CMPUT 307 Lab-5

MoCap Manipulation, Rigging and Skinning

Submission Deadline: **February 25**

You are NOT allowed to import any other package to finish this lab. Please only submit your own work and give credit to any material you referenced.

PART 1

Download any MoCap Player (see "External Resources" section below).
Download the skeleton file (18.asf) and the 3 motion files (*.amc) we provided.
Download the skeleton code file and make sure you understand the code provided.

a) Concatenate the 3 motions in random order and generate a new MoCap file. See external resources for information on the structure of asf and amc files. Note that you need to generate additional transition frames between each two different MoCap files so the transitions are smooth. For example, if you just combine those three MoCap files together and view the result in MoCap Player, you will see sudden changes of the position and pose. Finish implementation of `concatMoCap`. You can use `parse_amc()` to help you finish this part.

```
concatMoCap(input_filenames, n_transition_frames, out_filename):
concatenate the input MoCap files in random order, generate
n_transition_frames transition frames using interpolation between
every two MoCap files, save the result as out_filename.
```

Use MoCap player to visualize the skeleton and the combined animation, make sure the transitions between two motions are smooth.

b) From the course slides, we learned how to do rotation in the local coordinate system. But as you can see from the skeleton file, the coordinate system used for rotation might be different from the local coordinate system. The coordinate system used for rotation is defined by "axis" in the skeleton file, which are the angles to rotate the local coordinate system to the new coordinate system. Answer the following questions (you need to show your work and explain your answers for all 4 questions):

1. Suppose we are using the skeleton defined in the given file. Given the axis parameters $[a, b, c]$, what is the composite rotation matrix M, to rotate the local coordinate system to the coordinate system defined by "axis"? You can use $R_x(x), R_y(x), R_z(x)$ to indicate rotation matrices of angle x.

2. Denote the local coordinate system as $\mathcal{A}$, and another arbitrary coordinate system as $\mathcal{B}$, rotation matrix from $\mathcal{A}$ to $\mathcal{B}$ as M, a joint's coordinates in $\mathcal{A}$ as p, the rotation matrix in $\mathcal{B}$ defined by amc files as N. Derive the formula for calculating the point's new coordinates in $\mathcal{A}$ after applying the rotation N in an arbitrary coordinate system $\mathcal{B}$, show your work

and explain your answer. You can denote the final composite transformation matrix as N'. Hint: you can rotate the coordinate systems.

3. p is defined in the skeleton file by "length", the length of the bone (that is, the distance from the parent joint to the child joint), and "direction". Denote the length by l, and the direction vector by d. Derive the formula for calculating p.

4. The transformations are applied following the hierarchy defined in the skeleton .asf file. From the last two steps we obtained the coordinate rotation matrix M and local rotation matrix N'. Suppose the current bone is a child of a parent bone, and the transformation matrix of the parent's local coordinate system to global coordinate system (the root coordinate system) is M_0. Derive the formula for calculating the coordinates of p in global coordinate system after applying the rotation described in 2. (plug in the formula for p from 3.)

From the result of b) 4. we can calculate the global coordinates of all bones/joints, then we can visualize the animation using those coordinates.

PART 2

1. Install one of these animation software: MotionBuilder or Blender.
Note: You can use the user interface of your animation software to complete this part.

2. Import the skeleton file and any one motion file from Part 1.

3. Download, import or create a 3D human model supported by your animation software.

4. Investigate how to Skin your 3D Human model using the skeleton file.

5. Animate the model using the associated Motion File.

6. Create a video showing the animation of the 3D human model.

Submit the following via eClass (as separate files, **DON'T** zip them):
1. combined.amc: Combined MoCap file you generated for Part 1 a)
2. lab5.py: Part 1 Code
3. lab5.pdf:
   - Answers for Part 1 b)
   - A link to video recorded for Part 2 at the end. You can upload the video to you ualberta google drive and set proper sharing permissions. DON'T upload the video to eClass.

Grading:

- Part 1: 80%
    - a): 40%
        - Code Quality: 5%
        - Comments: 5%
        - Motion Concatenation: 10%
        - Motion Interpolation: 20%
    - b): 40%, 10% x 4
- Part 2: 20%

External Resources you might find helpful:

A sample MoCap Player: http://graphics.cs.cmu.edu/software/mocapPlayer.zip

Understanding the asf/amc file structure:
https://research.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/ASF-AMC.html
http://graphics.cs.cmu.edu/nsp/course/cs229/info/ACCLAIMdef.html

A Matlab toolbox with functions for analysing and visualizing motion capture data:
https://www.jyu.fi/hytk/fi/laitokset/mutku/en/research/materials/mocaptoolbox

CMU Graphics Lab Motion Capture Database: http://mocap.cs.cmu.edu/
Motion Capture Database HDM05: http://resources.mpi-inf.mpg.de/HDM05/

Animation Tool links: https://www.autodesk.com/products/motionbuilder/overview
Motion Builder: https://www.blender.org/download/
3D models:
https://www.turbosquid.com/Search/Motion-Capture/free/fbx
https://free3d.com/
Create a Human tool MotionBuilder Tutorial:
http://www.makehumancommunity.org/