Motion Capture Interpolation Report

1. Overview

In this project 4 interpolation algorithms have been implemented, including Linear interpolation for Euler angles, Spherical Linear(SLERP) interpolation for quaternions, Bezier Spherical Linear interpolation for quaternions and Bezier interpolation for Euler angles. This report contains the evaluations of all these 4 techniques besides their drawbacks and advantages. In the third part of this report I compared four methods using graphs and video screenshots.

1. Evaluation
2. Linear interpolation for Euler angles

Linear interpolation for Euler angles technique use linear interpolation for both root position and the joints’ rotations.

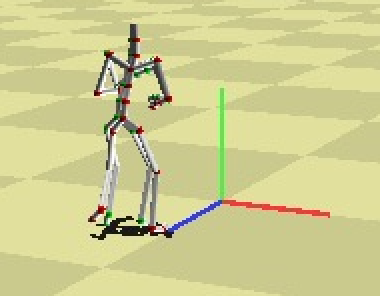
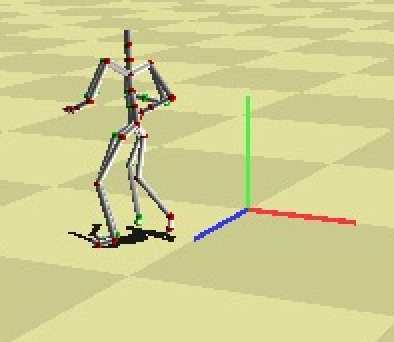
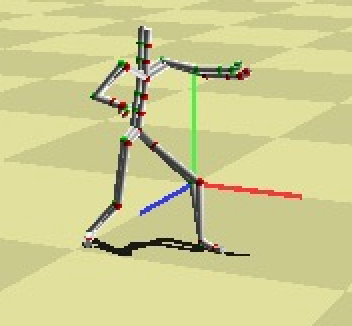
It is the fastest one among these 4 methods. It takes less computation time and when N (the number of consecutive frames waiting to be interpolated) is not huge it is able to generate good motion sequences which are not very far from the input frames. While when the N is larger than the tolerance it might generate very unstable rotations (sometimes the figure will be upside down due to the bed interpolation conditions).

1. Spherical Linear(SLERP) interpolation for quaternions

The only difference between this method and the above one is that it uses SLERP to interpolate quaternions rotations. It generates smoother rotation change.

Because SLERP takes more time than linear Euler angle interpolation, this method costs more time than linear interpolation for Euler angles (around 5 times longer than latter, it will be shown by graph in next part) while it also behaves better which eliminates the upside-down situation.

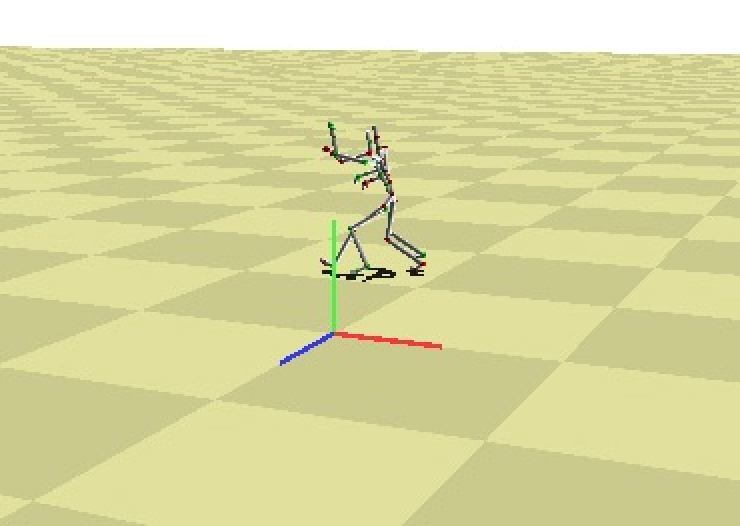
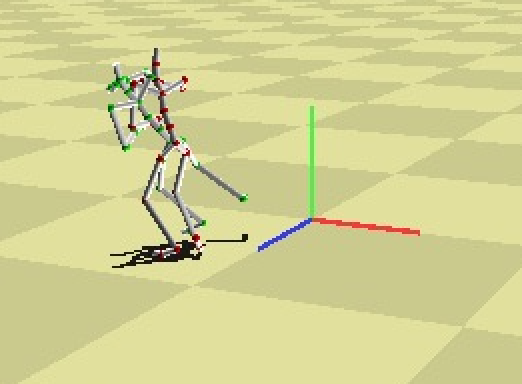
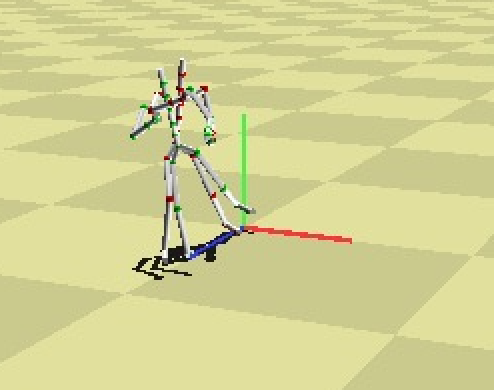
This method deals with rotation interpolations better than Euler angles. Pictures below are from video2.



