TELKOMNIKA, Vol. 11, No. 5, May 2013, pp. 2605 ~ 2611
ISSN: 2302-4046

A Robot Control System Based on Gesture Recognition Using Kinect

Biao MA*[1,a], Wensheng XU[2,b] Songlin WANG[2,c]
[1]School of Automation, Beijing Institute of Technology, Beijing 100081, China
[2] chool of Mechanical, Electronic and Control Engineering, Beijing Jiaotong University, Beijing 100044, China
*Corresponding author, e-mail: mabiaoeddy@gmail.com[a], shxu@bjtu.edu.cn[b], 1121421@bjtu.edu.cn [c]

Abstract

The Kinect camera is widely used for capturing human body images and human motion recognition in video game playing, and there are already some research works on gesture recognition. However, to achieve the anti-interference performance, the current recognition algorithms are often complex and tardiness, and most of the applications are based on the incomplete gesture library and not all hand gestures can be recognized. This paper explores a new method and algorithm which can describe all five fingertips for each hand in any time for hand gesture recognition with the Kinect system. The hand images are processed to build the hand models which are then compared with the gesture library for gesture recognition. After hand gestures are recognized with high accuracy and less computing, control commands corresponding to hand gestures are sent from the hand gesture recognition system to a hexagon robot controller wirelessly, the hexagon robot can then be controlled wirelessly and change its shape according to the hand gesture command. Thus the robot can interact with humans promptly through the gesture recognition system.

Keywords: *robot control, gesture recognition, Kinect, depth images*

1. Introduction

The studies of gesture recognition techniques started from 1977 when Electronic Visualization Laboratory created the first data glove which is called the Sayre Glove [1]. For the past thirty-five years, researchers gradually adopt the camera to implement the human-computer interaction (HCI). Unlike the data glove, gesture recognition through cameras makes it more natural and direct to realize HCI. There are three tasks for the gesture recognition system: segmentation, feature extraction and recognition. For the segmentation task, the camera is usually used for two different tasks: capturing RGB images and capturing depth images. For capturing RGB images, the system utilizes the characteristics of the human complexion to separate the gestures from the background. Lin detects the skin candidate regions on the color image with Gaussian Mixture Model (GMM) skin model [2]. And in Kramberger's research, to improve detection accuracy of pixel-based skin color segmentation, a parametric skin color model is used [3]. For capturing depth images, the depth information is employed to found a column diagram to distinguish the human body (the nearest object) and the background (the furthest object).

For the feature extraction and recognition task, recent researches are normally in three categories: (1) probability graph model based methods, which involve Hidden Markov Model (HMM) [4] and Dynamic Bayesian Network [5, 6]; (2) template based methods, which are related to the template matching algorithm [7, 8]; and (3) rule based methods, which contain the finite-state machine (FSM) [9] and the stochastic context-free grammar [10, 11]. By employing these algorithms, some researchers implement the recognition task based on the skeleton. Their systems can capture the main points of the human skeleton and recognize the motions. Meanwhile, others achieve the interaction based on the static hand gesture, such as the “victory” or “ok” gestures. However, these gestures are recognized by the extended fingers, which mean the bent fingers' information is lost in recognition process.

In 2010, Microsoft launched an infrared range-sensing camera – Kinect, which can provide users with raw depth data. Wan et al. [12] propose to set a threshold value and

Received January 14, 2013; Revised March 13, 2013; Accepted March 23, 2013

separate the hand from the background combining the depth data which is provided by the Kinect. Compared with others' algorithms, this approach is more direct and efficient. Like Wan et al., many researchers' works are based on the Open Computer Vision library (OpenCV) [13] and Open Natural Interaction (OpenNI) [14] while there are less studies based on the Kinect Software Development Kit (SDK) which was released in 2011.

In this paper, utilizing the depth data from the Kinect, a gesture recognition system is proposed and developed which can effectively recognize hand gestures with less computing but high accuracy. The gesture recognition system is developed with Microsoft Visual Studio 2010 and the Kinect Windows SDK. A new algorithm– the slot algorithm is proposed in our system and it can capture the information of both the bent fingers and extended fingers. And the system is based on a gesture library which is constructed on an algorithm with much lower complexity. Using this real-time recognition system, a hexagon robot developed by us can receive the gesture information wirelessly and can change its shape accordingly, so the robot can interact with people promptly through human hand gesture recognition.

