

Finger Controller: Natural User Interaction Using Finger Gestures

Unseok Lee and Jiro Tanaka

Department of Computer Science, University of Tsukuba,
Tennodai, Tsukuba, 305-8577 Ibaraki, Japan
{leeunseok, jiro}@iplab.cs.tsukuba.ac.jp

Abstract. We present a new natural user interaction technique using finger gesture recognition and finger identification with Kinect depth data. We developed a gesture version drawing, multi-touch and mapping on 3d space interactions. We implemented three type interfaces using their interaction such as air-drawing, image manipulation and video manipulation. In this paper, we explain finger gesture recognition method, finger identification method and natural user interactions in detail. We show the preliminary experiment for evaluating accuracy of finger identification and finger gesture recognition accuracy, evaluating user questionnaire for interaction satisfaction. Finally, we discuss the result of evaluation and our contributions

Keywords: NUI, Human Computer Interaction, Finger Gesture Recognition, Finger Identification.

1 Introduction

The interaction using hand gestures is a popular field in Human Computer Interaction(HCI) and consequently many related research papers have been proposed. Some of them propose media player manipulation interaction using hand motion gesture [3] and glove-based hand gesture interaction. Vision-based hand gesture recognition system was proposed as well [5], [8]. However, the research approaches that propose the mounting of additional device on the body [6] are usually troublesome and not natural. The vision-based hand gesture recognition is not practical for robust hand gesture recognition because of much influence by light and background clutter. Recently, new horizons are open to the HCI field with the development of sensors and technology [2] such as Kinect, DepthSense and Leap motion. This development has made possible robust recognition, like finger gesture recognition in bad conditions such as dark light and rough background. This depth-based sensor and technology provide a robust recognition, but many research works using them do not provide proper natural interaction [1], [4]. They mostly provide simple hand gesture interactions or hand motion interactions. They do not provide a natural interaction. On the other hand, Finger gesture recognition with finger identification can provide more practical experience and natural interaction than hand gesture and hand motion [2].

In this paper, we propose a new natural user interaction technique using finger gestures with finger identification. It is called Finger Controller. We implemented three types of interface such as air-drawing, image manipulation and video manipulation using designed natural user interactions. Figure1 shows the interfaces overview.



Fig. 1. Air-Drawing(Left), Image Manipulation(Middle), Video Manipulation(Right)

2 Related Work

Many hand gesture recognition techniques are proposed to interact with objects for natural user interface(NUI). They are implemented using various devices and techniques. Vision based, glove based and depth based are widely used in hand gesture recognition [2]. Vision based hand gesture recognition techniques and interfaces are continuously being proposed.

Chu et al [5] presents self-portrait interface using vision based hand motion gesture. They provide interesting interaction to control camera by the user. However, the recognition is difficult in bad conditions, such as darkness, because they use skin color segmentation. In addition, system functions are limited when user moves far from the camera because they do not use depth data for recognition. Kenn [6] presents an interface for wearable computing applications using glove based finger recognition. His paper implemented finger identification for hand gesture. The research provides rich user experience based on accurate recognition rate. However, it is not practical in all situations because the user cannot always carry a glove. The glove is both unaesthetical and heavy. This condition is not suitable for natural interaction.

Yang et al, [3] proposed a hand motion gesture recognition system using Kinect depth data. They designed hand motion gestures like wave, forward/backward, move up/down, left/right in a media player application. The designed gesture interaction session is performed based on assumptions of the user's intentions. The system used a 3D feature vector for examining the hand trajectory and HMM for hand gesture recognition. This system shows the possibility of using Kinect for hand motion gesture recognition in a contact-less UI. However, the implemented system was not able to recognize fingertip. Therefore, they were not able to provide delicate and natural gestures, such as pinch gesture, spread gesture and flick finger. Raheja et al, [4] proposed a method to recognize and track fingertips and center of palm using Kinect. They track the fingertip by calculating depth image segmentation of hand regions.

