02561 COMPUTER GRAPHICS IMM . DTU

Exercise 02561-02 **Projections – Virtual Camera**

Readings

ANG: 1.3.1, 1.4.1, 1.5, 1.6.2, 2.2.1, 2.6, 5.1-5
PRIM: 4.1-6+8, 5.1-6

Purpose

The purpose of this exercise is to understand the various method of setting up the virtual camera and to be able to adjust the parameters of the camera. We will get more acquainted with defining the matrices in the viewing pipeline and to concatenate them into the viewing matrix. Secondly, we will make different pictures of the scene using various projection method based on central projection (Front, X and 3-point perspective) and Orthographic parallel projection (Isometric, Dimetric, and Trimetric axonometric). Many of the principles will be demonstrated in OpenGL.

Part 1

A cube is placed simple in the World Coordinate System (Ow,Xw,Yw, Zw) with a diagonal through the vertices (0,0,0) and (1,1,1).
We want to make an isometric view (based on orthographic parallel projection) of the cube.
Use (0,0,0) as the Point of Interest (At-point) and the Zw-axis in defining the Up-vector. Make an appropriate selection for the "Eyepoint".
Make an OpenGL program that creates the isometric view. Use the function gluLookAt in the program. You may use the program 02561-02-00.cpp as a template.

Part 2

The function gluLookAt automatically sets up (the concatenated matrix) of the viewing transformation. In the viewing transformation you in a series of transformations transform the World Coordinate System to the Eye Coordinate System (or alternatively use the method in Angel).
Define the individual transformations in the viewing transformation and set up the corresponding matrixes.
Replace the gluLookAt function in the OpenGL program with a series of modelling transformations that transform the World Coordinate System to the Eye Coordinate System. You may find it helpful to start with the OpenGL default position and orientation and then check the current view for each transformation step you introduce. The axis may help you not to get lost in space.

Part 3

Find the viewing transformation in part 1 and 2. First find a general expression - it may include finding a series of transformations.

Part 4 (optional)

Make in OpenGL a highly oversimplified aircraft in which the body, the wings, and the horizontal and vertical stabilizers each consist of a box. Use the transformation functions **translate, rotate and scale** to position, orientate, and scale the boxes in an appropriate way. The link
http://www.grc.nasa.gov/WWW/K-12/airplane/airplane.html
defines term used for an aircraft and shows a detailed picture.
Add to the wings two aileron (simplified as boxes) which you can rotate around an edge (in order to get the aircraft to roll).
Optional you may also add boxes to simulate elevators which can change pitch (up and down) and rudders which can change yaw (side to side).

NJC.09.09.2009