Computer Graphics

LECTURE 08

MAHAM KHAN

Last Class

- Ray Tracing

Today's Agenda

- Viewing
  - Perspectives
  - Projections

Viewing

- Process of seeing a scene is regarded as viewing.
  - The world or scenes are in three dimensions (3D)
  - A 3D Scene has to be “projected” in 2D
  - A Synthetic camera has to be modeled

Viewing Problems

- Where a viewer is located (Location)
- Viewing Plane
- The visible portion of the scene
  - i. e., what can be seen (clipping)
- Maintaining relation between objects
  - Parallel lines
  - Angles
  - Distances
- Relation to the viewer

Scenes and Objects

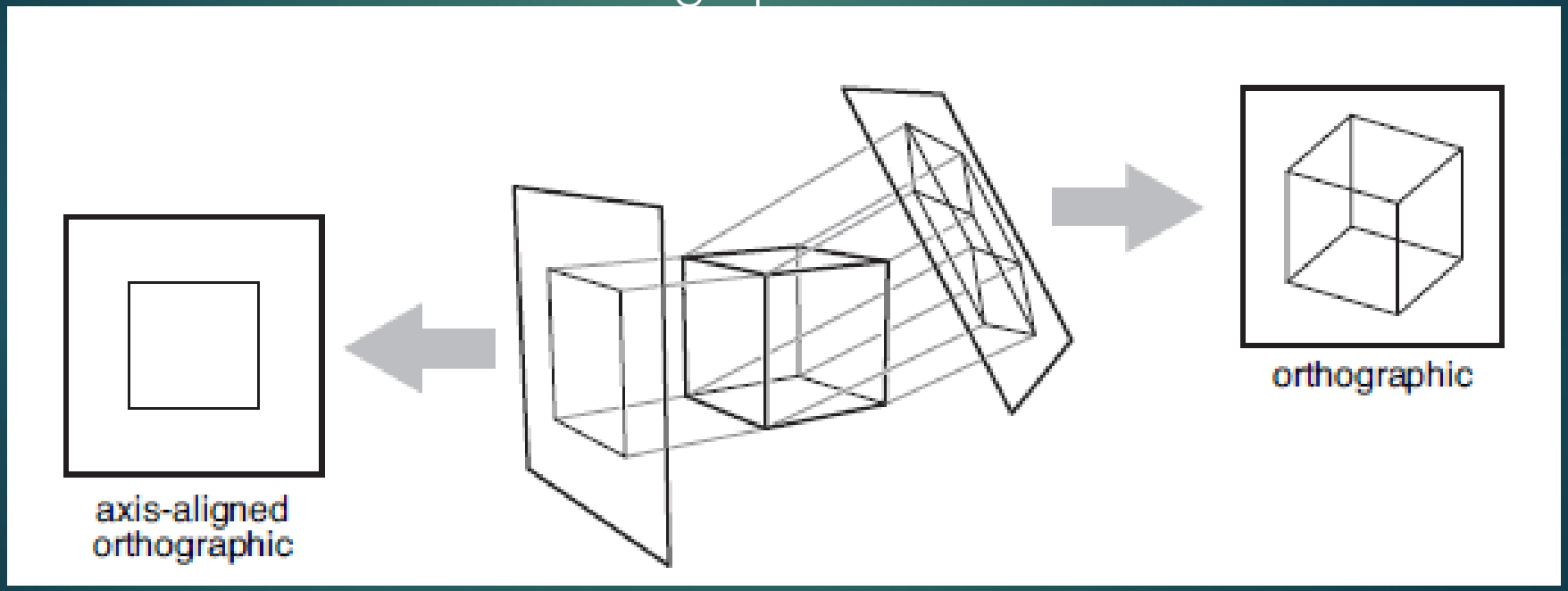
- Viewing a scene or object
  - Some viewing techniques perform better in viewing objects than a whole scene.
- External Viewing: Viewing an object from outside e. g. a building
- Internal Viewing: viewing from inside. e.g. internal of a building specially in games.
-

