International Journal of Artificial Intelligence and Machine Learning
Volume 11 • Issue 1 • January-June 2021

Shape-Based Features for Optimized Hand Gesture Recognition

Priyanka R., Anna University, India
Prahanya Sriram, Anna University, India
Jayasree L. N., Anna University, India
Angelin Gladston, Anna University, India

ABSTRACT

Gesture recognition is the most intuitive form of human-computer interface. Hand gestures provide a natural way for humans to interact with computers to perform a variety of different applications. However, factors such as complexity of hand gesture structures, differences in hand size, hand posture, and environmental illumination can influence the performance of hand gesture recognition algorithms. Considering the above factors, this paper aims to present a real time system for hand gesture recognition on the basis of detection of some meaningful shape-based features like orientation, center of mass, status of fingers, thumb in terms of raised or folded fingers of hand and their respective location in image. The internet is growing at a very fast pace. The use of web browser is also growing. Everyone has at least two or three most frequently visited website. Thus, in this paper, effectiveness of the gesture recognition and its ability to control the browser via the recognized hand gestures are experimented and the results are analyzed.

KEYWORDS

Convex Defect, Hand Gestures, Human Computer Interface, Shape Features, Website Launch

1. INTRODUCTION

In the world of almost 7 billion people more than 500 million suffer from some physical, sensory or mental disability states United Nations organization as on December 3, 2019. Their lives are often hampered by some disabilities which turns down them full participation in society and even their enjoyment of equal rights and opportunities. Sign language is used commonly for the deaf and the dumb. Sign language is an efficient alternative to talking, where the former is replaced by hand gestures. Hand gestures are combination of hand shapes, orientations and movement of the hands, alignments of the fingers and positioning of the palm which are used to express gracefully a conveyer's thoughts. Signs are used to communicate words and sentences to audience. Our objective is to optimize an algorithm for recognition of hand gestures with reasonable accuracy, where the input to the pattern recognition system will be given from the hand.

DOI: 10.4018/IJAIML.2021010103

Gesture recognition aims to recognize meaningful movements of human bodies, and is of utmost importance in intelligent human computer or robot interactions. Hand gesture recognition system has got good attention now-a-days because of easy interaction between human and machine. The focus of developing hand gesture is to enhance the communication between the humans and the computer for conveying various information, which will help accomplishing multiple task. It is an efficient way of interacting with machines making it more popular and applicable for many purposes. The hands, arms, body and face are used for gesture recognition to perceive critical demeanours of movement by a human. Gesture recognition is mainly applicable for video conferencing, sign language recognition, distance learning as well as in forensic identification. Hand gesture recognition research has been gaining more and more popular among worldwide researchers. Hand gesture recognition has become an important research topic in human-computer interaction (Hu et. al., 2019; Li et. al., 2018; Patil et. al., 2018; Soe et. al., 2018; & Chaikhumpha et. al., 2018).

Hand gesture recognition system provides us an innovative, natural, and user friendly way of interaction with the computer which is more familiar to the human beings. Generally, the process of vision based hand gesture can be divided into four stages: sample image capturing, image pre-processing, feature analysis and parameter extraction, classification and recognition. Wherein, feature extraction aims to find out a feature or a set of features which can describe the specified hand gesture uniquely and help in better classification. The most commonly used features of static hand gesture include: gradient histogram, image subspace projection, and shape features. The traditional gradient histogram is easy to calculate and implement, it has the invariance of translation but not rotation. Image subspace projection is able to remove the correlation of higher-order statistics and make relatively comprehensive representation of the local features of training sample images, but it very much relies on the position scale and rotation. Shape based features such as contour, silhouette, fingertips are free of translation, size and rotation of hand gesture, and feature extraction algorithm based on shape is most commonly used currently.

The main contributions in this paper are it introduces a series of fingertip related features based on convex defect detection; presents a real time system for hand gesture recognition on the basis of detection of some meaningful shape-based features; combines browser control with recognized hand gestures and evaluates the effectiveness of the system. This hand gesture recognition system is an efficient system which will be able to control the browser via the recognized hand gestures i.e. different hand gestures will be used to open different sites. The concept behind the detection of hand gestures based on shape features and the controlling capability to open corresponding web sites are described in system design section.

The remainder of this paper are organized as follows: Related work section summarized the works that has been carried out related to gesture recognition. Experimental design section explains the proposed hand gesture recognition system, experimental results section describes the implementation details of the proposed method and the results obtained. Result analysis section presents the analysis of experimental results along with the conclusions drawn and the next section concludes the work.

