Ray Tracing

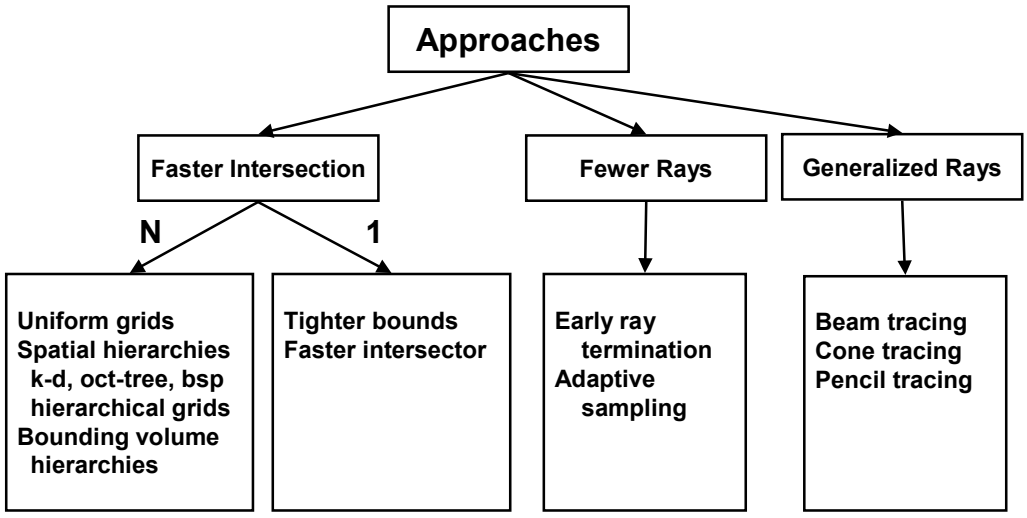
Ray Tracing 1

- Basic algorithm
- Ray-surface intersection (triangles, spheres, …)

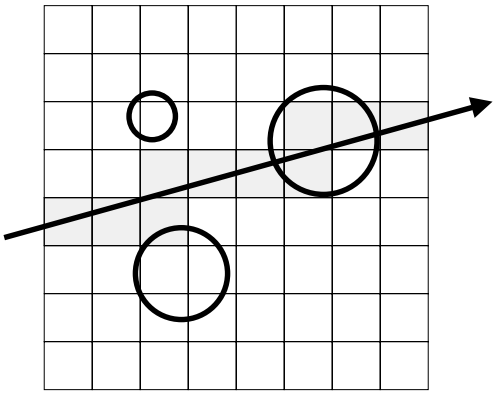
Ray Tracing 2

- Acceleration data structures

Ray Tracing Acceleration Techniques



Uniform Grids



Preprocess scene

1. **Determine resolution**

   D |cells| ~ |objects|

2. **Place object in cell if object overlaps cell**

Traverse grid

3D line – 3D-DDA

Caveat

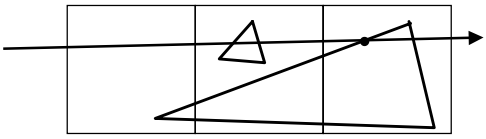
Optimize for objects that overlap multiple cells

Caveat 1:

Intersection must be within bound

Caveat 2:

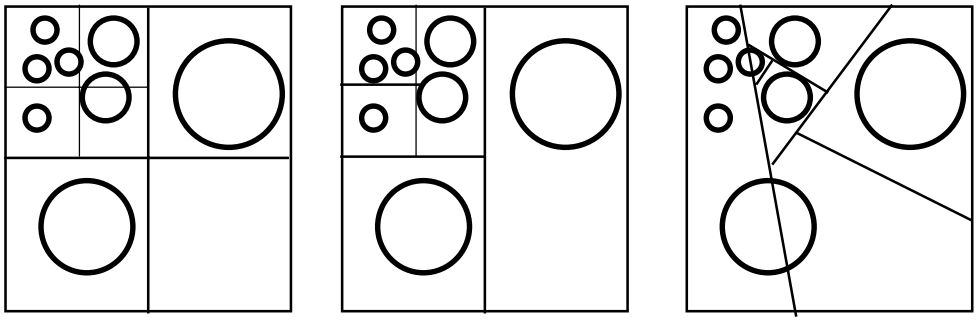
Redundant intersection tests



Mailboxes

- **Assign each ray a number**
- **Object intersection cache (mailbox)**
  - **Store ray number**
  - **Intersection**

