Visual Perception

Kinect Gesture Recognition project

Introduction

RGB-D cameras provide depth information in addition to the RGB stream, this is useful in many applications including 3D reconstructions, robotics vision, gesture recognition and many more, Kinect device is yet a good and cheap RGB-D camera, Kinect uses structured light to compute the depth information, the aim of this project is to be able to robustly recognize gestures and use them to control a 3D application, ideally on a big screen.

Methodology

The algorithm is taking advantage of the depth information we have and identify gestures as relationships between human joints depth values, the details for the gestures implemented are as the following:

- Move forward gesture: We measure the difference between right and left knees and check if it is bigger than a certain threshold.
- Move backward gesture: We measure the difference between left and right knees and check if it is bigger than a certain threshold.
- Move left: We measure the difference between left and right shoulders and check if it is bigger than a certain threshold.
- Move right: We measure the difference between right and left shoulders and check if it is bigger than a certain threshold.

The above mentioned gestures are to be detected one at a time, they can't co-exist, however the following two gestures can co-exist with each other and the first one can co-exist with any other gesture of the above mentioned gestures, this is because the above 4 will generate a keyboard "key pressed" event, and the following two will generate mouse events "move and click events".

- Move the mouse gesture: We measure the difference between the right shoulder and the right hand and check if it is bigger than a certain threshold.
- Click gesture: If the "Move the mouse gesture" is detected, we measure the absolute difference between the right hand and the left hand and check if it is smaller or equal to a certain threshold.

And lastly we assume that we are in a stand-by mode when no gesture is detected.

Implementation

We are using the official Microsoft Kinect SDK version 1.7, and the Kinect library developed by coding4fun.com, this last one was mainly used to refine the plotting and the scaling of the joints onto the RGB stream in the first prototype we developed. This is developed using WPF and C# on .NET platform.

We acquire all the frames from the Kinect, i.e. the RGB, depth and skeleton frames, we then extract depth values (from the depth frame) of the joints (from the skeleton frame) and feed these values to the algorithm, we may plot these points on the RGB frame. Skeleton frame provides a large variety of joints information.
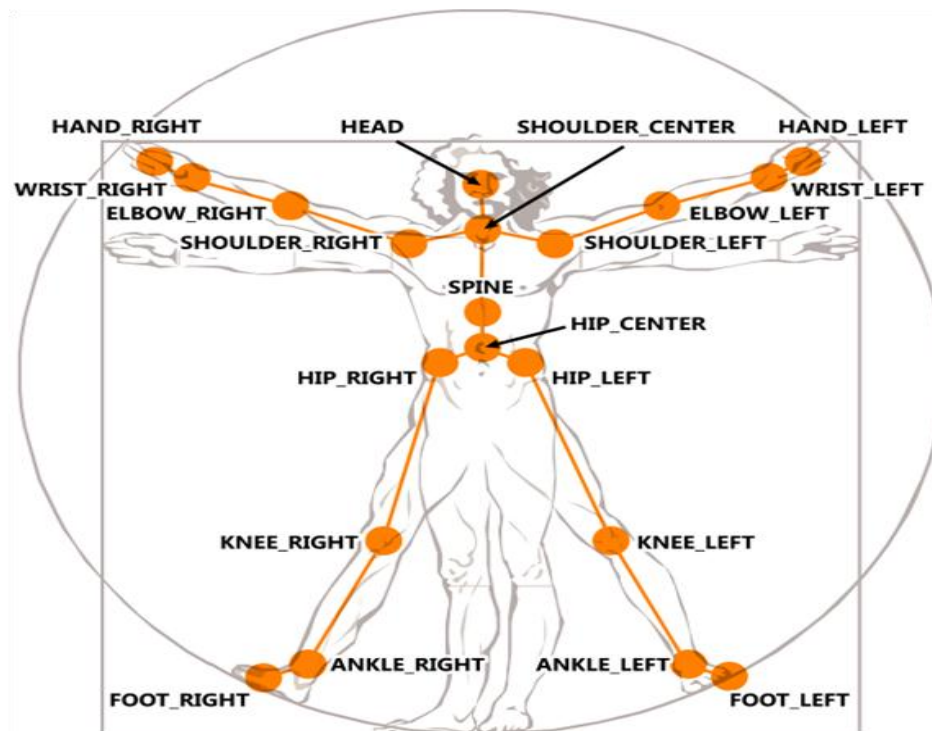
After we identify a gesture, we simulate an event (Mouse or keyboard) and we send it to the operating system, for the mouse movement control, we used a library called KinectMouseController, the library uses the X and Y information of the

right hand (the right hand is the joint responsible for mouse moving gesture detection) with respect to the primary screen size to control the mouse position.

Results and discussion

Since the software was checking every frame and in order to provide robust results we identify any gesture only after certain threshold, a number of consequent frames that confirms that gesture. This works very well and gave great results at the level of gesture detection, however this was still costly in computation time, we changed the code, that it does check only every certain number of frames only one time, the accuracy difference was negligible but the computations cost reduced. This is applied only for keyboard gestures, but for Mouse gestures, we are checking every single frame to update mouse pointer position continuously – and smoothly - when a mouse movement control gesture is detected. We tested that on Google Earth StreetView and Minecraft 3D game, using not-so-powerful computer, the performance wasn't bad, but we anticipate that it could be better using stronger computer.

References:

- Jang-Wook kang, Dong-jun Seo and Dong-seok Jung. "A study on the control method of 3-dimensional space application using Kinect system. IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.9, September 2011.

MOHAMMED Musaab

FLORIAN Jampy

KRISHNA Sai