# Image Manipulation Interface using Depth-based Hand Gesture

## UNSEOK LEE[†]    JIRO TANAKA[†]

Vision-based tracking is popular way to track hands. However, most vision-based tracking methods can't do a clearly tracking under some natural conditions like fast hand motion, cluttered background, poor light condition. On the other hand, Depth-based tracking is able to track under natural conditions. In this paper, we propose a new image manipulation interface by using depth-based hand gesture. For hand gesture interaction, we track fingertips and palm with KINECT depth data. Our interface provides resizing, rotating, and dragging interactions with images. In result, our system implemented natural user interface for image manipulation.

## 1.  Introduction

Hand gesture recognition is an important research issue in the field of Human-Computer-Interaction, because its extensive application in virtual reality, sign language recognition, and computer games [4]. Existing vision-based approaches [6], [8] are greatly limited by the quality of the input image from optical cameras. Variations in lighting and background clutters would only worsen the problem. Consequently, these systems have not been able to provide satisfactory results for hand gesture recognition. On the other hand, depth-based approaches are able to provide satisfactory results for hand gesture recognition even poor light and cluttered background condition.(e.g. [1] , [6]) One of the more popular devices used to do depth-based approach is Microsoft's Kinect, which has sensors that capture both RGB and Depth Data. The advent of relatively cheap image and depth sensors has spurred research in the field of hand tracking and gesture recognition. Accordingly, it was required various natural user interface for hand gesture interactions.

In this paper, we proposed natural user interface by fingertips and palm tracking with KINECT depth data. It called Hand Controller Interface. This interface provides resizing, rotating, and dragging image interactions by detecting fingertips number and tracking position both hands and palm. Moreover, performance can be improved by using k-means clustering algorithm and convex hull methods together when tracking more than two hands.

In result, our system provided new intuitive interface for image manipulation, It was

---

[†] Graduate School of Systems and Information Engineering, University of Tsukuba

improved hand gesture recognition comparing with vision-based approach as well.

## 2.  System Overview

The KINECT is comprised of IR light, Depth Image CMOS for depth and colour Image CMOS for RGB data. Sensors compute the depth of the hand while the RGB camera is used to capture the images. The system is comprised of display and Kinect sensor. Because of Kinect's resolution limitation(640x480),  the distance from hands to sensor is limited(see Figure 1).
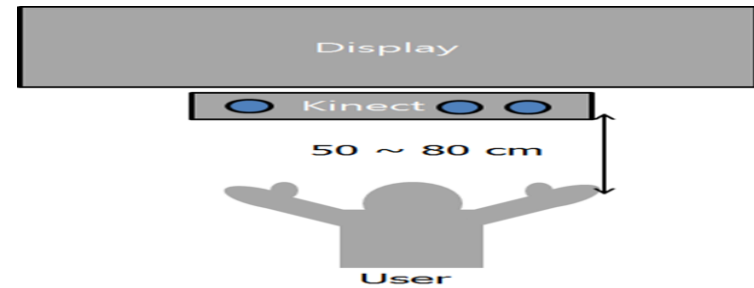


Figure 1. System Configuration

In application, there are three steps to image manipulation by hand gesture. First, Image is displayed on screen. User's hands shape are detected and displayed when the hands within given distance by system(yellow color line, see Figure 2).  Second, The system detect fingertips and palm. The red points are detected fingertips. Green circle is detected palm by system. Third, user can controls image objects by three types of hand gestures to resize, rotate and drag.
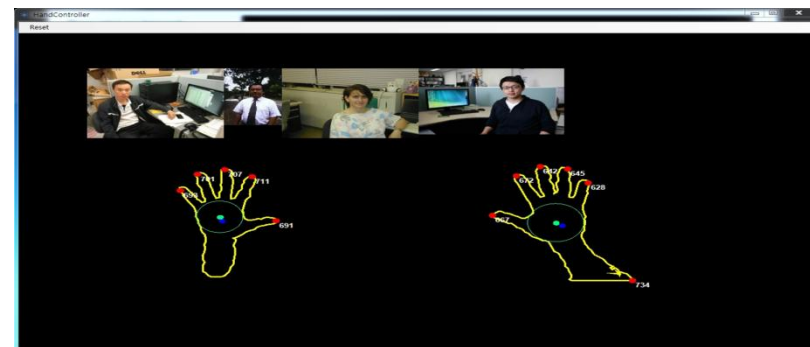


Figure 2. System Interface

## 3. Fingertips and Palm Tracking

The aim of the system is to let users manipulate images only by hand gestures, and with no wearable device requirement or complex construction. It is implemented by few steps for detecting and tracking.

