

CS 112



The Rendering Pipeline



Rendering Pipeline

- Input – 3D Object/Scene Representation
- Output – An image of the input object/scene
- Stages (for POLYGON pipeline)
 - Model view Transformation
 - Projection Transformation
 - Clipping and Vertex Interpolation of Attributes
 - Rasterization and Pixel Interpolation of Attributes



Rendering Pipeline

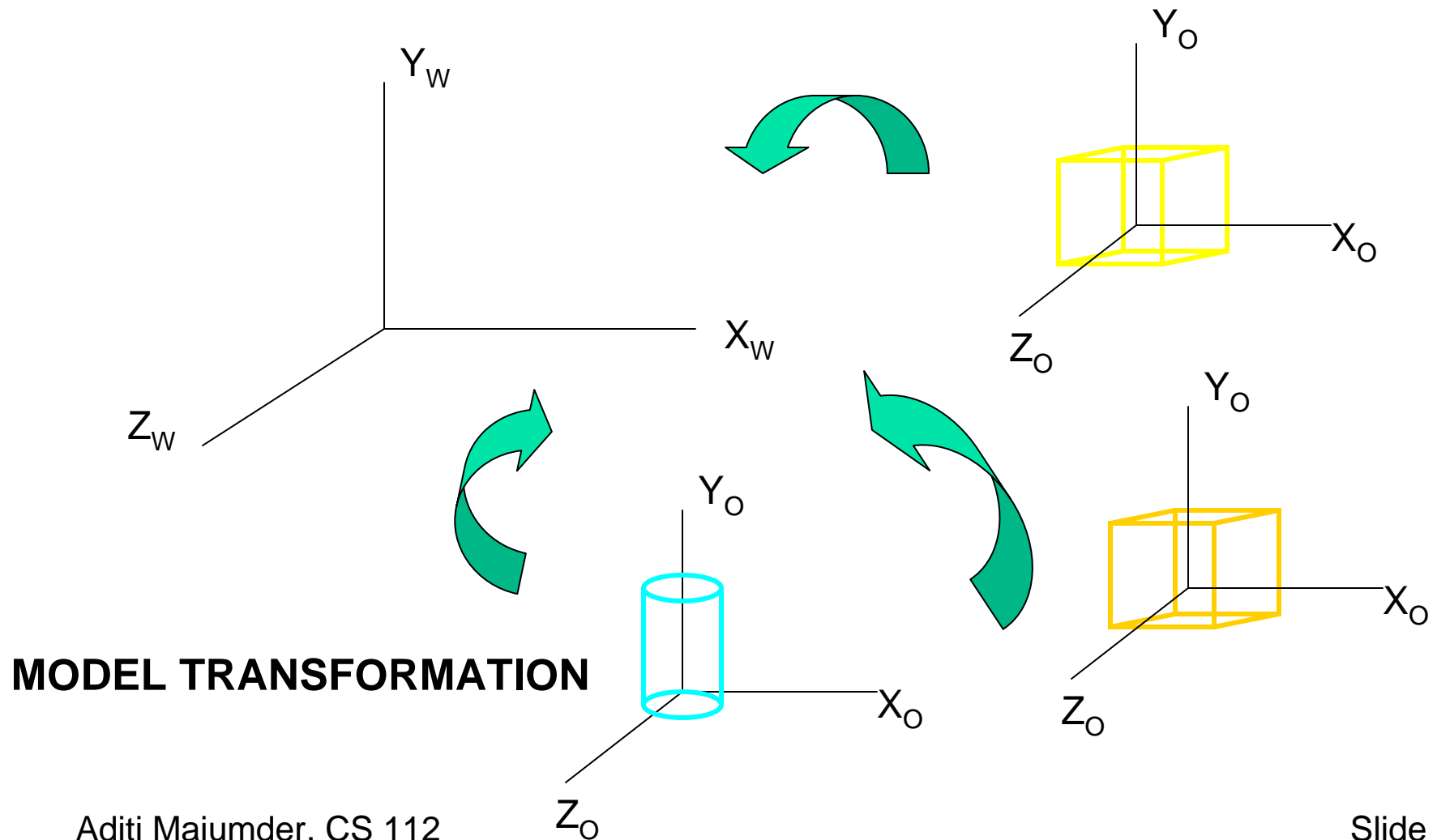
- Input – 3D Object/Scene Representation
- Output – An image of the input object/scene
- Stages
 - Model-view Transformation
 - Projection Transformation
 - Clipping and Vertex Interpolation of Attributes
 - Rasterization and Pixel Interpolation of Attributes



