ISSN Print: 2394-7500
ISSN Online: 2394-5869
Impact Factor: 5.2
IJAR 2016; 2(4): 656-661
www.allresearchjournal.com
Received: 16-02-2016
Accepted: 18-03-2016

Akshay Bhandari
Department of Information Technology, Mumbai University Ramrao Adik Institute of Technology, Nerul, India.

Anmol Chopra
Department of Information Technology, Mumbai University Ramrao Adik Institute of Technology, Nerul, India.

Sagar Rishi
Department of Information Technology, Mumbai University Ramrao Adik Institute of Technology, Nerul, India.

Gesture based control system

Akshay Bhandari, Anmol Chopra, Sagar Rishi

Abstract

Virtual environments have always been considered as a means for more efficient human computer interaction by a diversified range of applications. Evolution of ubiquitous computing, current user interaction approaches with keyboard, mouse and pen are not sufficient for the still widening spectrum of Human computer interaction. Gloves and sensor based trackers are unwieldy, constraining and uncomfortable to use. Due to the limitation of these devices the useable command set based diligences is also limited. This paper centralizes on the efforts of implementing an application that employs computer vision algorithms and gesture recognition techniques which in turn results in developing a low cost interface device for interacting with objects in virtual environment using hand gestures.

Keywords: Virtual environment, Haar Classifier, Edge Detection, Color Filtering, OpenCV, Gesture recognition

1. Introduction

Hand gesture recognition is one of the growing fields of research today which provides a natural way of human machine interaction. Gestures are some forms of actions which a person expresses in order to express information to others without saying it. In our daily life, we can observe few hand gestures frequently used for communication purpose like thumbs up, thumbs down, victory, directions etc. [1]. As computers become more pervasive in society, facilitating natural human-computer interaction (HCI) will have a positive impact on their use [2]. With the help of serious improvements in the image acquisition and processing technology, hand gestures became a significant and popular tool in human machine interaction (HCI) systems [3].

The primary goal of our Gesture Based Control System is to create a system which can identify specific human gestures and use them to convey information or for device control. A gesture may be defined as a physical movement of the hands, arms, face, and body with the intent to convey information or meaning. Gesture recognition, then, consists not only the tracking of human movement, but also the interpretation of that movement as semantically meaningful commands.

In this paper we have used web cam to take the input in form of a colorful image and then convert it to greyscale to perform Haar Classification. The Haar classifier then calculates the Region of Interest (ROI) and pass this region for Edge Detection and Color Filtering. The collective result is then fed for Feature Extraction which is in turn used for controlling the system. The advantage of using both Color Filtering and Edge Detection is to overcome the limitation of each individual technique.

2. Literature Survey

In the literature of hand gesture recognition, there are two important definitions that need to be cleared [4, 5].

- Hand posture. A hand posture is a static hand pose and hand location without any movements involved.
- Hand gesture. A hand gesture refers to a sequence of hand postures that are connected by continuous motions over a short time span with the intent to convey information or interact with computers.

Correspondence
Akshay Bhandari
Department of Information Technology, Mumbai University Ramrao Adik Institute of Technology, Nerul, India.

Pathari and Bojewar [6] proposed the common approach to use generic method for background subtraction algorithm. This method has phases like pre-processing the video input file then use background subtraction algorithm onto it and then go for further operations. In this paper in generic method we have added a new phase called as post processing which will help to remove noise from the output video before it has been sent to display output.

Bretzner, Laptev and Linderberg [7] presents algorithms and a prototype system for hand tracking and hand posture recognition. Hand postures are represented in terms of hierarchies of multi-scale color image features at different scales, with qualitative inter-relations in terms of scale, position and orientation. In each image, detection of multi-scale color features is performed. Hand states are then simultaneously detected and tracked using particle filtering, with an extension of layered sampling referred to as hierarchical layered sampling.

Chen and Georganas [8] proposes a new approach to solve the problem of real-time vision-based hand gesture recognition with the combination of statistical and syntactic analyses. The fundamental idea is to divide the recognition problem into two levels according to the hierarchical property of hand gestures. The lower level of the approach implements the posture detection with a statistical method based on Haar-like features and the AdaBoost learning algorithm.