### 3.1. Getting the Depth Image from KINECT

KINECT has infrared camera and PrimeSense sensor to compute the depth of the object while the RGB camera is used to capture the images. As Frati [19] stated "It has a webcam–like structure and allows users to control and interact with a virtual world through a natural user interface, using gestures, spoken commands or presented objects and images", it is clear that KINECT is a robust device and could be used in different complex applications. 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 of 11 bit with 2048 levels of sensitivity [21]. The depth value draw of a point in 3D can be defined as calibration procedure [19] $d = K\tan(H_{draw}+L) - O$ where d is the depth of that point in cm, H is $3.5\text{x}10\text{-}4$ rad, K=12.36 cm , L = 1.18 rad and O=3.7 cm.

### 3.2. Hand Detection and Tracking

It is tracking one or two hands using depth data gathered from the KINECT sensor. It implemented a simple k-means clustering algorithm, which divides all points of a frame that are closer than 80 cm into clusters. The blue dots are the centers of those clusters (see Figure 3). For performance gains, only every 25th point is measured.



Figure 3. Hand tracking with k-means clustering

### 3.3. Fingertips Detection and Tracking

The key was to use the hand's contour and then combine this information with the points in the convex hull. For each point in the hull (candidates for fingertips), find the nearest point in the contour curve. Let's call this set C.
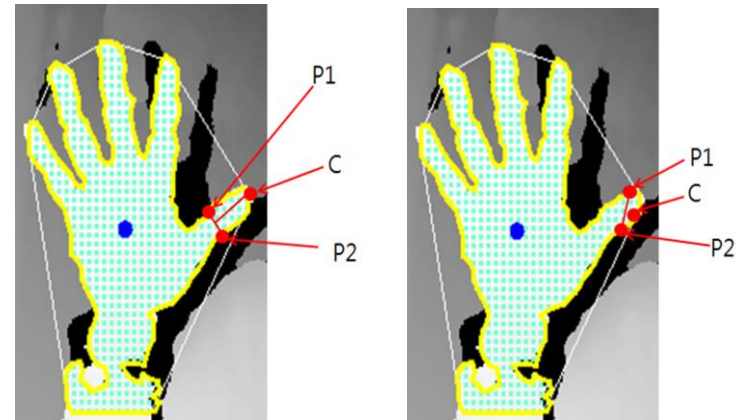


Figure 4. Fingertip(Left), Not Fingertip(Right)

For each point c in C, take the two points P1, P2 in the two different directions along the contour that are in a given distance to c (the ideal distance has to be found experimentally and depends on the hand shape's size). If these three points are aligned, then it's not a fingertip point. To find out if they are aligned, find the center of p1 and p2 and calculate the distance to c. If this distance is bigger than a certain value (to be found experimentally), the points are not on a line and the candidate point c is a fingertip point.

### 3.4. Palm Detection and Tracking

The solution is to find the center of the palm, which is quite stable during rotating, opening and closing the hand .This is done by finding the biggest circle inside the hands contour. The center of this circle is in most cases the center of the palm. This circle can be found by identifying the point inside the contour that maximizes the distance to the points in the contour line. The line through the center of the cluster and the center of the palm could also be used as hand orientation indicator.

## 4. Hand Gesture Interaction

### 4.1. Selecting Interaction

When user's hands enter detection area(see Figure 1), the system tracks the fingertips and palm. It can make selecting interaction with image when five fingertips and palm are detected(see Figure 5). Thereafter, system starts to capture user's hand gestures.
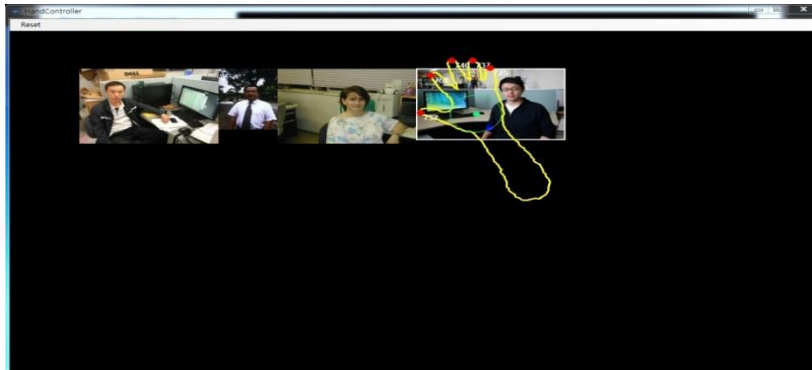
Figure 5. Selecting Interaction

