \cdot (\mathbf{p} + t\mathbf{d}) - 1 = 0$$

  - this is a quadratic equation in *t*

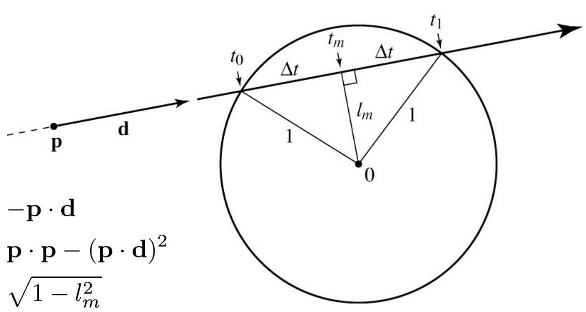
Ray-sphere intersection: algebraic

- Solution for *t* by quadratic formula:

$$t = \frac{-\mathbf{d} \cdot \mathbf{p} \pm \sqrt{(\mathbf{d} \cdot \mathbf{p})^2 - (\mathbf{d} \cdot \mathbf{d})(\mathbf{p} \cdot \mathbf{p} - 1)}}{\mathbf{d} \cdot \mathbf{d}}$$
$$t = -\mathbf{d} \cdot \mathbf{p} \pm \sqrt{(\mathbf{d} \cdot \mathbf{p})^2 - \mathbf{p} \cdot \mathbf{p} + 1}$$

  - simpler form holds when **d** is a unit vector
    but we won't assume this in practice (reason later)
  - discriminant intuition?
  - use the unit-vector form to make the geometric interpretation

Ray-sphere intersection: geometric



$$t_m = -\mathbf{p} \cdot \mathbf{d}$$
$$l_m^2 = \mathbf{p} \cdot \mathbf{p} - (\mathbf{p} \cdot \mathbf{d})^2$$
$$\Delta t = \sqrt{1 - l_m^2}$$
$$= \sqrt{(\mathbf{p} \cdot \mathbf{d})^2 - \mathbf{p} \cdot \mathbf{p} + 1}$$
$$t_{0,1} = t_m \pm \Delta t = -\mathbf{p} \cdot \mathbf{d} \pm \sqrt{(\mathbf{p} \cdot \mathbf{d})^2 - \mathbf{p} \cdot \mathbf{p} + 1}$$

Normal for sphere