Abstract—In this paper, the hand gesture of a person is being recognized and being identified that which hand is raised. This paper basically differentiates between the left and the second hand of the person. The human figure is first detected in the frame and then the skin color is taken into account to recognize the hands and the face of the person. The hands and the face are differentiated on the basis of the area and the centroid of the skin color map that represents basically the position and thus the hand, whether left or right hand. Camera is the only input device used in the process and then the captured digital image is being processed. The algorithm is being developed for both the captured and the real time images. The root algorithm that is being used in this paper is Mixture of Gaussian (MoG) for background subtraction that is an effective background mixture model for the background subtraction or foreground detection. Hence, a Human Machine Interface (HMI) has been developed by exploiting this method further.

Index Terms—Skin color map, Mixture of Gaussian (MoG), Background Subtraction, Human Machine Interface (HMI).

I. INTRODUCTION

A primary goal of gesture recognition research is to create a system which can identify specific human gestures and use them to convey information or for device control. To help understand what gestures are, an examination of how other researchers view gestures is useful. How do biologists and sociologists define “gesture”? How is information encoded in gestures? We also explore how humans use gestures to communicate with and command other people. Similarly on the same basis, if we want to communicate with a machine then there need to be a mechanism which could make exchange of information possible between human and the machine.

As humans have five receptors in their body to communicate with the outer world, and send information to the brain to process it. A machine also needs some reception mechanism to make this type of communication possible. Regardless of having so many sensors for different purpose, we need a replicate of a human eye that is the most powerful and the essential receptor of the human body and in the artificial world the same role is played by the Camera.

Recognizing gestures as input allows computers to be more accessible for the physically-impaired and makes interaction more natural in a gaming or 3-D virtual world environment. Hand and body gestures can be amplified by a controller that contains accelerometers and gyroscopes to sense tilting, rotation and acceleration of movement -- or the computing device can be outfitted with a camera so that software in the device can recognize and interpret specific gestures.

Gesture recognition can be seen as a way for computers to begin to understand human body language, this building a richer bridge between machines and humans than primitive text user interfaces or even GUIs (graphical user interfaces), which still limit the majority of input to keyboard and mouse.

In the paper, the human hand gesture is being recognized and identified whether the left or right hand or both hands are raised. This can be used for many smart applications and in distance communication in which hand raising can be made a special sign and on the basis of which hand has been raised the particular action can take place.

During the implementation of the algorithm, it has been realized that the hand gesture recognition algorithm may fail if the camera captures the image when the hand is the moving position. So in order to reduce the error as much as possible we used the MoG background subtraction method [1] and the foreground detection techniques for the object detection.

The most popular literature technique approach to detect moving object from video sequences is background subtraction. This approach utilized mathematical model of static background and comparing it with every new frame of video sequence. In this paper, background subtraction technique using Mixture of Gaussian (MoG) method is conducted for detection of moving object at outdoor environment. Focus is specified at the five parameters of MoG namely background component weight threshold (TS), standard deviation scaling factor (D), user-define learning rate (α), Total number of Gaussian components (K) and Maximum number of components M in the background model (M) to give significant impact in producing the optimize background subtraction process.
Experimental results showed that by varying each of the parameter can produce acceptable results that enable us to propose suitable parameter range of each parameter for detection of moving object in an outdoor environment.

II. LITERATURE SURVEY

A. Some previous work related to hand gesture recognition:

Gesture recognition was first proposed by Myron W. Krueger as a new form of interaction between human and computer in the middle of seventies [2]. It has become a very important research area with the rapid development of computer hardware and vision systems in recent years. The important role of hand gesture for different applications has increasingly being acknowledged, although some approaches still focus only on static hand poses rather than dynamic use of more general types of gesture in context. Currently, there are several available techniques that are applicable for hand gesture recognition, which are either based on auxiliary devices or computer vision. A typical widespread device based example is VPL data glove, which is developed by Zimmerman in 1987 [3]. In this system, user wears a VPL data glove that is linked to the computer. The glove can measure the bending of fingers, the position and orientation of the hand in 3-D space. Data glove is able to capture the richness of a hand’s gesture.