3. Theoretical Background

A. Canny Edge Detection

This is one of the best edge detection techniques but little complex than other edge detection techniques. The major advantage of this technique is its performance. In case of other edge detection techniques only one threshold is used, in which all values below the threshold were set to 0. Thus, we must be very careful while selecting the threshold. Selecting the threshold too low may result in some false edges which are also known as false positives. Whereas if the threshold selected is too high, some valid edge points might be lost, this is also known as false negatives. But canny edge detection technique uses two thresholds: a lower threshold, TL and a higher threshold, TH thus eliminating problem of false positive and false negative. Steps involved in this type of detection are:

- The input image is smoothened with a Gaussian filter after which the Gradient magnitude and angle images are computed.
- Non-maxima suppression is applied to the gradient magnitude image.
- And finally detection and linking of the edges is done using double thresholding and connectivity analysis [1].

B. Haar Classifier

After modelling and analysis of the input hand image, gesture classification method is used to recognize the gesture. Recognition process affected with the proper selection of features parameters and suitable classification algorithm. Initially, the algorithm needs a lot of positive images (images of hand) and negative images (images without hand) to train the classifier. Then we need to extract features from it. For this, Haar features shown in below image are used. They are just like our convolutional kernel. Each feature is a single value obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle.

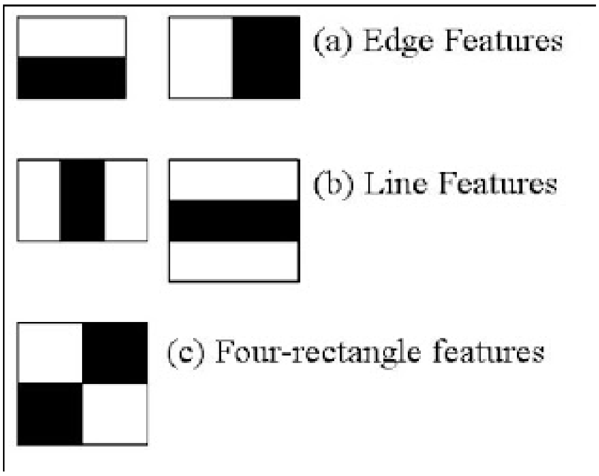


Fig 1: Haar - like Features

There are two motivations for the employment of the Haar-like features rather than raw pixel values. The first is that the Haar-like features can encode ad hoc domain knowledge, which is difficult to describe using a finite quantity of training data. Compared with raw pixels, the Haar-like features can efficiently reduce/increase the in-class/out-of-class variability, thus making the classification easier [9].

4. Working

A. Hand Recognition

Before the user starts performing gesture on the screen the hand must be recognized. For this the user must place the hand before the camera and be detected, this allows only the hand to be recognized and no other object, hence noise is eliminated from input. We use the HAAR classifier to perform the hand recognition.

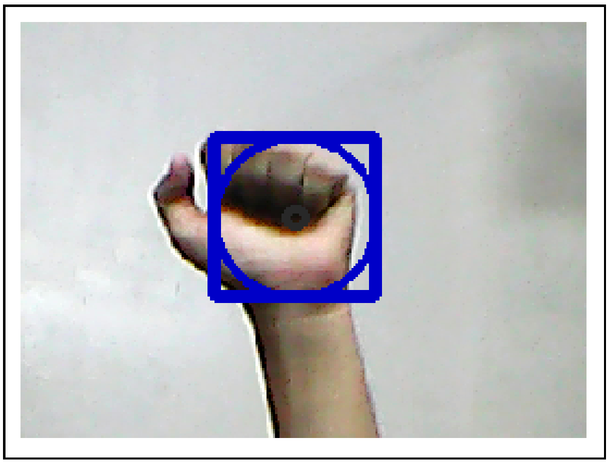


Fig 2: Hand Recognition using Haar classifier

Intel developed an open source library devoted to easing the implementation of computer vision related programs called Open Computer Vision Library (Open CV). The Open CV library is designed to be used in conjunction with applications that pertain to the field of HCI, robotics, biometrics, image processing, and other areas where visualization is important and includes an implementation of Haar classifier detection and training. Fig.2 show the recognition of hand when the hand is brought in the front of the screen using the Haar classifier.

Considering the problems faced by color/shape-based approaches, as well as the poor repeatability for hand postures due to high DOFs of the hand and the difficulty to duplicate the same working environment such as backgrounds and lighting conditions, we decided to employ a statistical approach based on a set of Haar-like features, which focus more on the information within a certain area of the image rather than each single pixel.

B. System Flow

The block diagram representing the system is given in Fig. 3. The details of the system along with explanations and figures of each step are given below. Fig 4 represents different gestures used in our system.

