Robust Part Based Hand Gesture Recognition Using Kinect Sensor

[1]Sohan Belsare, PG scholar in VLSI & Embedded Systems,
[2]Sujatha k. Professor,
[1,2]Shree Ramchandra College of Engineering, Pune, India.

Abstract: **Hand gesture based Human Computer Interaction (HCI) is one of the most natural and intuitive ways to communicate between people and machines, since it closely mimics how human interact with each other. In this, we present a hand gesture recognition system with Kinect sensor, which operates robustly in uncontrolled environments and is insensitive to hand variations and distortions. Our system uses both the depth and color information from Kinect sensor to detect the hand shape, we demonstrate the performance of our system in real life application which is controlling various devices using kinect sensor.**

***Index terms*- Human Computer Interaction, Hand Gesture Recognition, Kinect sensor. Software development kit.**

I. INTRODUCTION

There is always a need to communicate using sign languages, such as chatting with speech and hearing challenged people. Additionally, there are situations when silent communication is preferred for example, during an operation, a surgeon may gesture to the nurse for assistance. It is hard for most people who are not familiar with a sign language to communicate without an interpreter. Thus, software that transcribes symbols in sign languages into plain text can help with real time communication, and it may also provide interactive training for people to learn a sign language. Gesture recognition has become an important research field with the current focus on interactive emotion recognition.

Traditionally, gesture recognition requires high quality stereoscopic cameras and complicated computer vision algorithms to recognizes hand signals the systems often turn out to be expensive and require extensive setup. Despite lots of previous work, traditional vision based hand gesture recognition methods [1,2] are still far from satisfactory for real life applications. development of inexpensive depth cameras, the Kinect sensor [3], new opportunities for hand gesture recognition emerge. Instead of wearing a data glove, using the Kinect sensor can also detect and segment the hands Due to the low resolution of the Kinect depth map, typically of only 640*480, although it works well to track a large object, e.g., the human body, it is difficult to detect and segment a small object from an image with this resolution, Microsoft Kinect provides an inexpensive and easy way for real time user.

ISSN: 2278 – 7798

Kinect, originally designed for gaming on the Microsoft Xbox platform.

This system has several key features:

• Capable of capturing images in the dark.

• Identifying up to two hands, under all reasonable rotations of the hands.

• Translating and displaying gestures in real time.

• Allowing user to choose different scenarios.

The main contributions of this paper are as follows: We demonstrate our hand gesture recognition algorithm in one HCI applications. The proposed system operates accurately and efficiently in uncontrolled environments. It is applicable to other HCI applications.

II. RELATED WORK

A.Concept for hand recognition

Gesture recognition is a fundamental element when developing Kinect based applications or any other Natural User Interfaces. Gestures are used for navigation, interaction or data input. The most common gesture examples include waving, sweeping, zooming, joining hands, and much more. Unfortunately, the current Kinect for Windows SDK does not include a gesture-detection mechanism out of the box, it is always good to define it.

Kinect provides you with the position X, Y and Z of the users' joints 30 times (or frames) per second. If some specific points move to specific relative positions for a given amount of time, then you have a gesture. So, in terms of Kinect, a gesture is the relative position of some joints for a given number of frames. People wave by raising their left or right hand and moving it from side to side. Throughout the gesture, the hand usually remains above the elbow and moves periodically from left to right. fig.1 shows a graphical representation of the movement. The kinect sensor will recognize the position with coding which done using kinect for java.

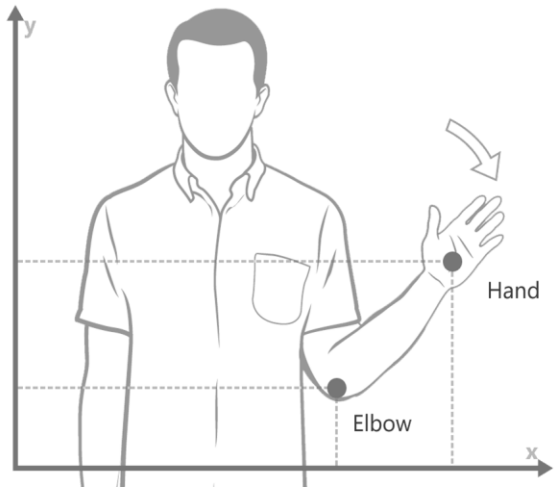


Figure 1: Hand and elbow detection using of x and y axis.

