Cameras and Triangles

CS6965 Fall 2011

Camera models

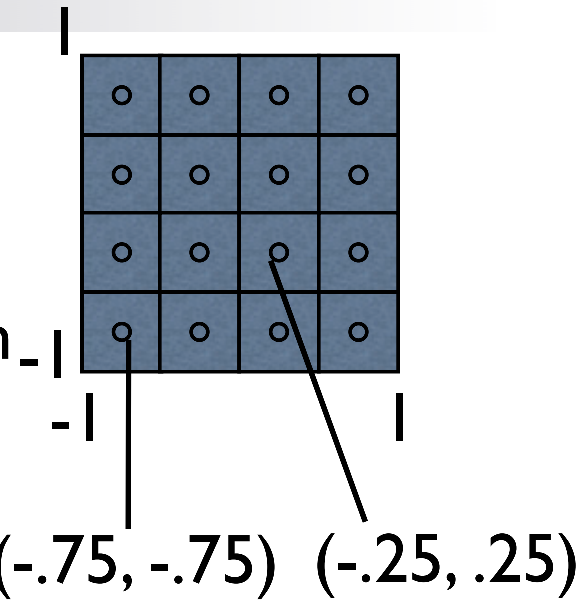
- The camera maps pixels to rays
- What kind of camera models might we want?

Camera models

- Typical:
  - Orthographic
  - Pinhole (perspective)
- Advanced:
  - Depth of field (thin lens approximation)
  - Sophisticated lenses ("A realistic camera model for computer graphics," Kolh, Mitchell, Hanrahan)
  - Fish-eye lens
  - Arbitrary distortions

Camera models

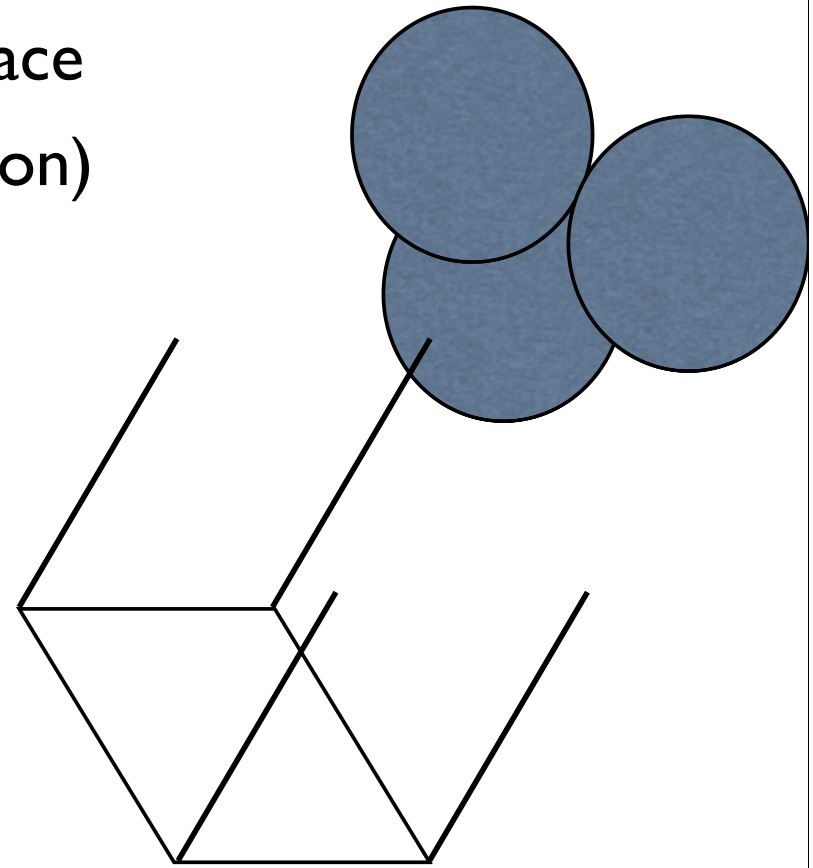
- Map pixel coordinates -1 to 1
- Pay careful attention to pixel centers
- Can be combined with ray generation



- Non-square images
  - Longest dimension is -1 to 1, shorter is smaller (still centered at 0)
  - Or camera knows about aspect ratio

Orthographic projection

- “Film” is just a rectangle in space
- Rays are parallel (same direction)



Orthographic projection

- Specify with center (P) and two vectors (u, v)

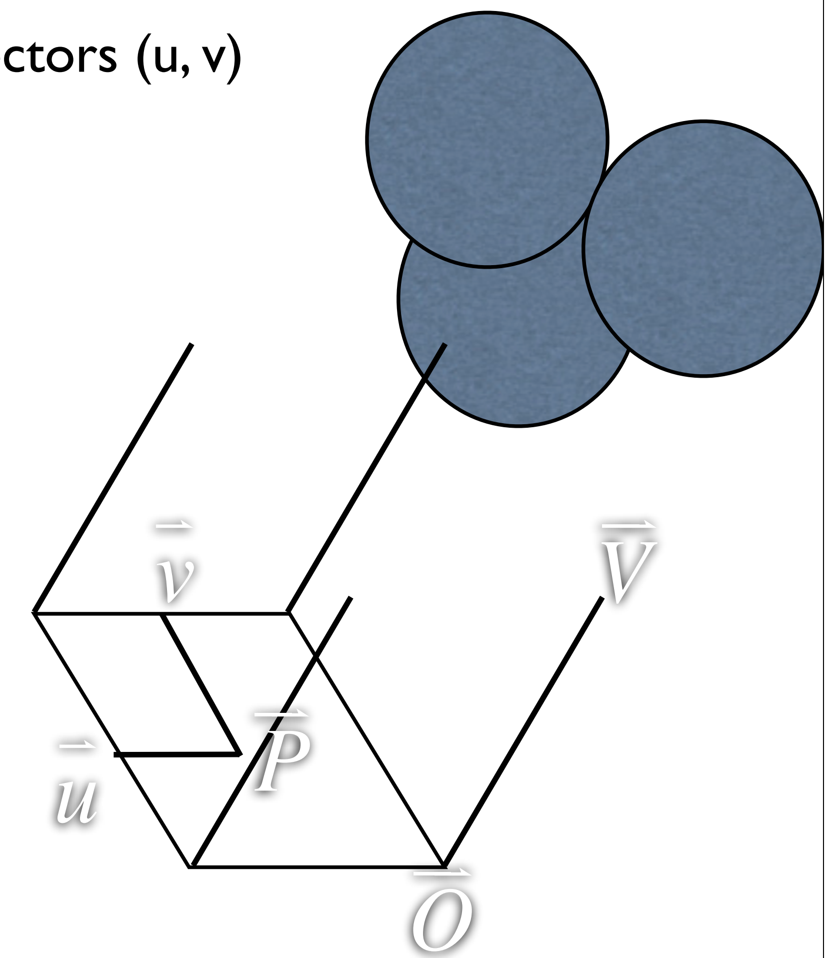
$$\vec{O} = \vec{P} + x\vec{u} + y\vec{v}$$

$$\vec{V} = \vec{u} \times \vec{v}$$

$$\|\vec{u}\|, \|\vec{v}\| : \text{ image size}$$

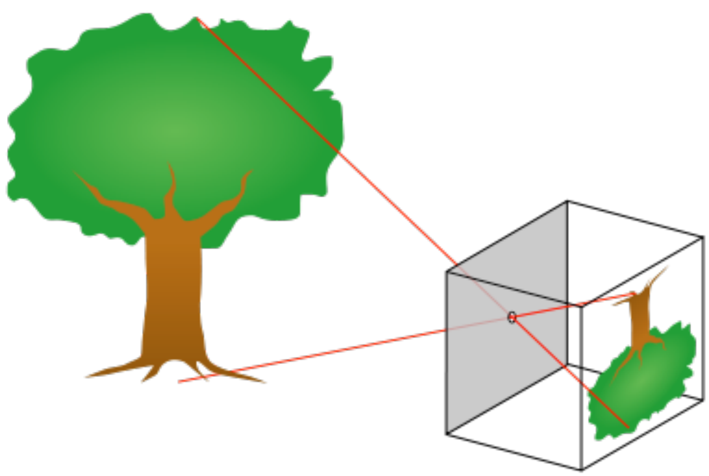
$$\frac{\|\vec{u}\|}{\|\vec{v}\|} = aspect\ ratio$$