2. System Structure

The robot control system includes four parts as shown in Figure 1:

(1) a gesture recognition system running on a laptop computer;
(2) a Kinect camera connected with the laptop computer;
(3) a hexagon robot and the robot controller;
(4) a pair of wireless communication modules connected with the gesture recognition system and the robot controller respectively.

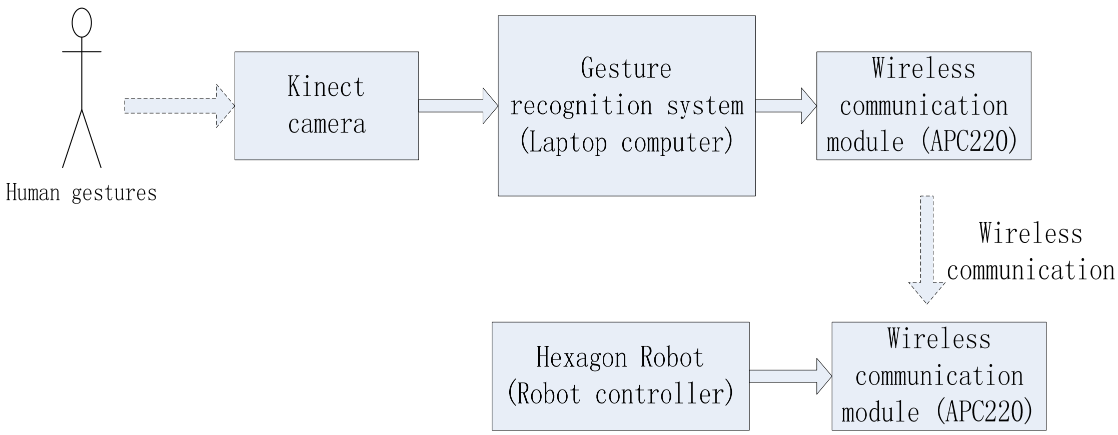

Figure 1. The system structure of the robot control system based on gesture recognition

The Kinect camera is used to obtain the image data of the human palm and fingers. Then the data are processed to recognize how many fingers are extended and straight. Each gesture is corresponding to a different robot control command. Then the APC220 wireless module is used to send these different robot control commands to the robot controller. Accordingly, the hexagon robot will do actions according to different human hand gestures, thus human-robot interaction can be achieved.

The system is developed with C# of Microsoft Visual Studio 2010, using the SDK instead of the OpenNI. As the newest kit, the SDK provides a set of powerful algorithms which enables the system to extract the depth data and transform the scene images into binary images, thereafter to build a body skeleton model. However, the 20-point-skeleton model cannot satisfy the needs for hand gesture recognition. Thus, the depth data of the two palms are utilized for hand details processing and recognition.

3. Recognition of Hand Gestures

3.1. The Capture and Process of Static Hand Images

In order to extract the features of hand gestures, first we need to divide the hand images from the background. A lot of researchers have proposed different algorithms to use the complexion features. Our method combines these complexion algorithms with the depth information and palms skeleton information to accomplish the gesture model. The palms skeleton information is provided by the SDK. Thus our main work is extracting the hand images from the background. The main method to separate the hand images is using a depth threshold which stands for the thickness of the hand and comparing the current point's depth value with the previous one's. Then we get the contour of the hand gestures. Let P denotes the previous point, C denotes the current point and T denotes the thickness of the hand. The hand extracting algorithm is shown below.

(1) P and C are set in a 3×3 pixel area. Assume P is the center of the area, C moves clockwise.
(2) At the beginning, utilizing the SDK, we gain the palms skeleton information of the humans. We simply use the palm point as the first P. And the upper pixel of P is the first C.
(3) If C.depthValue >= P.depthValue + T, we believe that C is the background point, then C moves to the next place while P's position is recorded in the contour set K.
(4) If C.depthValue < P.depthValue + T, we believe that C is inside the hand contour, then C becomes a new P while the previous P starts to move as a new C.

After we get the contour of the hand, we use the k-mean algorithm to divide the points into two hand groups and extract the features of the hand [15]. Some researchers propose a method which describes the hand with extended fingers and a palm center. They mainly use the binary images to identify the gestures. However, the binary images just simply provide the contour of the hand and a lot of information is lost. For instance, when we recognize the "victory" gesture, the binary image will lose the information of the thumb, ring finger and little finger. To solve this problem, we propose an algorithm called slot algorithm. In the slot algorithm, each hand is described by the position of five fingertips and a palm center even though some fingers are bent. The slot algorithm is divided into five steps:

(1) Traverse the contour set K and find the deepest point D.
(2) Find the top point, the bottom point, the point from the very left and the point from the very right. Use these points to build a rectangle. Put the points which are inside the rectangle into a set R. Find the closest point A within this set.
(3) Define the thickness T_2 of the gesture. T_2 = D.depthValue - A.depthValue.
(4) Define the slot width W which has the half value of T_2. Use the checking algorithm to build a set R', let $T = T_2$ and A be the starting point.
(5) Use the k-curvature algorithm to calculate the set K and R', then we have the set P_k (fingertips of extended fingers) and P_r (fingertips of bent fingers).