B. Coding concept

For the recognition of hand gesture we will consider the head as the middle pointer and with the point from the head we are set the condition with the minimum parameter. The condition will be always false so we will be recognize the hand position.

```
if(GlobalData.showSkeleton)
 head[1] = -head[1];
 left[1] = -left[1];
 right[1] = -right[1];
scaledHead[0] = (int)((GlobalData.ww/2) +
 ((head[0] * (GlobalData.ww/2 + 180))));
```


ISSN: 2278 – 7798

written the condition to match with properties of kinect sensor using software development kit for java.

```
public class GlobalData

public static int wwFeed = 640;

public static int hhFeed = 480;

public static int ww = 1280;

public static int hh = 720;
```

For selecting the panel for interfacing with hardware and it's condition we will written the following code for selecting port.

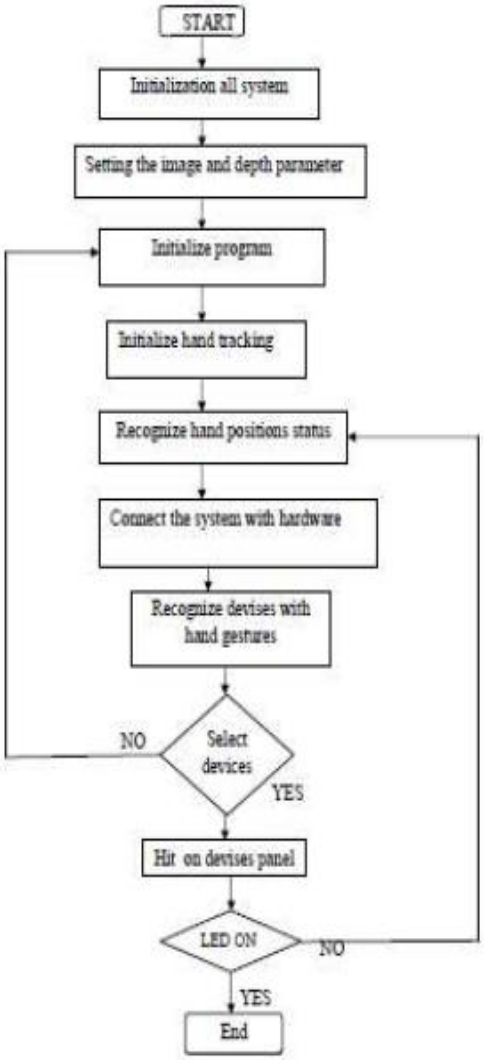
```
Timer myTimer = new Timer();

boolean running;

 public CommPortIdentifier portId;

 public Enumeration portList;

 public InputStream inputStream;

 public OutputStream outputStream;

 public SerialPort serialPort;
```

Also for the button status condition we are shows some code which check the condition of buttons for both hands.

```
 BufferedImage biButtons;

 BufferedImage biLayer1Back, biHandL, biHandR;

 int buttonStatus[][];

 double prevDiffL[], prevDiffR[];

 int selectedButton; //0 --- 7

 int selectionStatus;  //0 inactive,

 public MyForm(String commPort)

biButtons = new BufferedImage[2][2];

 for(int i=0;i<8;i++){

 buttonStatus[i][0] = 0;

 buttonStatus[i][1] = 0;
```

C .Flow chart

The hand gesture recognition procedure using kinect sensor and it's implementation with hardware operation and the various condition for selecting devices step show in following flow chart.



ISSN: 2278 – 7798

III. REQUIREMENTS AND SET UP

A. Hardware Requirement

- A Intel Core™ 5 Processor
- Minimum Quad 2.66 GHz CPU
- Minimum3 GB of RAM.
- Windows 8 operating system.
- Need high speed port of 3.0
- Kinect sensor.

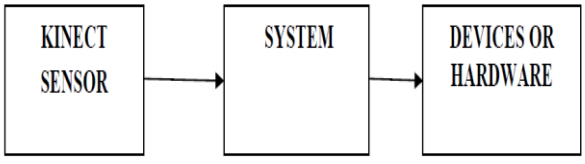
B. Software tools

Netbeans 7.1: NetBeans is a software development platform written in Java. The NetBeans Platform allows applications to be developed from a set of modular software components called modules. Applications based on the NetBeans Platform, including the NetBeans integrated development environment (IDE), can be extended by third party developers

Express PCB 7.0: Express PCB is a CAD (computer aided design) software designed to help you create layouts for printed circuit boards. The program can be really useful for engineers but also students that need to create PCB's for personal projects.