B. Background Subtraction or Foreground Detection

Background subtraction is the method used in computational to separate foreground objects from the background in the sequence of video frames using the fundamental logic of Frame Differencing. Non-recursive techniques such as Frame differencing, Median Filter, Linear predictive filter and Non-parametric model proposed by researchers as general is a technique that uses a sliding window approach for background estimation [6]. This technique maintain a buffer of previous video frames and estimate a background model based on the statistical properties of these frames that causes this technique consume high memory [7].

Different approach utilized in recursive techniques which is approximated median filter, Kalman filter. Mixture of Gaussian (MoG) do not maintain a buffer for background estimation but updated a single background based on input frame [1]. Recursive can be described as a process of repeating objects in a self-similar way. In addition of that, it maintains a single background model that is updated with each new video frame that makes this technique have minimal memory requirements as compare to non-recursive and computational efficient [6].

The method will detect moving object and classify the process of pixels as foreground and background. It detects the foreground and compares the input video frame with the background model, and identifies it pixels from the input frame. However, the detection rate of the foreground depends on the method chosen. Normal approach is to verify the input pixel and compare it with corresponding background estimation. , the moving object can be detected for K(total number of Gaussian component)= M(maximum number of components M in the background model).

The images required for this component are the current background model B(x, y) and the current incoming image frame from the camera I(x, y) t after pre-processing. The most commonly used approach for foreground detection is to check whether the input pixel is significant different from the corresponding background estimate. The creation of a binary mask M(x, y) t for a greyscale video frame with height h and width w can be described by:

\[
\text{for } x = 1 \text{ to } w \\
\text{for } y = 1 \text{ to } h \\
\text{if } |I(x,y) - B(x,y)| > T \text{ then } M(x, y) t = 1; \\
\text{else } M(x, y) t = 0 \\
\text{endfor;}
\]

where T is a threshold value.

C. Background Subtraction using MOG

Mixture of Gaussians (MoG) modelling [9] is a popular approach to background subtraction in video sequences. This algorithm has two components, one the online MoG algorithm for pixel density estimation and another is the binary classification to decide whether each pixel is background or foreground. In MoG, the background is known as parametric frame of values where each pixel location is represented with number of Gaussian functions as probability distribution function. It maintains a parametric density function P, for each pixel at time t. The value of a pixel at time t is denoted by xti. Here we assume x is gray-scale intensity and can be directly extended to multidimensional color or other feature spaces if we assume a diagonal covariance matrix. The pixel distribution P(x) is modelled as a mixture of K Gaussians:

\[
P(x|\theta) = \sum_{i=1}^{k} \omega_i G(x|\mu_i, \sigma_i)
\]
Where, \( \theta_t = \{ \omega_{i,t}, \mu_{i,t}, \sigma_{i,t} \}_{i=1} \) is the \( i \)-th Gaussian component with mean \( \mu_{i,t} \) and standard deviation \( \sigma_{i,t} \), \( \omega_{i,t} \) is the weight of the \( i \)-th Gaussian component. For each new pixel \( x_t \), a match is found if \( |x_t - \mu_{i,t}| < f \sigma_{i,t} \) for any \( i=1,2, \ldots, K \).

\[
\sigma_{i,t}^2 = (1 - \rho)\sigma_{i,t-1}^2 + \rho(x_t - \mu_{i,t})^2 \tag{2}
\]

Where, \( \alpha \) is the learning rate for the weight and \( \rho \) is the learning rate for the distribution.

After \( P(x|\theta_t) \) is obtained, the Gaussians are ranked according to their associated term \( \omega_i \). The background is then modelled by the first \( B \) largest Gaussians chosen as follows:

\[
B = \text{arg min} (\Sigma \omega_i > T) \tag{3}
\]

Where, \( T \) denotes the portion of the data we assume belongs to the background.

So the background model \( P_t(x|BG) \) is given by Equation (4) as follows:

\[
P_t(x|BG) = \Sigma \omega_{i,t}G(x|\mu_{i,t}, \sigma_{i,t}) \tag{4}
\]

Thus for each new video frame, the MoG algorithm will be as follows:

1. Perform binary classification based on background model \( P_t(x|BG) \) given in by equation 4.
2. Update \( P_t(x|\theta_t) \) using equations 1.
3. Update \( P_t(x|BG) \) using equation 3.