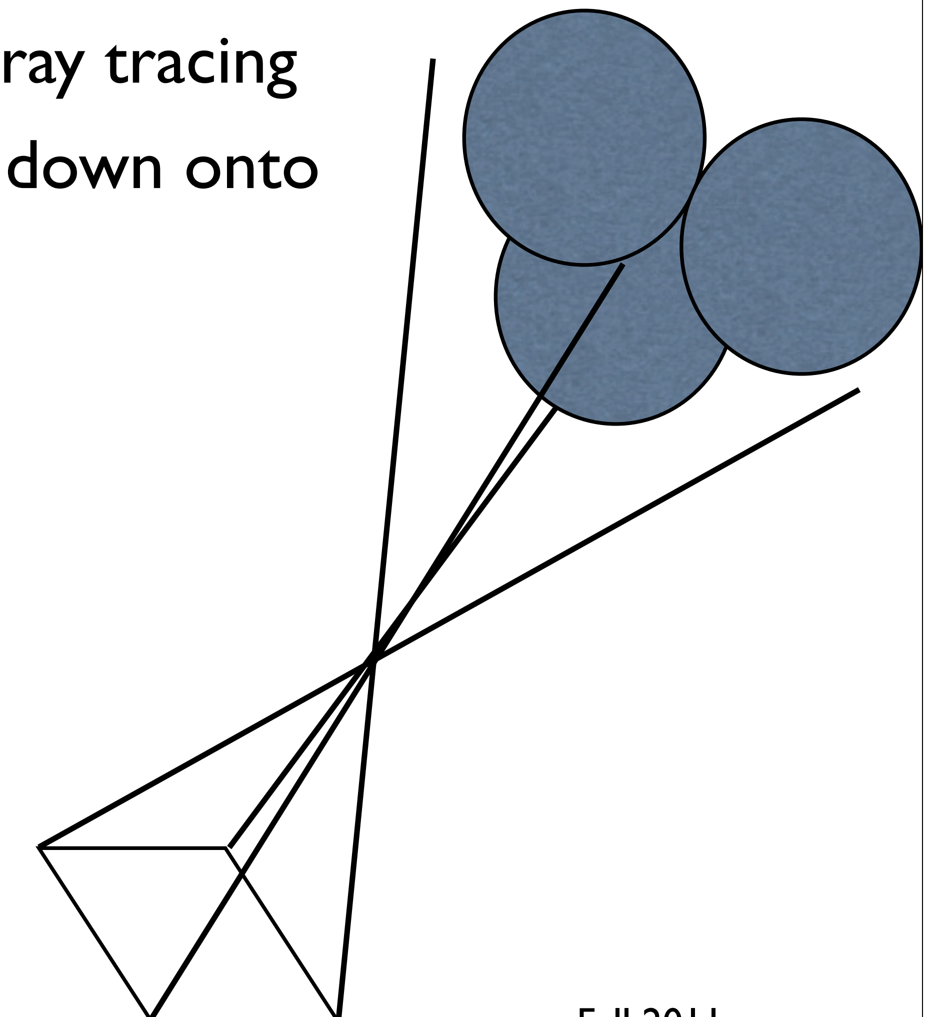
$$\text{square image: } \vec{u} \cdot \vec{v} = 0$$



Pinhole camera

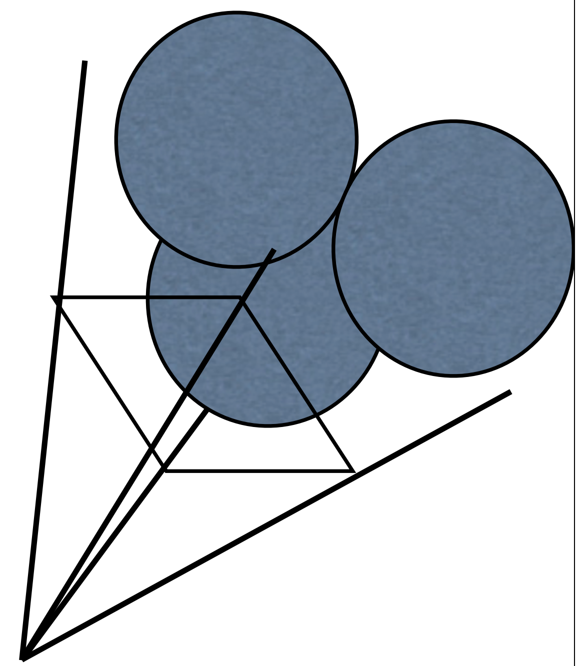
- Most common model for ray tracing
- Image is projected upside down onto image plane





Pinhole camera

- Easier to think about right side up
- Focal point is also called the eye point



Pinhole camera

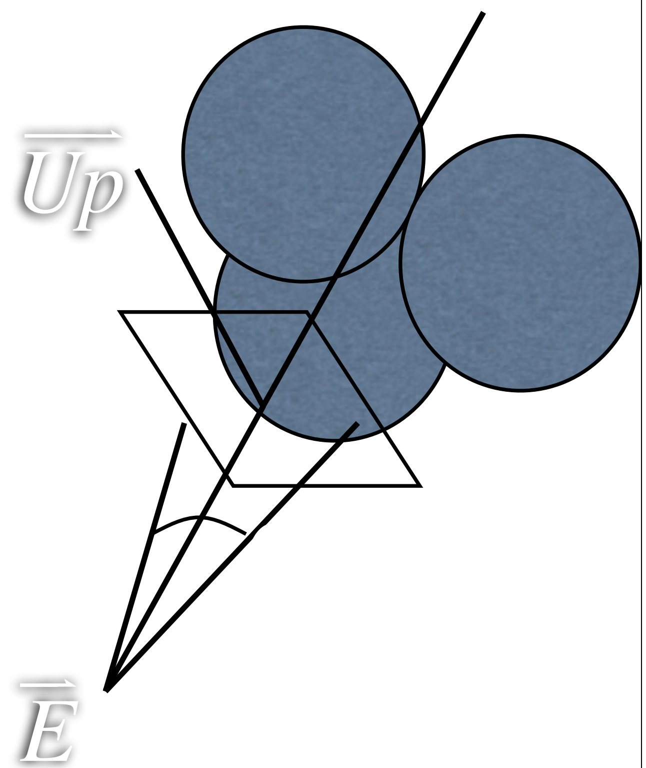
- Parameters:

$\vec{E}$: Eye point (focal point)

$\vec{C}$: Lookat point

$\overrightarrow{Up}$: Up vector

θ: Field of view



Pinhole camera

- Construction:

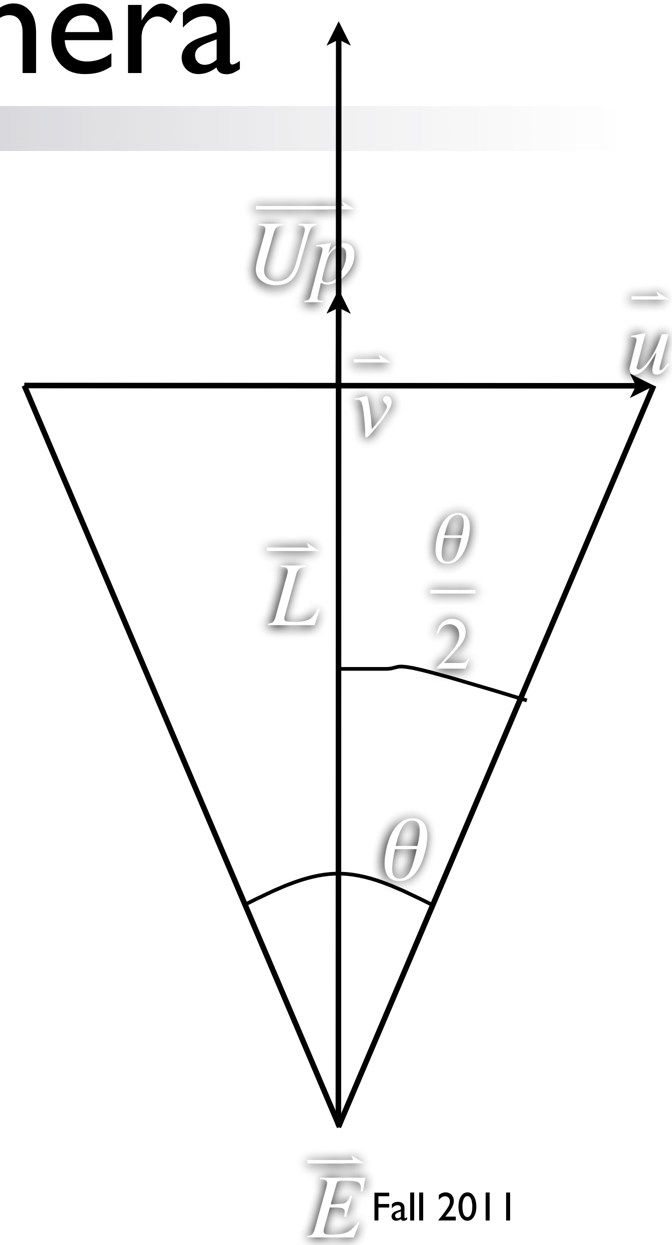
$\vec{L} = \vec{C} - \vec{E}$ (look or gaze direction)

$$\overrightarrow{L_n} = \frac{\vec{L}}{\|L\|}$$

$$\overrightarrow{u_{tmp}} = \overrightarrow{L_n} \times \overrightarrow{Up}$$

$$\overrightarrow{v_{tmp}} = \overrightarrow{u_{tmp}} \times \overrightarrow{L_n}$$

