COSC422 Assignment 2

Skeletal Animation

Maxwell Clarke

23rd October 2018

Introduction

In this assignment I implemented a program to render animated 3D models using skeletal animation data.

The included 3D models and base code were provided with the assignment specification and lab 11 respectively. The program also utilises the Assimp library to load 3D model files, and the DevIL library to load image files.

I was successful at all tasks however had some trouble with Task 2, as I will describe later.

Controls

The controls for my program are as follows:

1 – Switch to Task 1

2 – Switch to Task 2

3 – Switch to Task 3

Spacebar – Cycle through animations (Tasks 2 and 3 only)

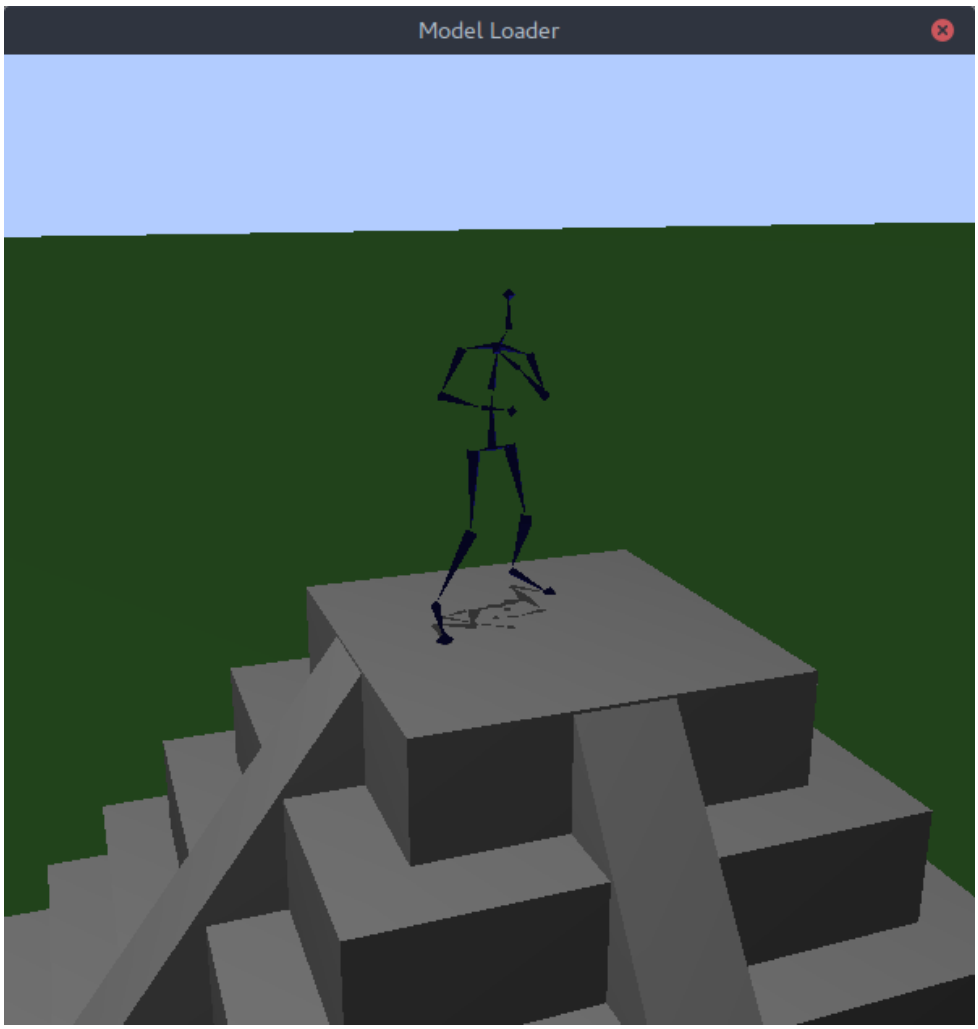
Task 1 – Bio-Vision Hierarchy

Bio-Vision Hierarchy (BVH) files are generated from Motion Capture data and are used in the film industry. It is a text format which defines a skeleton and uniform keyframes defining the rotations of each bone, and the position of the root bone.

It does not include any meshes, so Assimp generates a placeholder mesh.

For Task 1 I have used the provided Boxing.bvh file. The animation plays on a loop.

I have included a few additional features. The first is the Mayan pyramid that the boxing figure stands on. The second is a a shadow – this is implemented by rendering the figure a second time, scaling the Y axis to 0, and coloring the result a mid-grey.