Then the palm of the hand would be subtracted from the depth image, so that only the fingers are left. Under most situations, fingertip is minimum depth in each finger. The proposed method provides robust fingertip recognition rate. However, they don't implement any interactions, gesture and finger identification. Thus, this approach is not suitable for NUI. In short, most of the proposed hand gesture interaction techniques have limitation for natural interaction using finger gesture.

3 System Overview

In this section, we will discuss the system hardware we used. Fingertip tracking and finger identification method to detect interactions using finger gesture will be discussed as well.

3.1 Hardware

Our system is comprised of Microsoft Kinect for Xbox 360 sensor and large display. The kinect sensor is used with input data for capturing user gestures using CMOS camera(640x480 pixels) with 30 FPS and Depth camera. For improving system performance, we limit the distance from hands to sensor. We found the appropriate value of distance(0.5m to 0.8m) experimentally.

The large display that we used has a size of 30 inches. The display shows interaction feedback from the implemented application using user gesture.

3.2 Depth Based Fingertip Tracking Method

Our system uses depth data information for the robust finger tracking of finger gesture for the natural interaction. The algorithm designed for the finger tracking is described in the following.

The first step is to gather tracked hands depth data from the kinect sensor, and separate hands from the depth image background. A simple k-means clustering algorithm is implemented for dividing hand point of a frame into cluster. Second, the system makes the hand's contour using a modified Moor Neighbor Tracing algorithm, and compute convex hull by Graham Scan algorithm on detected depth cluster data. Then, the system combines contour information and point in convex hull. Third, for each point in the hull that is a candidate for fingertips, we find the nearest point in the contour curve. We call this set P. For each point in P, we take the two supporting points(P1, P2) in two opposite directions along the contour. P1 and P2 positions are detected by analyzing the hand contour shape pattern, because most hand shapes are similar. Then, the system determines whether these three point(point in P, P1 and P2) is aligned or not. Because we can ensure that it is not fingertip in the case of aligned, system doesn't need to calculate. If these three points are not aligned, the system calculates the distance from midpoint of P1 and P2 to the point in P. The distance is a certain value that was found experimentally, the candidate point in P is a fingertip point (see figure 2(c)). Figure 2-(d) shows that the distance is smaller than a certain value, thus that it is not a fingertip point.

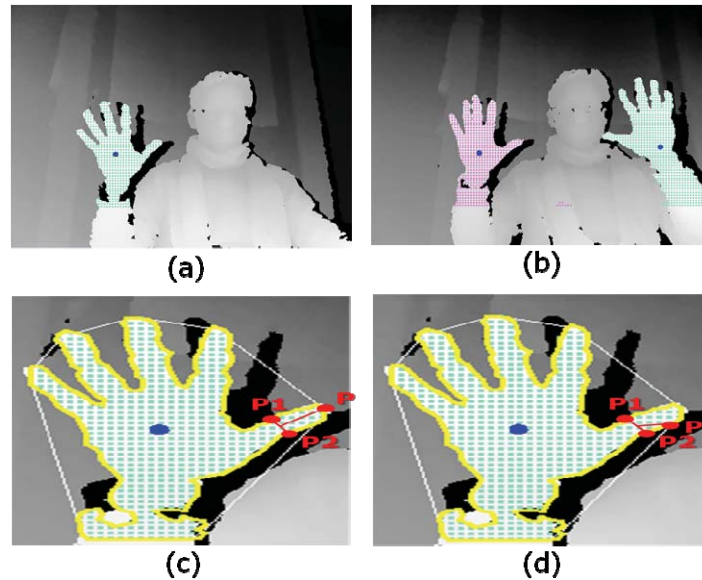


Fig. 2. Hand and Fingertip Detection. (a) Hand Detection using clustering (b) Two hands Detection using clustering (c) Fingertip (d) Not Fingertip.

3.3 Depth Based Finger Identification Method

In this section, we describe a finger identification method for finger gesture recognition. Our system makes interaction based on pre-defined finger gestures such as pinch, spread and rotate gesture, so that the finger identification is important for such delicate recognition. We have motivation of developing identification method that when the people manipulate object such as smart pad and phone devices, they are mainly used thumb and index finger for interaction. Our system tracks fingertip using the method explained in 3.2, then calculates mathematically their coordinates and depth value for finger identification. The method is mainly divided into three parts, described as follows.