2. RELATED WORK

This section presents survey of related works that have been carried out. The literature survey conducted provides an insight into the different methods that has been adopted and implemented to achieve hand gesture recognition. It also helps in understanding the advantages and disadvantages associated with the various techniques.

In the past few decades gesture recognition has become a very influencing term. There were many gesture recognition techniques developed for tracking and recognizing various hand gestures. The older one is wired technology, in which users need to tie up themselves with the help of wire in order to connect or interface with the computer system. In wired technology user cannot freely move in the room as they connected with the computer system via wire and limited with the length of wire.

Instrumented gloves also called electronics gloves or data gloves is the example of wired technology. The approach of gesture recognition (Panwar, 2012 & Ji-Hwan et. al., 2009) uses input extraction through data gloves. These instrumented gloves made up of some sensors, provide the information related to hand location, finger position orientation etc. through the use of sensors. These data gloves provide good results but they are extremely expensive to utilize in wide range of common application. Data gloves are then replaced by optical markers. These optical markers project infra-red light and reflect this light on screen to provide the information about the location of hand or tips of fingers wherever the markers are wear on hand, the corresponding portion will display on the screen. These systems also provide the good result but require very complex configuration.

The other commonly used methods of capturing input from the user hand belts and cameras. A hand belt with gyroscope, accelerometer and a Bluetooth was deployed to read hand movements are used (Hung et. al., 2016; Hung et. al., 2015). The authors (She et. al., 2014) used a creative Senz3D Camera to capture both colour and depth information and (Lee et. al., 2010) used a Bumblebee2 stereo camera. A monocular camera was used by (Dulayatrakul et. al., 2015). Cost efficient models like (Tsai et. al., 2015; Hussain et. al., 2014 & Huong et. al., 2015) have implemented their systems using simple web cameras. The methods (Chen, Luo, Chen, Liang, & Wu 2015; Wang et. al., 2015) make use of a Kinect depth RGB camera which was used to capture colour stream. As depth cameras provide additional depth information for each pixel at frame rate along with the traditional images (Chen, Wu & Lin 2015; Wong et. al., 2015).

Soe et.al., (2018) used faster region-based convolutional neural network for real time hand pose recognition. In this system used 10 types of posturesto control VLC Media Player. The average accuracy is 89.95% on NUS Hand Pose Dataset and 86.12% on images from webcam. Nevertheless, the only drawback cannot detect very tiny hand objects. They presented Hough transform and neural network for spatio-temporal approach. In the spatiotemporal feature extraction contains spatial features of the static hand gesture based on the skin color, geometrical features and Fourier descriptor of temporal features for the dynamic gesture by using Hough transform. The accuracy of this research is 94% on Cambridge database and 98% on Sebastien database.

Chaikhumpha et.al, (2018) studied condensation and hidden markov models for real time two hand gesture recognition. In their paper, the processes are classified into two categories: hand tracking and gesture recognition to recognize hands. In the first process, condensation density propagation is used to localize and track hands which are centered at the center of palms when they are moving. This system recognized 8 gestures used and the accuracy achieved 96.25%.

A large number of methods have been utilized for pre-processing the image which includes algorithms and techniques for noise removal, edge detection, smoothening followed by different segmentation techniques for boundary extraction i.e. separating the foreground from the background. The authors (Hussain et. al., 2014; Suriya et. al., 2014) used a morphology algorithm that performs image erosion and image dilation to eliminate noise. Gaussian filter was used to smoothen the contours after binarisation (Huong et. al., 2015; Chanda et. al., 2015). To perform segmentation, in (Lee et. al., 2010) a depth map was calculated by matching the left and right images with the SAD (Sum of Absolute Differences) algorithm. In (Lee et. al., 2010), the Theo Pavildis Algorithm which visits only the boundary pixels was used to find the contours. This method brings down the computational costs. In (Hussain et. al., 2014; Chen, Wu, & Lin 2015; Suriya et. al., 2014) the biggest contour was chosen as the contour of the hand palm after which the contour was simplified using polygonal approximation. Classification is a process in which individual items are grouped based on the similarity between the items. The approach (Luzhnica et. al., 2016) uses Euclidean distance based classifier to recognise 25 hand postures. Support Vector Machine (SVM) classifier was used in (Chen, Ding, Chen & Wu 2015; & Chen, Luo, Chen, Liang, & Wu 2015).

The works discussed above use hand markers such as gloves. But in this paper, we propose a hand gesture recognition system which deviates from other traditional methods without using any hand markers such as gloves for gesture recognition. Fingertip detection based on convex defect is adopted. In this hand gesture recognition system developed, the webcam available in the laptop is used for capturing hand gestures, in place of any additional cameras thereby making the system cost effective. Further, this hand gesture recognition system is effectively integrated with web services so as to enable the user get his web service activated on appropriate hand gesture recognition, which definitely finds an application in our day-to-day life.