Model-view Transformation

- Model Transformation
- View Transformation
- World Coordinate System
- Object Coordinate System

World and Object Coordinates

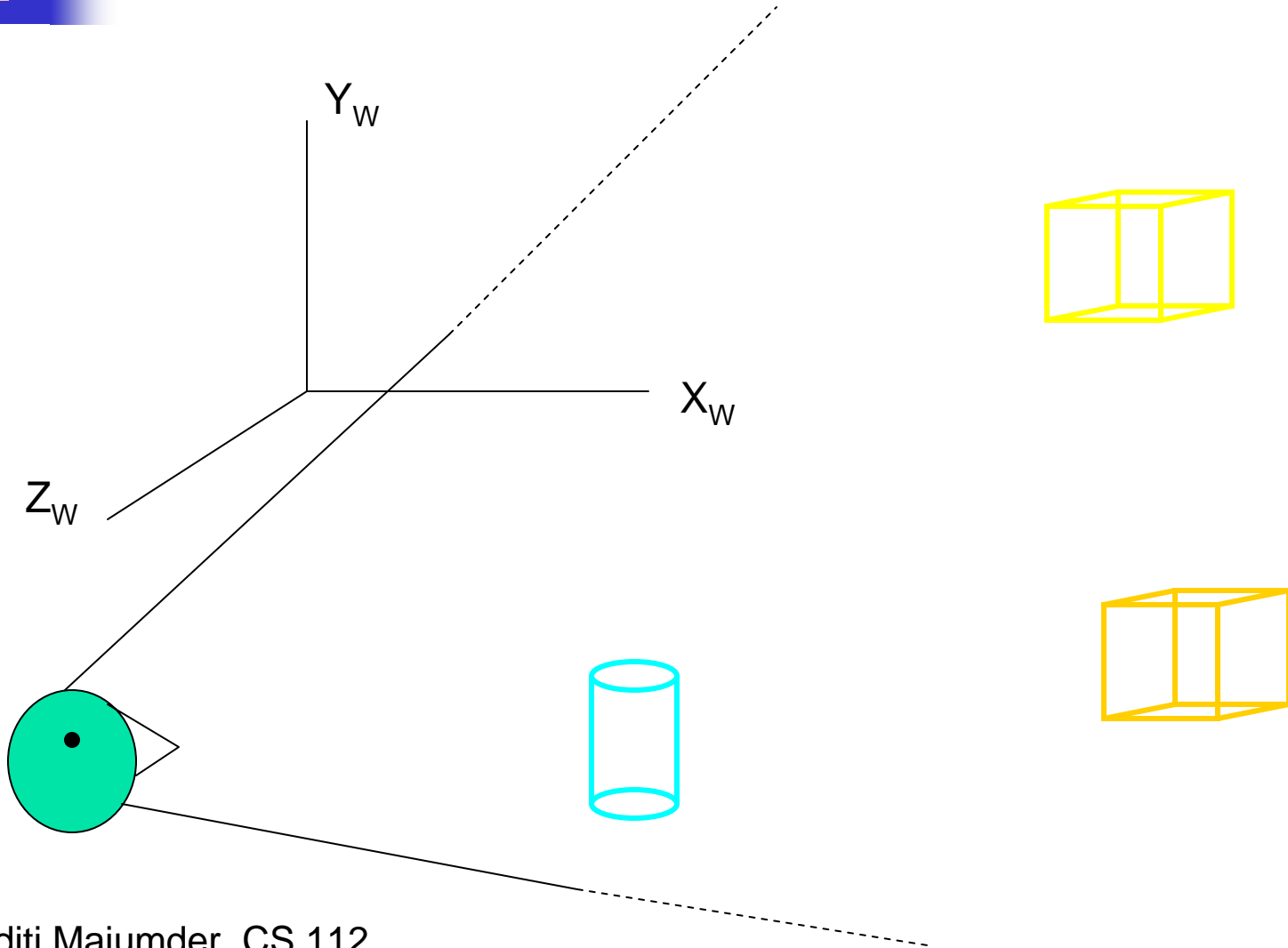




Model Transformation

- Transforming from the object to world coordinates
 - Placing the object in the desired **position, scale and orientation**
- Can be done by any kind of transformations
 - Graphics hardware/library support only linear transformations like translate, rotate, scale, and shear

