Research of hand gesture using Kinect based on finger recognition

Zhuo Shi[1, 2, 3], Yinghui Li[2], Ke Yu[2], Yuanquan Cheng[2], Changshao Zhou[2*]

[1]*South China University of Technology, Guangzhou, Guangdong 510275, China*

[2]*Computer Science & Engineering of the Guilin University of Electronic Technology, Guilin, Guangxi 541004, China*

[3]*Guangxi Key Laboratory of Trusted Software (Guilin University of Electronic Technology)*

Received 1 March 2014, www.cmnt.lv

Abstract

Using the depth image from Kinect, we present a novel method to archive the Hand Gesture Recognition (HGR). Firstly we overview the major technical components of the complete method. Then we elaborate several key challenges such as palm recognition, contour analysis and fingertip detection. We present the Biggest concave point detect schemes to contour analysis which is the key to achieving the HGR. Finally, we demonstrate the feasibility and effectiveness of the method by running the system.

Keywords: human-computer interaction, hand gesture recognition, finger recognition, Kinect

1 Introduction

In our life, we sometimes have to communicate with others by using hand gesture or sign language. For example, the place where quiet is required, and the cooperation between doctor and nurse in the operating room, etc. But most of us are not familiar with some special gesture, let alone sign language. In order to solve this problem, we need a way to help us get the meaning of some gesture or even sign language. In this paper, we get the depth data from Kinect, then we focus on the hand area, after a group of processes, we get finger recognition, finally meaning of hand gesture is display on the screen.

In order to archive a believable HGR system, several challenges have to address. First, the skeletal tracking in Kinect for Windows SDK cannot provide the finger recognition, so we need to focus on the research of finger tracked. Second, we are only interested in the hand region, thus it must be segmented from the background. Third, a believable finger recognition algorithm is needed. After read a lot of paper, we opt to use an improved Convex and Concave Angle Recognition Algorithm [1].

We will first discuss related previous work in Section 2. We then give an overview of our HGR system in Section 3 followed by details of the above mentioned key components. Lastly, we present our HGR system, and discuss the limitations of the current system for potential future improvements.

2 Related Work

There are about two main methods for dynamic hand gesture recognition: one is based on hardware, for example, data glove; the other one is based on the computer vision. At first, almost all the HGR are based on the wearable device [2-4]. By using data glove, we can directly get details of the hand joints. When the computer gets the data, after some analysis and process, we can quickly get feedback from computer, Human-Computer Interaction (HCI) complete. In China, there is already some research on Chinese Sign Language by using data glove or some other devices [5, 6]. For a period time, it is really useful for HGR or HCI by using extra hardware. But then people found it was not convenient, not only for researchers, but also for users. Because this hardware is expensive, the common user cannot afford one. What's worse is that users should be wearing the device all through the recognizing. It seems inconvenient indeed. In order to solve this problem, researchers noticed our natural hands, and then they focused on the recognition of natural hands only by computer vision. HGR based on computer vision is mainly divided into two aspects: coloured markers at the tip of fingers [7] and natural hand recognition [8, 9]. The former one requires coloured markers at tip of users' fingers; it is still complicated for us. Nowadays HGR with non-invasive, natural and convenient features is the trend of the research. According to the different technologies, it can be divided into 2D images and 3D images. With common camera, we can quickly get 2D images and then analyse what we are interested in. It seem to be simple and will not bother users, but there are lots of difficulties in distinguishing the target region from the complicated background, let alone recognizing gestures, and it's not easy to make computer get the meaning of any movement. The recognition rate that based on 2D images cannot satisfy the researchers

[*]***Corresponding author*** e-mail: shzh.cn@gmail.com

despite of growing amount of excellent algorithms and methods. It begins a new era to see the existing problems in the area of HGR, as the sensors which can give depth information, are available. Microsoft Kinect, which is released in 2010, is one of the examples and it is able to detect individual motion. The Kinect cannot provide the finger recognition, but we focus on the depth data, combining with the traditional algorithms on the 2D images, then we optimize the algorithm and enhance believability and feasibility of the system.

