

# human-Computer Recognition using Gestures

Jaeseong Lee, Praful Gupta,

**Abstract**—Human-Computer Interaction (HCI) exists ubiquitously in our daily lives. It is usually achieved by using a physical controller such as a mouse, keyboard or touch screen. It hinders Natural User Interface (NUI) as there is a strong barrier between the user and computer. This report mainly discusses the design and implementation of hand gesture recognition and its four different applications which are Pacman game using optical flow vectors, virtual Drawing, Computer control and Gesture classification. We made a dedicated MFC Hand gesture recognition application on Visual Studio 2013 which means that it can run on any Windows platform with required libraries and input images. The main aim of this application is to make human computer interaction more efficient and easy in terms of usage since gestures are an important aspect of human-interaction, both interpersonally and in the context of man-machine interfaces.

**Index Terms**—image processing, hand gesture recognition, Open CV.

## I. INTRODUCTION

The discipline of human-computer interaction is studied to determine how to make computer technology more interactive for efficient human use. In the past years, computers used to be very expensive and were almost used by technical people, but now, they are not so expensive and the majority of people using them are non-technical. Thus it is very important to make human-computer interaction more user-friendly. This is the motivation behind our project. The organization of this report is as follows: Section 1 discusses the

## II. METHODOLOGY

### A. Pacman game using optical flow vectors

In this application, we compute local optical flow vectors to determine the magnitude and direction of the movement of pacman. The idea is to move the pacman using local motion vectors and make it eat apples which are randomly floating in the background. The user increases his score by a point everytime pacman eats an apple. The user has to make pacman eat all 20 apples in a limited time

frame, else he loses.

We used Lucas-Kanade method to compute optical flow vectors. This differential method of calculating optical flow vectors is widely known. It assumes that the flow is essentially constant in a local neighbourhood of the pixel under consideration, and solves the basic optical flow equations for all the pixels in that neighbourhood, by the least squares criterion. By combining information from several nearby pixels, the LucasKanade method can often resolve the inherent ambiguity of the optical flow equation. It is also less sensitive to image noise than point-wise methods. On the other hand, since it is a purely local method, it cannot provide flow information in the interior of uniform regions of the image. The concept behind the optical Flow computation is as follows:

$$I_x(q_i) + I_y(q_i)V_y = -I_t(q_i)$$

where  $q_i$  is each pixel inside the window and  $I_x, I_y$  and  $I_t$  are the partial derivatives of the image  $I$  with respect to position  $x, y$  and time  $t$ , evaluated at the point  $q_i$  and at the current time. This system has more equations than unknowns and thus it is usually over-determined. The LucasKanade method obtains a compromise solution by the least squares principle.

$$A^T A v = A^T b$$

$$v = (A^T A)^{-1} A^T b$$

where  $v$  represents the final optical flow vectors.

### B. Gesture Recognition

We use the webcam output as the input for our preprocessing model. We implement various functions in preprocessing model which are discussed in detail.

1) *Background Subtraction:* Background subtraction is performed to effectively segment the user from the background. This is done to eliminate any object that is present in the background and resemble skin color. Typical methods use a static background image and calculate the absolute difference between the current frame and the background image. We calculate the absolute difference in YCrCb color space and split it into three channels (Y, Cr and Cb); then process each channel independently. We use a different min and max threshold for each channel, and then remove noise in each channel using a morphology operator. Finally, we merge these channels back together by an addition operator and then masking it with the original frame by using an AND operator. The output is then a YCrCb image containing skin color like face and hand.

both hand and face regions will be extracted. Therefore, we utilize Haar-like features by Viola Jones to detect the face region and then remove it by simple flood fill. Hence, face region will not be extracted during the next step. Haar-like features is efficient and it can detect human faces with high accuracy and performance. Preparing a good Haar classifier is a time consuming task, so we use the well-trained classifier provided in OpenCV library.