1. Bezier Interpolation for Euler angles

This method uses Bezier curves for root position interpolation which requires more calculation while it still uses linear interpolation for rotation so it keeps some unstable behaviors when N is large.

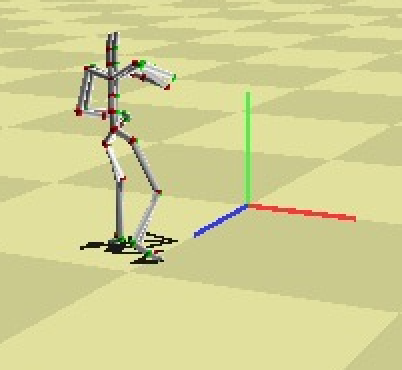
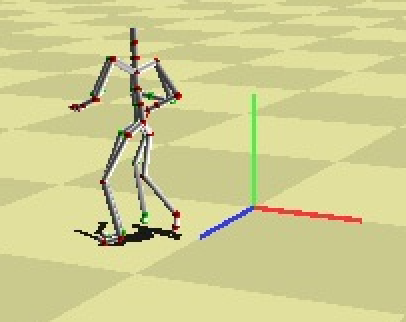
As video1 shows, Bezier Euler method will meet some problems when rotation changes frequently during a short time.



1. Bezier Spherical Linear interpolation for quaternions

Regardless of time cost it shows the best interpolation among 4 methods. It utilizes Bezier curves both in rotation interpolation and the root position interpolation. It works very well even the N is 40 (the graphs will be shown in next part) while it also takes the longest computation time.

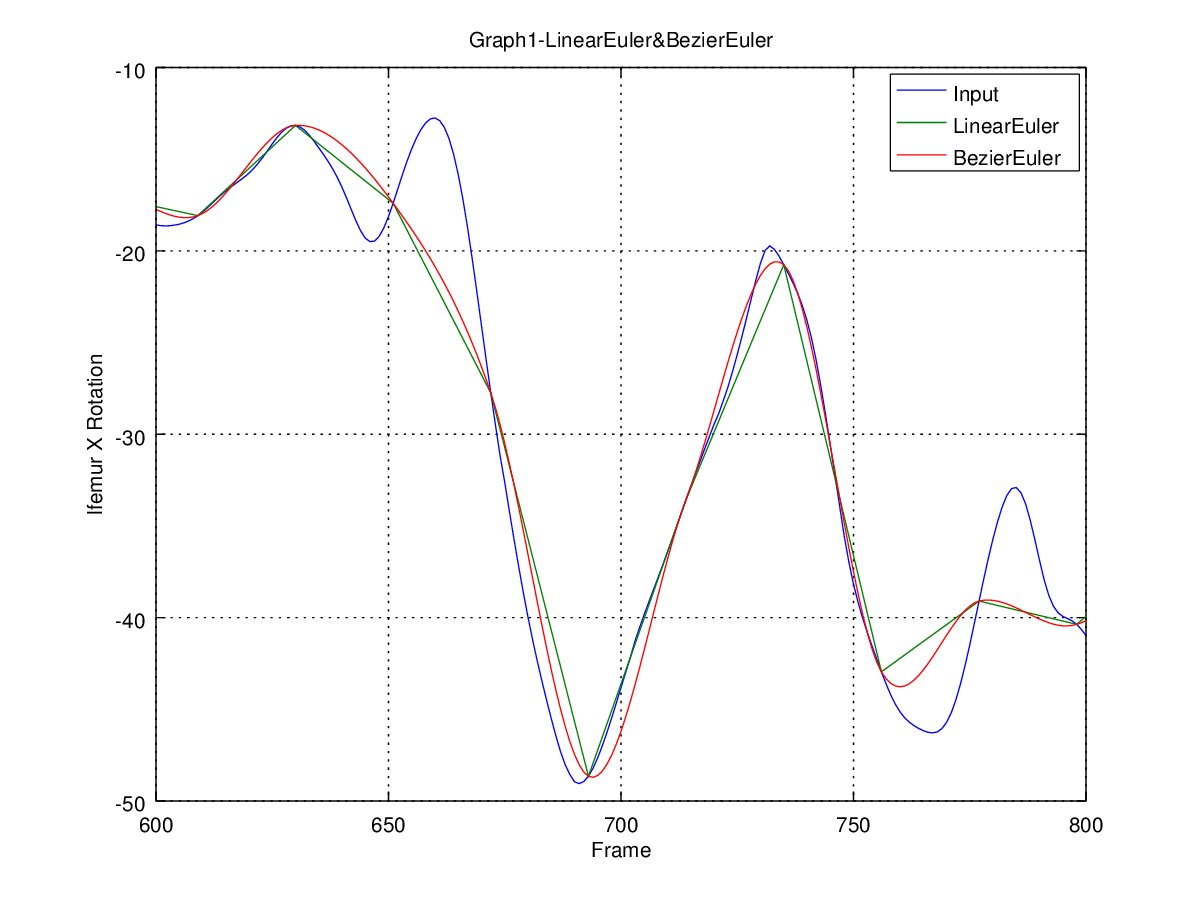
 As video3 shows, this technique works well even in some complicated situation.



