International Journal of Research and Reviews in Computer Science (IJRRCS)
Vol. 3, No. 1, February 2012, ISSN: 2079-2557

© Science Academy Publisher, United Kingdom
www.sciacademypublisher.com

Identifying Fingertips for Human Computer Interaction

R.M. Arunachalam, M. Ashok Gowtham, and R. Aarthi

Department of Information Technology, Amrita VishwaVidyapeetham, Coimbatore, India

Abstract – With the explosive evolution of computer technology, Human-computer interfaces are finding an increased importance in daily life. The use of input devices such as the mouse and keyboard limits the friendliness of the user interaction. Therefore, to increase the interaction with the system, a method is proposed which employs tracking the tip of the fingers using a single camera. It employs a graphical technique involving angle and points. It describes a technique by which the location of fingertip is mapped with that of the monitor. The outcome of this technique in a real time scenario is found to be efficient. Identifying a finger with the help of a fingertip paves way for increased expression of gestures.

Keywords – Segmentation, Fingertip, Gestures, Tracking, Human computer interaction

1. Introduction

Hand gestures play a fundamental role in human computer communication. The gestures make use of a hand motion tracking system. Data gloves are commonly used as input devices [1][4] but are expensive and inhibit free movements. The major disadvantage of glove-based devices is that the user cannot wear these devices as human hands vary in size and shape [5]. So the most natural form of hand pose tracking without using any gloves can be used [4]. Identifying hand gestures in a complex background is one of the challenging tasks in hand gesture recognition. This method does not require storage of any data or training to detect the input hand gestures. Most of the hand interactive systems can be considered to comprise of three layers: detection, tracking and recognition [2][3]. The detection layer is responsible for defining and extracting visual features that can be attributed to the presence of hands in the view of the camera. The tracking layer is used to get data association between successive image frames, so that, at each moment in time, the system may be aware of what is where. Recognition is used to interpret the hand location, posture, or gesture conveys.

Various approaches have been proposed for hand detection using skin color segmentation. A commonly employed color space includes RGB, normalized RGB, HSV, YCbCr, YUV, etc. [6][8][9]. Occurrence of skin coloured objects in the background may lead to error. One common solution to address this issue is background subtraction. Some of the other approaches are based on the characteristic shape of hands, learning detectors from pixel values, 3D model-based detection, Motion-based hand detection, etc. [2][7].

The number of open fingers can be used as descriptor for basic applications. In this paper we propose a method to find the number of active fingers in a segmented hand image. The finger count is found by detecting the finger tips and counting them. The finger tips are detected using boundary analysis by finding the maximum curvature points.

2. Architecture

The proposed method consists of the following stages as in Figure 1.

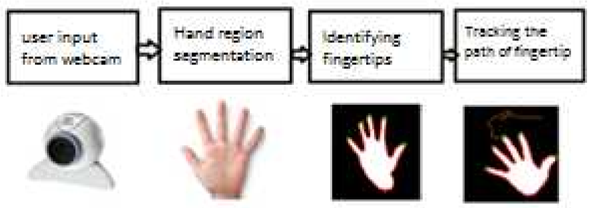


Figure 1. Method architecture.

2.1. *User Input*

The input is taken with the help of the webcam attached to the computer. The gestures are captured such that their field of view is parallel to the field of view of the webcam. The video is processed and the hand region is segmented. There must be proper illumination so that the video processed gives the correct result. The minimum resolution of the webcam is 640 x 480. The limitation is that the user should not wear any accessories which are of skin color range.

2.2. *Hand region segmentation*

The perceived color of human skin varies greatly across people in different parts of the world or even between individuals of the same country. Additional variability may be introduced due to changing illumination conditions and/or camera characteristics. Hence color-based approaches for hand detection must employ some means of compensation to account for this variability. Other elements may also be present along with the hand, and thus background must be separated from the foreground. After separation, the skin segmentation algorithm is applied to create a binary image. Therefore, an adaptive dynamic segmentation technique can be used which allows the system to recognize skin color as the user changes. The disadvantage is that it consumes some time but produces an accurate result.