Top View



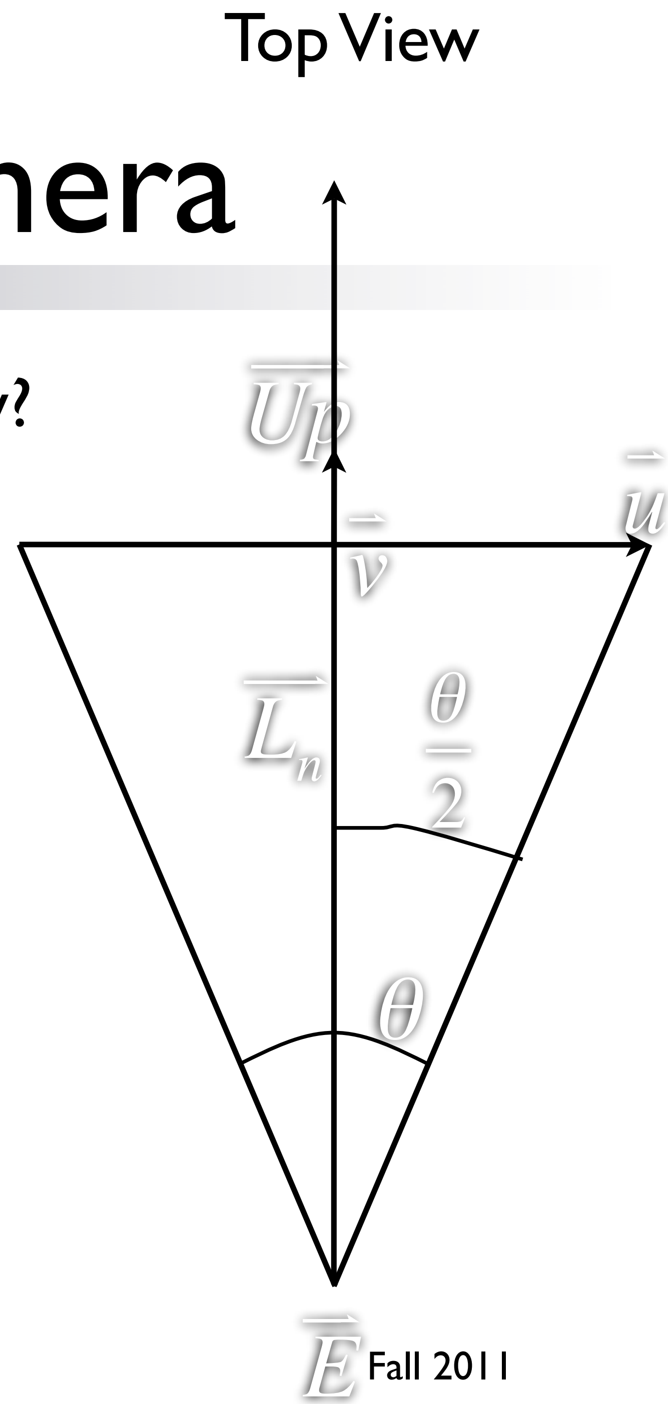
Pinhole camera

- How do we get the lengths of u/v?

$$\tan\frac{\theta}{2} = \frac{\left\|\vec{u}\right\|}{\left\|\overrightarrow{L_n}\right\|}$$

$$\left\|\vec{u}\right\| = \tan\frac{\theta}{2}$$

$$\vec{u} = \frac{\overrightarrow{u_{tmp}}}{\left\|\overrightarrow{u_{tmp}}\right\|}\tan\frac{\theta}{2}$$



Pinhole camera

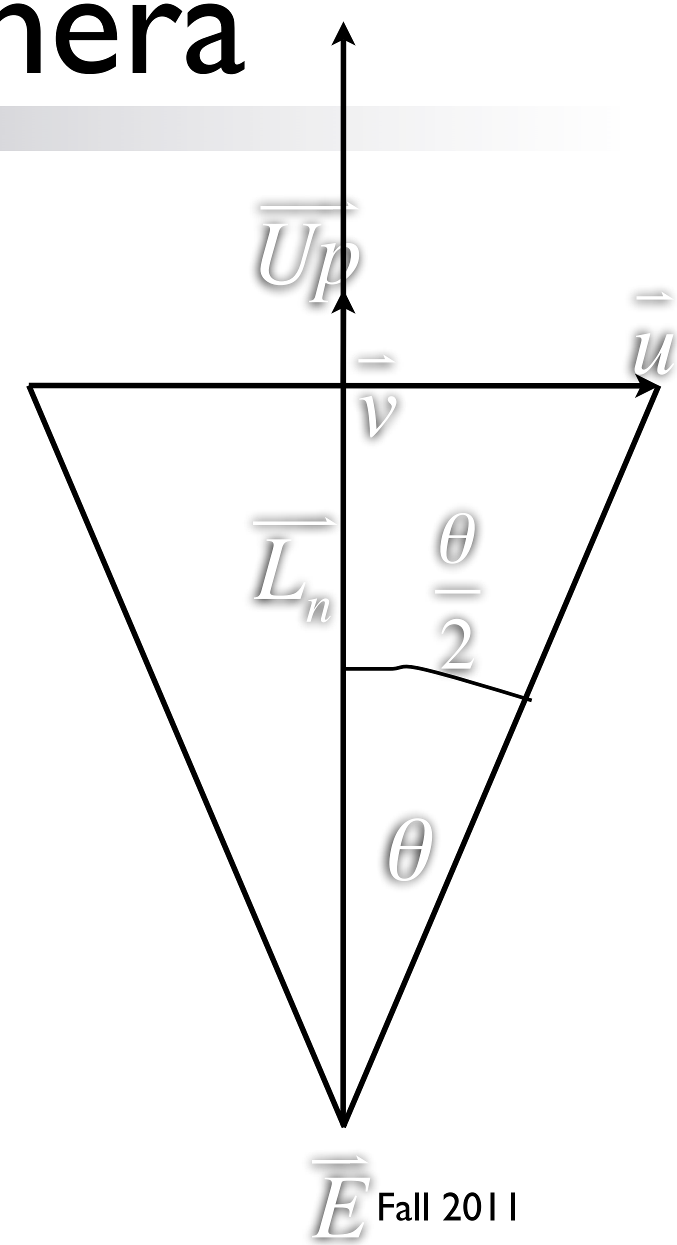
- What about v?

$$\text{aspect ratio} = a = \frac{\|\vec{u}\|}{\|\vec{v}\|}$$

$$\|\vec{u}\| = \tan\frac{\theta}{2}$$

$$\|\vec{v}\| = \frac{\tan\frac{\theta}{2}}{a}$$

$$\vec{v} = \frac{\overrightarrow{v_{tmp}}}{\|\overrightarrow{v_{tmp}}\|}\frac{\tan\frac{\theta}{2}}{a}$$

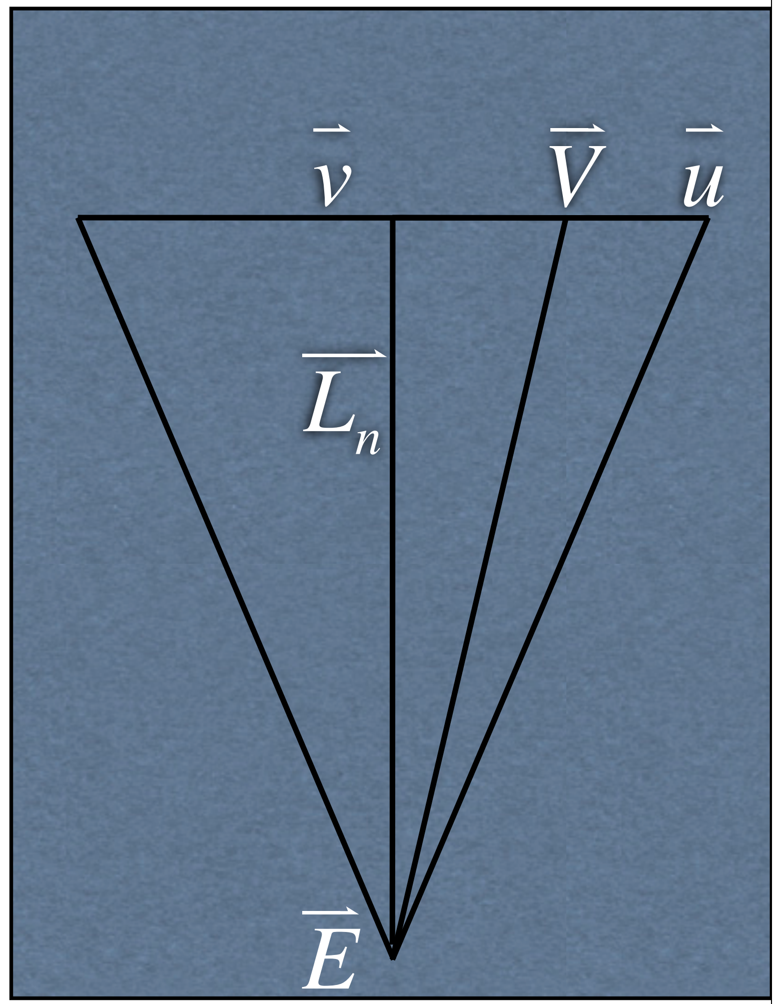


Pinhole camera

- Finally

$$\vec{O} = \vec{E}$$

$$\vec{V} = \vec{L_n} + x\vec{u} + y\vec{v}$$



Camera Implementation

- ulen is computed using a tan()
  - Just pass in the value (precompute)
  - ulen = tan(hfov/2.*PI/180.); // for degrees
- Normalize ray directions

Data structures: Camera

- Base class:
  - Method to compute ray for image x,y [-1,1]
- Pinhole camera:
  - Position
  - Lookat point or gaze direction
  - Up vector
  - Aspect ratio
  - Derived values: u, v, normalized gaze dir

Camera

- void makeRay(Ray& ray, float x, float y) const;

Group

- Iterate over all objects, calling intersect
- For now we can stick with what we have (list)
- More, better groups later

HitRecord

- bool hit(float t, const Primitive& hit_prim, const Material& hit_matl)
- Use like:
- hitrecord.hit(t, this, mymatl)
- Bool return value for optimizations (and more uses later)

