HAND GESTURE RECOGNITION SYSTEM FOR VIRTUAL MONITOR USING PYTHON

[1]Y Naga Durga Aravind, [2]Jaya Kumar Reddy, [3]Dr. E.Nagarjuna, [4]Avurupalli Samba Siva
[1,2,3]Assistant Professor, [4]Student, Dept. of Computer Science Engineering, Newton's Institute of Engineering, Macherla, Andhra Pradesh, India.

ABSTRACT

Now a day's computer is used by every person in their work. In this project our main aim is to make computers easily understand the human language. The project introduces an application using computer vision for Hand gesture recognition.

A camera records a live video stream, from which a snapshot is taken with the help of interface. The system is trained for each type of count hand gestures (one, two, three, four, and five) at least once. After that a test gesture is given to it and the system tries to recognize it.

Here we introduce a technique called OpenCV and PYTHON. By using this technique we have first detect pre-processing and recognize the hand fingers and the count. Then with the help of recognized fingers count, these gestures will help us to act as a mouse to perform the different operations of mouse. This hand mouse interface is known as "virtual monitor".

Keywords: Hyper local, Services, Transactions, Delivery, Communication, Correlation

INTRODUCTION

Recently in our daily life, communication between human and computer plays an important role. We are always looking for easy ways of interaction for machines.

Gesture recognition is a modern and innovative, it enables to communicate human with the machine by using this concept.

In human computer interaction mainly used input for system are mouse and keyboard.

The task of hand gesture recognition is one the important and elemental problem in computer vision. With recent advances in information technology and media, automated human interactions systems are build which involve hand processing task like hand detection, hand recognition and hand tracking.

Hand detection is related to the location of the presence of a hand in a still image or sequence of images i.e. moving images. In case of moving sequences it can be followed by tracking of the hand in the scene but this is more relevant to the applications such as sign language. The underlying concept of hand detection is that human eyes can detect object which machine cannot with that much accuracy as that of a human. From machine point of view it is just like a man fumble around with his senses to find an object.

Hand detection and recognition have been significant subjects in the field of computer vision and image processing during the past 30 years. There have been considerable achievements in these fields and numerous approaches have been proposed. However, the typical procedure of a fully automated hand gesture recognition system can be illustrated in the **Figure 1** below:

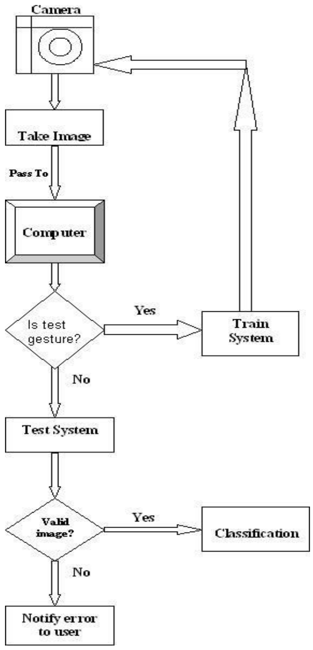


Figure 1: Hand Gesture Recognition Flow Chart

Hand gesture recognition research is classified in three categories. First "**Glove based Analysis**" attaching sensor with gloves mechanical or optical to transducers flexion of fingers into electrical signals for hand posture determination and additional sensor for position of the hand. This sensor is usually an acoustic or a magnetic that attached to the glove. Look-up table software toolkit provided for some applications to recognize hand posture.

The second approach is "**Vision based Analysis**" that human beings get information from their surroundings, and this is probably most difficult approach to employ in satisfactory way. Many different implementations have been tested so far. One is to deploy 3-D model for the human hand. Several cameras attached to this model to determine parameters corresponding for matching images of the hand, palm orientation and joint angles to perform hand gesture classification. Lee and Kunii developed a hand gesture analysis system based on a three-dimensional hand skeleton model with 27 degrees of freedom. They incorporated five major constraints based on the human hand kinematics to reduce the model parameter space search. To simplify themodel matching, specially marked gloves were used.

The Third implementation is "**Analysis of drawing gesture**" use stylus as an input device. These drawing analysis lead to recognition of written text. Mechanical sensing work has used for hand gesture recognition at vast level for direct and virtual environment manipulation. Mechanically sensing hand posture has many problems like electromagnetic noise, reliability and accuracy. By visual sensing gesture interaction can be made potentially practical but it is most difficult problem for machines.

Full American Sign Language recognition systems (words, phrases) incorporate data gloves. Takashi and Kishino discuss a Data glove-based system that could recognize 34 of the 46 Japanese gestures (user dependent) using a joint angle and hand orientation coding technique. From their paper, it seems the test user made each of the 46 gestures 10 times to provide data for principle component and cluster analysis. The user created a separate test from five iterations of the alphabet, with each gesture well separated in time.