Therefore, we can get the position of the five fingertips.

The way of using the k-curvature algorithm to find the fingertips is shown as follows [16]: When we have got the hand contour set K, we start to use the elements of it. For each K (i), we get the neighboring points, and utilize these points to generate two vectors and calculate the minimum angle which they form. The distance between the points is marked by m. And the vectors are formed by K(i)---K(i-m) and K(i)---K(i+m). If the angle is less than a specific value α, we believe that it is a fingertip. The more appropriate values found are m=22 and α= 40 degree. Based on the k-curvature algorithm, all fingertips can be identified.

3.2. Finger Gesture Recognition

Figure 2 shows the mirror image of a normal left hand taken by the Kinect camera and the corresponding recognized hand contour which is marked by the name of fingertips. The whole experiment situation is shown in Figure 3. Take the left hand for instance, the hand gesture recognition process is designed as three steps as follows:

(1) Thumb and little finger state judgement: Gestures are classified by whether the thumb or the little finger is extended or not. Get the position of point B from the very left from set P_k. If there is no point (regardless they are extended or bent) on the left of B, then the little finger is extended. Otherwise, the little finger is bent.

Among the fingers, the thumb is the only one that is far away from the others. If there is an element of set P_k which is far away from the other elements, then the thumb is extended. Otherwise, it is bent.

(2) Finger counting: Gestures are classified by the number of extended fingers. If the system still can not identify the gesture, the extended finger number and the state of thumb and little finger will be sent to the next step.

(3) Index finger state judgement: Ignore the gestures which are hard to make for normal people, we list every possible gesture in Table 1. For “1” denotes the finger is extended while “0” denotes the finger is bent. Among these 19 kinds of gestures, there are 4 pairs of gestures that are hard to distinguish with each other in the pair. However, they have different states of the index finger. Combine the set P_k and P_r. If the thumb is extended, find out the point that has three points on the left. Otherwise, find out the point that has four points on the left.

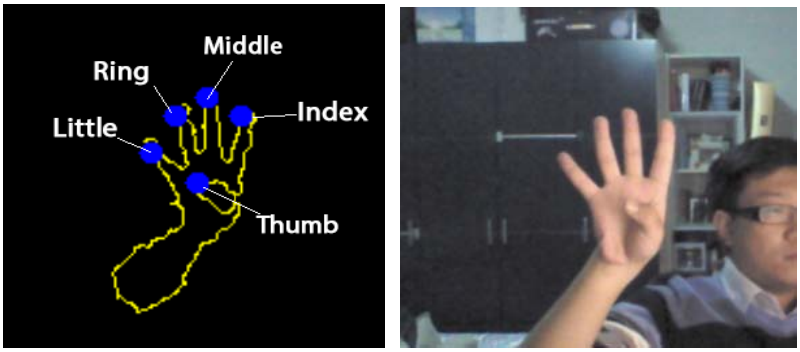

Figure 2. Recognition of the left hand fingers

Then if it comes from P_k, the index finger is extended. Otherwise, it is bent. These gestures are marked by black background in Table 1.

With the method above, the hand gestures can be recognized conveniently and efficiently.

Table 1. The classification of finger states in the gesture library

E-Num / L/T	0	1	2	3	4	5
1/0		10000	10010	11100/10110	11110	
1/1			10001	10011/10101	11101/10111	11111
0/1		00001	00011	00111		
0/0	00000	00010/00100	00110	01110		

E-Num: number of stretched-out fingers;
T/L: states of Thumb/Little finger
xxxxx: states of thumb, index, middle, ring, little finger

4. Robot Control with Gesture Recognition

After gesture recognition, each gesture is encoded and the coding is sent to the hexagon robot controller wirelessly. The hexagon robot developed by us can change its form into triangle, rectangle, or hexagon etc. and can roll on the ground from left to right or from right to left. Each shape or action is corresponding to a specific hand gesture. Thus the hexagon robot can interact with human hand gestures promptly. The gesture recognition system and the hexagon robot are shown in Figure 3. Several examples of the gesture recognition and the shape changes of the hexagon robot in our experiments are shown in Figure 4 to 6.