Ray-plane intersection

- To find the intersection of a ray with a plane, determine where both equations are satisfied at the same time:

$$\vec{N} \bullet \vec{P} + d = 0 \text{ and } \vec{P} = \vec{O} + t\vec{V}$$

$$\vec{N} \bullet \left(\vec{O} + t\vec{V} \right) + d = 0$$

$$\vec{N} \bullet \vec{O} + t\vec{N} \bullet \vec{V} + d = 0$$

$$t\vec{N} \bullet \vec{V} = -\left(d + \vec{N} \bullet \vec{O} \right)$$

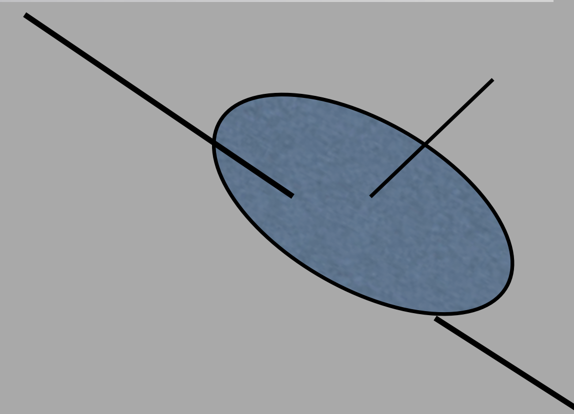
$$t = -\frac{\left(d + \vec{N} \bullet \vec{O} \right)}{\vec{N} \bullet \vec{V}}$$

Ray-plane intersection

- If $\vec{N} \bullet \vec{V} = 0$ (or close to it) then the ray is parallel to the plane
- The parameter t defines the point where the ray intersects the plane
- To determine the point of intersection, just plug it back into the ray equation

$$\left(\bar{O} + t\vec{V}\right)$$

Ray-disk intersection



Center: $\vec{C}$

Radius: r

Normal: $\vec{N}$

Plane equation: $\vec{N} \cdot \vec{P} - d = 0$

$d = \vec{N} \cdot \vec{C}$

intersect plane with ray: $t = \dfrac{d - \vec{O} \cdot \vec{N}}{\vec{V} \cdot \vec{N}}$

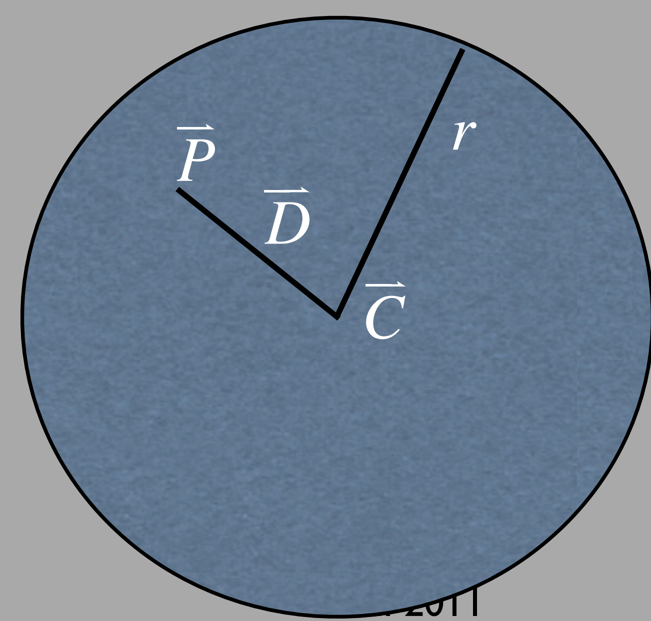
Determine if point is inside circle:

$\vec{P} = \vec{O} + t\vec{V}$

$\vec{D} = \vec{P} - \vec{C}$

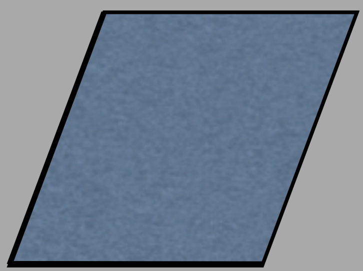
$if(\vec{D} \cdot \vec{D} < r^2)$

hit



Other planar objects

- A number of other planar object can be handled in the same way:
  - Ring (two circles)
  - Parallelogram (specified with point and 2 vectors)
  - Triangle



Triangle

- Several ways to do ray-triangle intersections:
  - Similar to disc (Dot products to perform inside/outside test)
  - Similar to disc (Project point to XY, XZ or YZ plane for inside/outside test)
  - Plücker coordinates
  - Barycentric coordinates

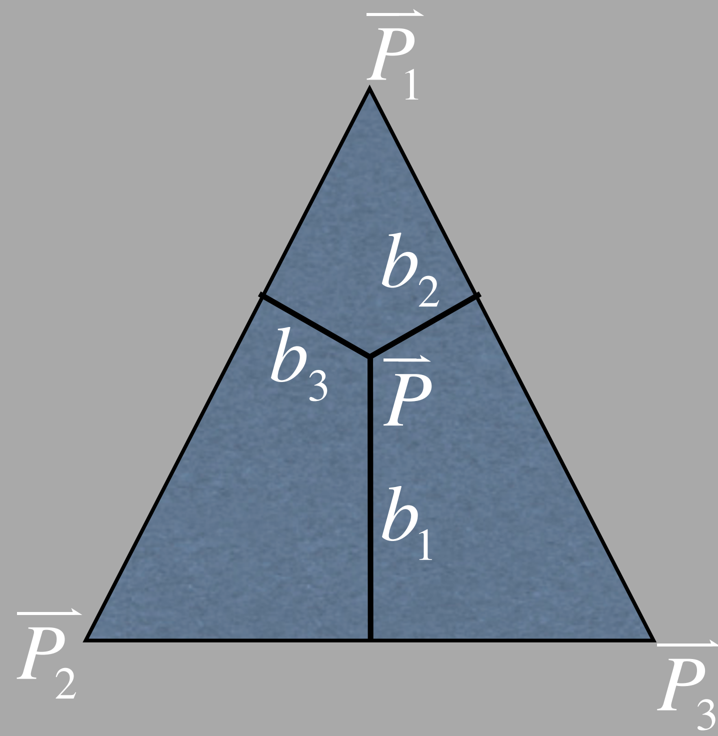
Barycentric coordinates

$$0 \le b_1, b_2, b_3 \le 1$$

$$b_1 + b_2 + b_3 = 1$$

$$\vec{P} = b_1 \vec{P_1} + b_2 \vec{P_2} + b_3 \vec{P_3}$$

$$= b_1 \vec{P_1} + b_2 \vec{P_2} + (1 - b_1 - b_2) \vec{P_3}$$



Barycentric coordinates

$$0 \le b_1, b_2, b_3 \le 1$$

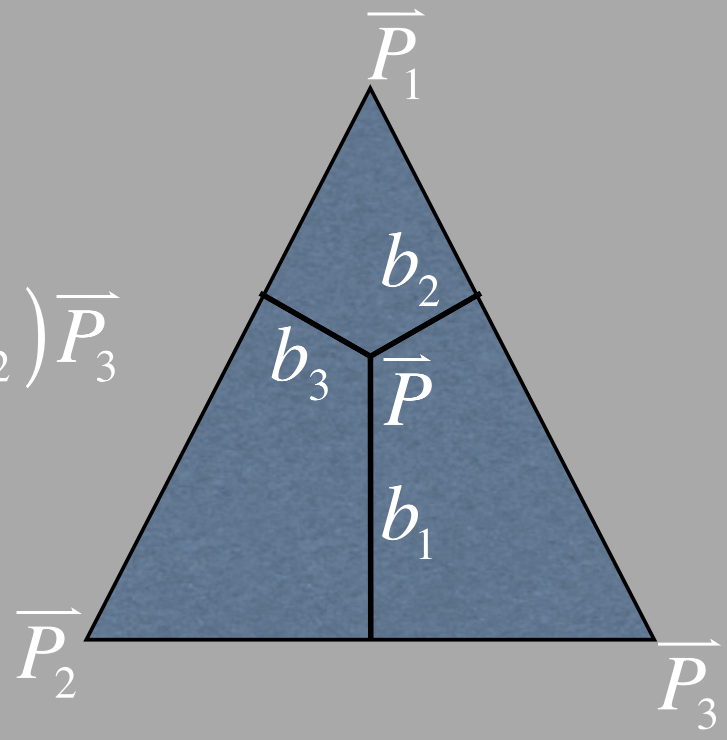
$$b_1 + b_2 + b_3 = 1$$

$$\vec{P} = b_1\vec{P_1} + b_2\vec{P_2} + b_3\vec{P_3}$$

$$= b_1\vec{P_1} + b_2\vec{P_2} + \left(1 - b_1 - b_2\right)\vec{P_3}$$

$$\vec{O} + t\vec{V} = b_1\vec{P_1} + b_2\vec{P_2} + \left(1 - b_1 - b_2\right)\vec{P_3}$$

(3 equations, 3 unknowns)



Ray-triangle intersection

$$\overrightarrow{O} + t\overrightarrow{V} = b_1\overrightarrow{P_1} + b_2\overrightarrow{P_2} + (1 - b_1 - b_2)\overrightarrow{P_3}$$

$$-t\overrightarrow{V} + b_1\left(\overrightarrow{P_1} - \overrightarrow{P_3}\right) + b_2\left(\overrightarrow{P_2} - \overrightarrow{P_3}\right) = \left(\overrightarrow{O} - \overrightarrow{P_3}\right)$$

$$\overrightarrow{\mathrm{e}_1} = \left(\overrightarrow{P_1} - \overrightarrow{P_3}\right)$$

$$\overrightarrow{e_2} = \left(\overrightarrow{P_2} - \overrightarrow{P_3}\right)$$

$$\vec{s} = \left(\overrightarrow{O} - \overrightarrow{P_3}\right)$$

$$\begin{bmatrix} -V_x & e_{1x} & e_{2x} \\ -V_y & e_{1y} & e_{2y} \\ -V_z & e_{1z} & e_{2z} \end{bmatrix} \begin{bmatrix} t \\ b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix}$$

