Ray Tracing I

Lecture 17

History

Ray Tracing came from the Physics of lens making. The process was that of drawing lines or rays through a glass shape to determine it's lens properties. It is also related to early perspective painting technique of Durer.

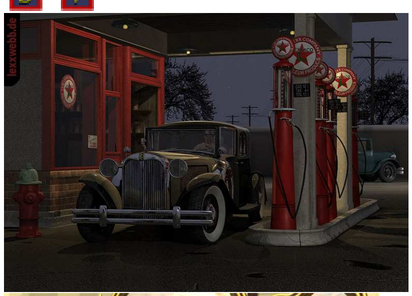
The ideas of using rays to make computers images was first tried in the early 60's, but with the computer power of that time was so much slower than other methods that it was considered not to worth the effort.

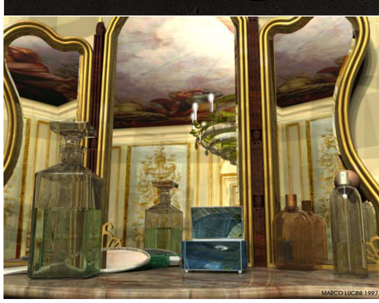
By the early 80's the computer power had developed and ray tracing was given another try. These images showed promise and lead to the research that has made today's images possible.

1

Internet Raytracing Competition

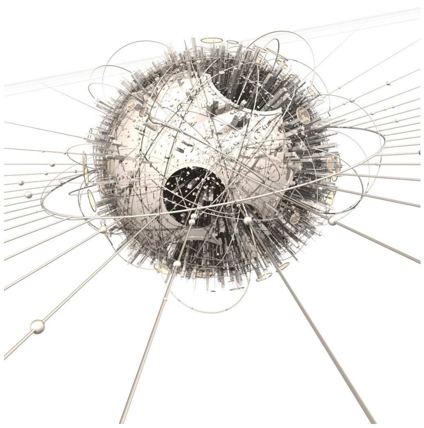
Lecture 17





Imagery

Lecture 17



- Casey Uhrig
- Povray
- Image 7200x10800
- 6 hours

3

Imagery

Lecture 17

Raytracing

Lecture 17

- Not in hardware
  - Full freedom on what you can do!
- Simple!
  - As sophisticated as you want
- Great way to learn graphics concepts

5

Ray Tracing I

Lecture 17

Ray Tracing

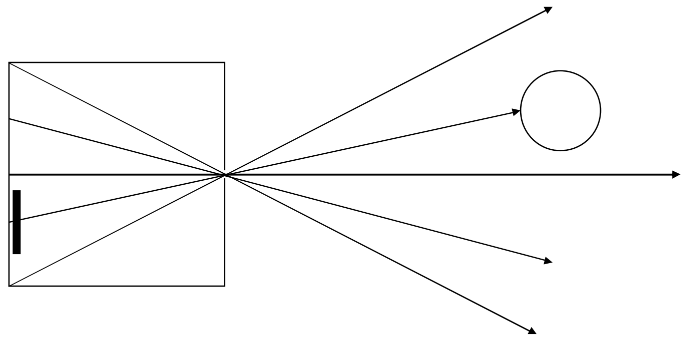
```
for all pixels (x,y) {
    for all objects {
        compare z
    }
}
```

6

Ray Tracing I

Lecture 17

Ray tracing uses geometric optics. We can look back at the pin hole camera to see the basics of ray tracing.

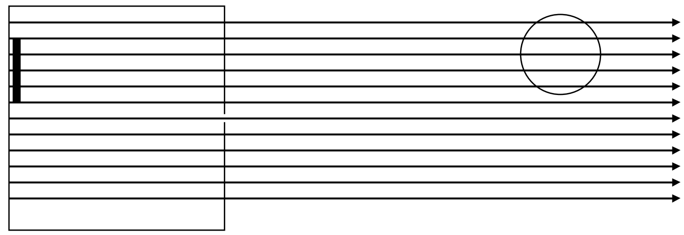


7

Ray Tracing I

Lecture 17

Ray tracing can also be used to generate parallel view images. In this mode we look from each pixel in the same direction.

