COSC422 Advanced Computer Graphics

Assignment 2

3D Character Animation

Due: 11:55pm, Friday, 18 October 2019

Maximum Marks: 25

I. Introduction:

Skeletal and character animation algorithms form an important part of rendering applications involving three-dimensional articulated character models. In this assignment you will implement character animation algorithms using the Open Asset Import Library (Assimp) to generate displays of animation sequences. The assignment consists of three tasks, and *all three* tasks must be completed: (i) animation of a rigged character model using an embedded animation sequence, (ii) animation of a rigged character model using motion data stored in a separate motion file, and (iii) animation retargetting using bvh files. Each of these tasks is outlined below.

Fig. 1. ArmyPilot.x

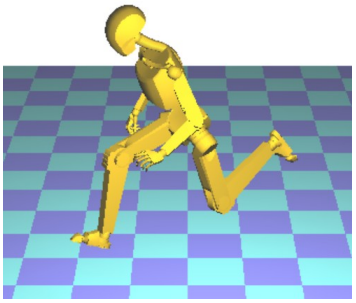

Fig. 2. Mannequin.fbx

Fig. 3. Dwarf.x

II. Task Description:

2.1 Animation of embedded motion data (6 Marks): The model file "ArmyPilot.x" (Fig. 1) contains several animation sequences. Develop an OpenGL program using Assimp data structures and functions to load the character model with textures, and animate it using the first animation sequence (animation index = 0). The scene should contain a floor plane, and the camera should follow the character model so that the model always stays within the field of view. Using the arrow keys, the user should be able to move the camera towards or away from the model, or change the view angle around the object towards left or right as shown in Fig. 4.

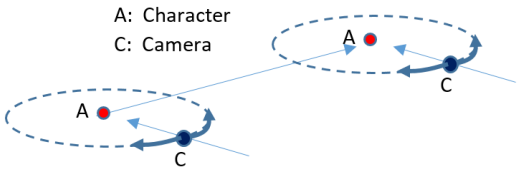

Fig. 4

2.2 Animation of a rigged character model using a separate motion file (6 Marks): Certain character model file formats (eg. FBX) store mesh data and the associated motion data in two separate files. The model file "Mannequin.fbx" (Fig. 2) and a motion file ("run.fbx") are provided. Develop a program using Assimp to load both files and animate the rigged character model using the animation sequence stored in the motion file. As given in Section 2.1, the scene should contain a floor plane, and the camera should follow the animated model.

2.3 Animation Re-targeting (6 Marks): The model file "Dwarf.x" (Fig. 3) contains only one embedded animation sequence. You are given a motion capture file "Walk.bvh" in BVH format. Both the model and the motion files have similar joint hierarchies. In this assignment task, you will use the joint angles for a few joints (eg. knee, ankle) from the motion capture file to generate a walk sequence for the character model. Develop a program using Assimp to load the model including textures, and to display both animation sequences (the embedded animation, and the remapped animation). Use key '1' to display the embedded animation sequence, and '2' to display the re-targeted animation). As given in Section 2.1, the scene should contain a floor plane, and the camera should follow the animated model when displaying the walk sequence.

2.4 Extra Features (3 Marks): You may implement any of the following additional features to gain up to a maximum of 3 marks.

- Continuous animation of walk or run sequences with the floor plane's position updated with character's motion (1 Mark). You may implement this feature for any one of the above tasks.

- Planar shadows (1 Mark). You may implement this feature for any one of the above tasks.

- Vertex blending using weights (3 Marks). Improper transformations resulting from using only unit weights with bone matrices can be clearly seen in the animation of the "ArmyPilot" model. Implement a vertex blending algorithm to rectify these defects.

III. Report (Max. 4 pages; Max. marks: 4):

Please prepare a report describing your work, and include the following sections:

- A brief outline of the implemented methods, including any problems/challenges faced and how you attempted to solve them. Please describe the methods used for positioning the model within the view frame and tracking it during motion. Please also describe the processes used for animation retargetting.
- A screenshot of each animation.
- The complete list of keyboard/mouse functions defined for user interaction.

- List of references to the sources of any additional materials (textures, mocap files, model files) used.

IV. Program Development:

Please develop your programs using the fixed function (legacy) version of OpenGL. You may use programs and other supplementary materials provided in the course (eg. ModelLoader.cpp).

Demo programs found on the Internet and other OpenGL resources should not be submitted as part of the assignment.

V. Assignment Submission

Submit your files using the assignment link on Learn (learn.canterbury.ac.nz) before 11:55pm on 18 October 2019. Your submission must contain:

1. The source codes and all supplementary files (bvh files, mesh files) needed to run your programs. Please do not include freeglut, opengl, glew, glm or assimp library files.

2. Your report in Word or PDF format.

Miscellaneous

1. Check regularly on the *Learn* system forums for spec updates and clarifications. You may submit up to one week late for a 15% penalty.

2. This is not a group project. Your assignment must represent your own individual work. In particular, students are not permitted to share program source code in any way. However, you may discuss ideas, implementation issues etc using the class forum on Learn.

3. Standard departmental regulations regarding dishonest practices and late submissions apply.