Use Cramer's rule to solve system

Solution

$$t = \frac{\begin{vmatrix} s_x & e_{1x} & e_{2x} \\ s_y & e_{1y} & e_{2y} \\ s_z & e_{1z} & e_{2z} \end{vmatrix}}{\begin{vmatrix} -V_x & e_{1x} & e_{2x} \\ -V_y & e_{1y} & e_{2y} \\ -V_z & e_{1z} & e_{2z} \end{vmatrix}}, b_1 = \frac{\begin{vmatrix} -V_x & s_x & e_{2x} \\ -V_y & s_y & e_{2y} \\ -V_z & s_z & e_{2z} \end{vmatrix}}{\begin{vmatrix} -V_x & e_{1x} & e_{2x} \\ -V_y & e_{1y} & e_{2y} \\ -V_z & e_{1z} & e_{2z} \end{vmatrix}}, b_2 = \frac{\begin{vmatrix} -V_x & e_{1x} & s_x \\ -V_y & e_{1y} & s_y \\ -V_z & e_{1z} & s_z \end{vmatrix}}{\begin{vmatrix} -V_x & e_{1x} & e_{2x} \\ -V_y & e_{1y} & e_{2y} \\ -V_z & e_{1z} & e_{2z} \end{vmatrix}}$$

We will also use this property:

$$\left|A^T\right| = |A|$$

Remember this slide?

- Scalar triple product
- $A \cdot (B \times C) = B \cdot (C \times A) = C \cdot (A \times B)$

- Can also be written as the determinant:
- | Ax Ay Az |
- | Bx By Bz |
- | Cx Cy Cz |

- This one is pretty cool, but we won't use it often.

Improved solution

$$denom = \begin{vmatrix} -V_x & -V_y & -V_z \\ e_{1x} & e_{1y} & e_{1z} \\ e_{2x} & e_{2y} & e_{2z} \end{vmatrix}$$

$$= -\overrightarrow{V} \cdot \left(\overrightarrow{e_1} \times \overrightarrow{e_2}\right) = -\overrightarrow{e_1} \cdot \left(\overrightarrow{e_2} \times \overrightarrow{V}\right) = \overrightarrow{e_1} \cdot \left(\overrightarrow{V} \times \overrightarrow{e_2}\right)$$

$$denom = \overrightarrow{e_1} \cdot \left(\overrightarrow{V} \times \overrightarrow{e_2}\right)$$

$$t = \frac{e_2 \cdot \left(\vec{s} \times \overrightarrow{e_1}\right)}{denom}$$

$$b_1 = \frac{\vec{s} \cdot \left(\overrightarrow{V} \times \overrightarrow{e_2}\right)}{denom}$$

$$b_2 = \frac{\overrightarrow{V} \cdot \left(\vec{s} \times \overrightarrow{e_1}\right)}{denom}$$

Ray-triangle intersection

$\overrightarrow{e_1} = \overrightarrow{p_1} - \overrightarrow{p_3}$

$\overrightarrow{e_2} = \overrightarrow{p_2} - \overrightarrow{p_3}$

$\overrightarrow{r_1} = \overrightarrow{V} \times \overrightarrow{e_2}$

$denom = \overrightarrow{e_1} \cdot \overrightarrow{r_1}$

$if\,(Abs(denom) < epsilon)$ miss, return;

$$invDenom = \frac{1}{denom}$$

$\vec{s} = \overrightarrow{O} - \overrightarrow{p_3}$

$b_1 = \left(\vec{s} \cdot \overrightarrow{r_1}\right) invDenom$

$if\,(b_1 < 0 \,\|\, b_1 > 1)$ miss, return;

$\overrightarrow{\mathrm{r}_2} = \vec{s} \times \overrightarrow{e_1}$

$b_2 = \left(\overrightarrow{V} \cdot \overrightarrow{r_2}\right) invDenom$

$if\,(b_2 < 0 \,\|\, \mathrm{b}_1 + b_2 > 1)$ miss, return;

$t = \left(\overrightarrow{e_2} \cdot \overrightarrow{r_2}\right) invDenom$

hit, save b1/b2 for normal/texture interpolation

	Add / sub / mult	*Compare*	*Divide*	$\sqrt{\ }$
$\vec{e_1} = \vec{p_1} - \vec{p_3}$	3			
$\vec{e_2} = \vec{p_2} - \vec{p_3}$	3			
$\vec{r_1} = \vec{V} \times \vec{e_2}$	9			
$denom = \vec{e_1} \cdot \vec{r_1}$	5			
$if\,(Abs(denom) < epsilon)$ miss, return;		2		
$invDenom = \frac{1}{denom}$			1	
$\vec{s} = \vec{O} - \vec{p_3}$	3			
$b_1 = \left(\vec{s} \cdot \vec{r_1}\right) invDenom$	6			
$if\,(b_1 < 0 \parallel b_1 > 1)$ miss, return;		2		
$\vec{r_2} = \vec{s} \times \vec{e_1}$	9			
$b_2 = \left(\vec{V} \cdot \vec{r_2}\right) invDenom$	6			
$if\,(b_2 < 0 \parallel b_1 + b_2 > 1)$ miss, return;	1	2		
$t = \left(\vec{e_2} \cdot \vec{r_2}\right) invDenom$	6			
hit, save b1/b2 for normal/texture interpolation		2		
total	20 / 29 / 45 / 51	2 / 4 / 6 / 8	0 / 1 / 1 / 1	0 / 0 / 0 / 0

Normal interpolation

true normal: $\overrightarrow{e_1} \times \overrightarrow{e_2}$

Smooth shading:

normal specified at each vertex:

$$\overrightarrow{\mathrm{N}} = b_1 \overrightarrow{N_1} + b_2 \overrightarrow{N_2} + \left(1 - b_1 - b_2\right) \overrightarrow{N_3}$$

HitRecord

- Add a scratchpad to your hitrecord:
- char data[MAXSIZE];
- template<typename T> getScratchpad() {
-     assert(sizeof(T) <= MAXSIZE);
-     return reinterpret_cast<T*>(data);
- }

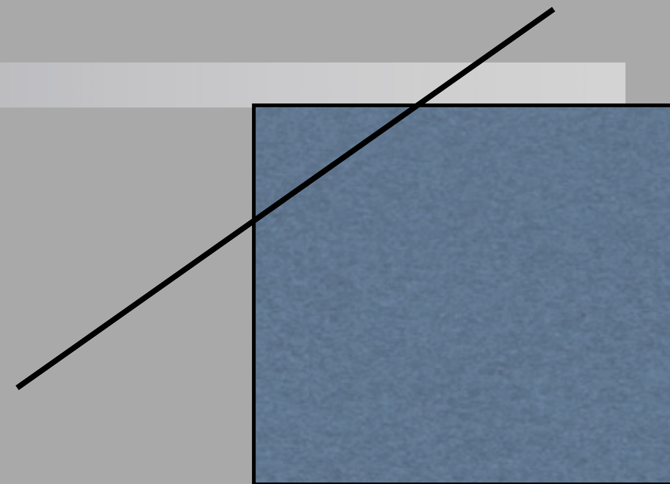
Scratchpad

- Use it like this:
- struct TriangleCoords{double b1, b2};
- if(hitrecord.hit(t, this, matl)){
-     // I am now the closest
-     TriangleCoords* tsave
-             =hitrecord.getScratchpad<TriangleCoords>();
-     tsave->b1=b1;
-     tsave->b2=b2;
- }

Normal computation

```
Triangle::normal(…)
{
 TriangleCoords* tsave
        =hitrecord.getScratchpad<TriangleCoords>();
  N=tsave->b1*N1+tsave->b2*N2
      +(1-tsave->b1-tsave->b2)*N3;
  N.normalize();
  return N;
}
```

Boxes



- Axis aligned boxes
- Parallelepiped

- 12 triangles?
- 6 planes with squares?