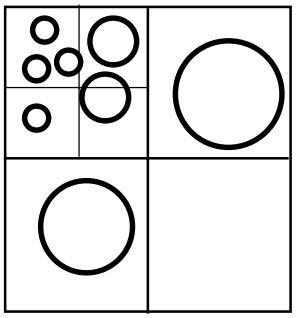
Spatial Hierarchies



oct-tree kd-tree bsp-tree

Spatial Hierarchies

```
insert(node,prim) {
  if( overlap(node->bound,prim->bound) )
    if( node->nprims > maxprims ) {
      subdivide(node); // and reinsert prims
      foreach child
        insert(child,prim)
    }
    else
      insert(node->prims,prim);
}
```



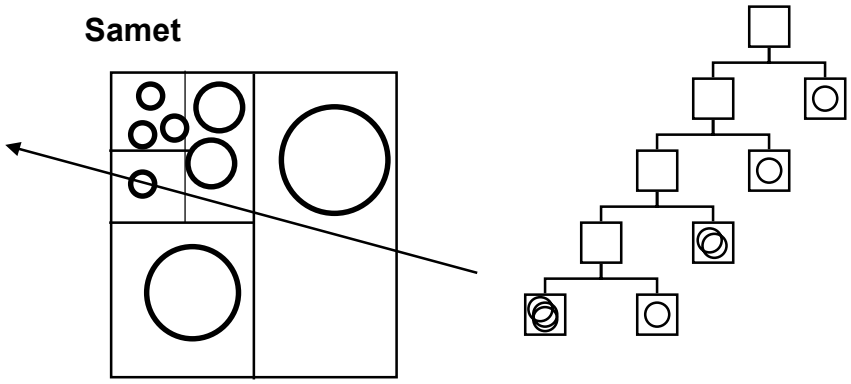
Ray Traversal Algorithms

1. Point location from root

   Glassner

2. Neighbor finding

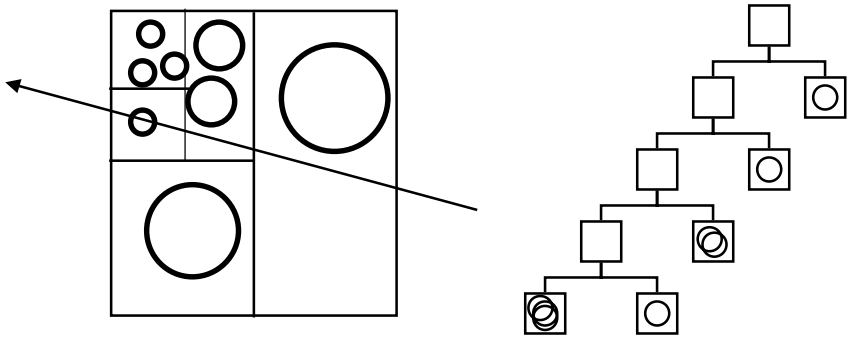
   Samet



Ray Traversal Algorithms

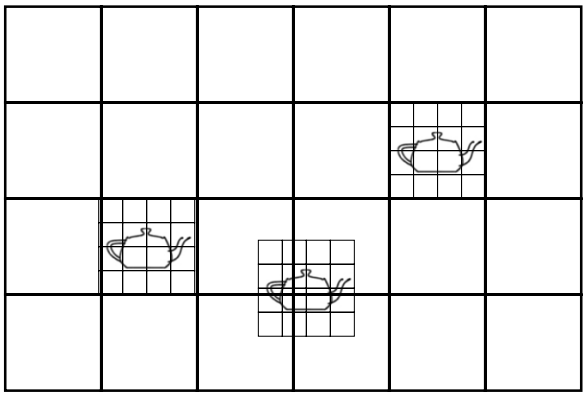
Recursive inorder traversal

Kaplan, Arvo, Jansen

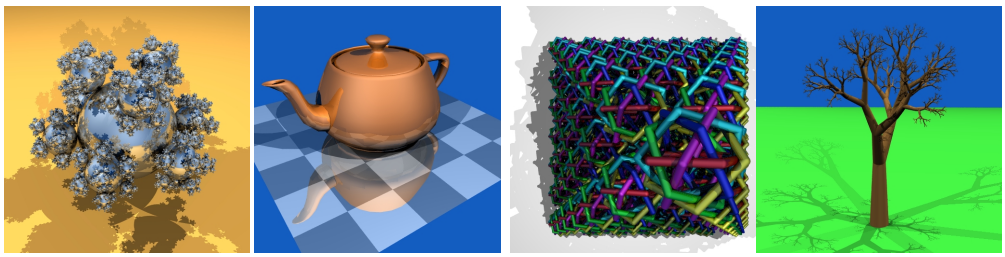


Hierarchical Grids

Good compromise preferred by many practitioners



Comparison



	Spheres	Teapot	Rings	Tree
Uniform Grid d=1	244	27	129	1517
d=20	38	19	83	781
Hierarchical Grid	34	25	116	34

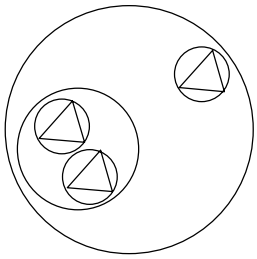
See http://www.acm.org/tog/resources/RTNews/html/rtnv12n1.html#art3

Hierarchical Bounding Volumes

Create of tree of bounding volumes

Children are contained within parent

- **Creation preprocess**