3. EXPERIMENTAL DESIGN

This section presents the detailed design of the proposed optimized hand gesture recognition system. The proposed optimized hand gesture recognition system consists of capturing input through camera, extracting features, recognition of hand gestures using the features extracted and then interfacing the recognized hand gestures with web services concerned. The detailed design of the proposed optimized hand gesture recognition system and the components involved are elaborated in the following sub sections.

The initial module is responsible for connecting and capturing input through the different types of image detectors and sends this image to the detection module for processing in the form of frames. The commonly used methods of capturing input are data gloves, hand belts and cameras. In our system, we use the inbuilt webcam which is cost efficient to recognize both static and dynamic gestures. The system has suitable provision to allow input from a USB based webcam as well but this would require some expenditure from the user. The image frames obtained are in the form of a video. This module is responsible for the image processing. The output from camera module is subjected to different image processing techniques such as colour conversion, noise removal, thresholding following which the image undergoes contour extraction. If the image contains defects, then convexity defects are found according to which the gesture is detected.

Next module is responsible for mapping the detected hand gestures to their associated actions. These actions are then passed to the appropriate application. The user interface created consists of three windows. The first window consists of the video input that is captured from the camera with the corresponding name of the gesture detected. The second window displays the contours found within the input images. The third window displays the smooth thresholded version of the image. The advantage of adding the threshold and contour window as part of the graphical user interface is to make the user aware of the background inconsistencies that would affect the input to the system and thus they can adjust their laptop or desktop web camera in order to avoid them. This would result in better performance.

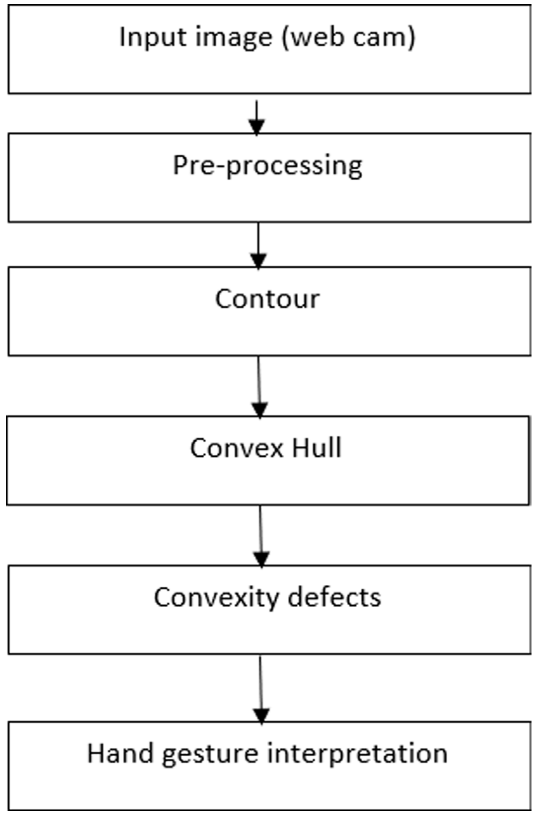
The overall flowchart of the hand gesture recognition system is shown in Figure 1 and the main steps involved in the optimized hand gesture recognition system are discussed in detail in the following sub sections.

In issues involving human hands such as for sign language recognition and gesture recognition, fingertip is one of the most popular characteristics because the number of fingertips can be considered to be the number of fingers and the direction of fingertips can effectively express the stretch information of fingers. Contour analysis is a commonly used method for fingertip detection, which achieves the location of fingertip based on geometric features of contour, such as the edge curvature method used in literature for contour detection, and the least square ellipse fitting method used in for fingertip detection, this kind of algorithms require high accuracy of contour and a large amount of computation, and are very dependent on the quality of gesture segmentation. In this paper, a method of fingertip detection based on convex defect is adopted.

3.1. Noise Removal and Image Smoothening

The input image, which is in RGB color space, is cropped to a size of 100 * 400 pixels. It is then converted into a gray scale image. This process is shown in Figure 2.