1. Comparison

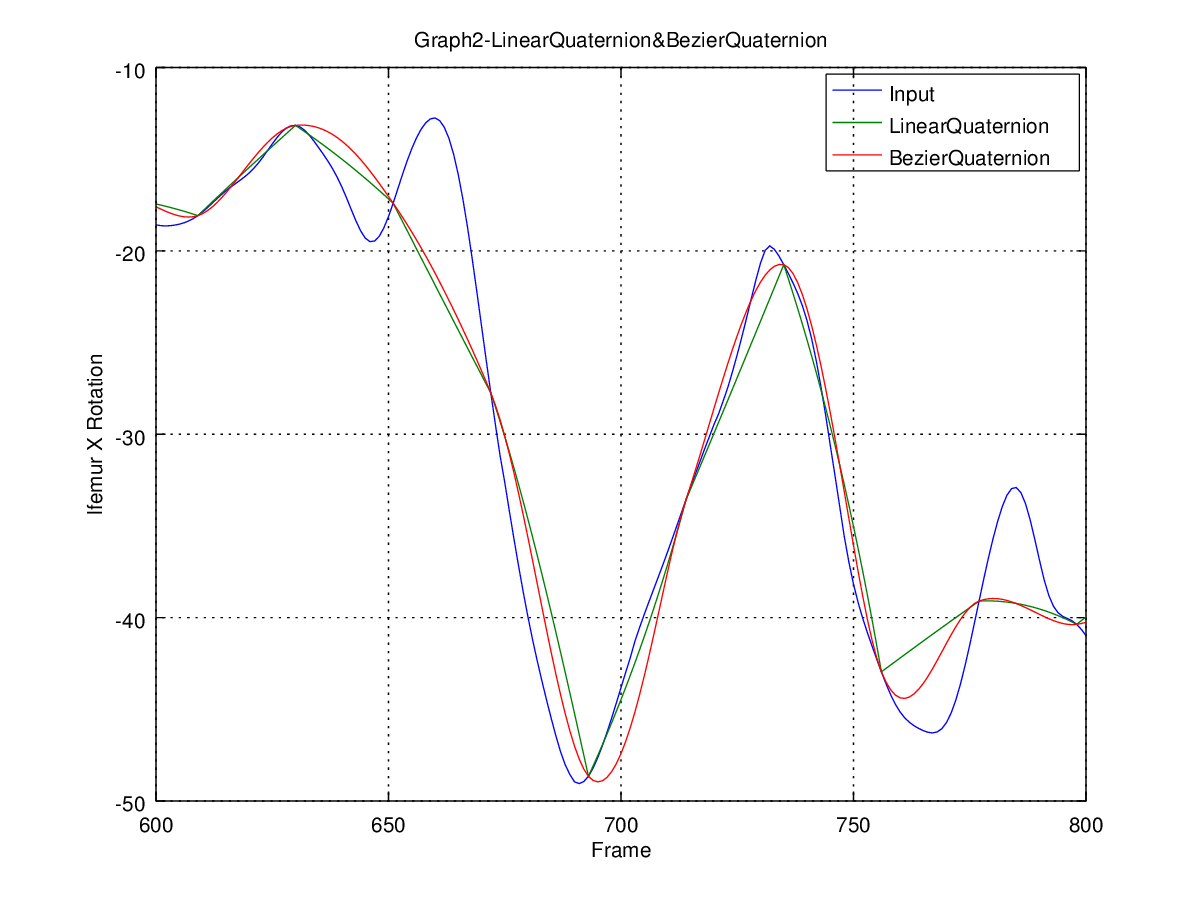
In this part of report I am going to compare these 4 methods.