Projections

- Project N-Dimensions Coordinates onto <N Dimensions Coordinates

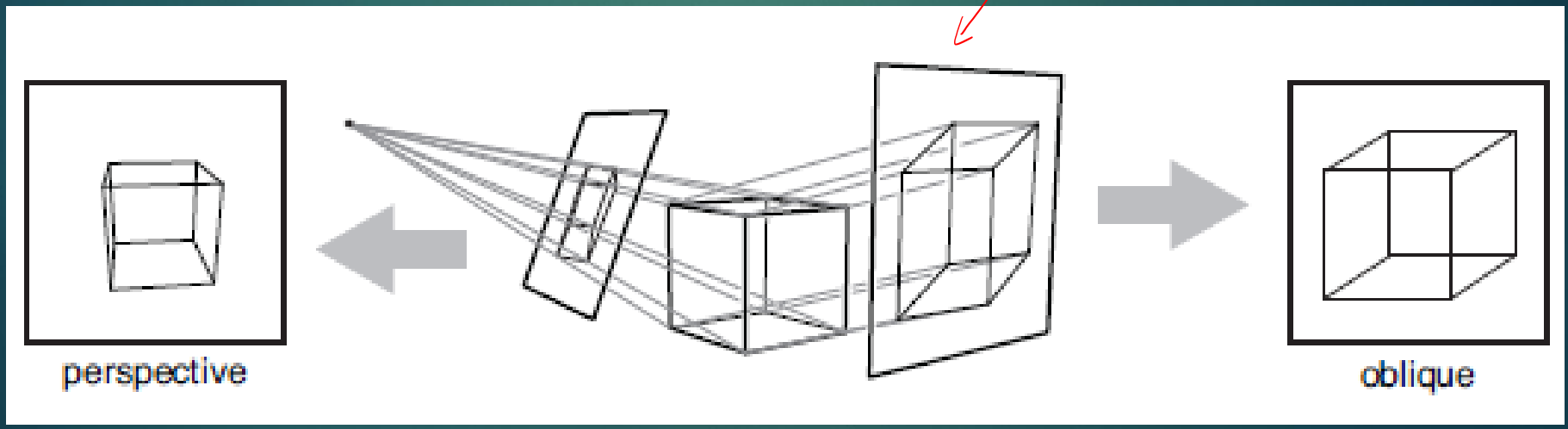
Projections

- When projection lines are parallel and perpendicular to the image plane, the resulting views are called orthographic.



Projections

- The image plane is perpendicular to the view direction, the projection is called *orthographic* otherwise it is called oblique



Perspective

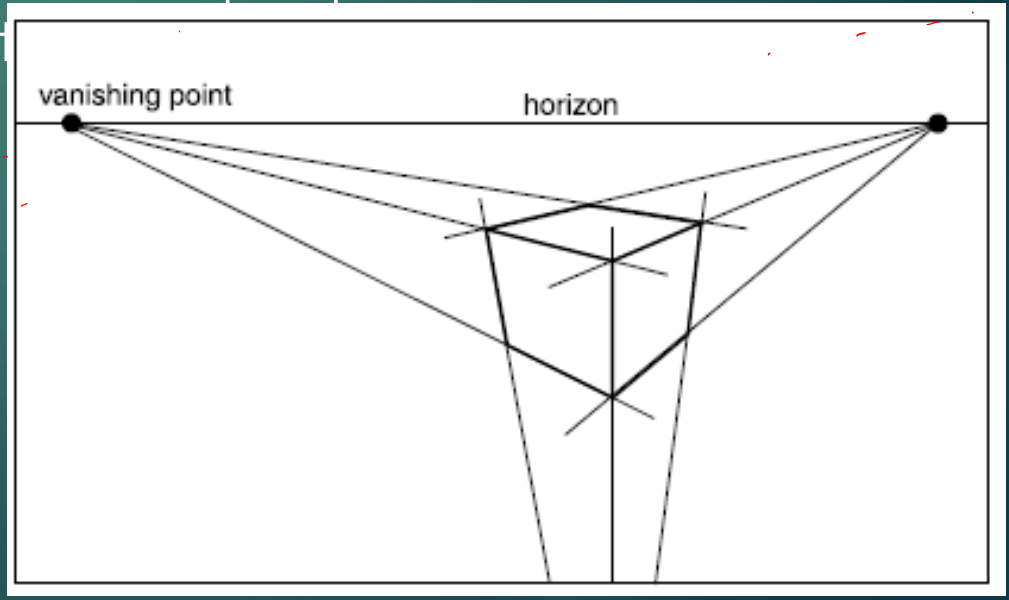
- The process of creating 2D images of 3D objects/scenes
- In linear perspective, 3D objects are projected onto an image plane in such a way that straight lines in the scene become straight lines in the image.
- Parallel Projection: 3D points are mapped to 2D by moving them along a *projection direction* until they hit the image plane

View Classification

- Parallel
  - Orthographic
    - Top (Plan)
    - Front
    - Side
    - Axiometric
  - Oblique
    - Cabinet
    - Cavalier
    - Other
- Perspective
  - One-Point
  - Two-Point
  - Three-Point

Creating View

- In three-point perspective, an artist picks "vanishing points" where parallel lines meet. Parallel horizontal lines will meet at a point on the horizon. Every set of parallel lines has its own vanishing points. These rules are followed automatically if we implement perspective based on the correct geomet



Computing Viewing Rays

- We start by 'shooting' rays from the camera out into the scene
- We can render the pixels in any order we choose (even in random order!), but we will keep it simple and go from top to bottom, and left to right
- We loop over all of the pixels and generate an initial *primary ray* (also called a *camera ray* or *eye ray*)
- The ray origin is simply the camera's position in world space
- The direction is computed by first finding the 4 corners of a virtual image in world space, then interpolating to the correct spot, and finally computing a normalized direction from the camera position to the virtual pixel

Shadow Rays

- Shadow rays behave slightly differently from primary (and secondary) rays
- Normal rays (primary & secondary) need to know the first surface hit and then compute the color reflected off of the surface
- Shadow rays, however, simply need to know if something is hit or not
- In other words, we don't need to compute any additional shading for the ray and we don't need to find the closest surface hit
- This makes them a little faster than normal rays

Summary

- Perspectives
- Projections

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.