Based on the overall experimental results, MoG method is used in this paper for moving object detection in an outdoor environment using acceptable parameters range as tabulated in Table 1.

### III. METHODOLOGY

The algorithm that is being reproduced in this paper is basically based on the MoG background subtraction and thus facilitating the gesture recognition by using the concept of centroid and area. Therefore the algorithm is named as “MoGCA Algorithm”.

The following flow diagram in fig 1 of the whole algorithm will explain the whole methodology that is being included for the hand detection and thus recognizing which hand is being lifted, either left or right hand.

Because understanding the gesture more semantically we need to first identify the hand i.e. left or right, then only we can move further to recognize the complete gesture.

For just testing purpose, some static images where being tested, by simply making them read.
To make sure that the system runs in real-time, the frame rate can be reduced in this stage by not processing each frame. Also a reduction of camera resolution is possible by rescaling each incoming image frames from the stream. Nowadays cameras are equipped with a CCD-sensor linearly transforms infinite-dimensional spectral colours space to a three-dimensional RGB colour space via red, green and blue filters.
Now as a result of this linearity and sensor type, the following characteristics of the output image hold.

I. Variation in colour: It rarely occurs that we observe the same RGB colour value for a given pixel over a period of time.

II. Clipping: The CCD sensors have limited dynamic range of responsiveness. All visible colours lie inside the RGB cube spanned by the R, G and B vectors with a given range. As a result some pixel values (outside the range) are clipped in order to lie entirely inside the cube. To reduce this unusual shape of the colour distributions $s_i$ is often set to a minimal value if a pixel’s $s_i$ is zero.

Band balancing: cameras typically have different sensitivities to different colours. A possible approach taken is to balance the weights on the three colour bands by performing a normalization [9]. The pixel colour is normalized by its standard deviation $s_i$ which is given by:

$$S_i = [\sigma_R(i), \sigma_G(i), \sigma_B(i)]$$ (5)

Where, $\sigma_R(i)$, $\sigma_G(i)$, $\sigma_B(i)$ are the standard deviation of the $i^{th}$ pixel’s red, green and blue values.

In the pre-processing step it can be advantageous to transform from the original RGB colour space to a more intuitive colour model that is the HIS colour system, which defines a value of the hue, saturation and intensity for each pixel. The values H, S and I are obtained from the R, G and B values in the following way.

$$H = \sqrt{3} \times \frac{(G - B)}{(R - G) + (R - B)}$$

$$S = 1 - 3 \times \min(r, g, b); \quad I = \frac{(R + B + G)}{3}$$

This model is not sensitive to surface orientation, illumination direction and illumination intensity.

While taking the images there is always a probability that the background may cause some sort of abnormality in the system, so the background must need to be subtracted which is done using the MoG method (see D) is chosen due to its low rate of complexity, memory consumption and suitability for outdoor environment along with its robustness and also it can handle multi-modal distributions. In MoG, the background is known as parametric frame of values where each pixel location is represented with number of Gaussian functions as probability distribution function. Moreover it can be made adaptive and thus more robust. In the background is known as parametric frame of values where each pixel location is represented with number of Gaussian functions as probability distribution function in the equation 6. And the table Table 1 is the experimented range of values used for this type of system.

$$F(i,t) = \sum_k \omega(i,t) \eta(\mu, \sigma)$$ (6)

Where,

$Ts =$ Background component weight threshold
$D =$ Standard deviation scaling factor
$\rho =$ Learning rate
$K =$ Total number of Gaussian components
$M =$ Maximum number of components $M$ in the background model.

### Table 1

**Experimented MoG Range**

<table>
<thead>
<tr>
<th>Test Parameters</th>
<th>Parameter Range</th>
<th>Proposed Parameter Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of Gaussian components (K)</td>
<td>(3 – 5)</td>
<td>(3 – 7)</td>
</tr>
<tr>
<td>Maximum number of components M in the background model (M)</td>
<td>3 (no range)</td>
<td>(3 – 7)</td>
</tr>
<tr>
<td>User-Define learning rate ($\alpha$)</td>
<td>(0 – 1)</td>
<td>(0.005 – 0.05)</td>
</tr>
<tr>
<td>Standard deviation scaling factor (D)</td>
<td>2.5 (no range)</td>
<td>(2-3)</td>
</tr>
<tr>
<td>Background component weight thresh (Ts)</td>
<td>0.25 or 0.75</td>
<td>(0.2 – 0.55)</td>
</tr>
</tbody>
</table>

### IV. MATLAB SIMULATION

During the simulation of this algorithm in MATLAB [10] we have to write three different codes in the MATLAB [10] script editor. One, Mixture of Gaussian Subtraction. Second, Skin detection.

