Finger rotation detection using a Color Pattern Mask

V. Shishir Reddy[1], V. Raghuveer[2] , R. Hithesh[3], J. Vamsi Krishna[4,] , R. Pratesh Kumar Reddy[5], K. Chandra lohit[6]
[1,2,3,4,5,6]Electronics and Communication, M.A.N.I.T, Bhopal, India.
[1]shishirvutukur@gmail.com, [2]vemuriraghu@yahoo.com

ABSTRACT

Gestures play a key role in making interaction of human with computers with ease. The idea of this paper is to use rotation of finger as a gesture. Since detection of rotation of bare finger is difficult, we employed a mask to be put on finger with a specific pattern of colors printed on mask. Whenfinger is rotated in front of camera then area ofrespective colors on mask get changed in images captured from camera.This change in area is taken as measure of rotation. So using image processing techniques we find out the rotation of the finger based on difference in areas of respective color regions in respective frames.

Keywords: component, Rotation, normalization, gesture.

INTRODUCTION

Gestures are one of the most important modes of communicating with computer in an interactive environment. Interpretation of human gestures by a computer or any processor is used for human-machine interaction in the area of computer vision. The main purpose of gesture recognition research is to identify a particular human gesture and convey information and control things to the user pertaining to individual gesture. From the corpus of gestures, specific gesture of interest can be identified and in from that specific gesture various parameters are calculated relative to previous frame involving gesture , and on the basis of calculated parameters, specific command for execution of action can be given to robotic system or interact with computer .

Hand gesture recognition can be used to enhance human–computer interaction without depending on traditional input devices such as joysticks and mouse. Color is one of the most distinctive clues in finding objects and is generally input as RGB information via a CCD or a video camera. The RGB color system is known to be sensitive to lighting conditions, since the RGB system includes the mixed information of color and intensity. So we used specific colors red, blue and black .And to make the model well verse with various lighting conditions, a special technique of normalization is employed to identify the colors perfectly.What we basically use is a mask consisting of 3 color strips as shown in figure 2. Mask is put on finger and when finger is rotated left the area of left side colored strip decreases and right side colored strip increases when rotated at a fixed position of finger and about finger as shown in figure 3 and figure 4 .Now when finger is rotated about its axis then area of color strips coming into camera vision increases or decreases. Now the difference in area of color strip is the measure of amount of rotation. So we use this criterion of area difference as a measure of rotation. Now when the finger is rotated at a fixed distance then there won’t be any problem in measurement of rotation, but problem occurs when finger moves forward or backward towards camera. In such cases of moving forward or backward then the color strip areas also change in camera vision. So we employ a black strip in the center as a reference strip. So now we use this reference black strip area, in case, to know if the finger is moved forward or backward. So we compare area of black strip between successive frames into consideration for further calculation of actual change in area of side color strips i.e. amount of actual rotation. So we calibrate the amount of rotation based on area change in black region from previous frame to current frame. If the difference in area of black is negligible directly the area difference is taken as measure of amount of rotation. Now to find rotation at different distances from camera the difference in area of black strip is considered and rotation is calculated using a simple formula as shown in figure.

Calibration and Initialization of Console

Before starting using the mask ,it needs to be initialised at which distance from the camera the user wants to operate the finger mask and to mark the initial position . And two parameters rotational sensitivity and lateral sensitivity are taken as input from user as per the comfort of the user.Rotational sensitivity indicates the amount of output the user wants for a particular degrees rotation of finger in rotation .In the same way the amount of output the user is expecting in lateral directions for a given input is taken from lateral sensitivity.Now depending on sensitivity parameters user can play or use the outputs as desired. Here we also calculate the area of black strip in this initial position to use it in calculaion of scaling factor.

ALGORITHM

It involves mainly four parts. These are:

A. Normalization of image and creation of new normalized image for further processing.

B. Extraction of Red,Black and Blue region areas of mask in consecutive two frames.

C. Extraction of parameters of red,black and blue regions.

D. Calculation of parameters and execution of pertaining task according to output parameters.

A. Normalization of image and creation of new normalized image for further processing.

The RGB color system is known to be sensitive to lighting conditions, since the RGB system includes the mixed information of color and intensity. So changing the given image into normalized image is a best solution to obtain the blue and red strip regions perfectly without any distortion or loss of information.

Step1: Getting red, green, blue components of image into a separate images or separate matrices.