3 System overview

3.1 GETTING DEPTH IMAGE FROM KINECT

The Kinect contains infrared camera and PrimeSense sensor to compute the depth of the object while the RGB camera is used to capture the images. The depth images and RGB image of the object could be getting at the same time. This 3D scanner system called Light Coding, which employs a variant of image-based 3D reconstruction. The depth output of Kinect is 11 bit with 2048 levels of sensitivity [10]. Supposing that the dimension in *x*-*y* coordinate system of image is $w \times h$, the real coordinate system is like Figure 1.

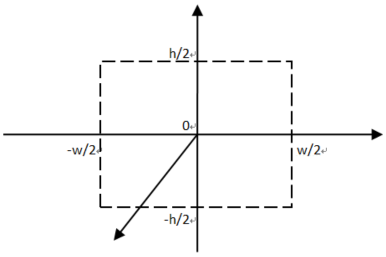

FIGURE 1 The Kinect coordinate diagram

The depth value of a point in 3D can be defined as calibration procedure [11].

$$d = K \tan\left(\frac{d_{raw}}{2842.5} + 1.1863\right) - O\,,$$

where d is the depth of that point in m, K = 0.1236m and O=0.037m.

According to the depth of the point, we can get the location (i, j, d) of the pixel in depth image from Kinect, then we transform (i, j, d) into (x, y, z), which is the location in three-dimensional system of coordinate in reality. The transform equation is as follows:

$$\begin{cases} x = \left(i - \frac{w}{2}\right)(d-10)\,s\,\frac{w}{h} \\ y = \left(j - \frac{h}{2}\right)(d-10)\,s \\ z = d \end{cases}.$$

3.2 HAND TRACKING AND SEGMENTING

From above, we can get the height of user, the schematic shows in Figure 2.

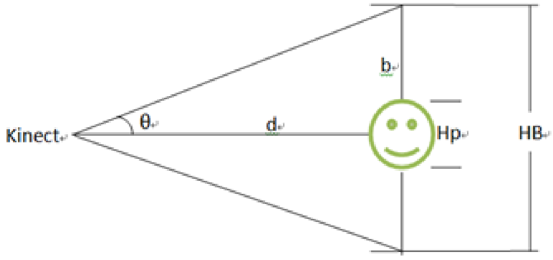

FIGURE 2 The principle diagram of the height measurement

The equation is as follows:

$$b = d\tan(\theta)\,,$$

$$\frac{Hp}{HB} = \frac{Hr}{2b} \underset{equals}{\Rightarrow} Hr = \frac{2bHp}{HB}\,,$$

d is the distance from the point to Kinect, θ is the vertical angle above the horizontal line, Hp is the height of user's pixels, HB is the height of background pixels and Hr is the height of user's body, what we want to get. According to the related research [12], we know that there are some statistical relations between the height of body and length of hand. From [12] we can get the following equation:

$$HL = \frac{Hr - 75.70178}{5.082380}\,,$$

where HL is the length of hand, Hr is height of user.

As we know, Kinect can track hand but finger detection. We now get the hand length, the hand data should be in the spherical area, which radius is half of the hand length, and the centre is hand point, the equation is as follows:

$$HandPoint = \left\{ p \,\middle|\, \left\| p(x,y,z) - p(x_0,y_0,z_0) \right\| \le \frac{HL}{2} \right\},$$

where $p(x_0, y_0, z_0)$ is hand point, HL is hand length. After this step, we can get the hand region what we want, and make convenient for hand processing.

FIGURE 3 Hand tracking and segmenting

3.3 PALMCONTOUR RECOGNITION

In last part, we have got the hand region. And now we need to simplify the coordinate system, so we put them on plane projection. We define the distance between two points like this:

$$D(P_1,P_2)=\sqrt{(x_2-x_1)^2+(y_2-y_1)^2} \ ,$$

$P_1(x_1,y_1)$, $P_2(x_2,y_2)$ are the two points.

Clustering is the task of grouping a set of objects in such a way that objects in the same group. We use *K*-means clustering algorithm [13]. We put *N* pixels into *K* pixel clusters, then we get. Every time before the new pixel gets into one cluster, it should be chosen the one by the value, which is mean value of all the points' range in the cluster. After the iteration we get the minimum value in different cluster

After the *K*-means clustering, we process these pixels by setting the threshold value of pixels in a cluster. In this way, we can eliminate the interference of noise pixels, and then we get the mask image like in Figure 4.