View Transformation





View Transformation

- Position and orientation of eye (9 parameters)
 - View point - POINT : (x, y, z) [3]
 - Normal to the image plane – VECTOR: (v_x, v_y, v_z) [3]
 - View Up – VECTOR: (u_x, u_y, u_z) [3]
 - Default: $(0, 0, 0)$, $(0, 0, -1)$, $(0, 1, 0)$
- Transformation to align
 - Eye with the origin
 - Normal to the image plane with negative Z axis
 - View Up vector with positive Y axis
 - Can be achieved by rotation and translation

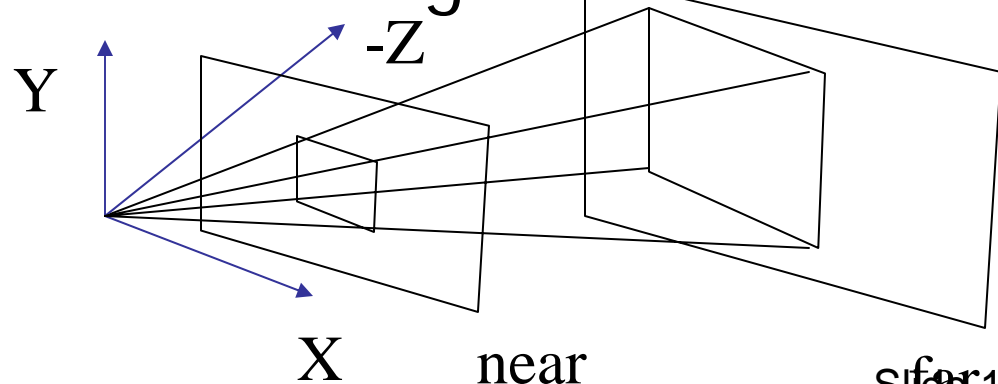


Rendering Pipeline

- Input – 3D Object/Scene Representation
- Output – An image of the input object/scene
- Stages
 - Model view Transformation
 - Projection Transformation
 - Clipping and Vertex Interpolation of Attributes
 - Rasterization and Pixel Interpolation of Attributes

Projection Transformation

- Define the “view frustum” (6 parameters)
 - Assume origin is the view point
 - Near and far planes (planes parallel to XY plane in the negative Z axis) [2]
 - Left, right, top, bottom rectangle defined on the near plane [4]