3) *Skin Color Extraction:* We use YCrCb color ranges for representing the skin color region. It also provides good coverage for different human races. The default values that we used as threshold for skin color extraction are as follows:

$$77 \leq Cb \leq 127 \quad \text{and} \quad 133 \leq Cr \leq 173$$

4) *Morphology operations and image smoothing:* In order to remove noise efficiently, we apply a morphology Opening operator (Erosion followed by Dilation) in several stages; during background subtraction and after skin extraction. After that, we apply a Gaussian filter and threshold binarization with proper value to smooth the contours and remove pixel boundary flickers.

5) *Contour extraction and polygon approximation:* We apply polygon approximation on the extracted contour to make it. Simple contours do not give much information about a hand, but we can find the interior contours and combine with other information to determine specific hand gestures.

6) *Distance Transform:* The distance map labels each pixel of the image with the distance to the nearest obstacle pixel. This method is very efficient in finding the center of the palm even when the entire hand is exposed to the webcam. We used the distance intensity at the center of the palm as the radius of the palm. This is illustrated in the figure.

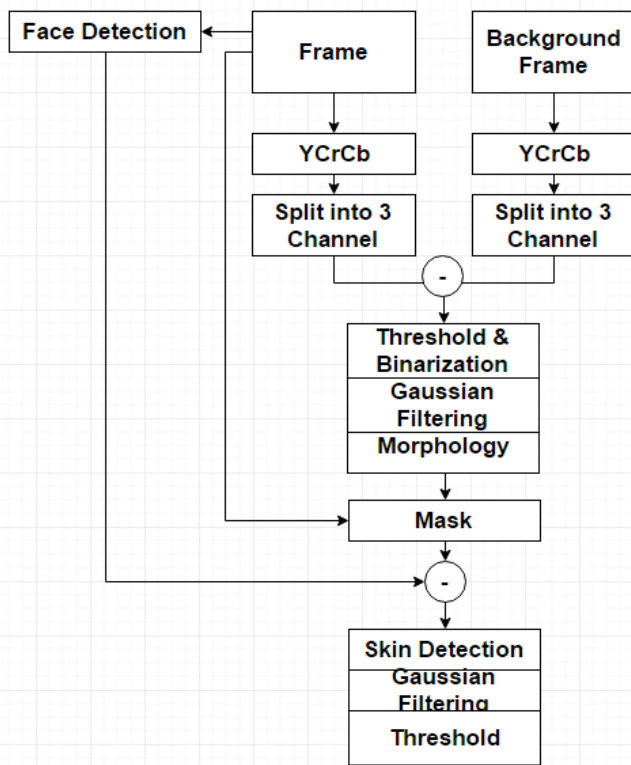


Figure 1: Gesture recognition flow diagram

2) *Face Removal:* Our hand region extraction method is based on skin color extraction; therefore,



Figure 2: Distance transform of hand

7) *Convex hull and convexity defects extraction:* A useful way of comprehending the shape of hand or palm is to compute a convex hull for the object and then compute its convexity defects. For a single convexity defect, there is a start point ( $p_s$ ), depth point ( $p_d$ ), end point ( $p_e$ ) and depth length ( $l_d$ ) as shown in Fig.

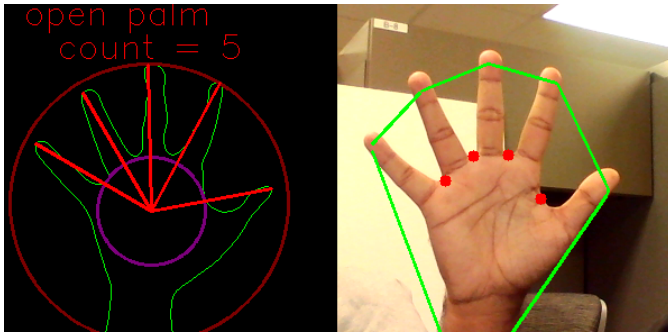


Figure 3: convex hull and convexity defect

8) *Simple hand gesture recognition:* Based on the number of convexity defects, number of fingers and ratio of inradius of the palm and the radius minimum enclosing circle of the hand, we classified different figures as shown in the figure.