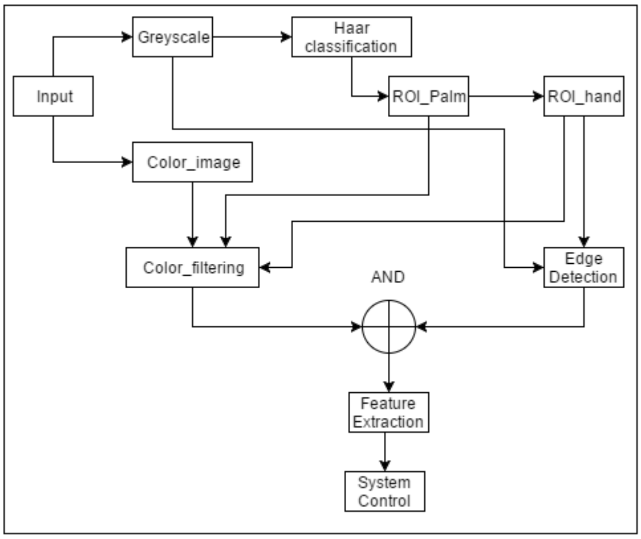


Fig 3: System Flow Diagram

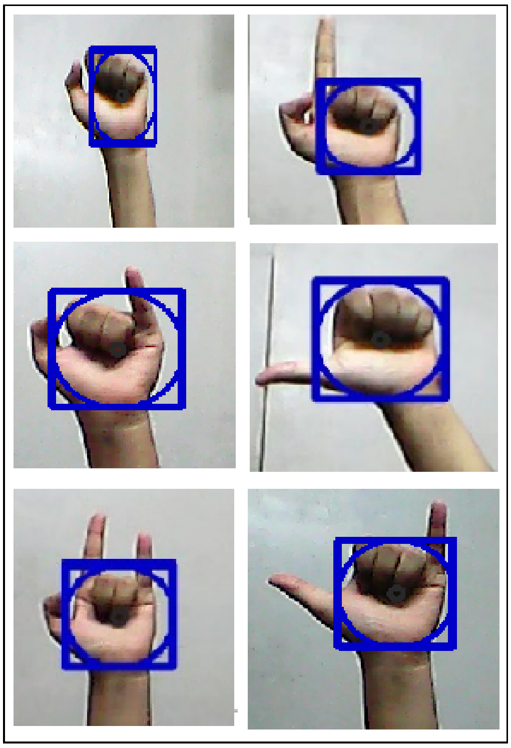


Fig 4: Implemented Gestures

C. Color Filtering

Color filtering is a process of finding regions with skin colored pixels from the image. This process has been used for detection of hand. Which is applied in different fields. The skin filtering process is shown in Fig. 5.

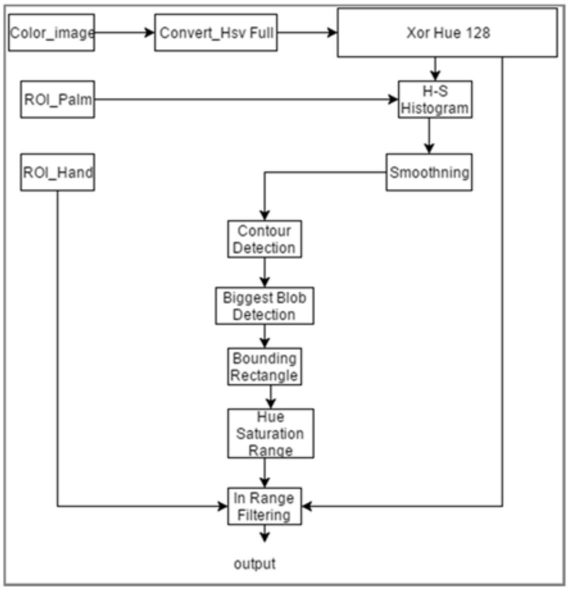


Fig 5: Skin Filtering Process

The first step involves the capturing of image using camera and conversion of the input RGB image to HSV Full colour space. This step is done because HSV model is more sensitive to changes in lighting condition. HSV which means Hue (H), Saturation (S) and the brightness (I, V or L) is shown in Fig. 6.