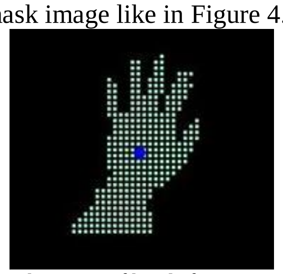

FIGURE 4 The mask images of hand after K-means clustering algorithm

Next we use the graham scan algorithm to detect the convex hull sets of the hand image above, and we choose Moore neighbourhood algorithm to detect the contour of the mask of hand. The two algorithms processes show in Figures 5 and 6. After contour recognition we define centre of palm circle as the centre of hand instead of the centre of hand cluster. Because centre point will be used to compute the direction of each finger, we need a more stable point. The differences are showed in Figure 7.

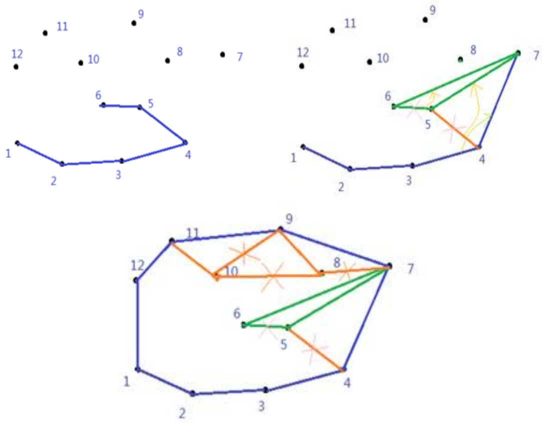

FIGURE 5 Graham scan algorithm process sample

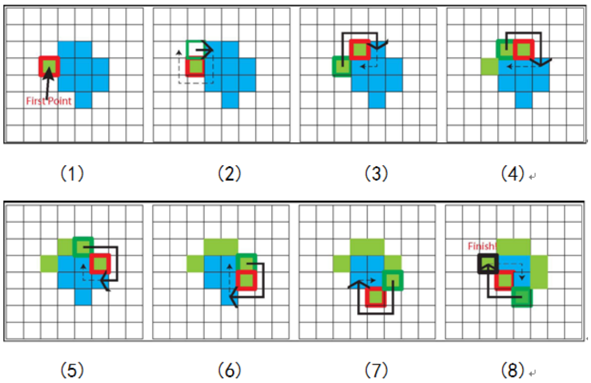

FIGURE 6 Moore neighbourhood algorithm process sample

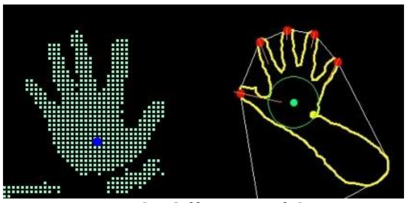

FIGURE 7 The differences of the centre

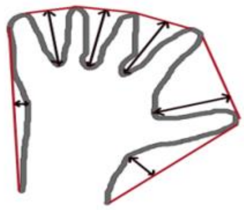

FIGURE 8 Biggest concave point detect schemes

3.4 FINGERTIPS DETECTIONS

Now we present recognition algorithm of concave and convex angle to detect fingertips. Firstly, we need to find the farthest concave point, which locates between two convex points. In order to get the points, we need the hand contour point chain table and concave point set chain table, both of the data structure are generated before. Secondly, we connect the one concave point to the adjacency convex point and connect it to the concave point on the other side, then get the angle of the two lines. If the angle is less than the threshold value, abandon it, or else the convex point is the fingertip we detect. The principle is showed in Figures 8 and 9, and the end of fingertip detection is showed in Figure 10.