The first step is counting the number of fingers. According to this number, the system determines whether it will start to calculate or not. The algorithm starts to calculate when all fingers are extended. Second, we identify the thumb and the index finger. The system uses the distance from the device and shape bases matching for identifying the thumb. In general, among all fingers the thumb has the shortest distance from the device when all fingers are extended. We use hand shape matching for more accurate thumb identification. The thumb can be found more accurately by combining these results, and then we make pairs of all neighbor fingers. System calculates largest distance among one to four (see figure 3(a)). The set of thumb and index finger has the largest distance, so we can determine the index finger. Third, the little finger is determined as the farthest finger away from the thumb, and then the

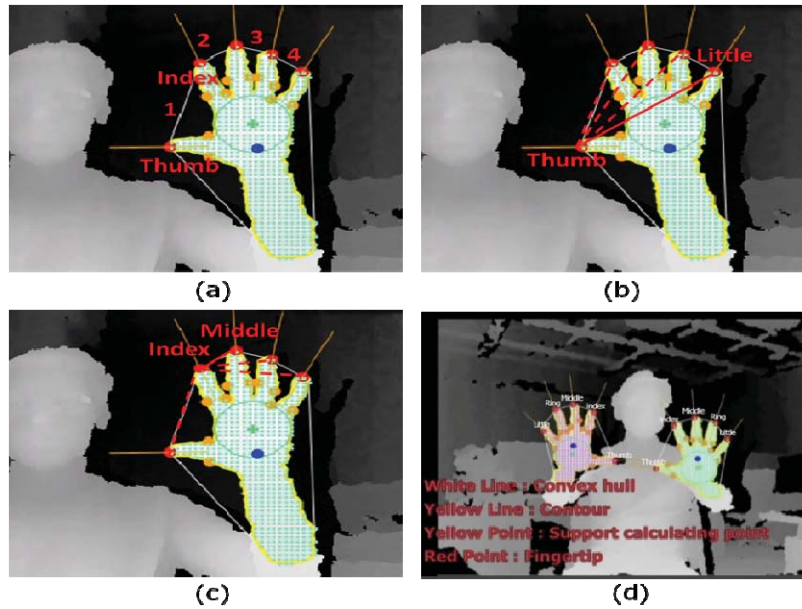


Fig. 3. Finger Identification. (a) Thumb and Index finger detection (b) Little finger detection (c) Middle finger detection (d) Identify fingers of two hands.

middle finger is determined as the closest finger from index finger (see (b) and (c) of figure 3). The remaining finger is determined as ring finger. The same method is used for identifying the fingers of both hands (see figure 3(d)).

4 Natural User Interface and Interaction

In this section, we explain the natural user interaction in each interface such as air-drawing, image manipulation and video manipulation. The interaction recognition method and feedback from the interface are explained in detail.

4.1 Air-Drawing Interface and Interaction

Our air-drawing interface implemented correctly finger painting interactions, by using depth-based finger gestures. We designed painting, drawing line gesture for these interactions. First, painting gesture is recognized when user extends his/her index finger, then the line is drawn along the path of movement.

Second, drawing line gesture is recognized when user extends their thumb and index finger, then the line is drawn depending on the coordinates value of the thumb and index finger (see figure 4(a)). The same interactions are made in the case of using both hands at simultaneously.

4.2 Image Manipulation Interface and Interaction

The Image manipulation interface is implemented well mapped with multi-touch interactions. It provides resizing (i.e. pinch, spread gesture) and rotating natural interaction with images.

In order to resize the image, the system tracks the number of fingertips. When system detects two fingertips (i.e. thumb and index finger of one hand or index finger of left and right hand), it changes to resizing interaction mode (see (c), (d) of figure 4). In resizing interaction mode, the system computes the distance between two fingertips. If the distance between fingertips becomes larger than a certain value found experimentally, zoom-in interaction will be performed and the size of the image is expanded. On the contrary, if the distance between fingertips is shorter than a certain value, zoom-out interaction will be performed and the size of the image is reduced.