Third, processing and decision making. The code for the above three script will incorporate the following algorithms respectively.

A. Mixture of Gaussian background subtraction

1. Set the source of video and frame size variables.
2. Set mog variables. No. of Gaussian components (typically 3-5), number of background components, positive deviation threshold, learning rate (between 0 and 1, 0.001 is chosen here), threshold, standard deviation.
3. Initialize weights array.
4. Calculate pixel mean, pixel standard deviation, difference of each pixel from mean.
5. Declare a parameter (z) to update mean and standard deviation.
6. Calculate the rank (weight/standard deviation) of the components.
7. Initialize component means and weights.
8. Process each frame of the source by calculating difference of pixel values from mean, updating Gaussian components for each pixel, update weight, mean, standard deviation and z.
9. Calculate component rank.
10. Sort the rank.
11. Calculate foreground and return its value.

B. Skin detection
1. Read the image and the capture the dimensions.
2. Initialize the output images.
3. Find the inverse of the average values of the RGB Component of the image.
4. Calculate the smallest average value and the scaling factor.
5. Scale the values.
6. Convert the image from RGB to YCbCr and find the Cb and Cr component.
7. Mark the skin pixels whose (b>=77 & b<=127 & c>=133 & r<=173).
8. Return the skin marked image.

C. Processing and decision making
1. Remove noise from the obtained image.
2. Find the smallest area from all the marked areas.
3. Compare the smallest area with the other areas.
4. Set the value of smallest area equal to some particular value (p).
5. If the area > p, then apply the condition on the both the hands, else only on the one hand.
6. If the centroid of smallest area lies in the middle of the centroid of other two areas then both hands are present.
7. Else if the centroid of smallest area lies to the left of the centroid of biggest area then only right hand is raised otherwise left hand is raised.
8. Return the result.

V. CONCLUSION
The gesture of the person is identified using the area and centroid concept. It requires less computation time. The concept used is easy to understand and the implementation is also not complex. It gives more accurate results. It requires less mathematical calculations. Unlike Kalman filter which tracks the evolution of a single Gaussian, the MoG method tracks multiple Gaussian distribution simultaneously and since MoG is parametric, the model parameters can be adaptively updates without keeping a larger buffer of video frames.

VI. FUTURE SCOPE
Gesture Recognition has been used in so many applications; it is very interesting and fascinating to be a part of future where without keyboards, mouse and other hard input devices, humans can interact with machines as they interact with each other’s. A huge knowledge base could make this system an effective one. Some learning algorithms using neural networks and good decision using fuzzy logic could make this system an expert vision system. For further development, work could be done to remove more noise. Since the simulation of this algorithm is done at a particular place with adequate light conditions and with the people not having the dark complexion. And only the hand is taken into account i.e. the algorithm works when the person is wearing the full sleeves. These restrictions can be removed if can design some another algorithm to differentiate between face, hands and arm. This can be done if we go for the shape detection approach or using probabilistic fusion of multiple visual cues [8].

REFERENCES
International Conference on Advanced Developments in Engineering and Technology (ICADET-14), INDIA.


AUTHOR’S PROFILE

Chirag Ahuja is currently pursuing dual degree with B.Tech (ECE) and M.Tech (Intelligesnt Systems and Robotics) since 2010-15 from School of ICT, Gautam Buddha University Gr. Noida. His research interest includes digital signal processing, soft computing, embedded systems, software development etc.

Chirag is a member of Institute of Electrical and Electronics Engineering (IEEE) USA and also currently holds the position of executive chair of IEEE-GBU student chapter.

Chirag Ahuja, Electronics and Communication Engineering, School of ICT Gautam Buddha University, Greater Noida, UP, India, +91-9971294921, (e-mail: cahuja1992@gmail.com).