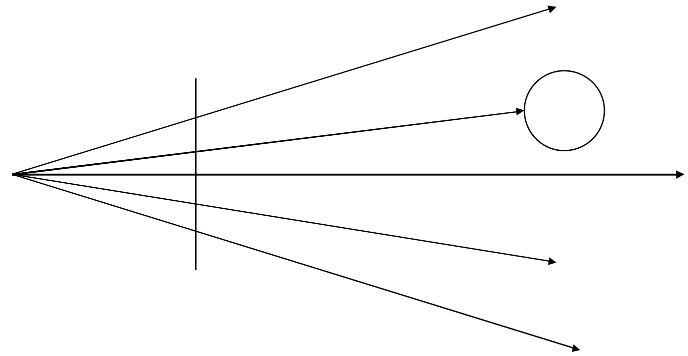


8

Ray Tracing I

Lecture 17

Remember the Camera Models.



9

Ray Tracing I

Lecture 17

The Camera Model is used to generate the initial rays. In ray tracing we do not use the Viewing Transformations.

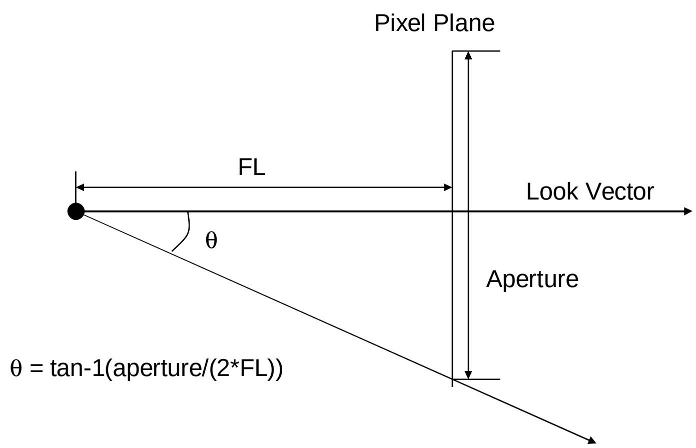
The relationship between pixel and initial ray direction is simple for the orthographic camera model. Each ray takes on the X-Y-Z position of the pixel and all rays have the same vector direction, that of the look vector.

The relationship between pixel and initial ray direction is more complex for the Perspective camera model. It is a rectangular to polar transformation.

10

Ray Tracing I

Lecture 17

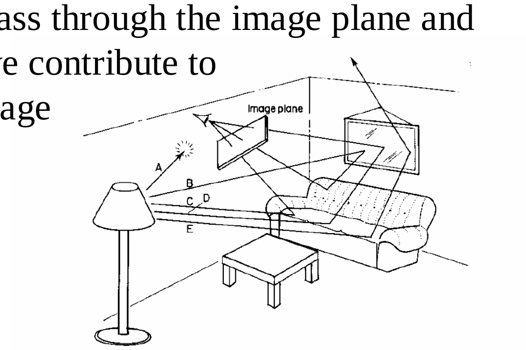


11

Forward Ray Tracing

Lecture 17

- Rays emanate from light sources and bounce around
- Rays that pass through the image plane and enter the eye contribute to the final image

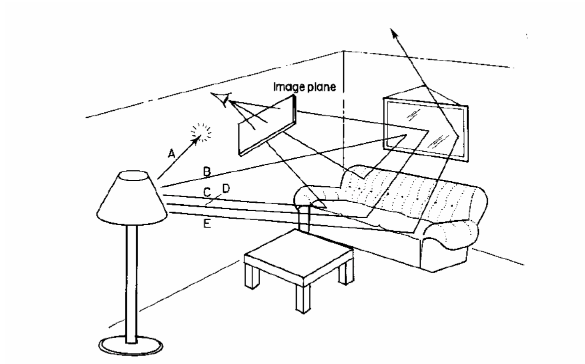


12

Forward ray tracing

Lecture 17

- So what is wrong with this forward ray tracing?