### III. RESULTS

We obtain the hand gesture recognition system with several following applications that people use

with electrical devices.

#### A. Applications

1) *pacman game:* The pacman game is demonstrated. the pacman moves with subject movement such as hand and body using the iterative Lucas-Kanade method with pyramid. The pacman is to eat apples moving within the camera frame and subjects are competing with others with time consumption for eating all apples. It is noted that control of pacman tracking is convenient and fast to play pacman game, so subjects enjoy the game.

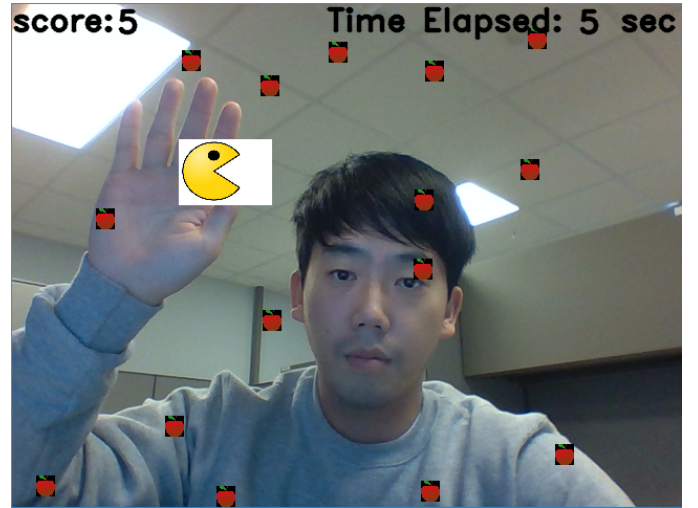


Figure 4: pacman game demonstration

2) *Virtual drawing:* We demonstrated virtual drawing, which is a platform for drawing with a certain color on the whiteboard and frame images. In frame image, it has buttons for selecting color (red, green and blue), erasing drawings and clearing all drawings. In addition, one finger is for clicking down selection and drawing and two fingers is for clicking up. It is noted that it is convenient and fast to draw and it is also easy to draw and erase.

3) *Virtual computer control:* We demonstrated computer control system with hand gesture. Similar with the virtual drawing, one finger is for clicking up and moving the mouse cursor, and two fingers is for clicking down. In addition, we can zoom-in and out with moving up and down with open hand, respectively. we search google maps, which is a web mapping service developed by Google with the