The experiments show that the hand gesture recognition method proposed above has high accuracy and stability in capturing the information of the whole hand, and it simplifies the algorithm of gesture identifying with the help of the gesture library. Previous researchers proposed several complex algorithms for gesture identification, such as Rashid's system with the Hu-Moments algorithm [17].

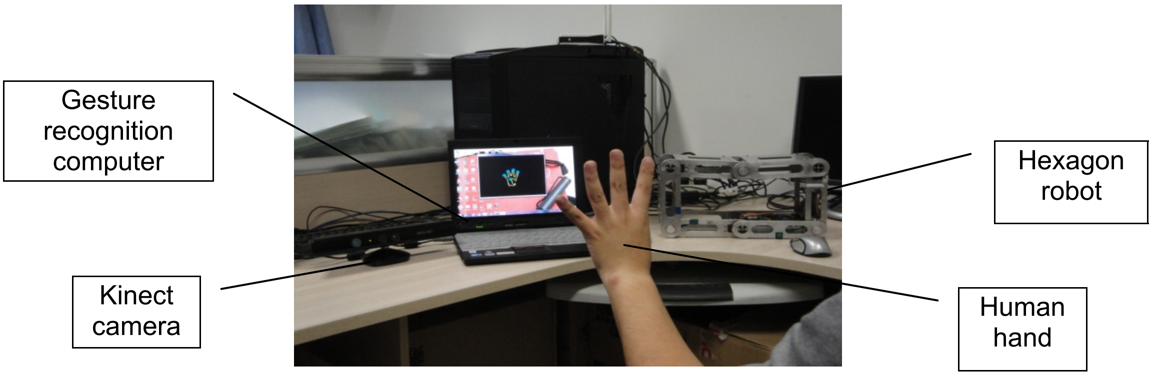

Figure 3. The gesture recognition system and the hexagon robot

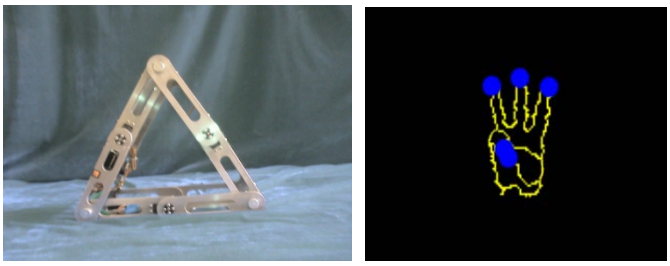

Figure 4. Recognition of gesture "three" and the shape change of the hexagon robot

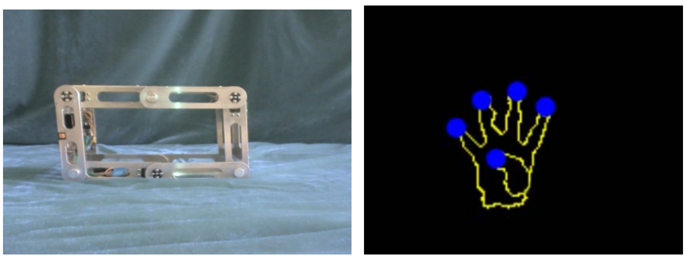

Figure 5. Recognition of gesture "four" and the shape change of the hexagon robot

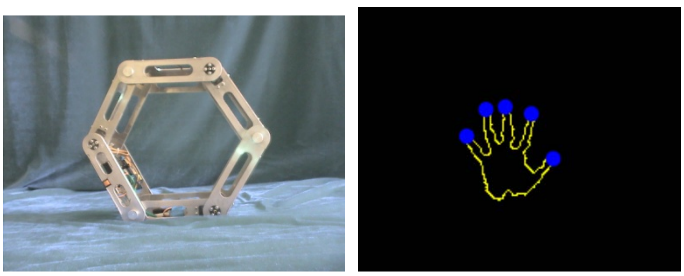

Figure 6. Recognition of gesture "all-extended" and the shape change of the hexagon robot

But their algorithms are complex and need more processing time, and they cannot recognize all gestures in all situations accurately. For instance, Rashid's system has difficulties in distinguishing whether an extended finger is an index finger or a middle finger. Our system is successfully built with the use of the intergrated information of all five fingertips, and it can recognize the status of every finger and it can ensure that the system has a stronger ability in distinguishing similar geatures.

