

Computer Graphics

LECTURE 07

MAHAM KHAN

Last Class

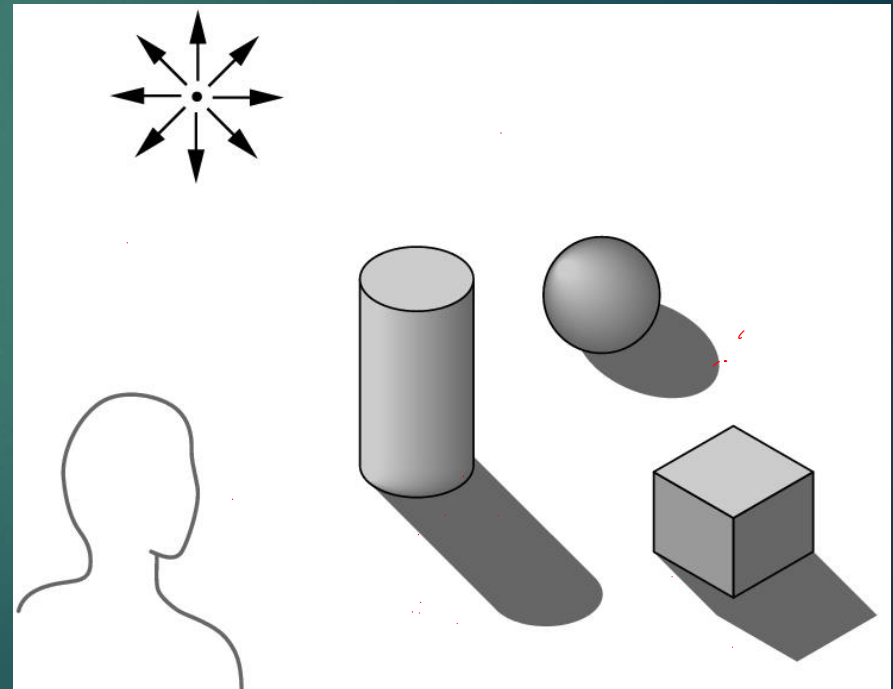
- ▶ Human Visual System
 - ▶ Illusions
- ▶ Ray Tracing

Today's Agenda

- ▶ Human Visual System
 - ▶ Illusions
- ▶ Ray Tracing

Ray Tracing

- ▶ Building an imaging model by following light from a source is known as Ray Tracing
- ▶ A Ray is a semi infinite line that emerges from a source and continues to infinity in one direction
- ▶ Part of ray contributes in making image.
- ▶ Surfaces
 - ▶ Diffusing
 - ▶ Reflecting
 - ▶ Refracting



Ray Tracing

- ▶ For each pixel intensity must be computed
- ▶ Contributions of all rays must be taken into account
- ▶ A ray when intercepted by a surface splits into two rays
 - ▶ Absorbed
 - ▶ Reflected

Ray Intersections

- ▶ Tracing a single ray requires determining if that ray intersects any one of potentially millions of primitives
- ▶ This is the basic problem of *ray intersection*
- ▶ Many algorithms exist to make this not only feasible, but remarkably efficient
- ▶ Tracing one ray is a complex problem and requires serious work to make it run at an acceptable speed
- ▶ Of course, the big problem is the fact that one needs to trace lots of rays to generate a high quality image

Lighting

- ▶ Once we have the key intersection information (position, normal, color, texture coordinates, etc.) we can apply any lighting model we want
- ▶ This can include procedural shaders, lighting computations, texture lookups, texture combining, bump mapping, and more
- ▶ Many of the most interesting forms of lighting involve spawning off additional rays and tracing them recursively
- ▶ The result of the lighting equation is a color, which is used to color the pixel

Ray Tracing Algorithm

- ▶ Ray generation, which computes the origin and direction of each pixel's viewing ray based on the camera geometry
- ▶ Ray intersection, which finds the closest object intersecting the viewing ray
- ▶ Shading, which computes the pixel color based on the results of ray intersection.

Ray Tracing Algorithm

- ▶ Structure of a ray tracing algorithm is
 - ▶ For each pixel do
 - ▶ compute viewing ray
 - ▶ find first object hit by ray and its surface normal ' n '
 - ▶ set pixel color to value computed from hit point, light, and n
 - ▶ **End**

References

- ▶ Fundamentals of Computer Graphics Third Edition by Peter Shirley and Steve Marschner
- ▶ Interactive Computer Graphics, A Top-down Approach with OpenGL (Third Edition) by Edward Angel.