Head and Face Gestures When people interact with one another, they use an assortment of cues from the head and face to convey information. These gestures may be intentional or unintentional, they may be the primary communication mode or back channels, and they can span the range from extremely subtle to highly exaggerate. Some examples of head and face gestures include: nodding or shaking the head, direction of eye gaze, raising the eyebrows, opening the mouth to speak, winking, flaring the nostrils and looks of surprise, happiness, disgust, anger, sadness, etc.

Hand and Arm Gestures

These two parts of body (Hand & Arm) have most attention among those people who study gestures in fact much reference only consider these two for gesture recognition. The majority of automatic recognition systems are for deictic gestures (pointing), emblematic gestures (isolated signs) and sign languages (with a limited vocabulary and syntax).

Explicitly Defined Skin Region

Following are some common ethnic skin groups and there RGB color space:

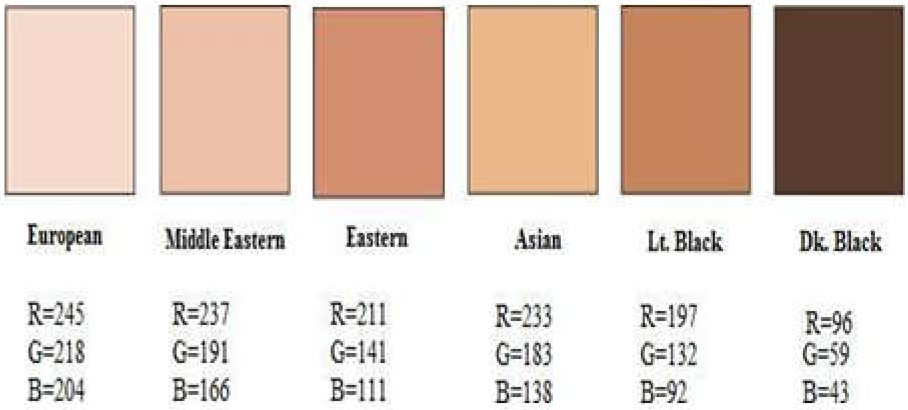


Figure 2: Different Ethnic Group Skin Patch

To build a skin classifier is to define explicitly through a number of rules the boundaries of skin color cluster in some color space. The advantage of this method is the simplicity of skin detection rules that leads to the construction of very rapid classifier. For Example [7]

(R,G,B) is classified as skin if:

$R > 95$ and $G > 40$ and $B > 20$ and $\max\{R,G,B\}-\min\{R,G,B\} > 15$ and

$|R-G| > 15$ and $R > G$ and $R > B$

In this classifier threshold defined to maximize the chance for recognizing the skin region for each color. If we see in **Figure 2** that Red color in every skin sample is greater than 95, Green is greater than 40 and Blue is greater than 20 in**.** So threshold can make this classifier easily detect almost all kind of skin.

This is one of the easiest methods as it explicitly defines skin-color boundaries in different color spaces. Different ranges of thresholds are defined according to each color space components in as the image pixels that fall between the predefined ranges are considered as skin pixels.

The advantage of this method is obviously the simplicity which normally avoids of attempting too complex rules to prevent over fitting data. However, it is important to select good color space and suitable decision rules to achieve high recognition rate with this method.

REMOVAL OF BACKGROUND

I have found that background greatly affects the results of hand detection that's why I have decided to remove it. For this I have written our own code in spite of using any built-in ones.

Before After

Figure.3. Removal of Background

CONVERSION OF RGB TO BINARY

All algorithms accept an input in RGB form and then convert it into binary format in order to provide ease in recognizing any gesture and also retaining the luminance factor in an image.

HAND DETECTION

mage could have more than one skin area but we required only hand for further process. For this I choose criteria image labeling which is following:

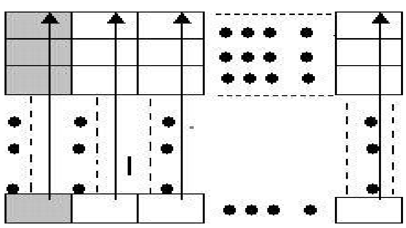
LABELING

To define how many skin regions that we have in image is by labeling all skin regions. Label is basically an integer value have 8 connecting objects in order to label all skin area pixel. If object had label then mark current pixel with label if not then use new label with new integer value. After counting all labeled region (segmented image) I sort all them into ascending order with maximum value and choose the area have maximum value which I interested because I assume that hand region in bigger part of image. To separate that region which looked for, create new image that have one in positions where the label occurs and others set to zero.

ROW VECTOR ALGORITHM

We know that behind every image is a matrix of numbers with which we do manipulation to derive some conclusion in computer vision. For example we can calculate a row vector of the matrix. A row vector is basically a single row of numbers with resolution 1*Y, where Y is the total no of columns in the image matrix.

Each element in the row vector represents the sum of its respective column entries as illustrated.