1. Graph1: compares linear Euler to Bezier Euler (and input)

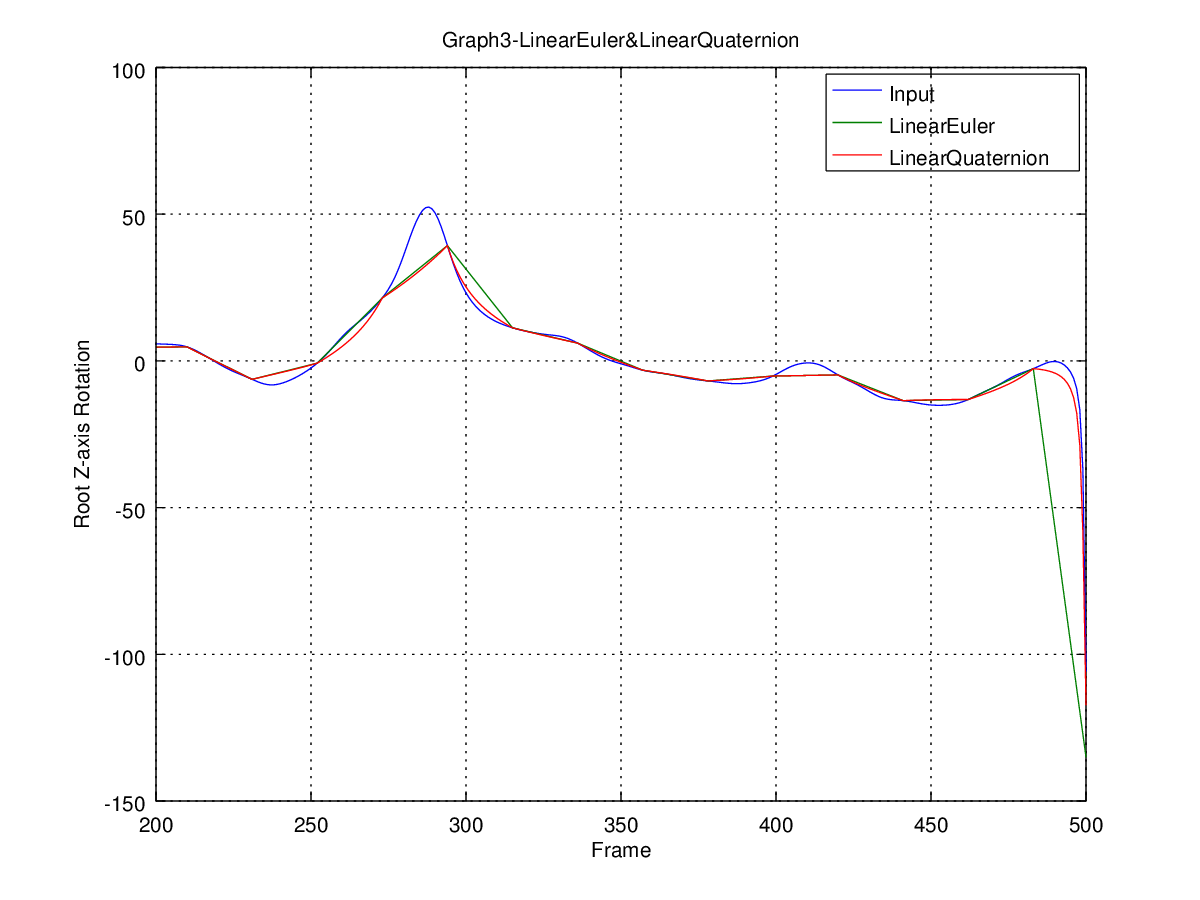


LinearEuler and BezierEuler are very similar but BezierEuler is obviously smoother than LinearEuler.