The input got through the webcam is in RGB model and by converting it to HSI color model, skin regions can be easily identified. The regions which lie in this range as shown in (1) are detected as skin. The hue value must be less than 0.1 or greater than 0.9 and the saturation must be in the range of 0.1 to 0.9.

The input got through the webcam is in RGB model and by converting it to HSI color model, skin regions can be easily identified. The regions which lie in this range as shown in (1) are detected as skin. The hue value must be less than 0.1 or greater than 0.9 and the saturation must be in the range of 0.1 to 0.9.

$$H < 0.1 \text{ or } H > 0.9 \text{ and } 0.1 < S < 0.9 \quad (1)$$

The skin regions are made white and all other regions are made black. The largest region which is detected as skin is taken as the hand region. This gives the segmented hand region which is the region of interest. The gesture recognition process depends on this region.

2.3. Identifying Fingertips

The segmented region of the hand is our region of interest. A bounding box enclosing this region is drawn. The boundary points are calculated.

At every boundary point a line joining the current point and the next l^{th} point is considered. In our experiment value of l is 20. The angle made by this line along with the x-axis is calculated using the formula as shown in (2),

$$\theta = \tan^{-1}\left[\frac{y_2-y_1}{x_2-x_1}\right] \quad (2)$$

Where θ is the angle made at the point P1 along with the x-axis. P1(x1, y1) is the current point and P2(x2, y2) is the next point taken after l points from P1. The readings recorded are plotted into a graph (Figure 2).

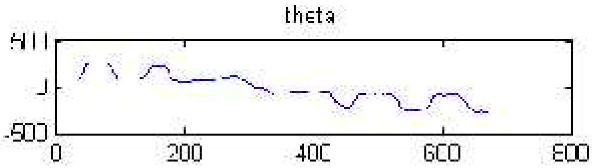


Figure 2. Graph between calculated angle and points.

The x-axis in the graph corresponds to the boundary point number and the y-axis stands for angle θ at that point. This graph is smoothened to remove mild changes or variations (Figure 3).

Then with the help of median filter the noise is removed (Figure 4). If the boundary points were taken in counter-clockwise direction then the position on the graph (Figure 4) where the curve falls denotes the tip of a finger.

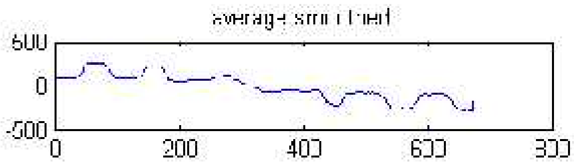


Figure 3. The smoothed curve.

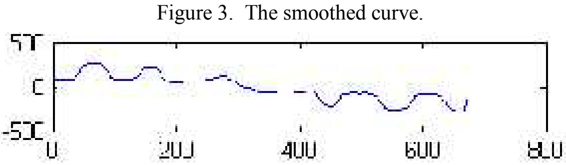


Figure 4. After applying median filter.

To compute the falling point, the noise free angle values are taken and the difference filter is applied as shown in (3) below (Figure 5).

$$[1\ -1] \quad (5)$$

Peaks in the differentiated graph whose heights are greater than a threshold value correspond to tips. In the curve the depth on both sides of the peak are considered for calculating peak height. The minimum of the depths is the peak height. Threshold is applied on the peak height. The points on the boundary corresponding to the peaks are marked as tips (Figure 6). It is used along with finger count to overcome the drawback in identifying gestures.

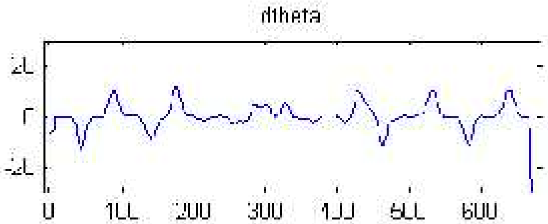


Figure 5. Differentiated graph.