Step2: Normalization of the red(r), blue (b), green (g) matrices by using a specific formula as shown below.

$$R = r/r + g + b$$
$$G = g/r + g + b$$
$$B = b/r + g + b$$

Step3: Concatenation of normalized component image matrices R, G and B to create a new RGBimage for further processing.

Step4: Now again step 1 is followed to extract the red, green and blue component images of the newly created frame.

B. Extraction of red, blue and black regions

Step5: Conversion of RGB image to gray scale image

Step6: Now subtracting the grayscale image from red contented matrix obtained from normalized image to get an image in which only red colored regions get a considerable intensity.

Step7: Employing proper thresholding to get a black and white image, we can get the red content as white pixels in black and white image.

Step8: Now eliminating the white regions which are having area less than a threshold area to eliminate detection of false regions and which in turn depend on the distance of operation of user and scaling factor.

Step9: Now, to obtain the blue region, subtracting thegray scale region from blue contented matrix obtained from normalized image to get an image in which only blue colored regions get considerable intensity. Repeating step7 and step 8 we get black and white image in which only our blue strip region appears in white pixels.

Step10: Now to extract black region we convert RGB image into gray scale image and later gray scale is converted into black and white image with a very low threshold such that almost every part in image turns to white except black region. Now complementing the obtained black and white image to get black strip regions as white .Now suppressing the white regions having area very less we will be left with the desired black strip region as white in black and white image to measure area and centroid of the region.

C. Extraction of parameters of red, black and blue regions

Step11: Now after extraction of black and white images in which red, black and blue respectively are in white region in all three black and white images .Now if no region of black or blue or red are present in image then it is taken as absence of finger mask or hiding of finger mask.

Step12: Now from all the black and white images we find out region properties like centroid and area to find out respective areas of blue, black and red regions

Step 13: Employing a frame grab interval we choose two frames every time after certain grab interval. The above steps are repeated for the every second frame to extract its parameters with which we compare parameters of first frame to executetasks.

D. Calculation of parameters and execution of pertaining task according to output parameters.

Step14: Now we have the areas of all the three regions of the mask in the two chosen frames. So now we will calculate the difference in areas of all the three regions .After calculating the differences there are four cases depending on the area of the black strip area difference.

Scaling Factor =

$$\left|\text{change in area of red or blue}\right| * \frac{\text{(area of black strip in the any one of current frames under comparision}}{\text{area of black strip during initialization}}$$

1. First case is when the difference in area of black strips is high and when difference in areas of blue strip is positive then the calculation of output parameter includes scaling factor.

2. Second case is when the difference in area of black strips is high and when difference in areas of redstrip is positive then the calculation of output parameter includes scaling factor

3. Third case is when the difference in area of black strips is negligible and when difference in areas of blue strip is positive then the calculation of output parameter excludes scaling factor.

4. Fourth case is when the difference in area of black strips is negligible and when difference in areas of red strips is positive then the calculation of output parameter excludes scaling factor