2) Graph2: compares SLERP quaternion to Bezier quaternion (and input)

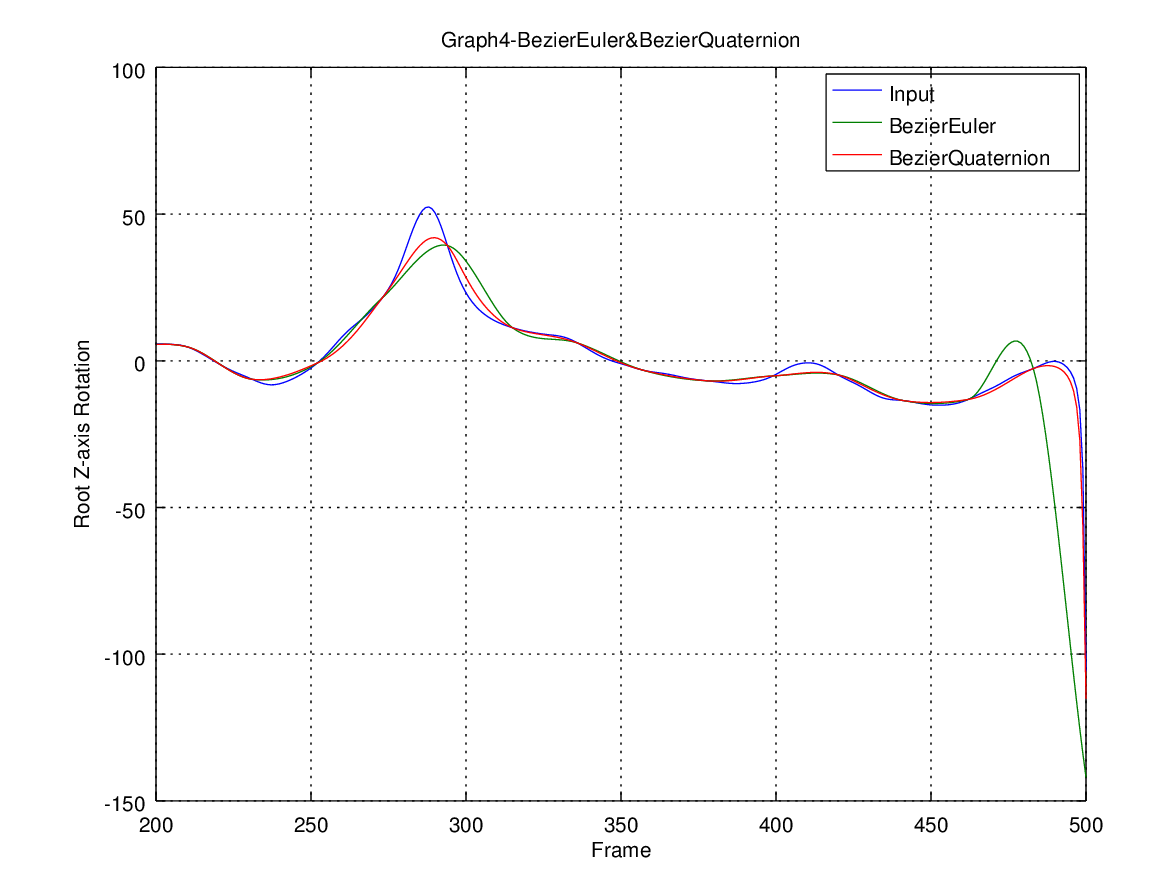


These 2 curse are also similar because they use the same rotation interpolation algorithm.