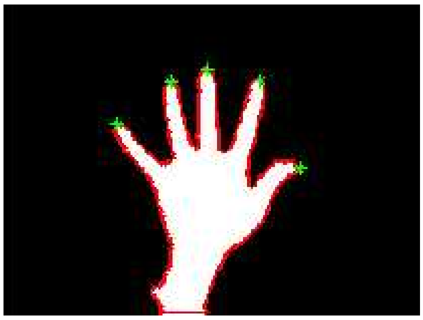
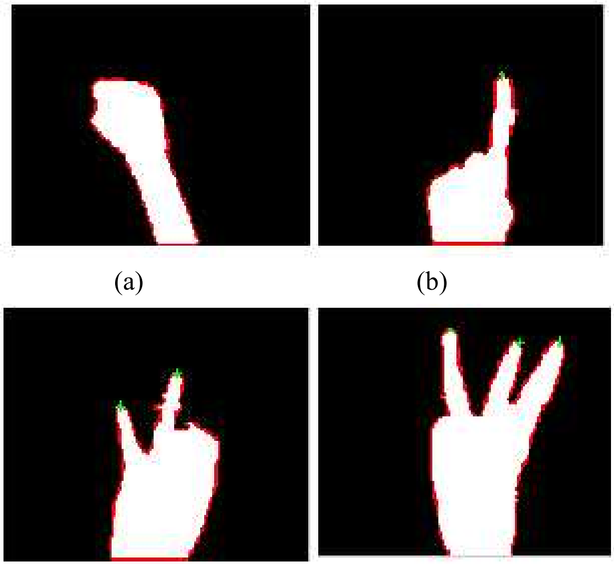


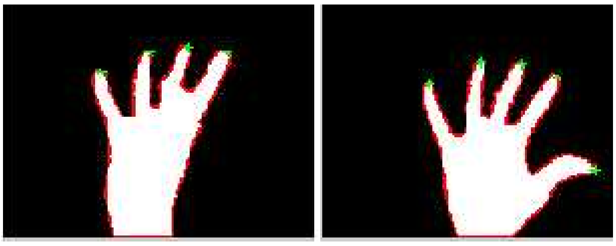
Figure 6. Tips identified.

Some of the inputs given and its results are shown in Figure 7.



(a) (b)

(c) (d)



(e) (f)

Figure 7. (a) No tips (zero) (b) One (c) Two (d) Three (e) Four (f) Five.

2.4. Tracking

Once the fingertip is detected, the first peak in counter clockwise direction from the tip touching the top of the bounding box is used to track the path. Using the value of x-coordinate the tips are sorted and labeled in counter clockwise direction. It is used to choose the path of a specified tip by its label. The path is made by connecting the coordinates in each frame (Figure 8). The x co-ordinate, y co-ordinate, and CPU time are stored. The CPU time is used to differentiate two similar gestures with a pause. Now these parameters can be given as input to HMM and compared with the gestures.

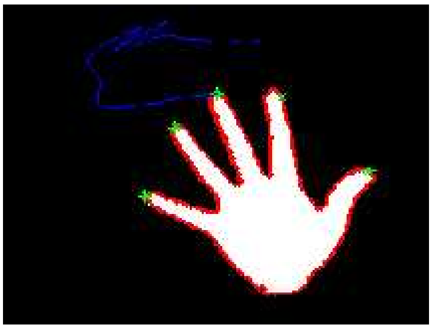
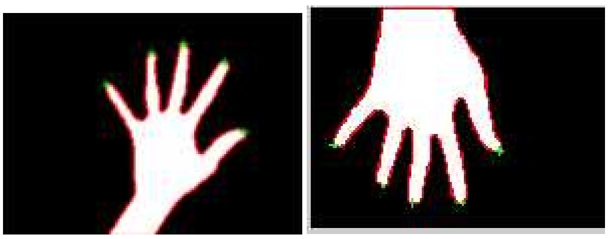


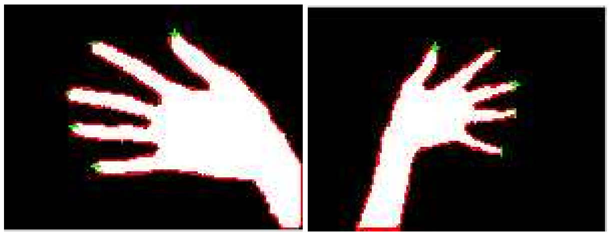
Figure 8. Tracking.