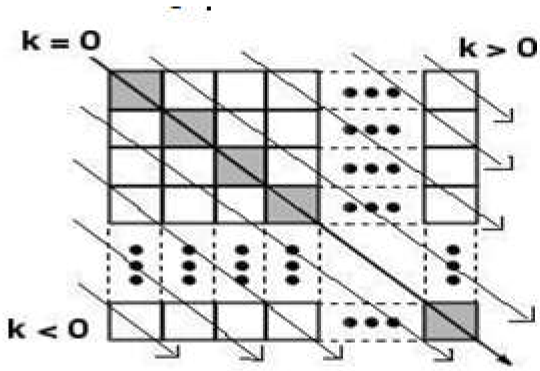
DIAGONAL SUM ALGORITHM

In the pre-processing phase, doing mentioned steps in methodology, skin modeling removal of the background, conversion of RGB to binary and labeling. The binary image format also stores an image as a matrix but can only color a pixel black or white (and nothing in between). It assigns a 0 for black and a 1 for white. In the next step, the sum of all the elements in every diagonal is calculated. The main diagonal is represented as k=0 i; the diagonals below the main diagonal are represented by k<0 and those above it are represented as k>0.

Here we introduce a technique called OpenCV using Convex Hull algorithm. In this project we create a Convex Hull around the hand using OpenCV library.

Working

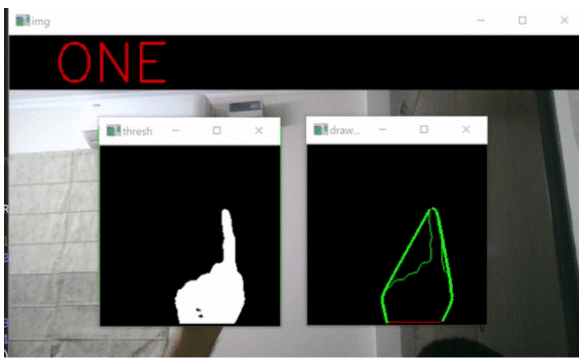
1. First we preprocess the image by applying operations like gaussian blur, thresholding.
2. Next step is to find the contours in the image.



3. The contour with the largest area(our hand) is used in drawing a complex hull.
4. We find the defects in that convex hull. These defects will be considered only if their angle is <=90 degree. (These defects will help us in finding the number of fingers).

- one defects = 1
- two defect = 2
- three defects = 3
- four defects = 4
- five defects = 5

Demo:



demo.gif

Convex hull function

```
# create hull array for convex hull points
hull = []
# calculate points for each contour
for i in range(len(contours)):
# creating convex hull object for each contour
 hull.append(cv2.convexHull(contours[i], False))
```

Finding Contours

Next, we find the contour around every continent using the **findContour** function in OpenCV. Finding the contours gives us a list of boundary points around each blob.

Finding contours for the thresholded image

im2, contours, hierarchy = cv2.find Contours (thresh, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)

CONCLUSION

a. Human hand gestures provide the most important means for non-verbal interaction among people.

b. At present, artificial neural networks are emerging as the technology of choice for many applications, such as pattern recognition, prediction, system identification, and control.

c. The ability of neural nets to generalize makes them a natural for gesture recognition

REFERENCES

1. Zhigang, F. Computer gesture input and its application in human computer interaction. Mini Micro Syst. **1999**, 6, 418–421. [**Google Scholar**]

2. Mitra, S.; Acharya, T. Gesture recognition: A survey. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **2007**, 37, 311–324. [**Google Scholar**] [**CrossRef**]

3. Ahuja, M.K.; Singh, A. Static vision based Hand Gesture recognition using principal component analysis. In Proceedings of the 2015 IEEE 3rd International Conference on MOOCs, Innovation and Technology in Education (MITE), Amritsar, India, 1–2 October 2015; pp. 402–406. [**Google Scholar**]

4. Kramer, R.K.; Majidi, C.; Sahai, R.; Wood, R.J. Soft curvature sensors for joint angle proprioception. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011; pp. 1919–1926. [**Google Scholar**]

5. Jesperson, E.; Neuman, M.R. A thin film strain gauge angular displacement sensor for measuring finger joint angles. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, New Orleans, LA, USA, 4–7 November 1988; pp. 807–vol. [**Google Scholar**]

6. Fujiwara, E.; dos Santos, M.F.M.; Suzuki, C.K. Flexible optical fiber bending transducer for application in glove-based sensors. IEEE Sens. J. **2014**, 14, 3631–3636. [**Google Scholar**] [**CrossRef**]

7. Shrote, S.B.; Deshpande, M.; Deshmukh, P.; Mathapati, S. Assistive Translator for Deaf & Dumb People. Int. J. Electron. Commun. Comput. Eng. **2014**, 5, 86–89. [**Google Scholar**]

8. Robust Part-Based Hand Gesture Recognition Using Kinect Sensor Zhou Ren, Junsong Yuan, Member, IEEE, Jingjing Meng, Member, IEEE, and Zhengyou Zhang, Fellow, IEEE, 15, AUGUST 2013

9. A Fast Gesture Recognition Scheme for Real-Time Human-Machine Interaction Systems . Ching-Hao Lai* Smart Network System Institute Institute for Information Industry Taipei City, Taiwan , 2010