Abstract:

By taking the project, Smart Mirror started in Spring 2016 and continued in Fall 2016, we are improving the functionality of the gesture software named *Kinect V2 Mouse Control (KV2MC) [1].* Improved functionality includes the following: left and right hand (dual-hand) recognition and a debug menu that returns values read by the Xbox One (XB1) Kinect to the window form of the program.

Initially, we were going to program the application from scratch, but have decided to use the source code from the *KV2MC* software which is stored in the github. From there we researched on how the kinect was used inside the *KV2MC* software and see how to incorporate dual hand recognition.



[A]



[B]

The code snippet above shows the improved hand checks in the developers application [A] from the original program. In the original program, it only recognized the right and left hands separately. In addition, where the "setcursor" function resides in picture [A], the original program had "duplicate" code for each left and right hands for setting the cursor on the screen. As shown in picture [B], it displays the "setcursor" function and is for the left and right hands to set the cursor on the screen. The code was moved into a function to help the functionality of the program perform better.

We did have some issues in implementing the dual hand recognition. One issue was when the program would hang when the program recognized the two hands. This was due to having to check whether either hand was raised inside the dual hand recognition portion of the hand checks.

First, we tried to implement a gesture such as gripping both hands close together then pulling both hands away from each other for a "zoom in" gesture and vice-versa for a "zoom-out" gesture. Unfortunately, we could not implement this as the program would bog down. To remedy this, we just checked whether or not if both hands were closed or either hands were closed.



[C]

As seen in picture [C], when both hands are recognized (not closed and are raised forward towards the kinect) the program would launch the application "Magnify.exe". This Magnify application would zoom in or out of the screen in specific placese where the cursor is currently at. When both hands are forward and are closed, the program Magnify is closed with the function "killmag()". When both hands are forward and the right hand is closed, the program will send a key-combination "Windows" with the "+" keys to zoom in onto the screen at the last known cursor location. When both hands are forward and the left hand is closed, the program will send a key-combination "Windows" with the "-" keys to zoom out of the screen.



[D]

The killmag function is shown in picture [D]. A try-catch is used as a measure if the program cannot close down the magnify application. The reasoning to closing the magnify application is to not have the magnify glass on the screen when the magnify application is not in use.



[E]

The code above lists sets up the Right hand (X,Y,Z), Left hand (X,Y,Z), and Spine (X,Y,Z) values. These values are used in the set cursor and the dual hands functions.

References:

[1] Kinect V2 Mouse Control:

Website: <http://tangochen.com/blog/?p=2137>

Github: <https://github.com/TangoChen/KinectV2MouseControl>