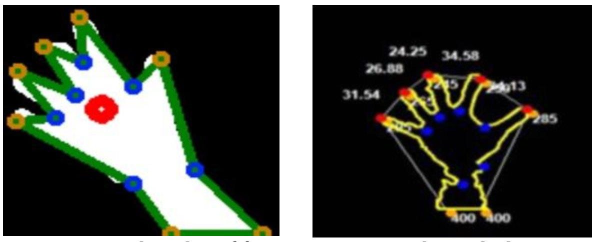

FIGURE 9 Algorithm of fingertip recognition by angle diagram sketch

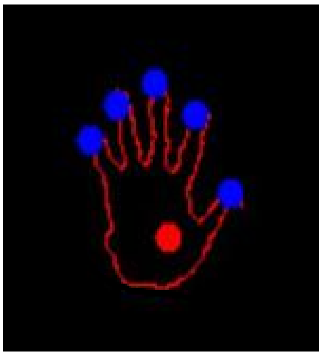

FIGURE 10 Fingertip Detection

3.5 HAND GESTURE RECOGNITION

In this part, firstly we recognize the different fingers, and then confirm the direction of each one. The direction is start from palm centre to fingertip. We use a vector to represent the direction. Suggesting where is the palm centre point, where is a fingertip, then is the vector. Secondly, we require the user to stretch their hands to make computer compute the distance of each fingertip, then with four steps showing in Figure 11, we can get the particular finger and display them on screen, like in Figure 12.

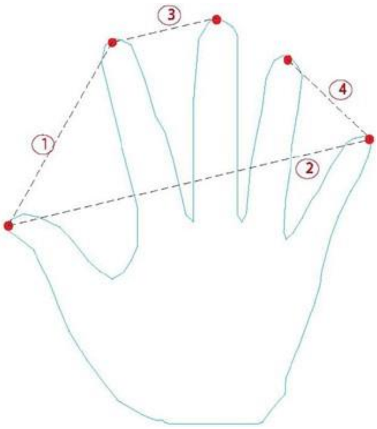

FIGURE 11 Four steps of identify fingers diagram

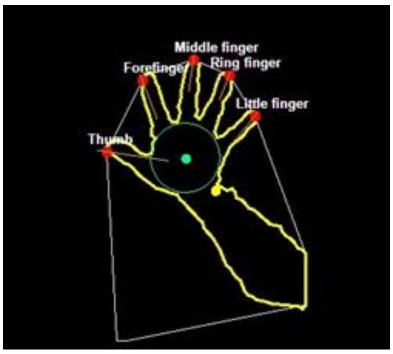

FIGURE 12 Finger recognition result

Thirdly we use a decision tree to classify hand gesture. The decision tree has three layers, the chance nodes of first layer are the finger counters, the second ones are names of finger, and the last are the angles of vectors. After the process above, we get HGR shown in Figures 13-15.

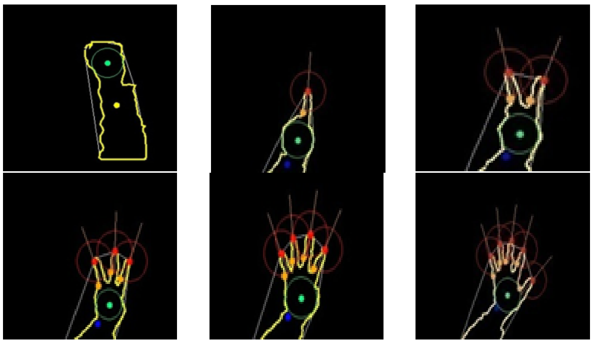

FIGURE 13 Number gesture recognition: zero to five

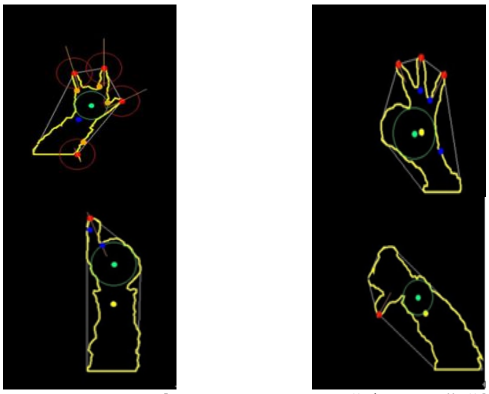

FIGURE 14 International gesture recognition: “I love you”, “Okay”, “Good”, “Bad”

FIGURE 15 Chinese number sign recognition: six, seven and eight

4 Conclusions

In this research we analyse and design the processing flow of gesture recognition system, using a means clustering algorithm for image pre-processing, contour recognition with Moore neighbourhood algorithm, Graham scanning algorithm and related algorithms for convex point detection, and we opt to three-point alignment algorithm and the bump maximum angle for fingertip recognition. According to the geometrical relationship of the feature points, we get gesture recognized. Finally, we complete the whole HGR prototype system.