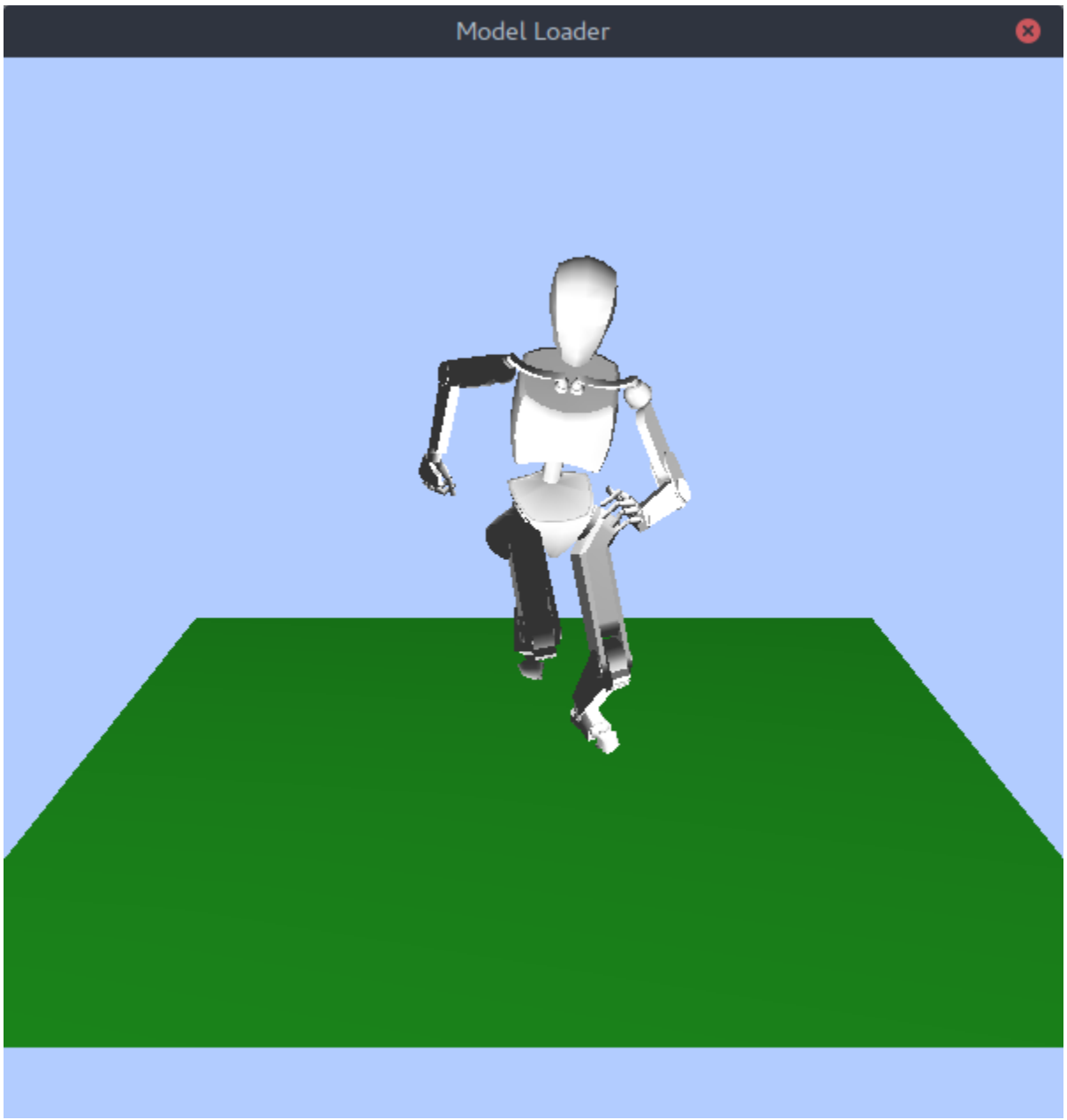
Task 2 – FBX files.

FBX is the format used by the AutoDesk products. It is a general-purpose format and can include both animations and models. In this case however the animations are provided as separate files: walk.fbx, run.fbx and jump.fbx.

Since this model does not have one mesh per node like the auto-generated model from Task 1, a different rendering method was used. This method involved transforming each vertex in a mesh in order to move it into the correct location. I did this by multiplying the transformations of the bone associated with the mesh, and all of it's parent bones, then multiplying the vertex by this product matrix to produce the transformed vertex.

I had trouble with getting this model to render properly. I think this is because the position of the model did not match the scale of the meshes – which resulted in a very small model rapidly moving through space. To make it work I disabled animation of the position of the model. I did this by overriding the position component of the root node with the identity matrix. I believe this introduces some distortion to the animation – especially with jump.fbx. I suspect that there is a hidden bone which has a rotation animation, and is normally offset by the position.

In addition I did not enable vertex weighting for this model – I hardcoded the vertex weight to 1. If I did not do this, I saw some rendering artifacts. Since this is a robot model, which has 1 mesh per bone, it does not require vertex weighting anyway.



Task 3 – X files

X files are an older file format associated with Microsoft's DirectX technology. I used provided the ArmyPilot.x file.

For this file it was necessary to correctly implement vertex weighting for the model to look correct, which I have done.

I have also implemented texturing for the model (this required overriding the "C:/Users/Nathan/..." file path embedded in the file) using the provided textures.

This file has many animations. In my program they can be cycled through with the spacebar.