Ray-box intersection

$$\overrightarrow{N} \cdot \overrightarrow{P} - d = 0$$

$$t = \frac{d - \overrightarrow{N} \cdot \overrightarrow{O}}{\overrightarrow{N} \cdot \overrightarrow{V}}$$

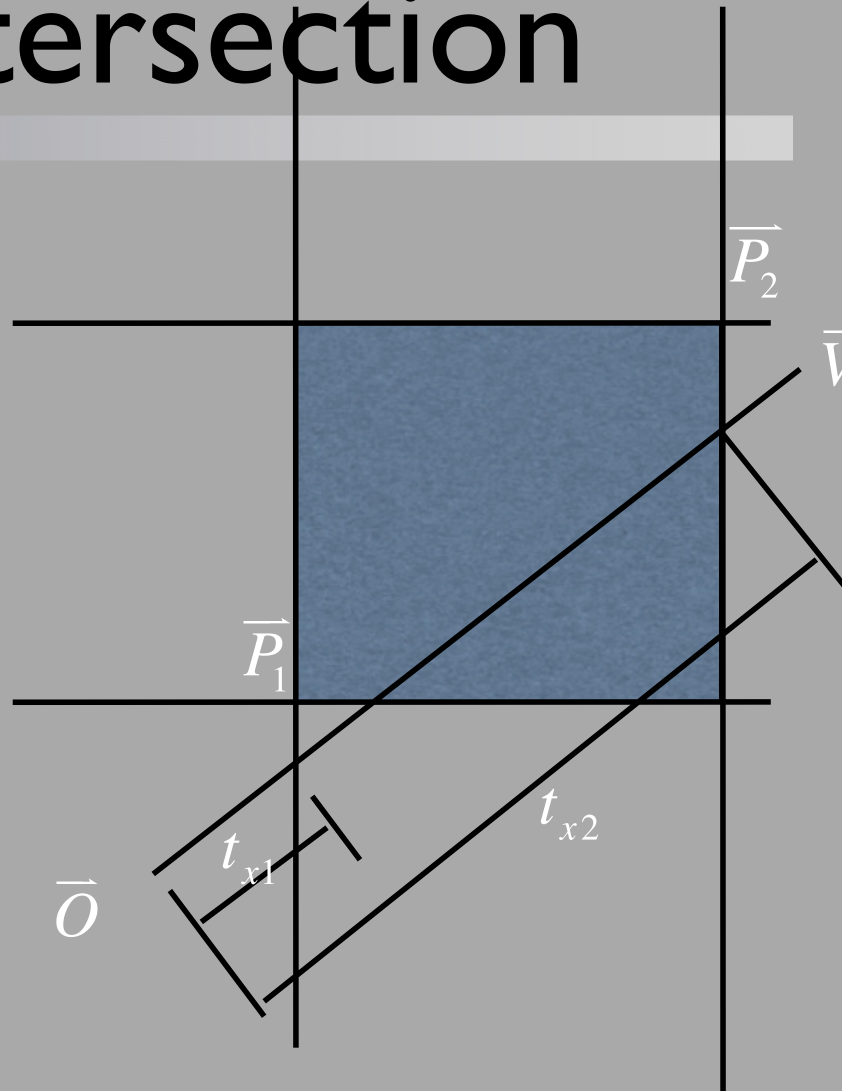
$$\text{x plane: } \overrightarrow{\mathrm{N}} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$$t = \frac{d - O_x}{V_x}$$

$$d_1 = P_{1x}, d_2 = P_{2x}$$

$$t_{x1} = \frac{P_{1x} - O_x}{V_x}, t_{x2} = \frac{P_{2x} - O_x}{V_x}$$

Same for y, z planes



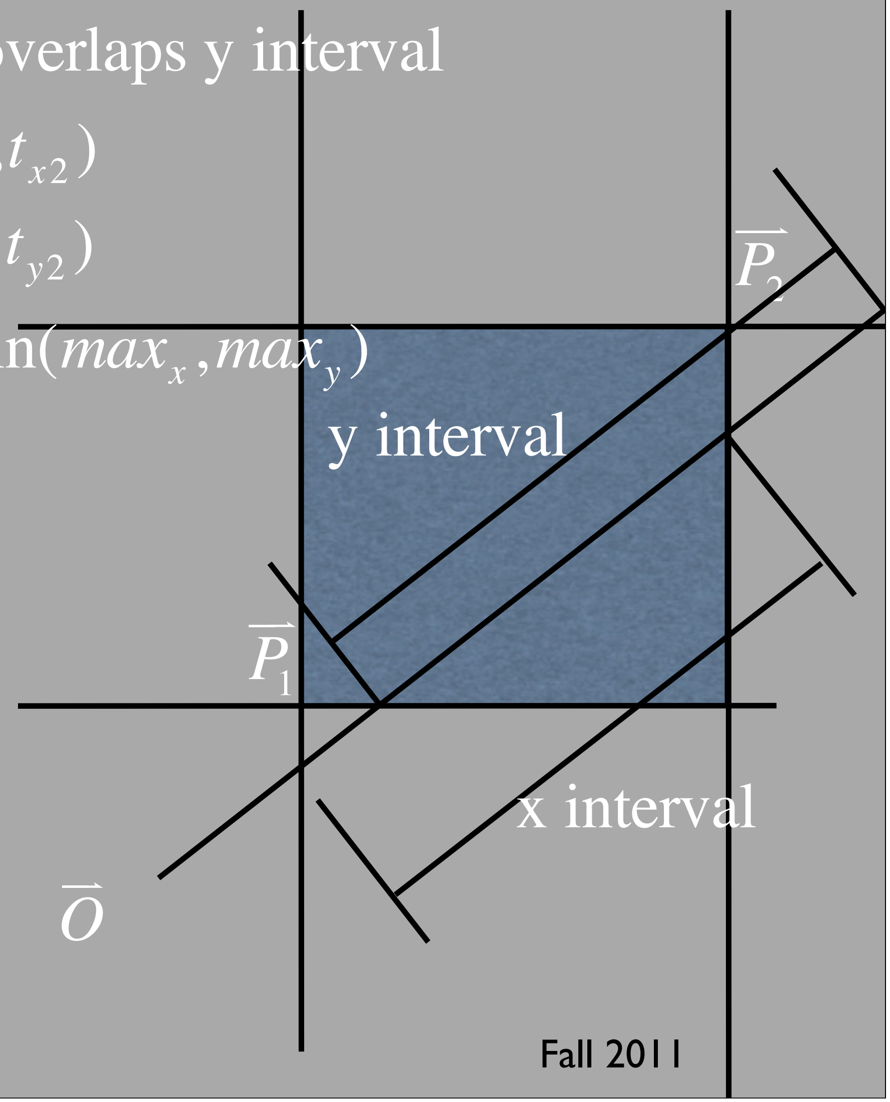
Intersection of intervals

Intersection occurs where x interval overlaps y interval

x interval: $\min(t_{x1}, t_{x2}) \le t \le \max(t_{x1}, t_{x2})$

y interval: $\min(t_{y1}, t_{y2}) \le t \le \max(t_{y1}, t_{y2})$

intersection: $\max(min_x, min_y) \le t \le \min(max_x, max_y)$



Better box

- http://www.cs.utah.edu/~awilliam/box/

Other cool intersections

- Extrusions
- Surfaces of revolution
- Swept spheres (Center, Radius vary)
- Metaballs
- Isosurface
- Torus
- Cones/Cylinders
- Superquadric
- Spline surfaces

Improved Sphere Intersection

Ray-sphere intersection

Ray: $\vec{P} = \vec{O} + t\vec{V}$

Sphere: $\left(\vec{P} - \vec{C}\right) \cdot \left(\vec{P} - \vec{C}\right)$

Solution: $t^2 \vec{V} \cdot \vec{V} + 2t\left(\vec{O} - \vec{C}\right) \cdot \vec{V} + \left(\vec{O} - \vec{C}\right) \cdot \left(\vec{O} - \vec{C}\right) - r^2$

Vector $\overrightarrow{\mathrm{OC}} = \vec{O} - \vec{C}$

$a = \vec{V} \cdot \vec{V}$

$b = 2\overrightarrow{OC} \cdot \vec{V}$

$c = \overrightarrow{OC} \cdot \overrightarrow{OC} - r^2$

$$roots: \quad \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Ray-sphere intersection (revisited)

$$\overrightarrow{\mathrm{OC}} = \vec{O} - \vec{C}$$

$$a = \vec{V} \cdot \vec{V}$$

$$b = 2\overrightarrow{OC} \cdot \vec{V}$$

$$c = \overrightarrow{OC} \cdot \overrightarrow{OC} - r^2$$

$$disc = b^2 - 4ac$$

$$if\,(disc > 0)$$

$$root = \sqrt{disc}$$

$$denom = 2a$$

$$t_1 = \frac{(-b + root)}{denom}$$

$$t_2 = \frac{(-b - root)}{denom}$$

Counting operations

Op	Add/Sub/Mult	Compare	Divide	$\sqrt{\ }$
$\overrightarrow{\mathrm{OC}} = \vec{O} - \vec{C}$	3			
$a = \vec{V} \cdot \vec{V}$	5			
$b = 2\overrightarrow{OC} \cdot \vec{V}$	6			
$c = \overrightarrow{OC} \cdot \overrightarrow{OC} - r^2$	7			
$disc = b^2 - 4ac$	5			
$if\,(disc > 0)$		1		
$root = \sqrt{disc}$				1
$denom = 2a$	1			
$t_1 = \frac{(-b + root)}{denom}$	2	2	1	
$t_2 = \frac{(-b - root)}{denom}$	1	2	1	
Total	26 / 30	1 / 5	0 / 2	0 / 1

First optimization

There are a lot of 2s in there - can we get rid of them?