5. Conclusion

Based on the slot algorithm proposed in this paper, the contour of both the extended fingers and bent fingers can be identified and the actual positions of the fingers can be got, thus hand gestures can be recognized accurately and efficiently. By sending the gesture information to the robot controller wirelessly, the hexagon robot can change its shape into triangle, rectangle or hexagon, etc. accordingly, thus human-robot interactions can be achieved. In future, by identifying more hand gestures and by adopting a more versatile robot, the interactions between the humans and the robot can be more abundant and flexible.

Acknowledgement

This work is funded by Beijing Jiaotong University Student Innovation Program.

References

[1] Sturman DJ, Zeltzer D. A survey of glove-based input. *IEEE Computer Graphics and Applications*. 1994; 14(1): 30–39.

[2] Lin Lan, Cong Yang, Tang Yandong. Hand gesture recognition using RGB-D cues. *Information and Automation (ICIA)*, 2012 *International Conference on. Shenyang*. 2012: 311-316.

[3] Kramberger I. Reflectance Measurement System for Skin Color Modeling in Chromaticity Color Space. 2009 16th *International Conference on Systems, Signals and Image Processing*. Chalkida. 2009: 1-4.

[4] Rizkha Emillia Nyoman, Suyanto NFN, Maharani Warih. Isolated word recognition using ergodic hidden markov models and genetic algorithm. *Telkomnika*. 2012; 10(1): 129-136.

[5] Budiharto W, Santoso A, Purwanto D, Jazidie A. Multiple moving obstacles avoidance of service robot using stereo vision. *Telkomnika*. 2011; 9(3): 433-444.

[6] Zhu Chun, Sheng Weihua. Realtime recognition of complex daily activities using dynamic Bayesian network. *IEEE International Conference on Intelligent Robots and Systems*. San Francisco. 2011: 3395-3400.

[7] Celik IB, Kuntalp M. Development of a robotic-arm controller by using hand gesture recognition. 2012 *International Symposium on Innovations in Intelligent Systems and Applications (INISTA)*. Piscataway. 2012: 5.

[8] Lai K, Konrad J, Ishwar P. *A gesture-driven computer interface using Kinect*. Proceedings of the 2012 IEEE Southwest Symposium on Image Analysis & Interpretation (SSIAI 2012). Santa Fe. 2012: 185-188.

[9] Dingli Alexiei, Attard Daniel, Mamo Ruben. Turning homes into low-cost ambient assisted living environments. *International Journal of Ambient Computing and Intelligence*. 2012; 4(2): 1-23.

[10] Mitomi H, Fujiwara F, Yamamoto M, Sato T. Bayesian classification of a human custom based on stochastic context-free grammar. *Systems and Computers in Japan*. 2007; 38(9): 716-726.

[11] Qing Chen, Georganas ND, Petriu EM. Hand gesture recognition using Haar-like features and a stochastic context-free grammar. *IEEE Transactions on Instrumentation and Measurement*. 2008; 57(8): 1562-1571.

[12] Wan Ting, Wang Yujun, Li Junke. Hand gesture recognition system using depth data. 2012 *2nd International Conference on Consumer Electronics, Communications and Networks (CECNet). Yichang*. 2012: 1063-1066.

[13] Zhang Baoyun, Yun Ruwei, Qiu Huaqing. Hand Gesture Recognition in Natural State Based on Rotation Invariance and OpenCV Realization. *Entertainment for Education*. *Digital Techniques and Systems. 5th International Conference on E-learning and Games, Edutainment 2010*. Berlin. 2010: 486-496.

[14] Cho Ok-Hue, Lee Won-Hyung. *Gesture recognition using simple-OpenNI for implement interactive contents*. Lecture Notes in Electrical Engineering. Vancouver. 2012; 2: 141-146.

[15] Li Y. *Hand Gesture Recognition Using Kinect*. Proceedings of 2012 IEEE 3rd International Conference on Software Engineering and Service Science. Beijing. 2012: 196-199.

[16] Mühendisligi Bölümü, Orta Dogu. *3D object recognition by geometric hashing*. 2009 IEEE 17th Signal Processing and Communications Applications Conference. Antalya. 2009; 932-935.
[17] Omer Rashid, Ayoub Al-Hamadi, Bernd Michaelis. Robust Hand Posture Recognition with Micro and Macro Level Features using Kinect. *2011 IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS 2011)*. Guangzhou. 2011; 3: 391-341.