Java 7 : One design goal of Java is portability, which means that programs written for the Java platform must run similarly on any combination of hardware and operating system with adequate runtime support.

C. Setup for experiment



Block diagram for experiment

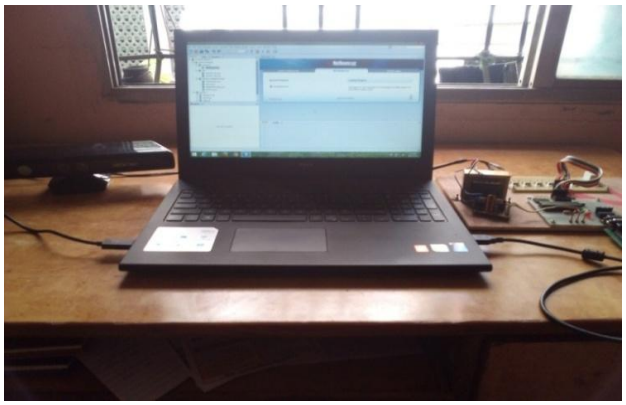
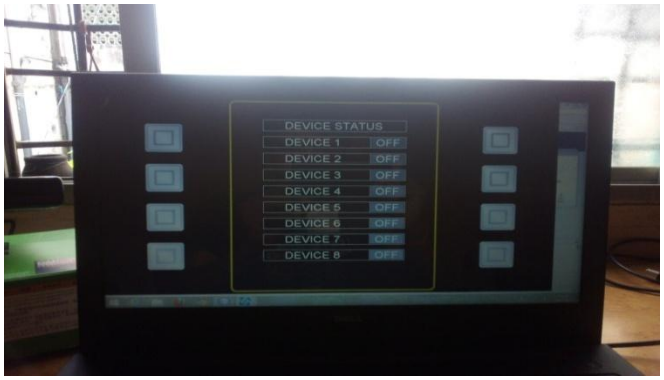


Figure2: Experiment setup image.

IV. RESULTS

Hand gesture based Human Computer Interaction (HCI) is one of the most natural and intuitive ways to communicate between people and machines, since it closely mimics how human interact with each other. In this application, we present a hand gesture recognition system with Kinect sensor.fig 3(a) is the initial status and also the control panel setting for the interaction with hardware.fig 3(b) are the running status in that the kinect sensor will recognize hand gesture which are shown in the yellow colour in the hand shape.

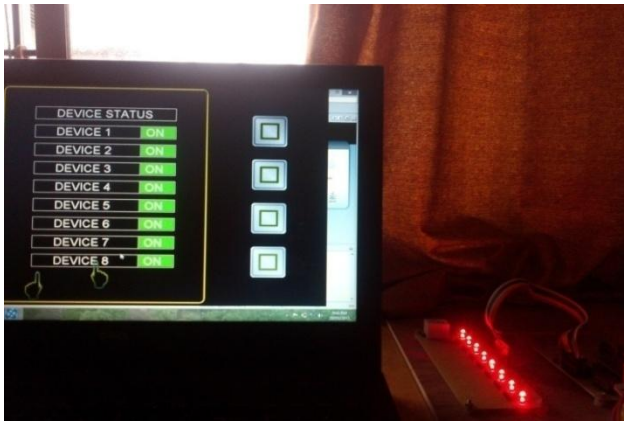
The left four panel are initialize first four devices using left hand. Also the devices from fifth to eight will initialize with right hand.



(a)

ISSN: 2278 – 7798

(b)



(c)

Figure 3: (a) When the project is running at that time shows the initial status of system. (b) After selecting the square device will selected. (c) when all device will selected that time on hardware all LED gets ON.

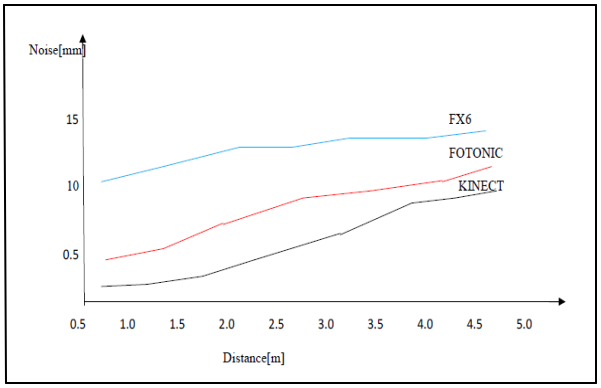


Figure 4: Distributed noise depending on the distance.