Vector $\overrightarrow{OC} = \vec{O} - \vec{C}$

$a = \vec{V} \cdot \vec{V}$

$b' = \overrightarrow{OC} \cdot \vec{V}$

$b = 2\overrightarrow{OC} \cdot \vec{V} = 2b'$

$c = \overrightarrow{OC} \cdot \overrightarrow{OC} - r^2$

$$roots: \quad \frac{-2b' \pm \sqrt{4b'^2 - 4ac}}{2a} = \frac{-b' \pm \sqrt{b'^2 - ac}}{a}$$

Counting operations

Op	Add/Sub/Mult	Compare	Divide	$\sqrt{\ }$
$\overrightarrow{\mathrm{OC}} = \vec{O} - \vec{C}$	3			
$a = \vec{V} \cdot \vec{V}$	5			
$b' = \overrightarrow{OC} \cdot \vec{V}$	5			
$c = \overrightarrow{OC} \cdot \overrightarrow{OC} - r^2$	7			
$disc = b'^2 - ac$	4			
$if\,(disc > 0)$		1		
$root = \sqrt{disc}$				1
$denom = a$				
$t_1 = \dfrac{(-b' + root)}{denom}$	2	2	1	
$t_2 = \dfrac{(-b' - root)}{denom}$	1	2	1	
Total	24 / 27	1 / 5	0 / 2	0 / 1

Unit vectors

What if $\left\|\vec{\mathrm{V}}\right\| = 1$? (unit direction)

Vector $\overrightarrow{\mathrm{OC}} = \vec{O} - \vec{C}$

$a = \vec{V} \cdot \vec{V} = 1$

$b' = \overrightarrow{OC} \cdot \vec{V}$

$c = \overrightarrow{OC} \cdot \overrightarrow{OC} - r^2$

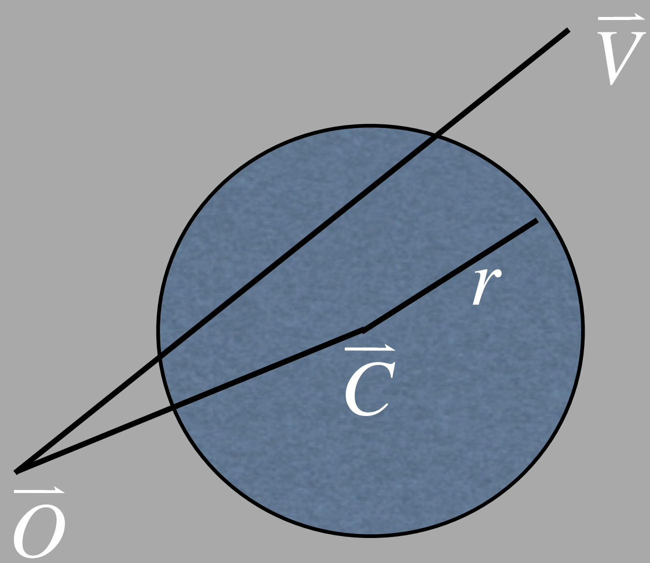
$roots: -b' \pm \sqrt{b'^2 - c}$

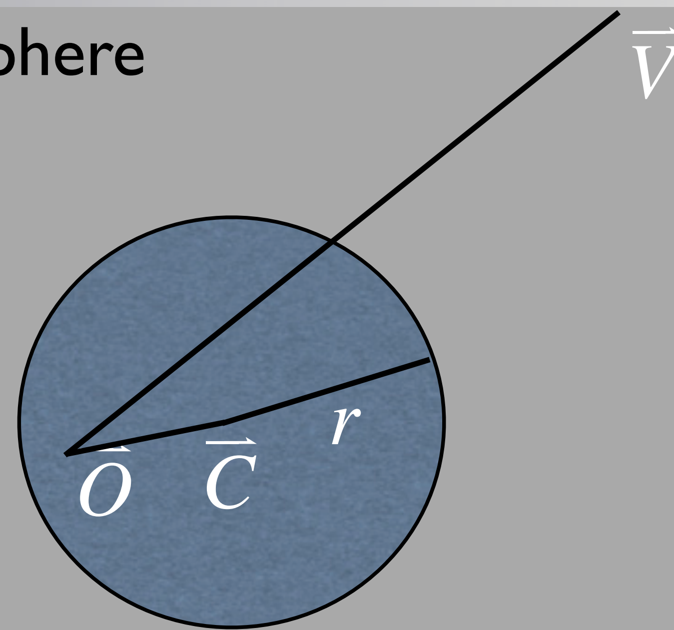
Counting operations

Op	Add/Sub/Mult	Compare	Divide	$\sqrt{\ }$
$\overrightarrow{OC} = \vec{O} - \vec{C}$	3			
$b' = \overrightarrow{OC} \cdot \vec{V}$	5			
$c = \overrightarrow{OC} \cdot \overrightarrow{OC} - r^2$	7			
$disc = b'^2 - c$	2			
$if\,(disc > 0)$		1		
$root = \sqrt{disc}$				1
$t_1 = \left(-b' + root\right)$	2	2		
$t_2 = \left(-b' - root\right)$	1	2		
Total	17 / 20	1 / 5	0	0 / 1

