

Assignment 1 : A Simple Keyframe-Based Animation for Human Head

NAME: LI YUWEI

STUDENT NUMBER: 45895086

EMAIL: LIYW@SHANGHAITECH.EDU.CN

1 INTRODUCTION

In this assignment, I create a simple graphics program for keyframe-based animation of human head using *nanogui* for GUI and *tinyobj-loader* for load model. The system GUI is shown in Fig.1. I accomplished the following functions:

- I create a window-based program for rendering with control panels shown in Fig. 2.
- The program is able to load and render a 3D model with texture.
- Users can drag left mouse to rotate the object, right mouse to move the object and use mouse scroll to change the scale of the model.
- In the control panel in Fig. 2, I used slider for timeline. Users can slide to a specific frame, then rotate, translate and scale the model.
 - Click *Add* to add current frame as a keyframe.
 - Click *Edit* to change the transformation of a keyframe.
 - Click *Delete* to delete a keyframe.
 - Click *Run* to play the animation. Transformation will be interpolated using adjacent keyframes.

2 IMPLEMENTATION DETAILS

Given a frame, first find adjacent two keyframes, a before frame and an after view. If there is no after view, then add the first keyframe as the after keyframe to make a smooth loop animation.

For each view, the transformation are stored in an class TransFactor. As shown in code segment 9, the class consists of three components of a transformation: scale, translation and rotation. For simplify, I use quaternion to represent rotation.

For interpolation, I simply use linear interpolation for scale and translation, slerp for rotation. The equation for interpolating between quaternion p_0 and p_1 at ratio t is shown as follow:

$$\text{Slerp}(p_0, p_1; t) = \frac{\sin[(1-t)\Omega]}{\sin \Omega} p_0 + \frac{\sin[t\omega]}{\sin \omega} p_1 \quad (1)$$

```
1 class TransFactor
2 float scale;
3 float transx;
4 float transy;
5 float transz;
6 float rotationx;
7 float rotationy;
8 float rotationz;
9 float rotationw;
```

3 RESULTS

The program UI is shown in Fig. 1. The object will move smoothly through the sequence during animation.

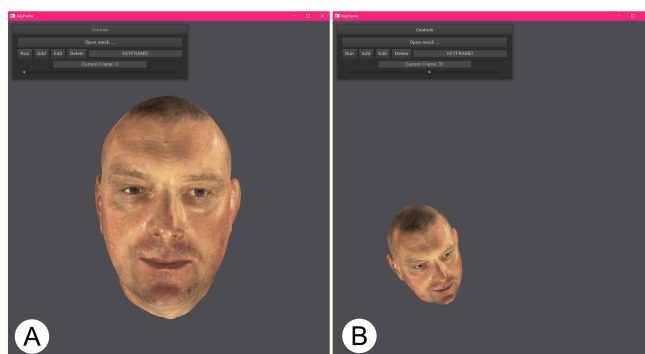


Fig. 1. Program snapshot. (a) Initial state. (b) Add current transformation as a keyframe.

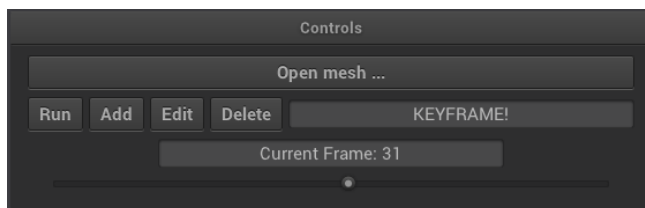


Fig. 2. The control panel of the program. Including a slider and some control buttons.