Thereafter, system starts to capture user's hand gestures. User can make dragging interaction with images when only palm is detected(see   Figure 6). It is possible even in case of using both hands, and it can select multiple images and drag in case of both hands.
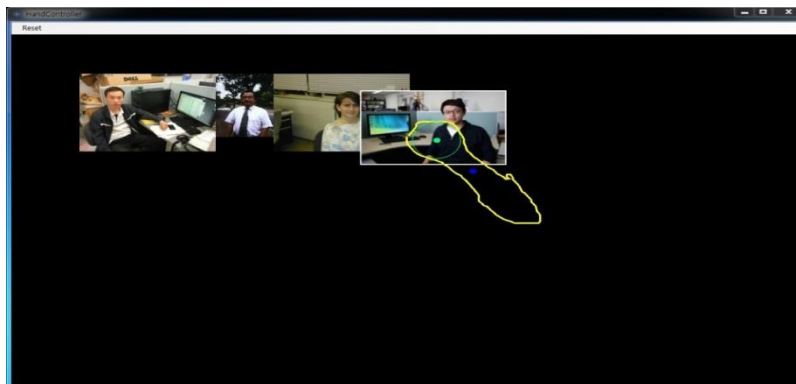
Figure 6. Grab and Drag Interaction

### 4.2. Resizing Interaction

In order to resize the image, system tracks fingertips number. It changes to resizing interaction mode when the system detects two fingers. In resizing interaction mode, system computes distance of each fingertip. It implements multi-touch function of the recent smart device with the hand gesture base.
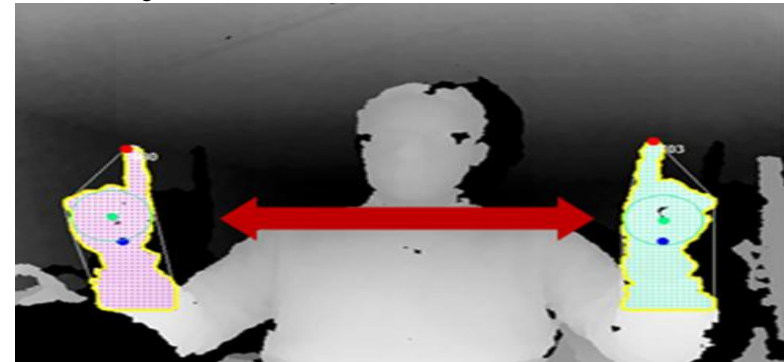
Figure 7. Zoom-in Gesture

As shown in Figure 7, if the distance of each fingertip becomes far to a certain value(to be found experimentally), zoom-in interaction will be performed(see Figure 8). Consequently, size of the image is expanded.

Figure 8. Zoom-in Interaction

On the contrary, if the distance of each finger becomes close at the certain value, zoom-out interaction will be performed and size of the image is reduced. Even in case of recognizing two fingertips in one hand, It is able to make same interaction with recognizing two fingertips

in two hands. However, it is restrictive due to the physical limitation comparing with the both hands.

### 4.3. Rotating Interaction

In order to rotate image, same process with resizing mode is required. For rotate interaction, detection of two fingers is required.
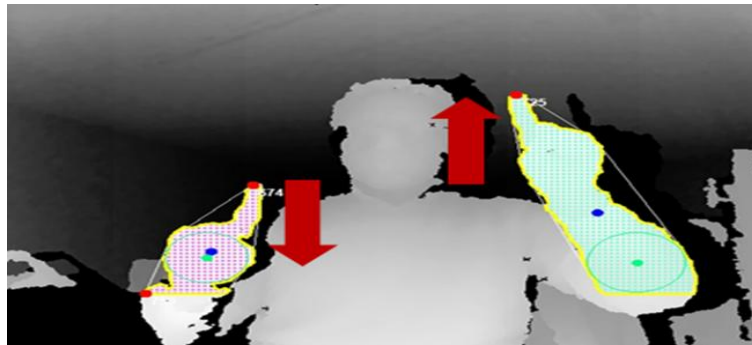