A different derivation

- Determine if ray is inside of sphere



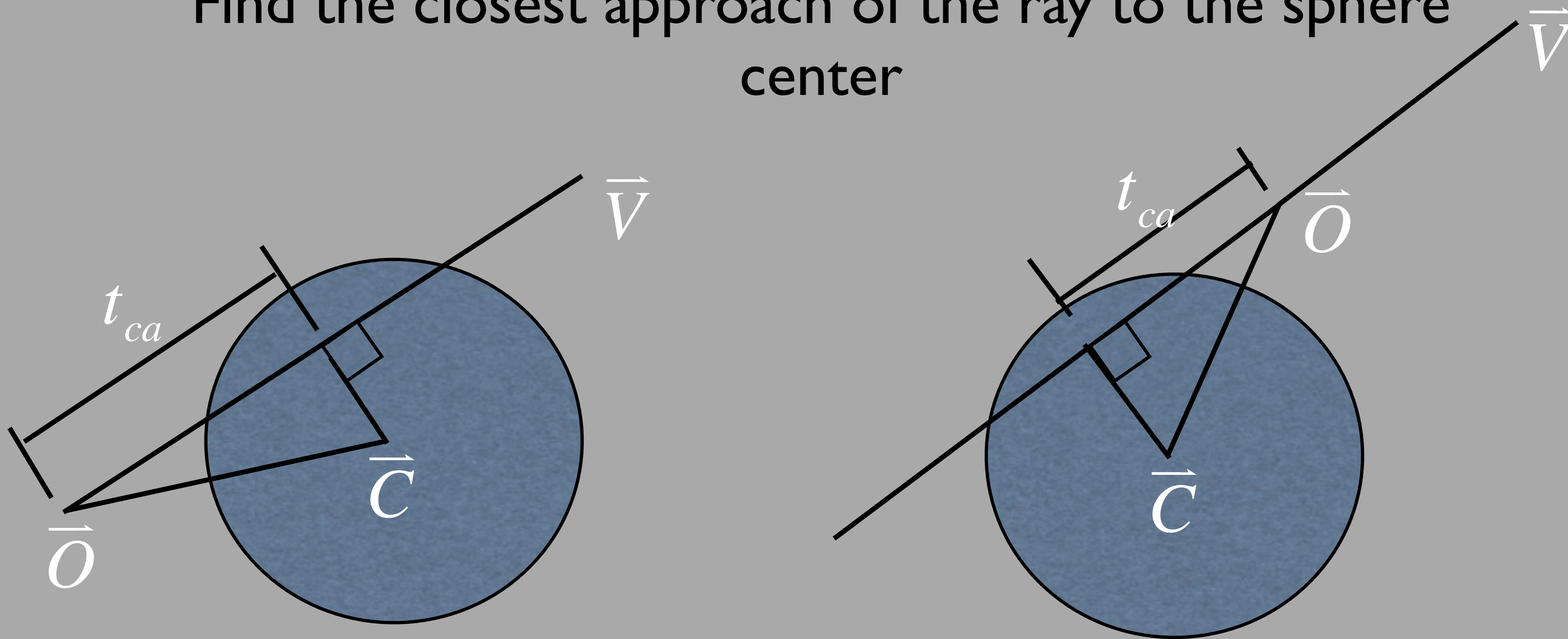


$$\left(\vec{C}-\vec{O}\right)\cdot\left(\vec{C}-\vec{O}\right) > r^2$$

$$\left(\vec{C}-\vec{O}\right)\cdot\left(\vec{C}-\vec{O}\right) < r^2$$

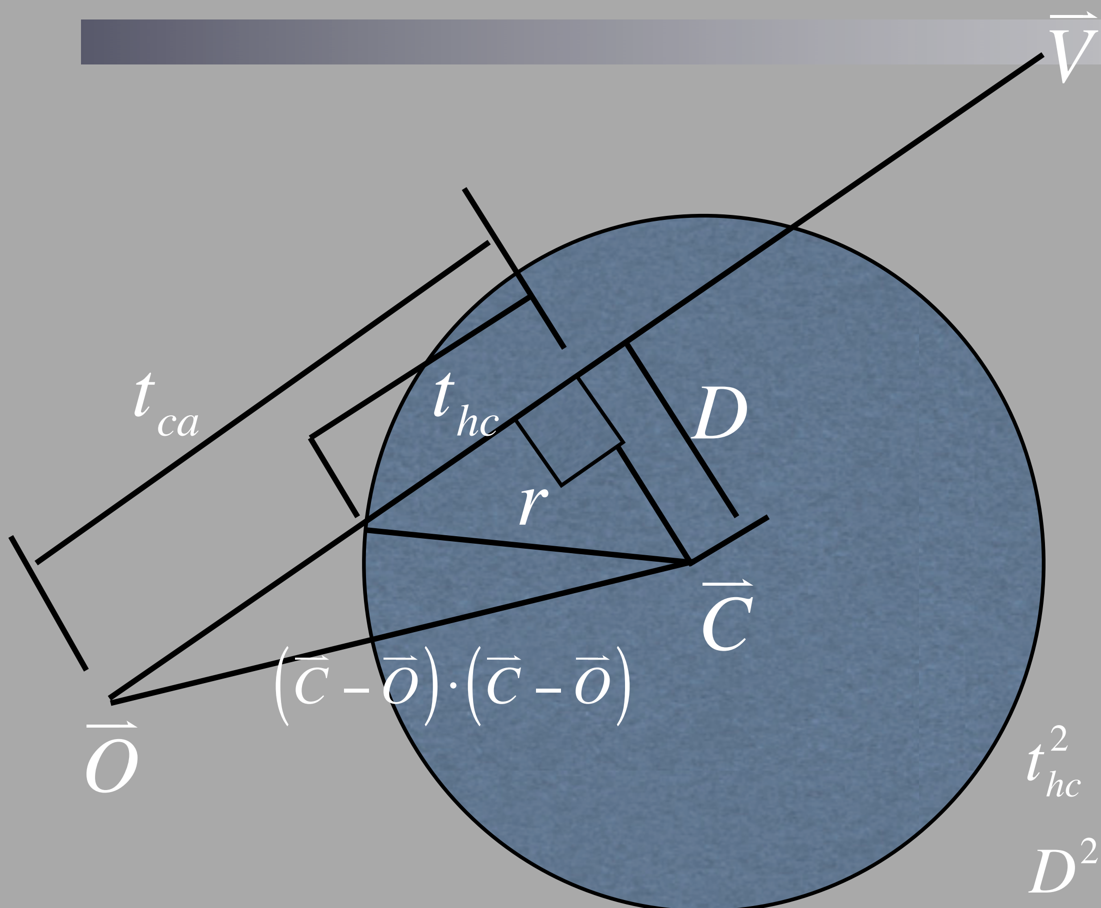
Closest approach

Find the closest approach of the ray to the sphere center



$$t_{ca} = \left(\vec{C} - \vec{O}\right) \cdot \vec{V}$$

Distance to surface

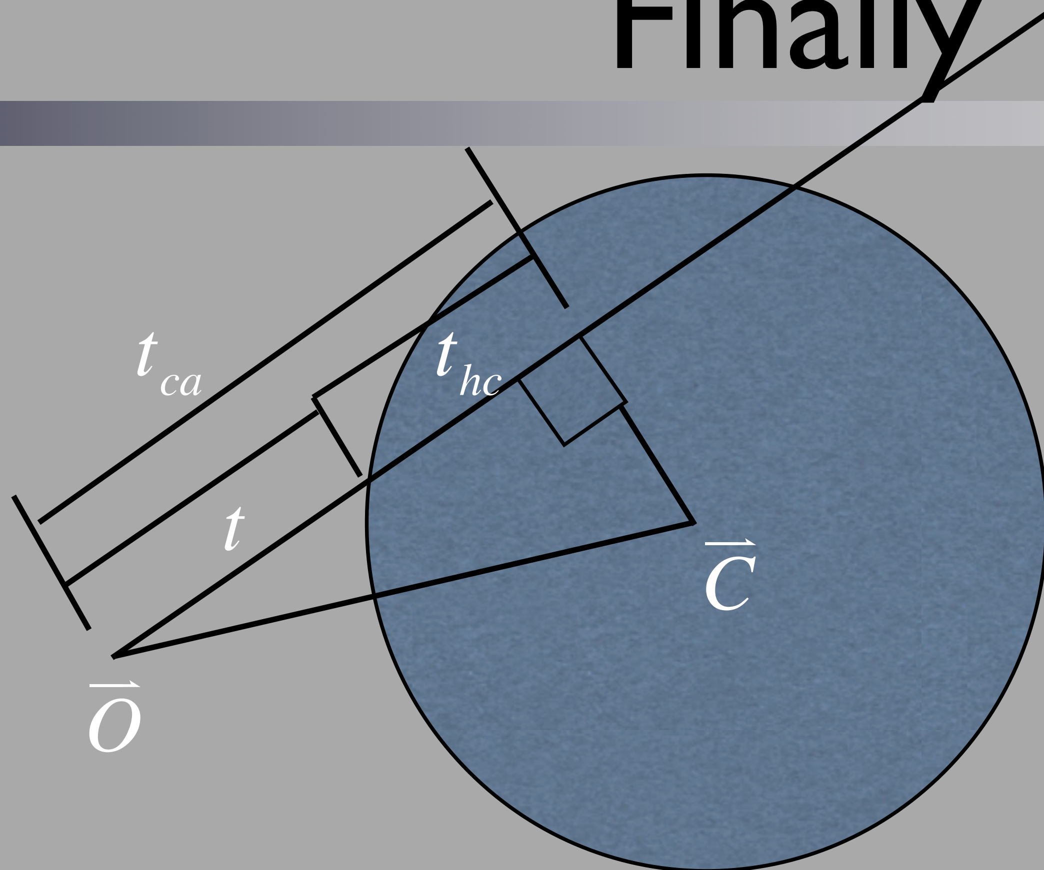


$$t_{hc}^2 = r^2 - D^2$$

$$D^2 = \left(\vec{C}-\vec{O}\right)\cdot\left(\vec{C}-\vec{O}\right) - t_{ca}^2$$

$$t_{hc}^2 = r^2 - \left(\vec{C}-\vec{O}\right)\cdot\left(\vec{C}-\vec{O}\right) + t_{ca}^2$$

Finally



origin outside sphere: $t = t_{ca} - \sqrt{t_{hc}^2}$

origin inside sphere: $t = t_{ca} + \sqrt{t_{hc}^2}$

Ray-sphere intersection

Vector $\overrightarrow{CO} = \vec{C} - \vec{O}$

$L_{co}^2 = \overrightarrow{CO} \cdot \overrightarrow{CO}$

$t_{ca} = \overrightarrow{CO} \cdot \vec{V}$

$if\left(L_{co}^2 < r^2\right)$

$\mathrm{t}_{\mathrm{hc}}^2 = r^2 - L_{co}^2 + t_{ca}^2$

root: $t_{ca} + \sqrt{t_{hc}^2}$

else

if(tca<0) behind ray, no roots

else

$\mathrm{t}_{\mathrm{hc}}^2 = r^2 - L_{co}^2 + t_{ca}^2$

if($\mathrm{t}_{\mathrm{hc}}^2$ <0) misses, no roots

else

root: $t_{ca} - \sqrt{t_{hc}^2}$

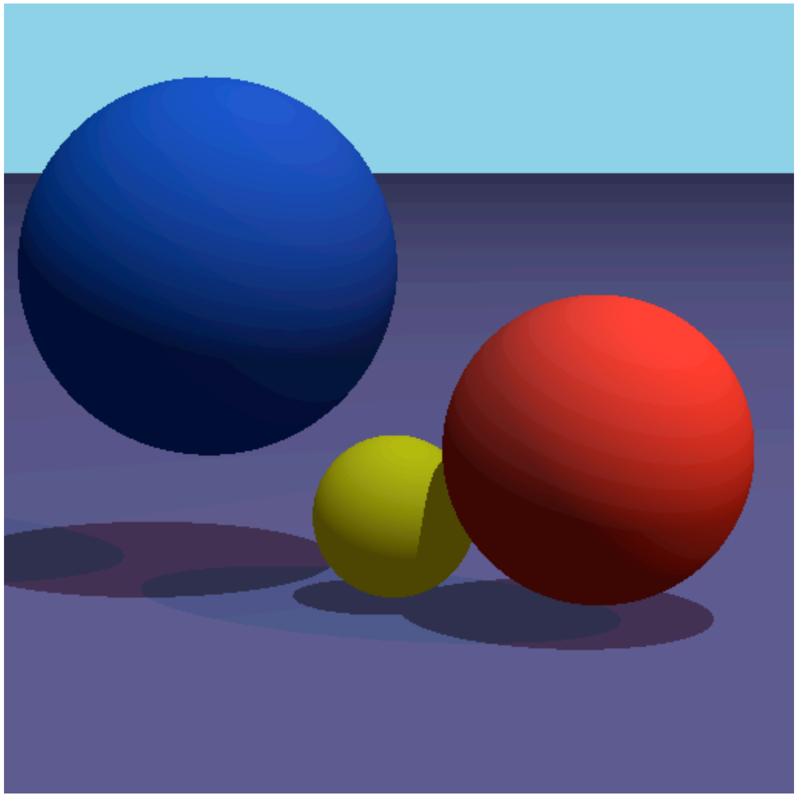
	Add / Sub / Mult	*Compare*	*Divide*	$\sqrt{\ }$
Vector $\overrightarrow{CO} = \vec{C} - \vec{O}$	3			
$L_{co}^2 = \overrightarrow{CO} \cdot \overrightarrow{CO}$	5			
$t_{ca} = \overrightarrow{CO} \cdot \vec{V}$	5			
$if\left(L_{co}^2 < r^2\right)$	1	1		
$\mathrm{t}_{\mathrm{hc}}^2 = r^2 - L_{co}^2 + t_{ca}^2$	3			
root: $t_{ca} + \sqrt{t_{hc}^2}$	1	1		1
else				
if(tca<0) behind ray, no roots		1		
else				
$\mathrm{t}_{\mathrm{hc}}^2 = r^2 - L_{co}^2 + t_{ca}^2$	3			
if($\mathrm{t}_{\mathrm{hc}}^2$ <0) misses, no roots		1		
else				
root: $t_{ca} - \sqrt{t_{hc}^2}$	1	1		1
Total	18 / 14 / 17 / 18	2 / 2 / 3 / 4		1 / 0 / 0 / 1

Results

- 16-23 ops vs. 18-26 ops
- Important early exit (sphere behind ray with no sqrt)
- Use this insight to do the same in the algebraic version
  - Use CO instead of OC (saves 1 op)
  - Early exit with $c<0$
  - Exercise left to reader

Program 2

- Due sometime
- Assigned sometime



End Slides