Report of Assignment 2 for CGRA350

Name: Zhenchuan Yu
ID: 300489509
Username: yuzhen1

1 Introduction

1.1 Functions

`skeleton_model::drawBone()` is the function for drawing the T-pose. This function will invoke `drawJoint()`, `drawLength()`, `drawAxis()` respectively.

`skeleton_model::calBoneTrans()` is invoked in completion and challenge part.

1.2 Program Compiling

The “build” folder has been emptied. It can be compiled by using the usual CMake approach.

2 Core

The “Priman button” is for drawing the whole skeleton:

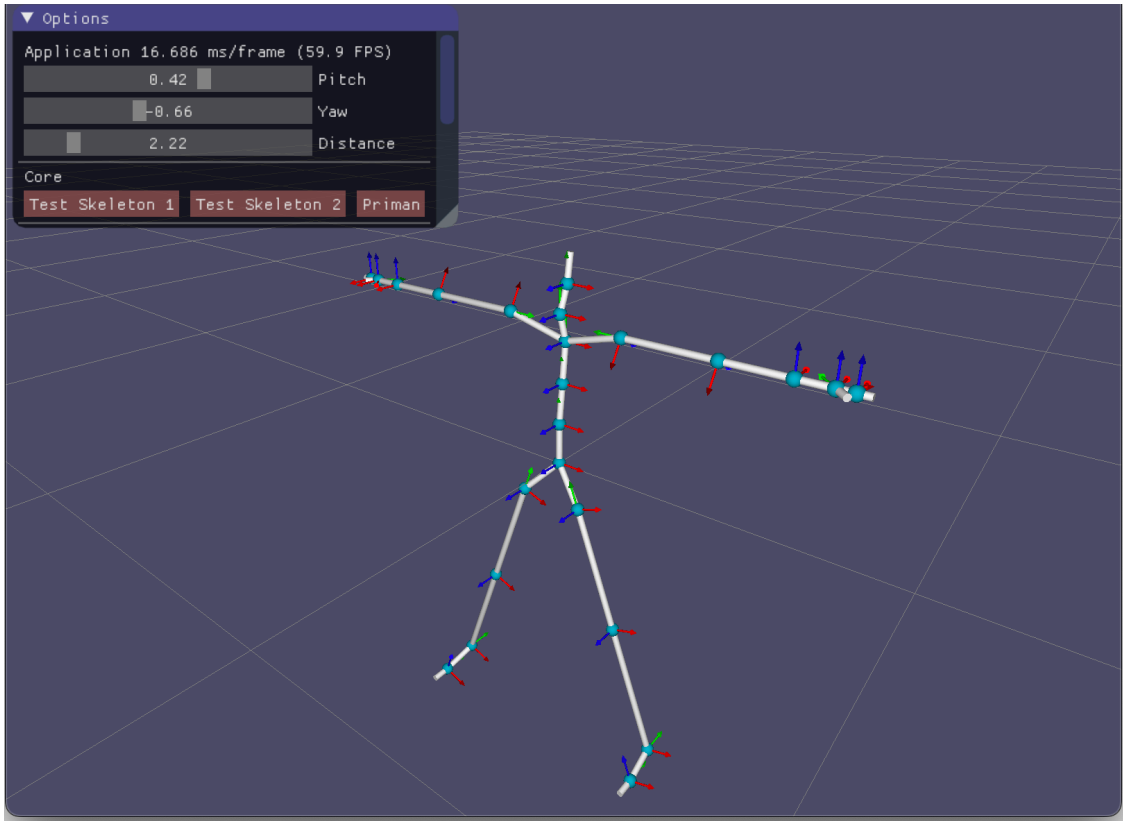

Figure 1: `priman.asf`

The “Test Skeleton 1” and “Test Skeleton 2” buttons are for drawing the test files:

Figure 2: `testskeleton1.asf and testskeleton2.asf`

3 Completion

The three required poses can be read from file: `comp_walking.amc, comp_sitting.amc, and comp_fighting.amc`. The poses can be generated by pressing “Walk”, “Sit”, and “Fight” button respectively. The arbitrary pose I chose is Black Widow’s super hero landing pose (from Iron Man 2).