Figure 9. Clockwise Rotate Gesture

As shown in Figure 9, system computes each fingertip's position. If the left hand is raised to the upside and the right hand is moved downwards, then the image is rotated in the clockwise. Even in case of two fingertips in one hand, It is able to make rotate interaction as well.
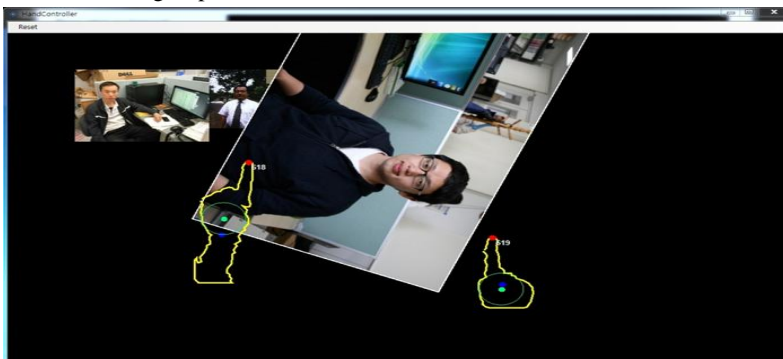


Figure 10. Clockwise Rotate Interaction

On the contrary, If the right hand is raised to the upside and the left hand is moved downwards, then the image is rotated in the counter clockwise.

### 5. Conclusion

In this paper, we presented Hand Controller, a natural user interface of image manipulation that allows user to resize, rotate and drag the image objects through hand gestures. Our work contributes a novel natural user interface for image controlling. The merits of our system are as follows. First, we implemented multi-touch function using hand gestures. Because of using the hand gestures, our system provided more intuitive and natural way to control images. Second, our system overcame limitation of Vision-based tracking problem under some natural conditions like fast hand motion, cluttered background, poor light condition. Third, performance was improved by using k-means clustering algorithm and convex hull methods together when tracking more than two hands.

### 6. References

1) Z. Ren, J. Meng and Z. Zhang. "Robust Hand Gesture Recognition with Kinect Sensor", MM'11, Proceedings of the 19th ACM international conference on Multimedia.(2011)

2) R. Urtasun and P. Fua. "3D Human Body Tracking Using Deterministic Temporal Motion Models", Lecture Notes in Computer Science, Vol. 3023, pp.92-106. (2004)

3) C. A. Pickering, K. J. Burnham, and M. J. Richardson. "Research Study of Hand Gesture Recognition Technologies and Applications for Human Vehicle Interaction", Automotive Electronics, 2007 3rd Institution of Engineering and Technology Conference, pp.1-15. (2007)

4) K. J. Wachs, M. Kölsch, H. Stern and Y. Edan. "Vision- based hand-gesture applications", Communications of the ACM, Vol. 54 Issue 2, pp.60-70. (2011)

5) S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, J. Shotton, S. Hodges and D. Freeman. "KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera". ACM UIST Proceedings ACM User Interface and Software Technologies, UIST'11, ACM, pp.1-9. (2011)

6) M. Tang. "Hand Gesture Recognition Using Microsoft's Kinect." Paper written for CS228, Winter 2010.Technologies, UIST'11, ACM, pp.1-9. (2011)

7) J. Sullivan and S. Carlsson. "Recognizing and Tracking Human Action." Lecture Notes in Computer Science, vol. 2350, 2002: 629-644.

8) X. Zabulis, H. Baltzakis and A. Argyros. "Vision-based Hand Gesture Recognition for Human-Computer Interaction." Computer Vision Techniques for Hand Gesture Recognition.

9) L. Bretzner, I. Laptev and T Lindeberg. "Hand Gesture Recognition using Multi-Scale Colour Features,   Hierarchical Models and Partical Filtering." Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002.

10) V. Frati and D. Prattichizzo. "Using Kinect for hand tracking and rendering in wearable haptics" IEEE World Haptics Conference(WHC2011), 21-24 June 2011, pp317-321.

11) http://en.wikipedia.org/wiki/Kinect