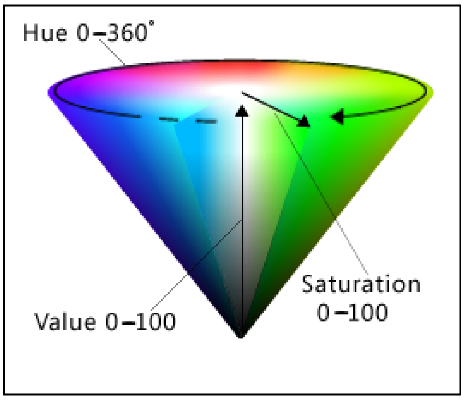


Fig 6: HSV Color Space

The HSV Full image is then passed for circular shift of hue by 128. The ROI Palm from the Haar Classifier and the output of circular shift is combined for making H-S Histogram. The resulting image was subjected to smoothening process in which erode and dilate functions are used.

The function erodes the source image using the specified structuring element that determines the shape of a pixel neighbourhood over which the minimum is taken. The function dilates the source image using the specified structuring element that determines the shape of a pixel neighbourhood over which the maximum is taken.

Erode is followed by dilate for removing noise whereas dilate is followed by erode is to fill-up holes.

From the smoothened image the contour is detected. Along with the desired hand image, other objects having skin coloured was also taken into consideration which needs to get removed. This was done by taking the biggest BLOB (Binary linked object). This biggest BLOB is then bounded by a rectangle according to its size.

We have specified the range (upper and lower limit) of Hue saturation. The In-range filter, filters the image according to the threshold value specified in hue saturation range. The output of color filtering process is a monochrome image.

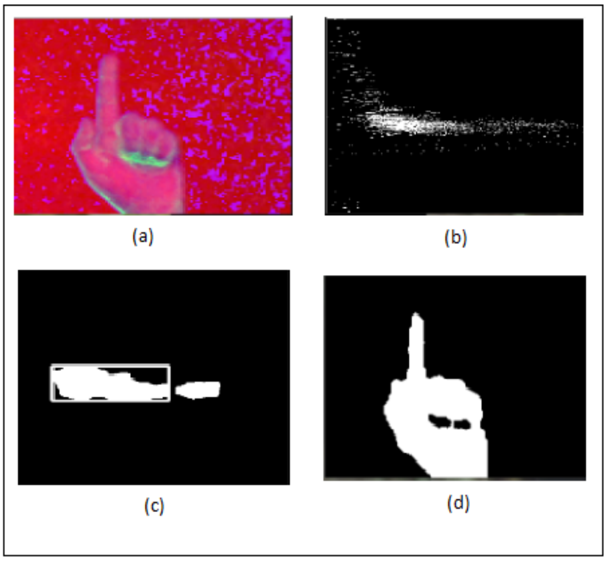


Fig 7: Skin Filtering
(a) HSV Hand, (b) H-S Histogram
(c)Threshold Range, (d) Output

D. Edge Detection

Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision.

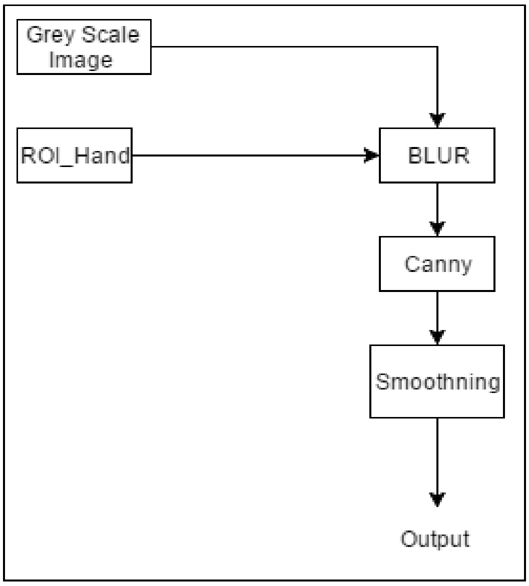


Fig 8: Edge Detection Process

The Edge Detection technique which is used takes two inputs Grey scale image and ROI Hand. These inputs are then passed to blur function which is use for removing noise. The resultant output is then passed to Canny Edge Detector which detects and links the edges using double thresholding. The output of Canny is then passed for smoothening to get the final output for edge detection process.

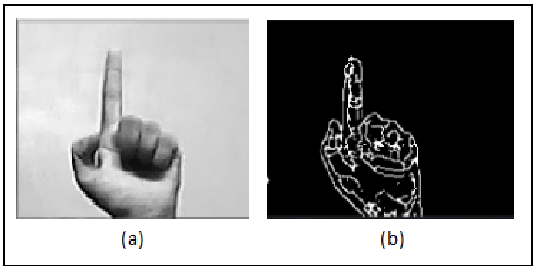


Fig 9: Edge Detection
(a) Grey scale image, (b) Output

E. Feature Extraction

In pattern recognition and in image processing, feature extraction is a special form of dimensionality reduction. The main goal of feature extraction is to obtain the most relevant information from the original data and represent that information in a lower dimensionality space.

