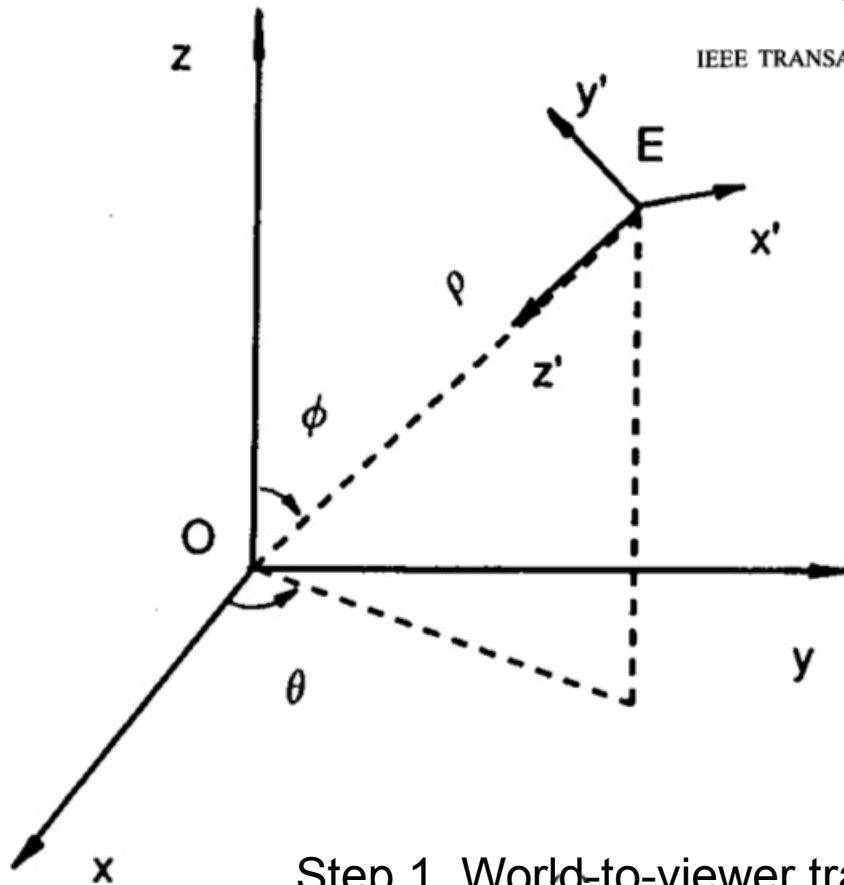


3D Transformation Pipeline Technique

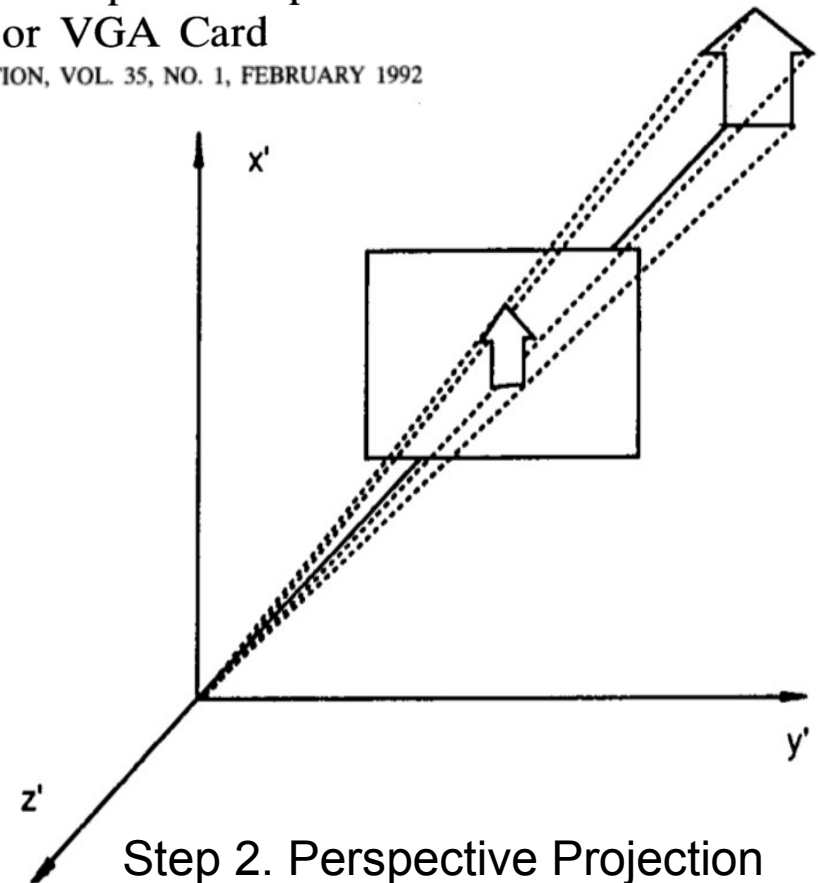
Reference: H. Li Three-Dimensional Computer Graphics