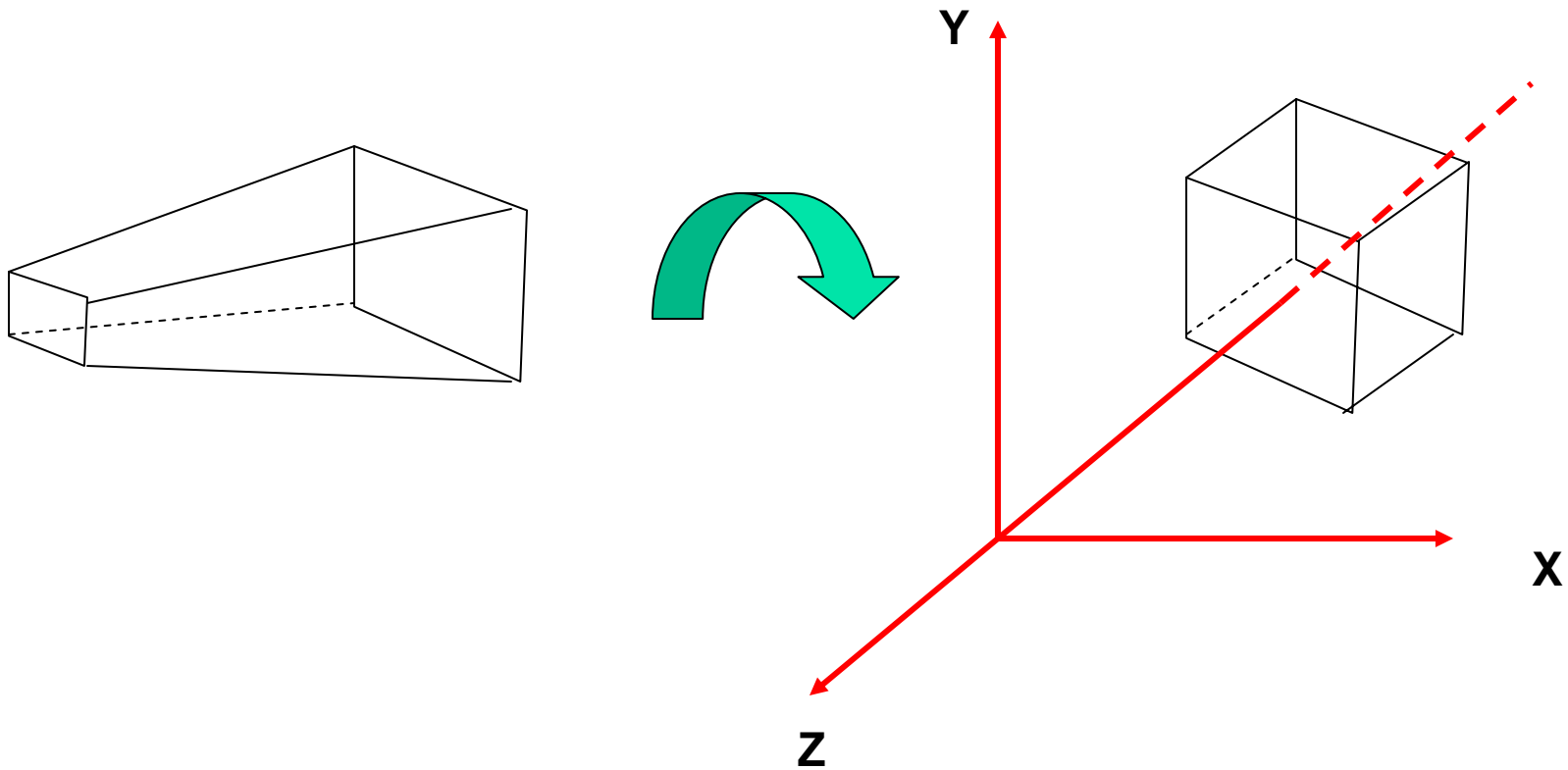


Projection Transformation

- Transforming the view frustum (along with the objects inside it) into a
 - cuboid with unit square faces on the near and far planes
 - the negative Z axis passes through the center of these two faces.
 - Projecting the objects on the near plane
- Consists of a “shear” and a “perspective projection” operations.



Projection Transformation





Important

- Every vertex undergoes the modelview and projection transformation
- Geometric transformation
 - Topology does not change
- Even when transforming triangles
 - Sufficient to transform the vertices

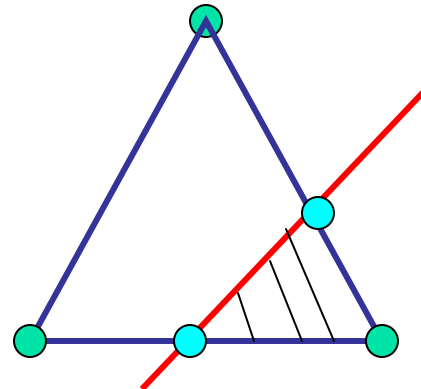


Rendering Pipeline

- Input – 3D Object/Scene Representation
- Output – An image of the input object/scene
- Stages
 - Model view Transformation
 - Projection Transformation
 - Clipping and Vertex Interpolation of Attributes
 - Rasterization and Pixel Interpolation of Attributes