In order to rotate the image, the system computes each fingertip's position (i.e. thumb and index finger coordinates of one hand or index finger coordinates for left and right hand). If the left index finger is raised upwards and the right index finger is moved downward then the image is rotated clockwise. On the contrary, if the right index finger is raised upwards and the left index finger is moved downwards, then the image is rotated counter clockwise (see figure 4(f)). When using only one hand, similar interactions are performed. If the thumb is raised upwards and the index finger is moved downwards, then the rotating interaction is performed in clockwise (see figure 4(e)).

4.3 Video Manipulation Interface and Interaction

Video manipulation interface implements interactions with video surface on 3D space using depth data. We designed the gestures for mapping, selecting surface and time shift interaction with media player.

Mapping interactions are made when system detects two fingertips on each hand. The video surface to play the video is created depending on the coordinates of the two fingertips of each hand (i.e. 4 sets of coordinates, thumb and index finger coordinate value for each hand). User can control the size of surface and mapping position by moving four points of P1, P2, P3 and P4 (see figure 4(b)). The surface's depth value can be controlled by moving a hand or both hands forward/backward. We can create a new surface in front of the surface that was created before, when both hands are moved forward. On the contrary, the new surface can be created behind the surface that was created before, when both hands are moved backward. In the case of moving a hand forward and the other hand backward, an almost diamond-shaped surface will be created. If shown from the front, it appears as a diamond surface. However, it can be shown as a rectangle surface when it is shown from different angles.

Selecting interactions occur when an index fingertip is on the surface that the user wants to control (i.e. it means that the depth value of index fingertip and the surface is same). This interaction is needed when multiple surfaces are created. The selected surface is shown with a white color border. Other interactions can be performed with the selected surface.

Time shifting interaction with selected surface occurs when the system detects five fingertips of the right hand and one fingertip of the left hand. The user can control the movie time bar by moving his/her left fingertip.

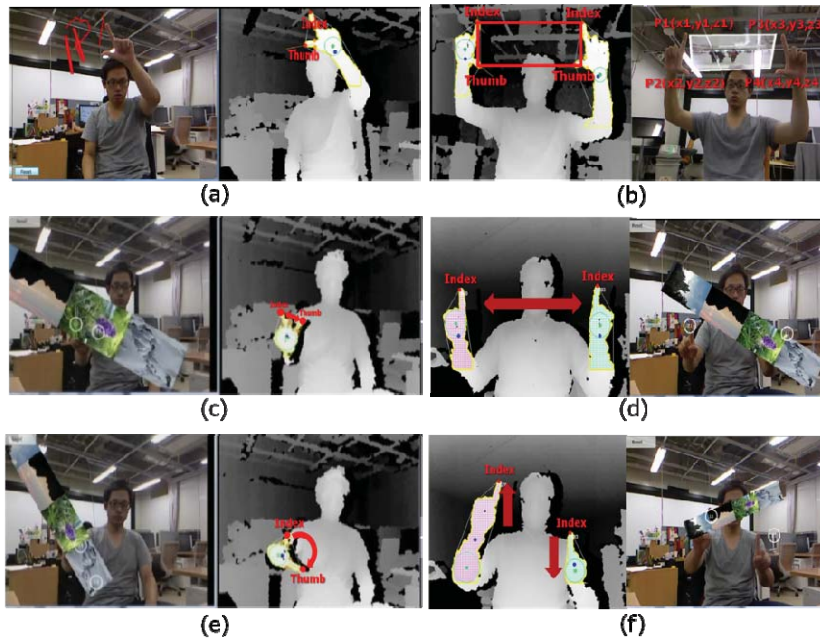


Fig. 4. Natural Interactions using Finger Gestures. (a) Drawing Gesture and Interaction (b) Mapping Gesture and Interaction (c) Spread Gesture and Zoom-in Interaction using a hand (d) Spread Gesture and Zoom-in Interaction using both hands (e) Clockwise Rotate Gesture and Rotating Interaction using a hand (f) Clockwise Rotate Gesture and Rotating Interaction using both hand.