For first and second case,

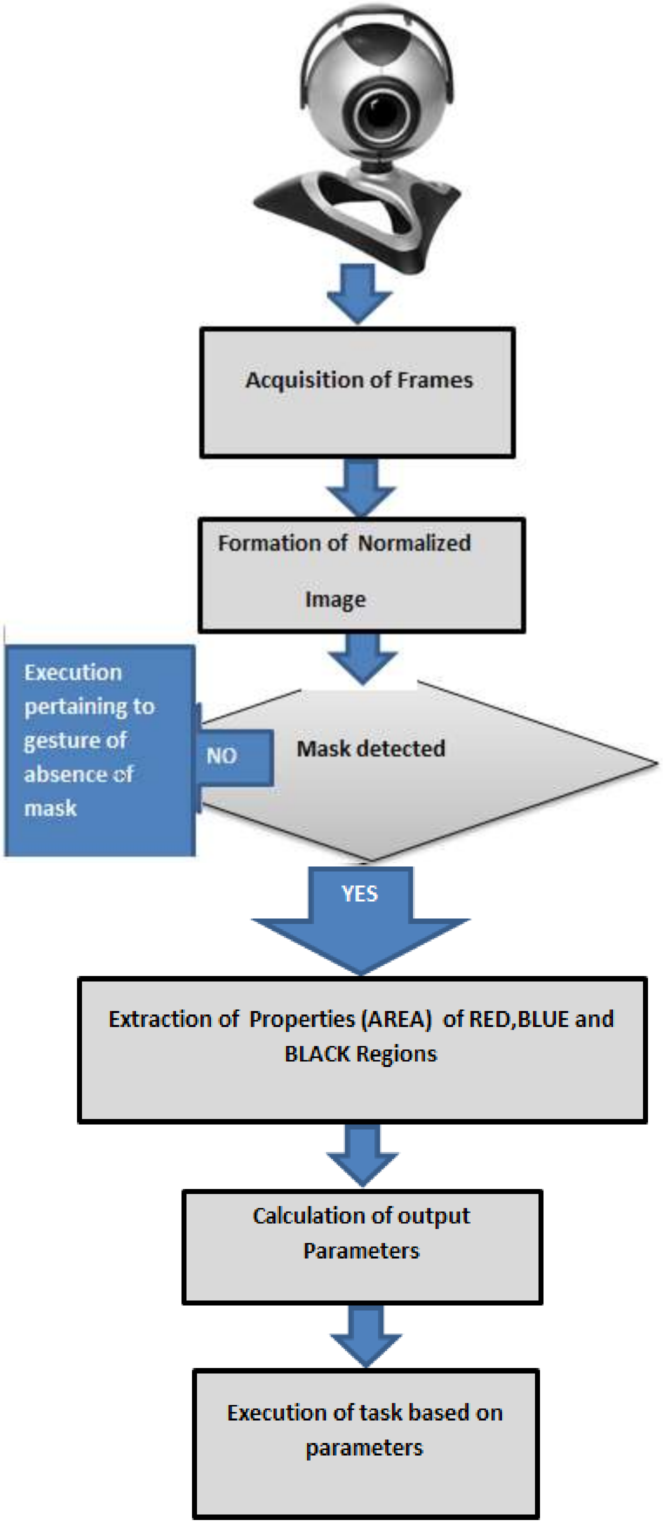
$$\text{Rotation} = \text{Change in area of red or blue} * \text{scaling Factor}$$

For third and fourth case,

$$\text{Rotation} = \text{Change in area of red or blue}$$

The final output obtained by the user is,

$$\textbf{Output Rotation Parameter} = \textbf{Rotation} * \textbf{Rotational Sensitivity}$$


Acquisition of Frames
Formation of Normalized Image
Execution pertaining to gesture of absence of mask
NO
Mask detected
YES
Extraction of Properties (AREA) of RED,BLUE and BLACK Regions
Calculation of output Parameters
Execution of task based on parameters

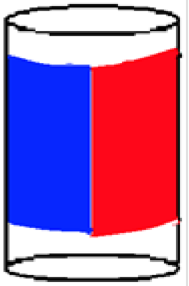
AUTO SCALABILITY

To ensure auto scalability we employ a black color strip at center of the mask. We place it at the center of mask so that entire black color strip always stays in the vision of the camera and the distance moved by finger towards camera can be known by the amount of area change that occurs in the black strip. This auto scalability feature is employed to avoid interpretation of wrong gesture or wrong calculation of parameters. Because when user moves his finger forward then there will be increase in area of red and blue strips also and in turn due to increase in the areas device may wrongly interpret that the area increase might be because of the rotation. So we use black strip area to know whether it has moved forward or the user is operating at the same initial distance. And when user moves his finger forward or backward then when he rotates his finger, then the area difference also be scaled according to the amount of increase or decrease in black strip area .Thus a scaling factor as a ratio of black strip area in the current frame to previous frame is employed to carry out exact calculations of output parameters.

Absence of gesture

It implies that when finger mask is not in view of camera, that is when it does not satisfy minimum threshold conditions of black, red and blue regions used to check the presence of the finger mask, and then it is taken as a new gesture. For example this gesture can be used as a brake while playing racing games.

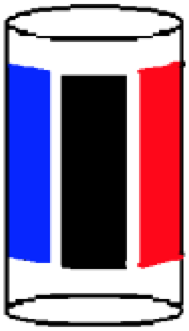
Figure 1



This application uses the rotation gesture and absence gesture within MATLAB program for interaction with game. Different gestures along with their assigned commands can be added (functions) to control the application that is further used for the game control.