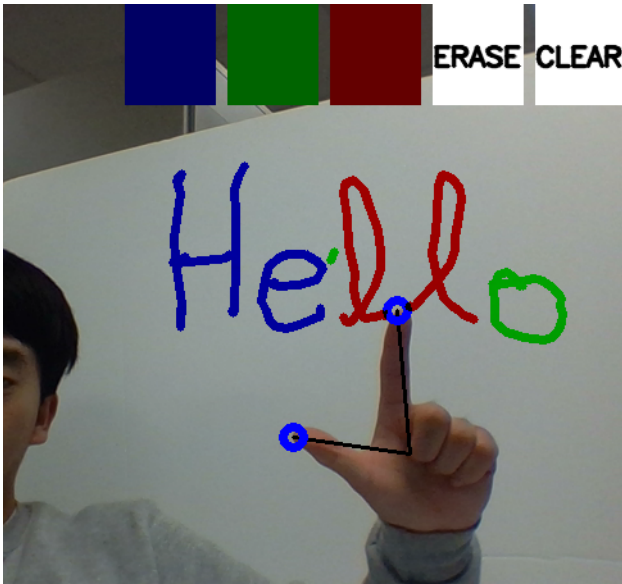


Figure 5: Virtual drawing

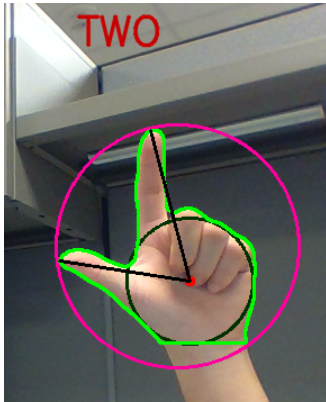


Figure 6: One finger up classification

function. It is noted that we obtained high accuracy of searching places that we are interested in on the Google Maps.

4) *Hand gesture classification:* We demonstrated Hand gesture classification. we assign 8 hand gestures, which are one to five finger up, open and close hand, and okay sign. It is noted that we obtained high accuracy of classifying gestures. In addition, we split arm area and hand area so that it function more robust.

#### IV. ACHIEVEMENTS

For skin color selection in each frame, certain range of color in YCbCr color demension is com-

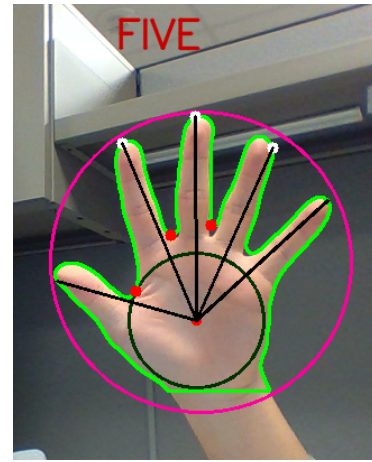


Figure 7: Five fingers up classification

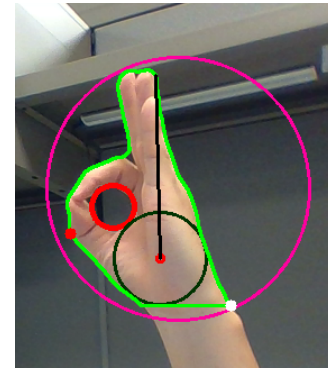


Figure 8: OK sign classification

monly used and it works with high accurate performance. However, since skin color is supposed to be detected differently in different environments, thresholding with a specific range of color for skin is not a very good approach for skin detection. it could limit the general performance for all human beings. So we came up with one idea for the thresholding color issue by assuming that skin color of face is same as hand color, and adapting the threshold value using face image histogram.

In addition, in order to generalize the system in various environment, we have achieved many ideas. at the beginning the performance was limited with many condition such as wearing full sleeves, face position and background-dependent performance. However, it has a trade-off issue with hardware speed for real-time usage. Therefore, making it simple and clear by implementing haar-cascade face detection algorithm for face removal and distance

transform to split hand and arm area, it works with high accuracy.

In addition, From the finger classification, we learned that finding convex hull and convexity defects can be efficiently used for hand recognition. Since the algorithm is different from the way that human being classifies fingers, convexity defects, at the beginning of the project, it was not stable and there were many incorrect convex hull and convexity defects. So we approach it into extracting the region of interest in image and we achieved stable well-located convex hull and convexity points.

We have indeed learned a lot from this project and consider all the time devoted to it absolutely worthwhile. The only thing one can do is to keep improving. There are so many times in this semester that we doubted our ambition and capacity. Those are the moments when determination and confidence became critical. Solving the problems one by one, step by step, we eventually reach the stage today where our project can actually function. Definitely we know that it's far away from perfect today, but we will keep working on it, exploring, and enjoying more delightful moments succeeding with image processing.

## REFERENCES

- [1] B. D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision. Proceedings of Imaging Understanding Workshop*, pages 121–130, 1981.
- [2] Hui-Shyong Yeo, Byung-Gook Lee, Hyotaek Lim, *Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware*, Springer, pp 2687-2715, April 2005.