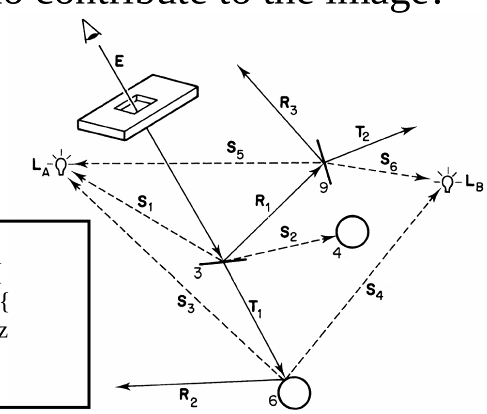


13

Backward ray tracing

Lecture 17

- Which rays do contribute to the image?



Ray Tracing

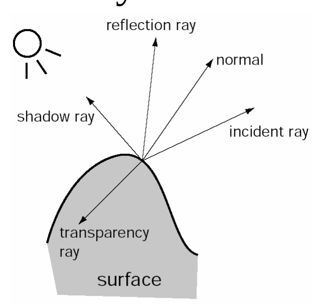
```
for all pixels (x,y) {
    for all objects {
        compare z
    }
}
```

14

Kinds of rays

Lecture 17

- Primary ray
  - leaves the eye and travels out to the scene
- When hit - spawn three new rays to "collect light"
  - shadow ray
    - towards lights
  - reflection ray
  - transparency ray

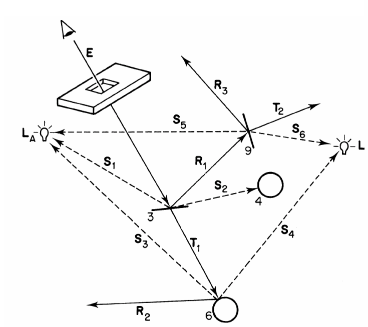


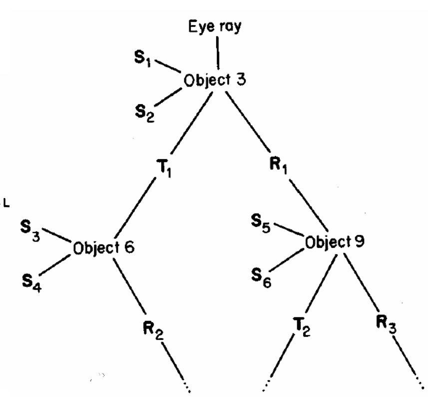
15

The ray tree

Lecture 17

- so we have a tree.



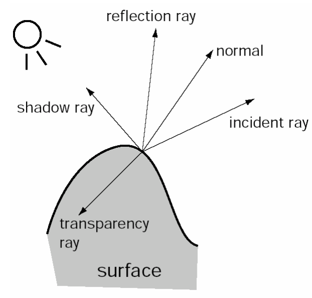


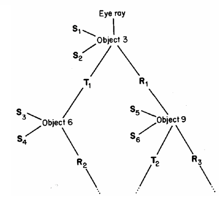
16

Raytracing is …

Lecture 17

- **recursive**
- I(incident-out) =
  I(shadow-local-in)
  + Kr * I(reflection-in)
  + Kt * I(transparent-in)
  - what is a range of Kr and Kt?
- Without recursion we have **raycasting**



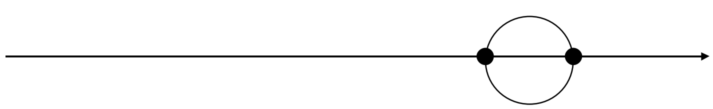


17

Ray Tracing I

Lecture 17

The core of any ray tracing systems, as well as the bottle neck, is the finding of the intersection points between an object an ray. All ray intersection problems boil down to the mathematical process of finding roots.