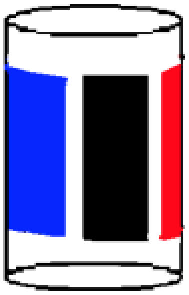
The game player can use rotation of fingermask to implement different gaming commands. Depicting the front and back side of mask in figure 1 and in figure 2 respectively

Figure 2



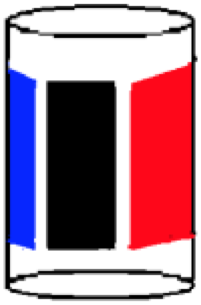
Depicting the left side moving gesture used in experimental application. Initial position and later position after rotation of mask are shown in figure 3.

Figure 3



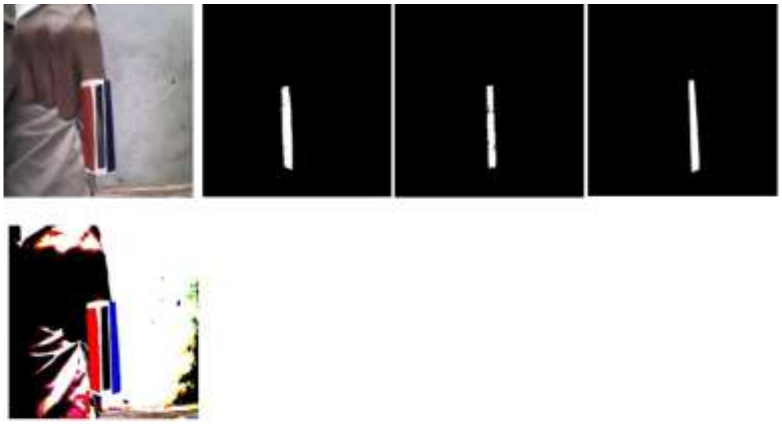
Depicting the right side moving gesture in experimental application. Initial position and later position after rotation of mask are shown in figure4.

Figure 4

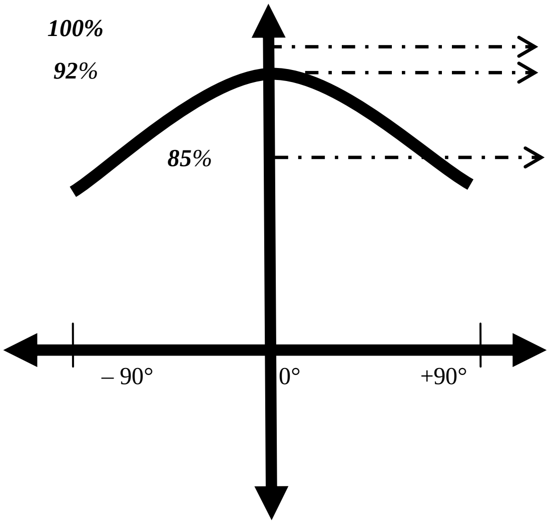


Result analysis

The image processing techniques used in application for interaction with cursor have been implemented in MATLAB. The web cam captures image sequence at the resolution of 640 X 480.
Practical experiments show that our application is implemented well in environment with little noise (i.e. existence of objects whose color is closer to red or blue) and with balanced lightening condition.



The above image depicts the normal position, in which, bottom image shows the normalized image, and the rest three binary images shows three colors on the mask.



X-AXIS–Angle of Rotation In Degrees

Y-AXIS—Accuracy of measurement at different angles of Rotation

The above image shows the rotate right position, in which, bottom image shows the normalized image, and the rest three binary images shows three colors on the mask.

The above image shows the rotate left position, in which, bottom image shows the normalized image, and the rest three binary images shows three colors on the mask.

Conclusion

The furtherance's in the field of Human-Computer Interaction is majorly been in quality of interaction. Hence instead of designing regular interfaces, research branches have had focus on the concepts of intelligent and adaptive interfaces rather than command or action based ones. Many of input providing technologies require physical touch and also there are other variations that provide input to the application without using physical touch as they may be based on other human characteristics like speech, hand gesture etc. The advantage of the usage of hand gesture based input mode is that this method provides the user ability of interacting with the application with ease and without using the traditional input devices like keyboard or mouse. The application of virtual game controlling through hand gestures proposed and implemented in the present paper provides a suitable efficient and user friendly interface between human and computer for interaction.

Acknowledgement

The corresponding author thankfully acknowledges the guidance given by Associate Professor Jyoti Singhai, ECE Deptt., MANIT, Bhopal. He also acknowledges the valuable and timely suggestions given by his co-authors and colleagues and all the concerned faculty of MANIT, Bhopal.

References

[1]. Rafael C.Gonzalez , Richard E. Woods, "Digital image processing".