1. Graph3: compares linear Euler to SLERP quaternion (and input) 

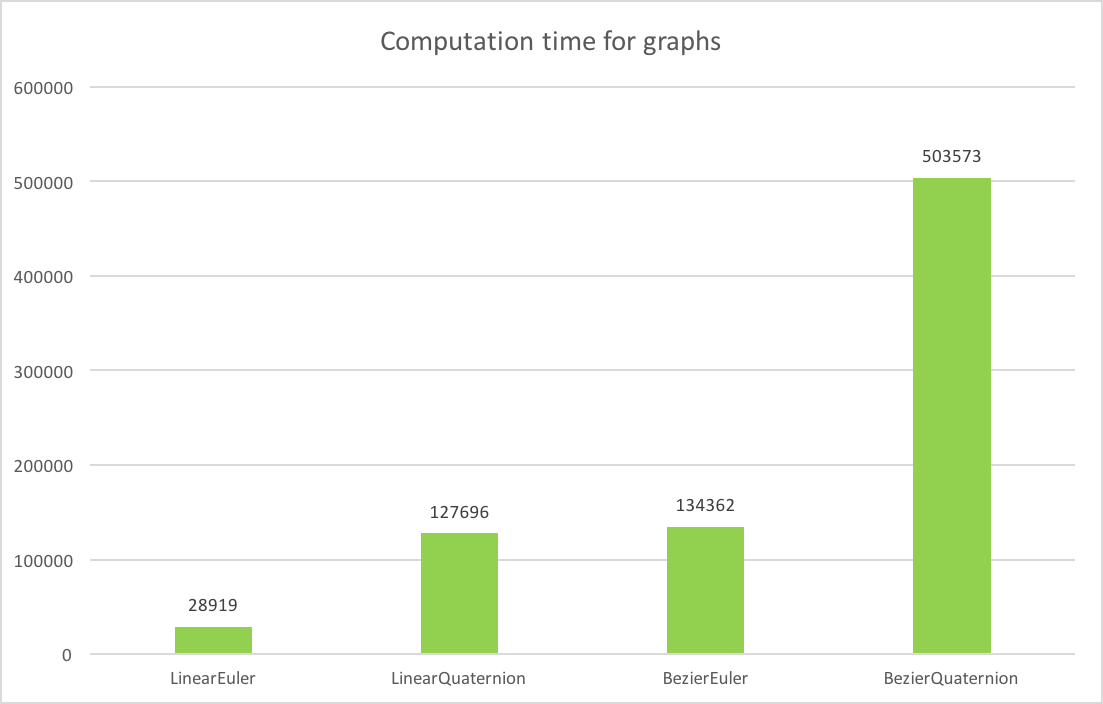
Due to Graph3 we can tell SLERP fits the input curve better than linear Euler interpolation and it especially obvious when it comes to frame 480～500.

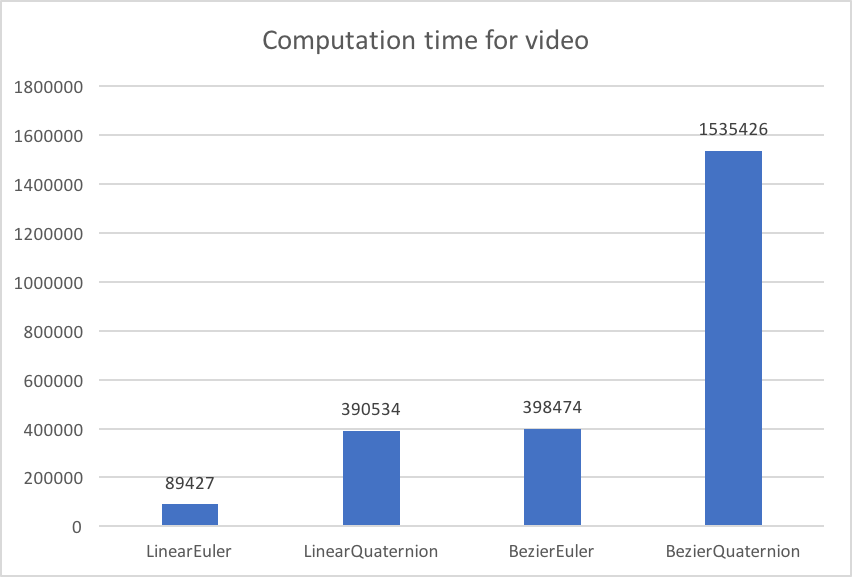
1. Graph4: compares Bezier Euler to Bezier SLERP quaternion (and input)



Graph4 shows us how well the Bezier SLERP quaternion interpolation works and we can also tell BezierEuler method is not bad at this situation. After comparing the time cost we may make a better trade-off between the behavior and the cost.