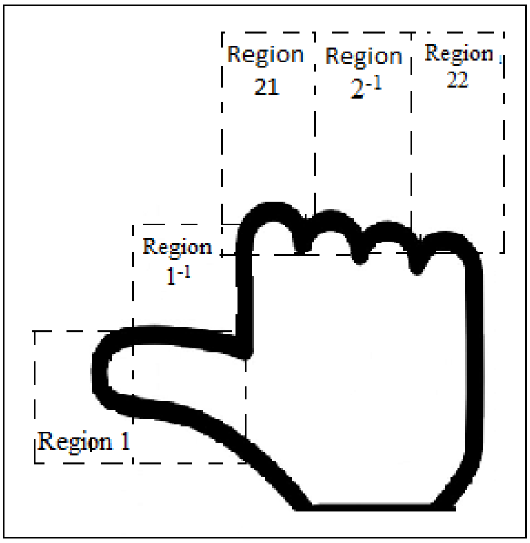


Fig 10: Feature Extraction

The above figure Fig 10 describes the regions to be used during feature extraction. Firstly we will calculate the average of values for a particular region. Then for calculating the thresholds we will use the following formulae:
Thumb =avg (r1) - avg (r^{-1})
Index finger = avg (r21) - avg (r2^{-1})
Little finger = avg (r22) - avg (r2^{-1})
After calculating these values the threshold used are:
Thumb = -5
Index Finger = 15
Little Finger = 10
We are using a buffer of size 6, which is maintained for all feature respectively. If a buffer have four or more value higher than threshold value then the feature is considered positive else negative. The reason we are using buffer is overcome fluctuation which occurs while feature extraction.
The output from Color Filtering and Edge Detection is ANDed which is given as an input to feature extraction. The reason we are using Color filtering and Edge detection rather than using multiple Haar Classifier for recognizing different gestures is because implementing Haar classifier is time consuming and costly as it requires many input datasets (positive and negative images). Whereas Edge detection and color filtering collectively provides highest accuracy and flexibility at cheapest implementation cost.

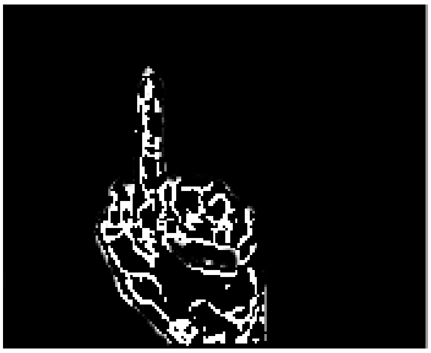


Fig 11: Input to feature extraction

F. System Control

The different gestures/features are passed to the system are as follows:

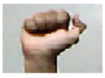
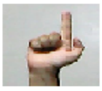
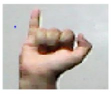
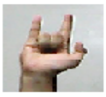
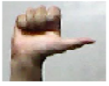
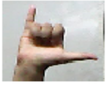
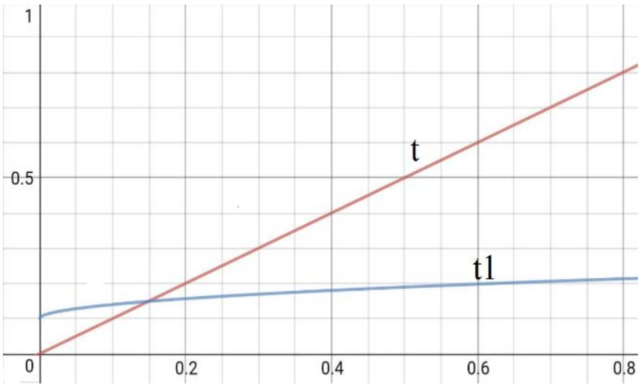
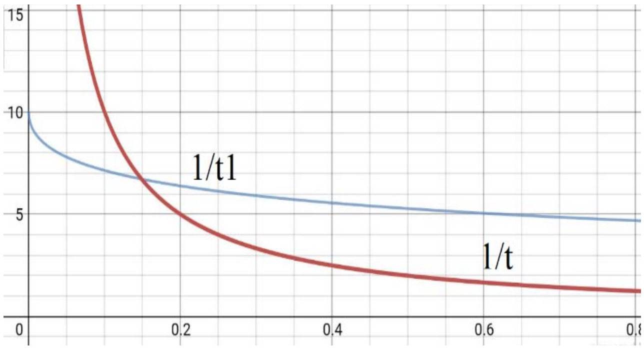
Gestures	Functions
	Mouse Pointer
	Left Click
	Right Click
	Scroll
	Middle Mouse Button
	Take Mouse Pointer to centre of screen

