Sean Yang

10/4/18

Pd 4

Journal Report 5

This week I reimplemented my code based on dlib’s base code for facial landmark detection. Instead of using opencv for face detection, this time I used dlib implementation. Based on research, dlib’s implementation has less misdetections. The key point detection is able to run fairly smoothly on crappy hardware.

There are 69 key points detected on face. Point 30, the point at the center of the nose is the position I chose to compare other points. It’s fairly centered and is one of the most accurate point to detect. The comparison is represented by 1x2 vector with x and y difference. I vertically stack the difference for all 68 points into a numpy array. Then I used Scikit-learn for principal component analysis, trying to reduce the dimensionality of the distance. It is able to output a 1x2 vector.

Currently, the main problems is that all the change in facial position has the same effect on my vector. I would like to make effect of movement of my mouse to be bigger as well as adjust for tilt of face messing with output. I will probably give mouse higher weight than other points.