  - **From model hierarchy**
  - **Automatically**



Search

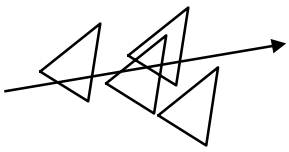
```
intersect(node,ray,hits) {
  if( leaf(node) )
    intersect(node->prims,ray,hits)
  else if( intersectp(node->bound,ray) )
    for each child
      intersect(child,ray,hits)
}
```


Theoretical Nugget 1

Computational geometry of ray shooting

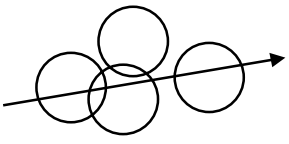
1. Triangles (Pellegrini)

- **Time:** $O(\log n)$
- **Space:** $O(n^{5+\varepsilon})$



2. Sphere (Guibas and Pellegrini)

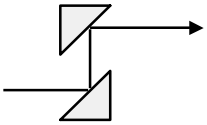
- **Time:** $O(\log^2 n)$
- **Space:** $O(n^{5+\varepsilon})$



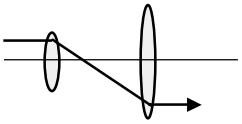
Theoretical Nugget 2

Optical computer = Turing machine

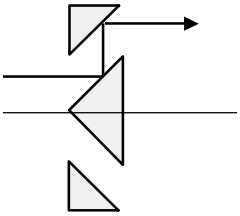
y=y+1



y=2*y



if(y>0)

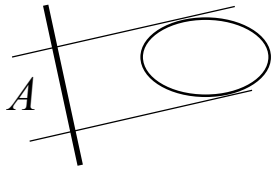


Theoretical Nugget 3

Probability that a ray in a given direction intersects a shape is equal to its projected area.

Probability that any ray intersects a shape is equal to its average projected area.

Theorem: For a convex body $\overline{A} = \frac{S}{4}$



Sphere: $\overline{A} = A = \pi r^2$

$S = 4\pi r^2$