5 Evaluation

5.1 Recognition Accuracy Experiment

In this experiment, we evaluated recognition accuracy with our proposed method for finger identification and designed finger gesture. The experiments were performed on a computer with Intel Core i5 CPU 2.67GHz and 4.0 GB RAM, using Microsoft Kinect for Xbox 360.

We performed the experiments with ten volunteers. Our experiments are designed to evaluate six gestures, i.e. extending all fingers for finger identification, drawing, pinch, spread, rotate and mapping gesture. After thoroughly explaining all our gestures, each volunteer performed a gesture 100 times, in each condition. We checked whether the system recognized the gesture or not. Figure 5 shows the average of recognition for the ten volunteers.

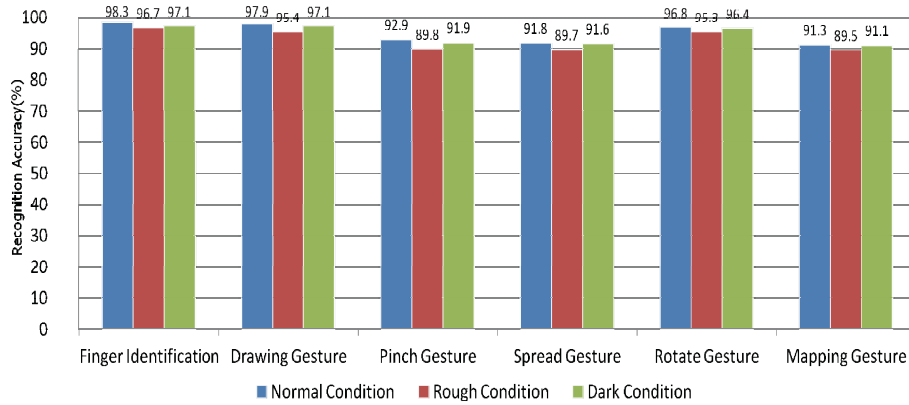


Fig. 5. The average result for ten volunteers

5.2 User Questionnaire

25 users participated in our system evaluation. They were all university students with various majors, 15 men and 10 women. They had experience in using Wii controller. We explained in simple terms our proposed interface's functions. The participants performed the proposed gestures and interactions freely, without any time limit. After completing their task, we asked them to answer a short questionnaire and offer a score from 1 to 5 (5 meaning very good) and comments.

Table 1. Average of 25 user's satisfaction for finger controller

Question	Average
Was it easy to interact using proposed gesture in each interface?	4.6
Was it natural to perform interaction and gesture using one hand?	4.64
Was it natural to perform interaction and gesture using both hands?	4.4
Was using the interface intuitive?	4.88
Was it natural compared with Nintendo Wii, Remote Game Conroller, Glove-based, etc?	4.92

5.3 Discussion

Our proposed method for finger identification and finger gesture recognition with depth data has shown high accuracy over 89 percentages for all gestures. Figure 5 shows almost the same accuracy with normal and dark conditions, and relatively low accuracy in rough conditions (e.g. many obstacles in background). However, the system was not influenced at all by dark and rough condition. We found that pinch,

spread and mapping gesture had a relatively low recognition rate. The pinch gesture was recognized as spread gesture because the users moved their hand rapidly, before the pinch gesture was recognized. The spread gesture was recognized as pinch gesture in the similar way. In addition, the mapping gesture has a high time complexity compared to other gestures, because the system calculates depth data from fingertips on both hands. Therefore, the recognition accuracy is decreased in case the user moves their hand rapidly, but it can be improved if we upgrade our system hardware.

Table 1 shows that our proposed interface and interaction are natural and intuitive. The final question had the highest score: it showed that our proposed wear-less interaction provides a more natural usage and intuitive feeling than wii, glove-based interface, and remote game controller. We received comments from the users about the need for applying this interaction in more practical situations, such as games, Google Street View, Google Earth. The users also suggested the use of our proposed interactions on larger displays.