Fig 12: Gestures and its functions

Movement for mouse pointer is calculated by following formulae:

- Time(Difference in frame) : t (in ms)
- Difference in hand position: dx, dy
- ROI_Palm_Width: h_w
- Webcam feed dimension: W * H
- Desktop Resolution: $S_w * S_H$
- $d_1=\sqrt{((dx/W)^2 +(dy/H)^2)}$
- $d_2=H_W/W$
- $t_1=(100 + 4\sqrt{(t*1000)})/1000$





- $Movement = d/(d_2*t_1)$
- $Movement_1=(Movement/10)^2$
- $Movement\_X=(Movement_1*S_w*dx)/d_1$
- $Movement\_Y=(Movement_1*S_h*dx)/d_1$

Acknowledgment

We hereby are grateful to present our Paper on "Gesture Based Control System" through this report. We thank our Honourable Principal Dr. Ramesh Vasappanavara for extending his full-fledged support and cooperation for development in this project. We express our gratitude towards Prof. Dipti Jadhav, Head, Information Technology Department and Prof. Reshma Gulwani, Project coordinator for her valuable advice and support. We would like to express our sincere respect to the most favouring faculty member & our Project Guide Prof. Anita Patil who helped us throughout the whole Project & gave us her valuable time & advice, contributing her experience and knowledge.

5. Applications

- Smart TV
- Virtual Library
- Modern Entertainment System

6. Conclusion

In present environment a number of facilities and various modes for providing input to any application are available. It is though unfortunate that with the ever increasing smart environments and corresponding input technologies still many applications are controlled using traditional input devices like keyboard and mouse. We have overcome this by implementing hand gesture to interact a computer and/or application. The most important advantage of the usage of hand gesture based input modes is that using this method the user can interact with the application from a distance without using mouse. The application of manipulating objects through hand gestures in virtual environment is being proposed and implemented in the present paper provides a suitable efficient and user friendly human computer interaction. With the help of this application the user can interact with the computer using hand gesture instead of any other physical input devices. As the application provides the flexibility to the users and specifically physically challenged users to define the gesture according to their feasibility and ease of use.

References

1. Joyeeta Singha, Karen Das. Hand Gesture Recognition Based on Karhunen-Loeve Transform, Mobile & Embedded Technology International Conference, 2013.
2. Rautaray SS, Agarwal A. Vision Based Hand Gesture Recognition for Human Computer Interaction: A

Survey. Springer Science & Business Media Dordrecht, 2012.

3. Ramakant Yadav. Fusion of Information from Data-Gloves and a Camera for Hand Gesture Recognition, M Tech thesis, 2011.
4. Liang RH, Ouhyoung M. A real-time continuous gesture recognition system for sign language, in Proc. 3rd Int. Conf. Autom. Face Gesture Recog., 1998, 558-565.
5. Corradini. Real-time gesture recognition by means of hybrid recognizers, in Proc. Int. Gesture Workshop Gesture Sign Lang. Human– Comput. Interaction. New York: Springer-Verlag 2001; 2298:34-46. Revised paper.
6. Ms. Rucha Pathari D, Asst. Prof. Sachin Bojewar M. Background Subtraction Algorithm with Post Processing in Video Surveillance, International Journal of Engineering Research & Technology (IJERT). 2014, 3(1). IJERT ISSN: 2278-0181.
7. Bretzner L, Laptev I, Lindeberg T. Hand gesture recognition using multi-scale color features, hierarchical models and particle filtering, Proc. Int. Conf. Autom. Face Gesture Recognition, 2002.
8. Qing Chen, Nicolas Georganas D. Hand Gesture Recognition Using Haar-Like Features and a Stochastic Context-Free Grammar, Ieee Transactions On Instrumentation And Measurement, 2008, 57(8).
9. Lienhart R, Maydt J. An extended set of Haar-like features for rapid object detection, in Proc. IEEE Int. Conf. Image Process., 2002; 1:900-903.