After testing the prototype system, we can find that it is proved to be high-performance. The system can correctly get the geometrical relationship of feature points on fingertips and palm, and then give a reliable feedback displayed on the screen. But the system also has some shortcomings: the method how Kinect capture depth images is the speckle, which reflected back with the infrared, is not good enough. The more distance between user and Kinect, the less reliability it will reach. Because part of palm cannot entirely reflect the infrared speckle, the

Kinect cannot get the accurate depth data of the palm; it will result in glinting slimly. Therefore when recognize fingertip in the depth stream again and again, the fingertip will also glint at a certain small range. The problem will be solved in Kinect 2.

Acknowledgement

This work was supported jointly by Project of Guangxi Key Laboratory of Trusted Software (kx201304), Innovation Project of Guangxi Graduate Education (YCSZ2012071), and Project of Guilin Industry Innovation Foundation.

References

[1] Mei P 2010 Fingertips Recognition based on Contour and curvature *Anhui: a dissertation for Master's Degree* University of Science and Technology of China
[2] Liang R, Ouhyoung M 1998 A Real-Time Continuous Gesture Recognition System for Sign Language *The Third International Conference on Automatic Face and Gesture Recognition* 558-65
[3] Nakamura T, Takahashi S, Tanaka J 2008 Double-Crossing: A New Interaction Technique for Hand Gesture Interfaces *Lecture Notes in Computer Science* **5068** 292-300
[4] Hu Y 2005 Gesture Recognition Technology Review *China Science and Technology Information* 41-2
[5] Wu J, Gao W, Chen X 1999 A System Recognizing Chinese Finger-Spelling Alphabets Based On Data-Glove Input PR&AI **12** 74-8
[6] Li Y, Chen X, Zhang X, Wang K, Wang Z J 2012 *IEEE Transactions On Bio-Medical Engineering* **59**(10) 2695-704
[7] Maes P, Mistry P 2009 Unveiling the "Sixth Sense" game-changing wearable tech *TED 2009* Long Beach CA USA
[8] Guo C, Zhang L, Sun L 2008 Sign Language Recognition and Classification Based on Computer Vision *Science & Technology Information* **17** 248-9
[9] Yan Y 2012 The study of Gesture Recognition Based on Video *Wuhan: a dissertation for Master's Degree* Central China Normal University.
[10] *http://en.wikipedia.org/wiki/Kinect.*
[11] Raheja J L, Chaudhary A, Singal K 2011 Tracking of Fingertips and Centres of Palm using Kinect *The Third International Conference on Computational Intelligence, Modelling and Simulation (CIMSiM)* 248-52
[12] Wang X, Wang S 1990 Applied Multivariate Statistical Analysis *Shanghai Science and Technology Press*
[13] Yan P, Kong Y, Kong D 2008 Sign Language Recognition based on K-means/SCHMM Multi-stage Classification *Microcomputer Information Press* 187-9

Authors

Zhuo Shi, born on August 25, 1978, Changzhou, China

Current position: Doctor of Computer Software and Theory, vice-professor and Master supervisor in Guilin University of Electronic Theology.
University studies: Mechanical and Electronic Engineering in Shandong University
Scientific interest: digital media, virtual reality and natural interaction.
Publications: 3 Patents, 30 Papers

Yinghui Li, born on August 28, 1988, Guilin, China

Current position: Master of Computer Technology in Guilin University of Electronic Technology.
University studies: Computer Science and Technology in Wuhan Polytechnic University.
Scientific interest: graphic algorithm and virtual reality.
Publications: 1 patents.

Ke Yu, born on April 22, 1979, Liuzhou, China

Current position: Guilin University of Electronic Theology.
University studies: Guangxi Arts Institute.
Scientific interest: virtual reality.
Publications: 3 papers.

Yuanquan Cheng, born on February 24, 1989, Guilin, China

Current position: Master of Mathematics in Guilin University of Electronic Technology.
University studies: Mathematics and Applied Mathematics in Fuyang Teachers College.
Scientific interest: graphic algorithm and virtual reality.
Publications: 1 patent.

ChangShao Zhou, born on November 21, 1978, GuangZhou, China

Current position: Teacher of Guangzhou College of Technology and Business.
University studies: Master of Software Engineering in Guilin University of Electronic Technology,.
Scientific interest: big data, graphic algorithm and virtual reality.
Publications: 1 patents, 2 papers.