6 Conclusions and Future Work

In this paper, we proposed a new natural user interaction using depth-based finger gestures. We presented finger tracking and finger gesture recognition techniques using finger identification with depth data. We use the clustering algorithm for hand detection and Graham Scan/Tracing algorithm for fingertip tracking. We propose our own method to identify fingers. After identification, the system detects the designed finger gesture from users and feedback. Our system provides gesture version of natural finger interaction and gestures such as drawing, resizing, rotating and mapping gesture. We presented the interfaces using our proposed interaction. These interfaces are : air-drawing, image manipulation and video manipulation.

We performed experiments for evaluating gesture recognition performance and we obtained satisfying results. They showed that finger gesture is promising for natural and complicate gesture recognition with high accuracy, i.e. over 89 percentages for all gestures. Our proposed interactions are natural and also intuitive.

In future work, we intend to support more practical functionalities with interaction. We also want to implement a 3d interface using HMD with finger gesture. It will need to apply object tracking for marker-less interface. We expect that this interface can include practical functions and will be more natural to use.

References

1. Ren, Z., Meng, J., Zhang, Z.: Robust Hand Gesture Recognition with Kinect Sensor. In: Proceedings of the 19th ACM International Conference on Multimedia, MM 2011 (2011)
2. Wigdor, D., Wixon, D.: Brave Nui World, pp. 9–15. Morgan Kaufmann (2011)
3. Cheoljong, Y., Yujeong, J., Jounghoon, B., David, H., Hanseok, K.: Gesture recognition using depth-based hand tracking for contactless controller application. In: 2012 IEEE International Conference on Consumer Electronics (ICCE), pp. 297–298 (2012)

4. Raheja, J., Ankit, C., Singal, S.: Tracking of fingertips and centers of palm using KINECT. In: 2011 Third International Conference on Computational Intelligence Modelling Simulation, pp. 248–252 (2011)
5. Chu, S., Tanaka, J.: Hand Gesture for Taking Self Portrait. In: Jacko, J.A. (ed.) Human-Computer Interaction, Part II, HCII 2011. LNCS, vol. 6762, pp. 238–247. Springer, Heidelberg (2011)
6. Kenn, H., Megan, F., Sugar, R.: A glove-based gesture interface for wearable computing applications. In: Proceedings of the IFAWC 4th International Forum on Applied Wearable Computing 2007, pp. 169–177 (2007)
7. Lenman, S., Bretzner, L., Thuresson, B.: Computer Vision Based Hand Gesture Interfaces for Human-Computer Interaction. Technical Report CID-172, Center for User Oriented IT Design, pp. 3–4 (2002)
8. Wachs, J., Kölsch, M., Stern, H., Edan, Y.: Vision-based hand-gesture applications. *Communications of the ACM* 54(2), 60–70 (2011)
9. Frati, V., Prattichizzo, D.: Using Kinect for hand tracking and rendering in wearable haptics. In: IEEE World Haptics Conference, pp. 317–321 (2011)
10. Graham, R.: An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letter*, 132–133 (1972)
11. Iwai, Y., Watanabe, K., Yagi, Y., Yachida, M.: Gesture recognition using colored gloves. In: IEEE Int. Conf. Pattern Recognition, vol. A, pp. 662–666 (1996)
12. Nanda, H., Fujimura, K.: Visual tracking using depth data. In: Conference on Computer Vision and Pattern Recognition Workshop, p. 37 (2004)
13. OpenNI organization. OpenNI User-Guide (2012)
14. He, G., Kang, S., Song, W., Jung, S.: Real time gesture recognition using 3D depth camera. In: 2011 IEEE 2nd International Conference on Software Engineering and Service Science (ICSESS), pp. 187–190 (2011)
15. Tang, M.: Hand Gesture Recognition Using Microsoft's Kinect. Paper Written for CS228, Winter 2010. Technologies, UIST 2011, pp. 1–9. ACM (2011)