18

Ray Tracing I

Lecture 17

Sphere Intersection

The vector is defined as:

$\mathbf{P}_{\mathbf{origin}} = \mathbf{P}_{\mathbf{o}} = [X_0, Y_0, Z_0]$
$\mathbf{P}_{\mathbf{direction}} = \mathbf{P}_{\mathbf{d}} = [X_d, Y_d, Z_d]$ where P_d is normalized.

This defines a ray as a set of points on the line
$\mathbf{P}(t) = \mathbf{P}_{\mathbf{o}} + \mathbf{P}_{\mathbf{d}} * t$, where $t > 0$

19

Ray Tracing I

Lecture 17

The sphere is defined as:

Sphere center = $\mathbf{S}_{\mathbf{c}} = [X_c, Y_c, Z_c]$
Sphere radius = $S_r = r$

There sphere's surface is defined as all points $[X_s, Y_s, Z_s]$ in the implicit equation:

$$(X_s - X_c)^2 + (Ys - Y_c)^2 + (Z_s - Z_c)^2 - S_r^2 = 0$$

20

Ray Tracing I

Lecture 17

The intersection of the sphere and the ray can now be found by placing the ray equation into the sphere equations.

$$X = X_0 + X_d * t$$
$$Y = Y_0 + Y_d * t$$
$$Z = Z_0 + Z_d * t$$

21

Ray Tracing I

Lecture 17

Resulting in:

$$(X_0 + X_d * t - X_c)^2 + (Y_0 + Y_d * t - Y_c)^2 + (Z_0 + Z_d * t - Z_c)^2 - S_r^2 = 0$$

Simplified this results in:

$$(X_d^2 + Y_d^2 + Z_d^2)t^2$$
$$+ 2*(X_d*(X_0 - X_c) + Y_d*(Y_0 - Y_c) + Z_d*(Z_0 - Z_c))*t$$
$$+ (X_0 - X_c)^2 + (Y_0 - Y_c)^2 + (Z_0 - Z_c)^2 - S_r^2 = 0$$

22

Ray Tracing I

Lecture 17

We now solve for t. Note that the above equation is a simple quadratic

$A*t^2$ where $A = (X_d^2 + Y_d^2 + Y_d^2)$
$+ B*t$ where $B = 2*(X_d*(X_0 – X_c) + Y_d*(Y_0 – Y_c) + Z_d*(Z_0 – Z_c))$
$+ C = 0$ where $C = (X_0 – X_c)^2 + (Y_0 – Y_c)^2 + (Z_0 – Z_c)^2 - S_r^2$

23

Ray Tracing I

Lecture 17

Because the direction ray is normalized A is always equal to 1. So the solutions for t can be found by using the quadratic formula:

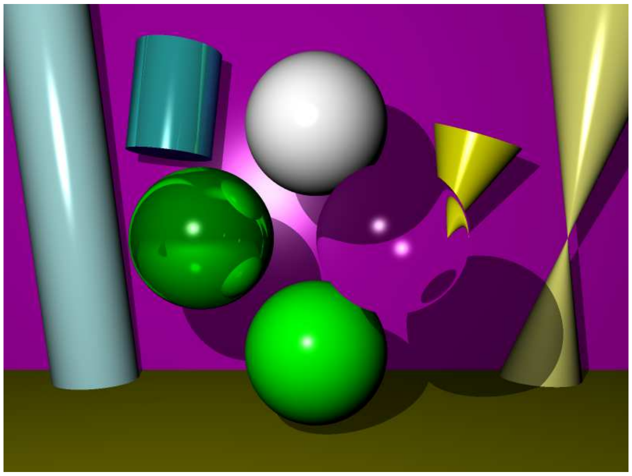
$$t_0 = (-B - sqrt(B^2 - 4*C))/2$$

$$t_1 = (-B + sqrt(B^2 - 4*C))/2$$

24

Cones and spheres and such

Lecture 17



http://www.c-wilkie.pwp.blueyonder.co.uk/index.html

25

Ray Tracing I

Lecture 17

•Only the real roots show that the ray and sphere intersect.