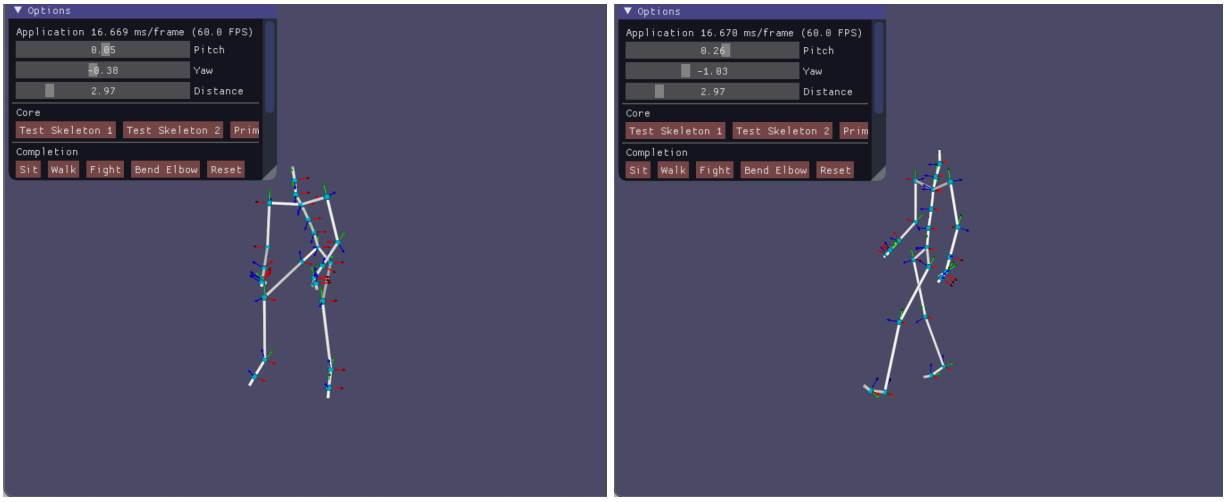

Figure 3: Two required poses

Explanation: I used the `asf` file to draw the pose at the beginning. However, I found that it was hard to rotate the axis to the correct direction. After implemented the challenge part, I found that using `amc` file is a better way to create the pose. The old `asf` files (`priman_walk.asf, priman_sit.asf, priman_fighting.asf`) are still kept in the “assets” folder.

Figure 4: Custom pose

(Image source: https://nypost.com/2021/07/08/black-widow-pokes-fun-at-classic-superhero-landing-pose/)

4 Challenge

The speed of the animation was implemented in a naïve way. By generating an increasing integer in the "big while loop" and passing into the draw function, so that this integer can retrieve the pose as the current frame from the transformation array(std::vector). There are videos in the "output" folder as well.

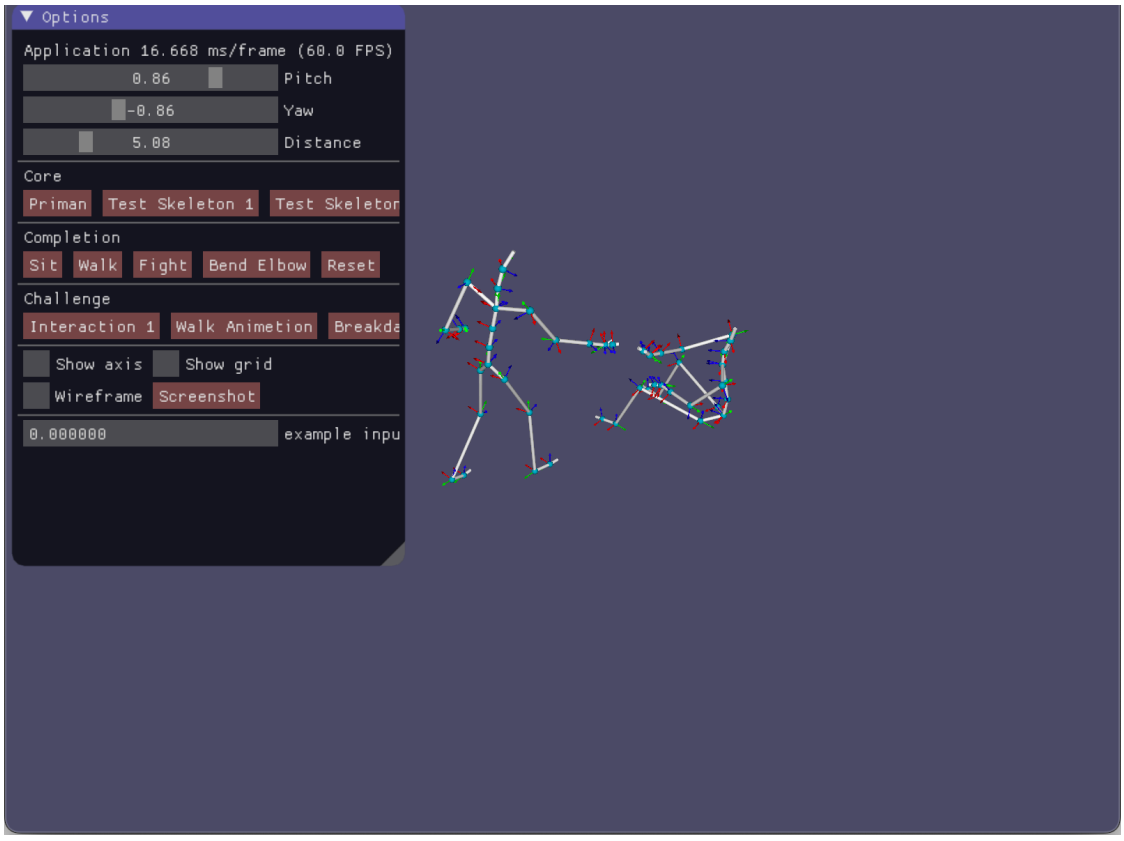

Figure 5: The screenshots of `interaction_s1.amc` *and* `interaction_s2.amc`