Using EGA or VGA Card

IEEE TRANSACTIONS ON EDUCATION, VOL. 35, NO. 1, FEBRUARY 1992



Step 1. World-to-viewer transform



Step 2. Perspective Projection

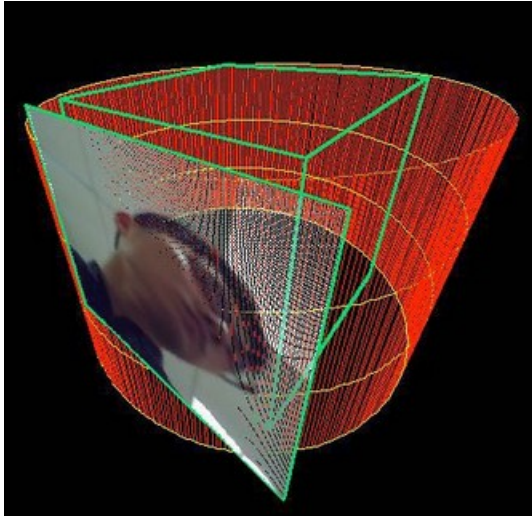
$$\mathbf{T} = \begin{bmatrix} -\sin \theta & \cos \theta & 0 & 0 \\ -\cos \phi \cos \theta & -\cos \phi \sin \theta & \sin \phi & 0 \\ -\sin \phi \cos \theta & -\sin \phi \sin \theta & -\cos \phi & \rho \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$x_p = x_e \left(\frac{D}{z_e} \right)$$

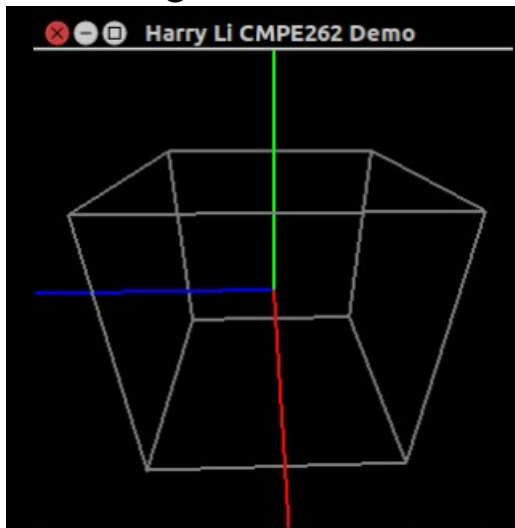
$$y_p = y_e \left(\frac{D}{z_e} \right)$$

Appendix: Example of 3D Transformation Pipeline Program (1)

OpenGL/lecWireframe



Create green frame above



```
/******  
 * Program: wireframe.c  for CMPE262          *  
 * Date: Sept 12, 2013                        *  
 * gcc main.cpp -o main.o -lGL -lGLU -lglut -lm *  
 * Note: linking be sure to have included math lib *  
 *      e.g., -lm                             *  
******/  
  
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <math.h>  
#include <GL/glut.h>  
  
void Display(void);  
void CreateEnvironment(void);  
void MakeGeometry(void);  
void MakeLighting(void);  
void MakeCamera(int,int,int);  
void HandleKeyboard(unsigned char key,int x, int y);  
void HandleSpecialKeyboard(int key,int x, int y);  
void HandleMouse(int,int,int,int);  
void HandleMainMenu(int);  
void HandleSpeedMenu(int);  
void HandleVisibility(int vis);  
void HandleIdle(void);  
void DrawTextXY(double,double,double,double,char *);  
void GiveUsage(char *);
```