• Non-real roots indicate the sphere was missed. Negative values of t are not used and indicate that the ray started in the sphere and only the positive roots are valid.

•We can then compare t_0 and t_1 to find out which is smaller indicating the point on the sphere that is closer to the ray starting point $\mathbf{P_o}$.

26

Ray Tracing I

Lecture 17

Once we have selected t_0 or t_1 we can then place it back in the ray definition and find the true intersection point in X,Y,Z.

$$\mathbf{P}(\text{intersection}) = \mathbf{P_o} + \mathbf{P_d} * t_{0/1}$$

We can then find the sphere's surface normal at this point and other properties needed to render the image.

27

Ray Tracing I

Lecture 17

Plane Intersection

For the plane use the same definition of the ray. The definition of a plane is:

$$\text{Plane} = A*x + B*y + C*z + D = 0$$
$$\text{where } A^2 + B^2 + C^2 = 1$$

The unit normal of the Plane defines A, B, C:

$$P_{normal} = P_n = [A\ B\ C]$$

28

Ray Tracing I

Lecture 17

D is defined as the distance from coordinate system origin [0 0 0]. The sign of D determines which side of the plane the system origin is located. Again we substitute the Ray into the plane definition.

$$A *(X_0 + X_d*t) + B*(Y_0 + Y_d*t) +C*(Z_0 + Z_d*t) + D = 0$$

29

Ray Tracing I

Lecture 17

We again solve for t:

$$t = -(A*X_0 + B*Y_0 + C*Z_0 + D) / (A*X_d + B*Y_d + C*Z_d)$$

We can write this is vector notation:

$$t = -(\mathbf{P_n} \bullet \mathbf{P_o} +D)/(\mathbf{P_n} \bullet \mathbf{P_d})$$

30

Ray Tracing I

Lecture 17

Now we can look at the components of the solution for t and find that we can test for simple things to save time and computations. First lets look at the bottom part:

$$V_d = \mathbf{P_n} \bullet \mathbf{P_d} = (A*X_d + B*Y_d + C*Z_d)$$

If this is equal to zero then the ray is parallel to the plane and there is no intersection. Even if the ray lays in the plane you can not have an intersection with the edge of a plane.

31

Ray Tracing I

Lecture 17

Once again we can put t back into the ray equation and find the intersection point.

$$\mathbf{P}(\text{intersection}) = \mathbf{P_o} + \mathbf{P_d} *t$$

We can then quickly determine if this intersection point is within a four sided polygon by testing min and max X,Y,Z points. These polygons are often used to bound complex shapes and object. The ray tracer would first see if the ray hits the bounding volume before it proceeds to find intersection of the complex objects contained in the volume.

32

Ray Tracing I

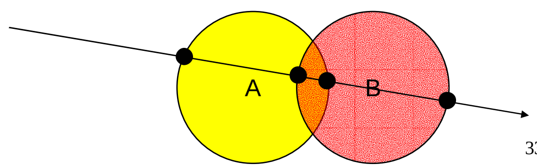
Lecture 17

Constructive Solid Geometry

- Find intersection along ray with each object that makes up the solid.

- Follow CSG tree to combine intersection pairs to find the final intersection pair.

•Example: two spheres t_{A1}, t_{A2}, t_{B1}, t_{B2}



33

Ray Tracing I

Lecture 17

References

An Introduction to Ray Tracing
by Andrew S. Glassner (Editor)
Academic Press; ISBN: 0122861604
Ray Tracing Worlds With Pov-Ray
Book and 2 Disks
by Alexander Enzmann, Lutz Kretzschmar, Chris Young
ISBN: 1878739646

Realistic Ray Tracing
by Peter Shirley
AKPeters Ltd. 2000

34