3. Experimental Results

The technique was implemented on a system with Intel dual core processor with speed of 2.53 GHz using MATLAB software. The experiment was conducted under various environments which were illuminated with different kind of light sources. The hand is give as input in all four directions (Figure 9) in real time and it work perfectly. The percentage of accuracy obtained is 100%, taking into account that no element in the surrounding is similar to skin color. Also the fingers need to be spread out. If user wears a band of skin color range it affects the output (Figure 10).



(a) (b)



(c) (d)

Figure 9. (a) Hand facing up (b) Down (c) From right (d) From left

Figure 10. User wearing a band.

The hand is also given as input in different orientation. As we change the angle of the webcam, the position of the hand parallel to the plane of the webcam is denoted as 0° angle. The hand can be oriented at different angles to the plane of the webcam. The orientation is calculated as shown below (Figure 11). The orientation of right hand is taken as positive and that of left hand is taken as negative. The accuracy of detection of fingertips at different orientations of hand in that plane is tabulated in Table 1. It doesn't work at 90° except for one finger. From the observation the technique works well in the range of -45° to +45°.

Table 1.Accuracy at Different Orientations

Fingers	Orientation			
	±0°	**±30°**	**±45°**	**±60°**
1	100%	100%	100%	100%
2	100%	100%	100%	85.71%
3	100%	100%	100%	75%
4	100%	100%	100%	62.50%
5	100%	100%	100%	42.81%

(a) (b)

(c)

Figure 11. (a) Orientation 0° (b) Orientation +45°
(c) Side view orientation 0°.

4. Conclusion

This method of HCI does not involve of storage of any images in database to identify the gestures. The hand gestures can be identified based on fingertips and tracking used to make a gesture. The accuracy of identifying a gesture decreases if the surrounding has elements with skin color. Along with the finger count technique, this method can be combined to make use of number of gestures and hence able to handle other applications which involve more gestures.

References

[1] Deliang Zhu, Zhiquan Feng, Bo Yang, Yan Jiang, Tiantian Yang "The Design and Implementation of 3D Hand-based Human-Computer Interaction Platform" Sch. of Inf. Sci. & Eng., Univ. of Jinan, Jinan, China, Vol. 2., pp. 485-489, November 2010.

[2] X. Zabulisy, H. Baltzakisy, A. Argyroszy "Vision based Hand Gesture Recognition for Human Computer Interaction."

[3] Antonis A. Argyros and Manolis I.A. Lourakis "Vision-Based Interpretation of Hand Gestures for Remote Control of a Computer Mouse."

[4] Robert Y. Wang "RealTime HandTracking as a User Input Device" Massachusetts Institute of Technology, Cambridge, MA USA Vol. 28, Issue: 3, ACM, pp: 1-8

[5] J. LaViola."A survey of hand posture and gesture recognition techniques and technology", Department of Computer Science, Brown University, Providence, Rhode Island, 1999.

[6] Sanjay Kr. Singh, D. S. Chauhan, Mayank Vatsa, Richa Singh "A Robust Skin Color Based Face Detection Algorithm."

[7] Mark Bayazit, Alex Couture-Beil, Greg Mori "Real-time Motion-based Gesture Recognition using the GPU." Simon Fraser University Burnaby, BC, Canada, May 2009

[8] T. S. Caetano, D. A. C. Barone," A probabilistic model for human skin color", IAP Conf. 2001.

[9] X. Zhu, J. Yang, and A. Waibel."Segmenting hands of arbitrary color ", March 2000.

[10] J. Rehg and T. Kanade, "Vision-based hand tracking for human-computer interaction." In Workshop on Motion of Non-Rigid and Articulated Bodies, Austin Texas, November 1994