Figure 1. Hand gesture recognition system



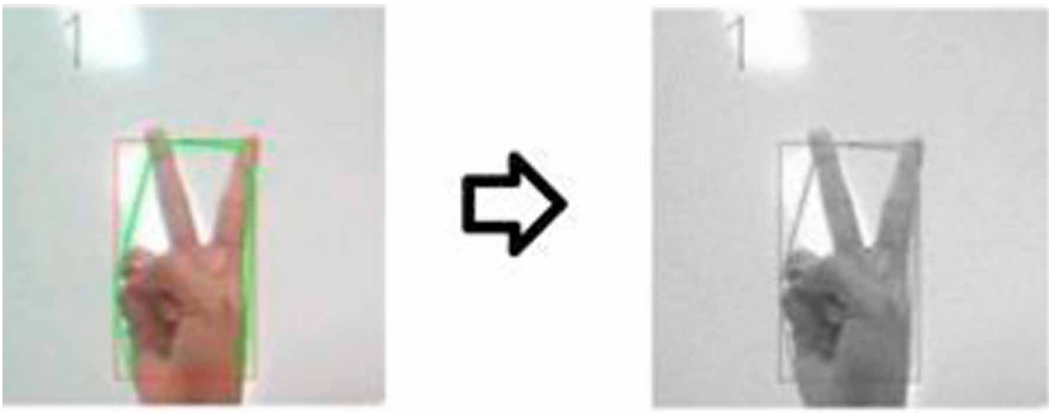
Process of cropping and converting RGB input image to gray scale is carried out. Noise in images can be defined as a random variation of brightness or colour information that is usually produced during the image acquisition process from the webcam. This noise is an undesirable aspect of the image and needs to be removed. In order to do this, Gaussian filter is applied. Gaussian filtering is performed by the convolution of Gaussian kernel with each point in the input array. These are then added to produce the output array. A 2D Gaussian kernel can be represented mathematically as shown in Equation 1.

$$G_0\left(x, y\right) = A_e \frac{-\left(x-\mu_x\right)^2}{2\sigma x^2} + \frac{-\left(y-\mu_x\right)^2}{2\sigma y^2} \tag{1}$$

3.2. Thresholding

Thresholding, which is a simple segmentation method, is then carried out. In the work by Granit et. al., (2016), K-means clustering has been used to segment the image from background. The K-means clustering algorithm is an iterative technique which is used to segment the image in to K clusters. K-mean computes centroid of each cluster in order to minimize the sum of distances from each object to its cluster. K-means iteratively minimizes the sum of distances from each object to its cluster centroid

Figure 2. Process of cropping and converting RGB input image to gray scale



until the sum cannot be decreased further. The result of K-means clustering is a set of clusters that are well separated from other clusters and compacted in their own cluster.

In our proposal, Otsu's binarization method is used for thresholding which gives good accuracy rate on par with that of the k-means clustering based method. Thresholding is applied to obtain a binary image from the gray scale image. Thresholding technique compares each pixel intensity value (I) with respect to the threshold value (T). If I<T, the particular pixel is replaced with a black pixel and if I>T, it is replaced with a white pixel. A threshold value (T) of 127 is used in our work which classifies the pixel intensities in the gray scale image. Maximum value of 255 is the pixel value used if any given pixel in the image passes the threshold value. The two types of thresholding that are implemented are Inverted Binary Thresholding and Otsu's Thresholding. Inverted Binary Thresholding inverts the colors, to be white image in a black background. This thresholding operation can be expressed as shown in Equation 2.

$$Dest\left(x,y\right)=\begin{cases}0 & if\ src\left(x,y\right)>T\\ maxVal\left(255\right)othewise\end{cases}. \tag{2}$$

So, if the pixel intensity src(x, y) is greater than the threshold value T, then the new intensity of the pixel is initialized to 0. Otherwise, the pixels are set to max*Val*. Nobuyuki Otsu has given us the Otsu's method. Clustering-based image thresholding is achieved from this method. Otsu binarization automatically calculates a threshold value from image histogram for a bimodal image, which is an image whose histogram has two peaks. In Otsu's method we try to find the threshold that minimizes the intra-class variance (the variance within the class), defined as a weighted sum of variances of the two classes as seen in Equation 3. Weights 0 and 1 are the probabilities of the two classes separated by a threshold t and 02 and 12 are variances. The class probability 0, 1 (t) is computed from the L histograms. This is shown in Equation 4.

$$\sigma_{\omega}^{2}\left(t\right)=\omega_{0}\left(t\right)\sigma_{0}^{2}\left(t\right)+\omega_{1}\left(t\right)\sigma_{1}^{2}\left(t\right) \tag{3}$$

$$\omega_0\left(t\right)=\sum_{i=0}^{t-1}p\left(i\right)$$
$$\omega_1\left(t\right)=\sum_{i=t}^{L=1}p\left(i\right) \tag{4}$$

Otsu shows that minimizing the intra-class variance and maximizing inter-class variance generates the same results as seen below in Equation 5.

