

CS4204 Computer Graphics Fall 2007

Project 3: Animation Authoring Tool

October 31, 2007

1 Due Date

Project 3 is due on Wednesday November 28th, 2007, 11:59pm.

2 Introduction

This project is the final project of the course, and it's a follow-up project based on your previous project (#2) on 3D Robot Modeling Tool.

In this project, you are asked to create an Animation Authoring Tool which can generate animation clips of your 3D robot. The animation technique used in the project is *Keyframing*, which was discussed in the class. Please review the class note (already posted online) for *Keyframing*, if you are not familiar with the technique.

The project is designed to be finished by **ONE PERSON TEAM** only.

3 Project requirement

3.1 Editing features

Your Animation Authoring Tool should have the following **editing** features.

1. (4 points) It uses the 3D robot from your project #2. You can edit the pose the the robot, save and load the pose you edited.
2. (16 points) It can create an animation clip, save and load your authored animation clips into animation files. (You should generate at least one animation file for project submission.) In the file, you need to store all the information you need to re-create the animation clip, which are

the key poses (keyframes). Each animation clip should be stored into only one file.

3. (2 points) It can specify the frame-rate (motion speed) of your animation clip. E.g. 30 frame per second.
4. (12 points) It can specify/set key poses (keyframes) for the editing animation clip by pressing spacebar of your keyboard. Each key pose includes the pose of the robot and its position in the time line (which frame number) . For example, key pose #4 is located at frame #23, and pose #6 is located at frame #60. The length (how many frames) of the editing animation clip is the frame number between the time frame (should always be frame #1) of your first key pose and the time frame of your last key pose. For example, if the last key pose is located at frame 125, the length of the animation clip is 125.
5. (7 points) It can edit and delete the previously set key pose. When using the playback function (described below) you come back to a frame with previously set key pose, you can edit this pose again. And you can also delete this key pose for this frame by pressing key "d".

3.2 Playback features

The playback functionality is essential for all animation tools. Your authoring tool should include the following **playback** features.

1. (11 points) Based on the frame-rate of the animation clip, it can play the animation clip, forward and backward. The keyboard shortcut for play forward and backward is "f" and "b". It can also stop playing the clip, by press the key "s".
2. (5 points) Using keyboard left-arrow and right-arrow, you can display the previous frame and next frame of the animation clip respectively.
3. (11 points) You should be able to use mouse to increase and decrease the "current frame" number of the animation clip. Basically, you can forward and backward play the animation clip by a certain amount frames. By holding "Shift" key and left mouse key, moving the mouse to the right can forward the animation clip; moving the mouse to the left can backward the animation clip.

Note: *Playback function is also useful when you are editing animation clip. For example, after set the key pose for frame #1, you need to increase the*

current frame to some other frame number in order to set next key pose. Also, after you set the key poses, you need to verify the animation by playing the animation.

3.3 Display features

On the OpenGL rendering window of the tool, you should display the following information.

1. (4 points) The frame number of which frame is currently being edited. When play back the animation, the currently frame number should be displayed.
2. (3 points) When you are increase or decrease the current frame number by playing forward, play backward or using arrow keys or using the combination of shift key and mouse, you need to indicate (show on the screen) when a key pose is attached to the current frame. For example, if we have a key pose set to frame 25. When we play the animation clip to frame 25 or drag the mouse forwarding the animation clip to frame 25, somewhere on the screen we need to show that frame 25 is a keyframe.

4 What to Submit

Put your solution in one or more C++ source files. The main file (which includes function main) should be named project3.cpp. Include all your source files in a zip file and send to your TA, Sean Ponce, via email ponce@vt.edu. Please do NOT send the compiled EXE files. VT email server will reject the email. Please also include a description file, called descriptions.txt that describes how to use your program.

Also submit one animation clip file that can be loaded into your animation authoring tool to demonstrate your designed animation. As I mentioned in the class, your animation should look "interesting". So, you might need to add some secondary animations to your robot motion. For example, a bullet is flying by. You can also add some sound effect.

5 Grading

Your overall grade will be based on the following criteria:

- (75 points) Completion of the feature list.

- (25 points) How interesting (in art sense) your submitted animation is.