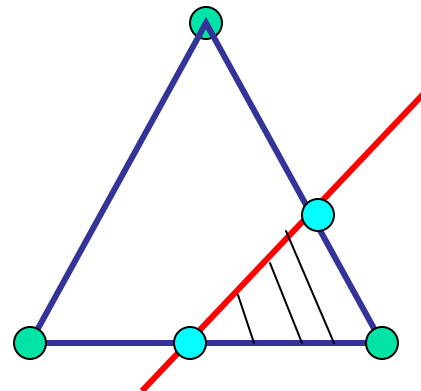
Clipping

- Removing the part of the polygon outside the view frustum
- If the polygon spans inside and outside the view frustum
 - introduce new vertices on the boundary



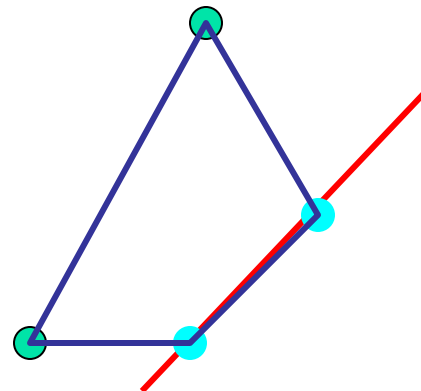
Interpolation of Attributes

- For the new vertices introduced
 - compute all the attributes
 - Using interpolation of the attributes of all the original vertices



Interpolation of Attributes

- For the new vertices introduced
 - compute all the attributes
 - Using interpolation of the attributes of all the original vertices





Window Coordinate Transformation

- Scale XY coordinates of unit cuboid to reflect size of window (relative pixel coordinates)
- Translate these coordinates to the position of the window on the monitor screen to represent the absolute pixel coordinates.
- Z value is used for resolving occlusion

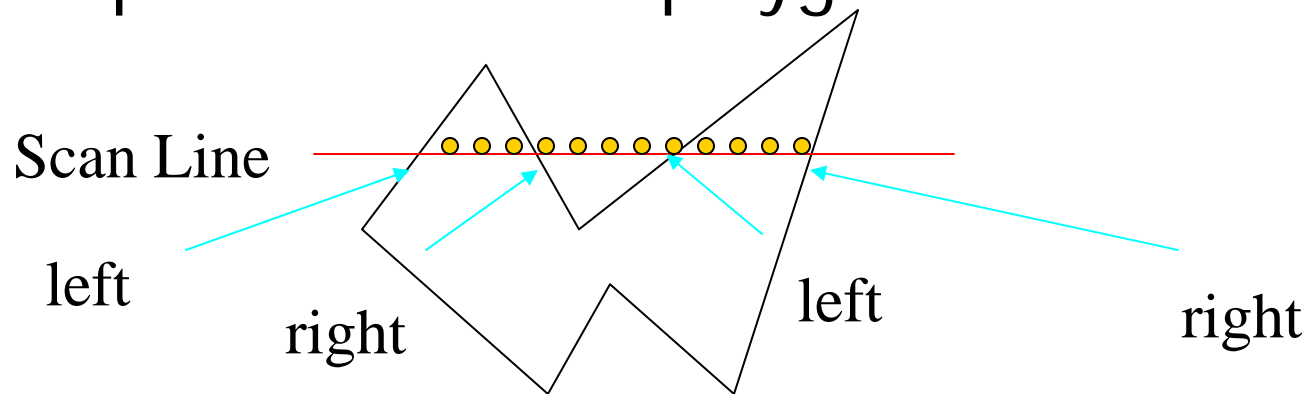


Rendering Pipeline

- Input – 3D Object/Scene Representation
- Output – An image of the input object/scene
- Stages
 - Model view Transformation
 - Projection Transformation
 - Clipping and Vertex Interpolation of Attributes
 - Rasterization and Pixel Interpolation of Attributes

Rasterization

- Process of generating pixels in the scan (horizontal) line order (top to bottom, left to right).
- Which pixels are in the polygon



Interpolation of Attributes

- Interpolate the colors and other attributes at pixels from the attributes of the left and right extent of the scan line on the polygon edge.
- Also in scan line order