The tested conditions are distance of the sensor to the object, environmental illumination and the surface of the object. It is found that for the most sensors, distance has the biggest influence. Further is found, that the absolute error of the radius increases with the distance for Time of Flight Sensors.

V. APPLICATION

A great emphasis on Human Computer Interaction (HCI) research to create easy to use interfaces by facilitating natural communication and manipulation skills of humans. Among different human body parts, the hand is the most effective interaction tool because of its dexterity. Devices controlling by kinect sensor:

In the figure 3(c) we are showing the various devices which controlling by the hand gesture. For the left hand recognition shows the control of device 1 to 4.and right hand shows the control of devices 5 to 8.we are setting the axis x, y for the hand recognition and the z axis for selecting the devices to turn it on and off.

VI. CONCLUSION

In this system based on Microsoft Kinect for Xbox is introduced. The system is motivated by the importance of real time communication under specific situations. such as chatting with speech and hearing challenged people. It is capable of working in the dark, invariant to signer's skin color, clothing, and background lighting conditions, and it can be easily transplanted to other applications.

In this demo, we present an efficient and accurate hand gesture recognition system using Kinect sensor as the input device. Both the depth and color information obtained from Kinect sensor are used for

ISSN: 2278 – 7798

hand detection gesture recognition module provides an effective mechanism for recognizing hand shapes with input variations and distortions. Such a hand gesture recognition system provides a robust solution in real life HCI applications, which can also be applied to many other hand gesture based HCIs.

REFERENCES

[1] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly, "Vision based hand pose estimation: A review," Computer. Vision Image Understand., vol. 108, pp. 52–73, 2007.

[2] G. R. S. Murthy and R. S. Jadon, "A review of vision based hand gesture recognition," Int. J. Inf. Technol. Knowl. Manage., vol. 2, pp. 405–410, 2009.

[3] J. Shotton, A. Fitzgibbon, M. Cook, and A. Blake, "Real-time human pose recognition in parts from single depth images," in Proc. IEEE Conf. Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 2011.

[4] Z. Ren, J. Yuan, and Z. Zhang, "Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera," in Proc. ACM Int. Conf. Multimedia, 2011, pp. 1093–1096.

[5] X. Bai, L. J. Latecki, and W.-Y. Liu, "Skeleton pruning by contour partitioning with discrete curve evolution," IEEE Trans. Pattern Anal. Mach. Intel., vol. 29, pp. 449–462, 2007.

[6] X. Bai and L. J. Latecki, "Path similarity skeleton graph matching," IEEE Trans. Pattern Anal. Mach. Intel., vol. 30, pp. 1–11, 2008.

[7] J. P. Wachs, M. Kolsch, H. Stern, and Y. Edan, "Vision based hand gesture applications," Commun. ACM, vol. 54, pp. 60–71, 2011.

[8] Antonis A Argyros and Manolis I A Lourakis. Binocular hand tracking and reconstruction based on 2d shape matching. fCPR, pages 207-21O, 2006.

[9] P. Buehler, M. Everingham, D. P. Huttenlocher, and A. Zisserman. Long term ann and hand tracking for continuous sign language TV broadcasts. In British Machine Vision ConjcTence, 2008.

[10] The Duy Bui and Long Thang Nguyen. Recognizing postures in vietnamese sign language with MEMS accelerometers. Ieee Sensors Journal, 2007.

[11] M.C. Su, "A fuzzy rule based approach to spatio temporal hand gesture recognition," IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 30, pp. 276–281, 2000.

[12] C. Kwok, D. Fox, and M. Meila, "Real-time particle filters," Proc. IEEE, pp. 469–484, 2004.

[13] A. Doucet, N. de Freitas, and N. Gordon, Sequential Monte Carlo in Practice. New York, NY, USA: Springer Verlag, 2001.

[14] C. Chua, H. Guan, and Y. Ho, "Model-based 3D hand posture estimation from a single 2D image," Image Vision Computer., vol. 20, pp. 191–202, 2002.

ISSN: 2278 – 7798