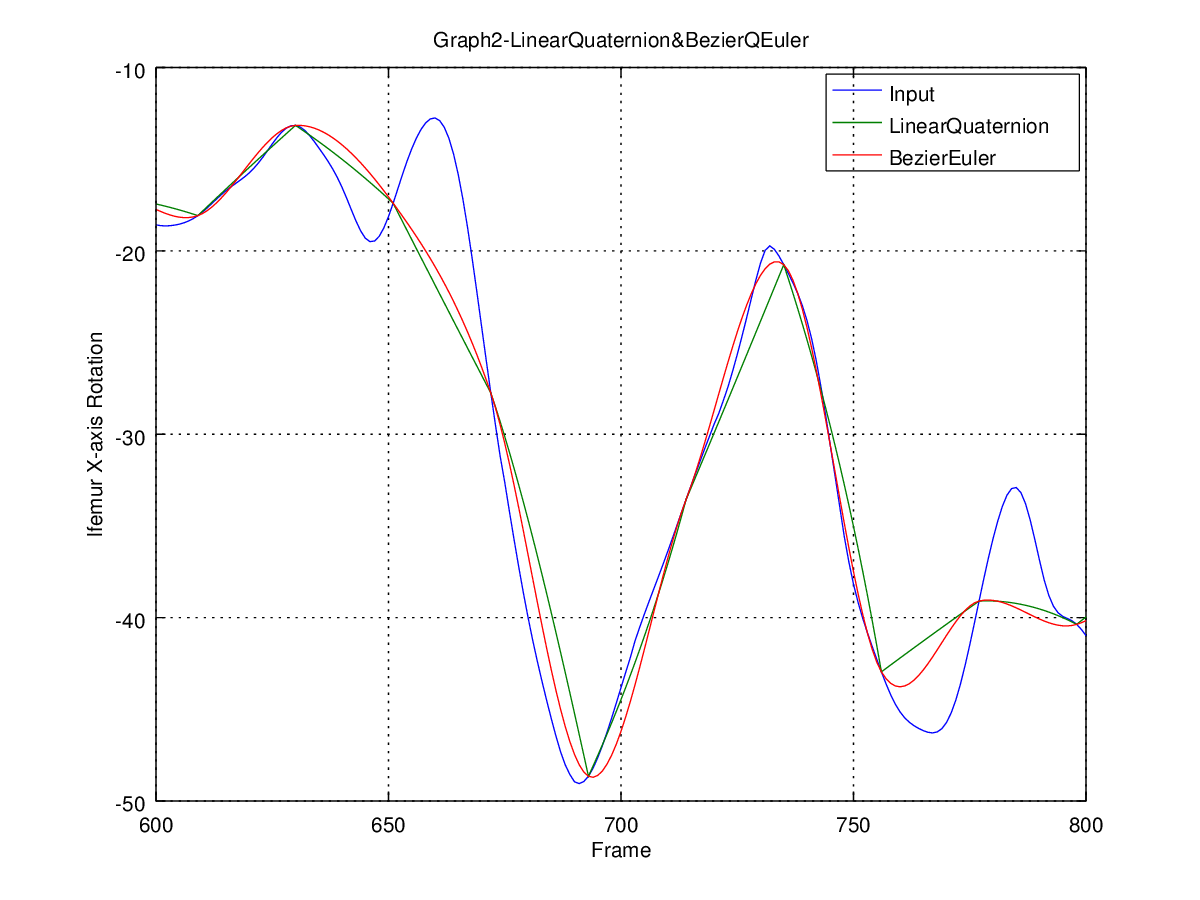
1. Computational time cost comparison





Graph condition is N=20 while video condition N=40. The y axis is time cost, we use clock ticks here which is the unit of tie of a constant but system-specific length. We can clearly know from the pictures above that LinearQuaternion and the BezierEuler cost almost the same computation time while BezierQuaternion costs nearly 200 times more than LinearEuler.

1. Linear Quaternion compares with Bezier Euler (and input)



Due to computation time comparison, I found Linear Quaternion and Bezier Euler have similar time cost. Then I set another experiment to see if the performances are also similar. From the picture above we can tell Bezier Euler is slightly smoother and fit better than Linear Quaternion while both are not bad.

1. Extra Credit
2. Counted the computation time for each kind of interporators;
3. Analyze the computation time of the different interpolation techniques;
4. Added computational texture for ground.

1. Notice

The project is using fltk-1.3.4-1