$$\sigma_b^2\left(t\right)=\sigma^2-\sigma_\omega^2\left(t\right)=\omega_0\left(\bar{\mu}_0-\mu_T\right)^2+\omega_1\left(\mu_1-\mu_T\right)^2=\omega_0\left(t\right)\omega_1\left(t\right)\left[\mu_0\left(t\right)-\mu_1\left(t\right)\right]^2$$

This is expressed in terms of for probabilities and for means. While the class mean $\mu_{0,1,T}\left(t\right)$ .can be expressed as shown in Equation 6.

$$\mu_0\left(t\right)=\sum_{i=0}^{t-1}i\frac{p\left(i\right)}{\omega_0}$$
$$\mu_1\left(t\right)=\sum_{i=t}^{L-1}i\frac{p\left(i\right)}{\omega_1} \tag{6}$$
$$\mu_T=\sum_{i=0}^{L-1}ip\left(i\right)$$

The following relations in Equation 7 can be easily verified.

$$\omega_0\mu_0+\omega_1\mu_1=\mu_T$$

$$\omega_0+\omega_1=1 \tag{7}$$

Before finding contours, threshold has been applied to the binary image to achieve higher accuracy.

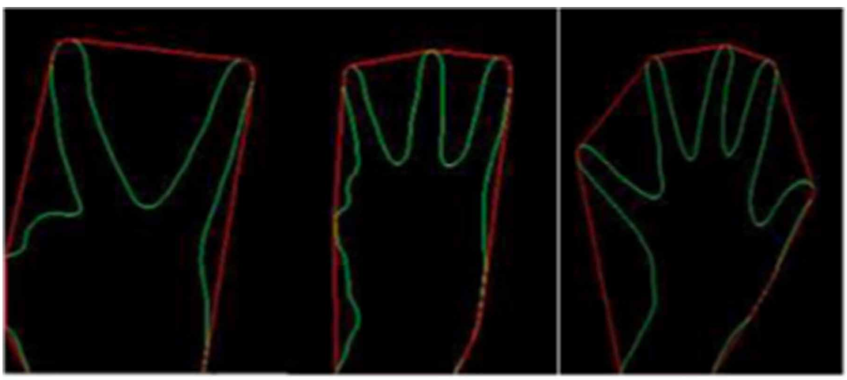
The image in Figure 3 shows the front end window that portrays the thresholded version of the user's gesture input.

3.3. Contour Extraction

Contours are a useful tool for object detection and recognition in image processing. In our work, we have used contours, to detect and recognize the hand from the background. The curves that link continuous points, which are of the same color, are called contours. Finding the contours is the first step which is like finding white object from black background in OpenCV. Hence, Inverted Binary Thresholding has been utilized during thresholding. The second step is to draw the contours which can be used to draw any shape provided the boundary points are known. Some gestures in our recognition system with their appropriate contours are shown in the below Figure 4.

Figure 3. Front end window showing the thresholded version of the input

Figure 4. Contour extraction



3.4. Convex Hull

Mathematically, convex hull of a set *X* of points in any affine space is defined as the smallest convex set that contains *X*. Any deviation of the object from this convex hull can be considered as convexity defect. The convex hull of a finite point set S can be defined as the set of all convex combinations of its points. In a convex combination, each point xi in S is assigned a weight αi and these weights are used to compute an average of the points. For each choice of weights, the resulting convex combination is a point in the convex hull. Convex hull can be represented mathematically as shown in Equation 8.

$$Convex\left(S\right)=\left\{\sum_{i=1}^{|S|}\alpha_i x_i \mid \left(\forall i:\alpha_i \geq 0\right)\wedge\sum_{i=1}^{|S|}\alpha_i=1\right\} \tag{8}$$

3.5. Convex Defect

The convex defect is defined as the difference between gesture convex hull and contour, they are contained in the convex hull but not hand area. The data structure of each of the convex defects contains three components: start contour point, end contour point and concave contour point. The fingertip is closely related to the convex defect, which is close to the start and end contour points of convex defect. Therefore, it is possible to detect the fingertips by using hand gesture contour and convex defects.

The count and position of fingertips are determined as follows:

- Conduct noise elimination on the obtained convex defects.
- Scan filtered convex defects clockwise, take the start contour point of the first convex defect and the end contour point of the last convex defect as the first and last fingertip respectively.
- Take use of the average position of the end contour point of current convex defect and the start contour point of next convex defect as the position of current fingertip.

3.6. Feature Extraction

For the binary image with convex defects, we can extract the convexity of gesture and relative position of fingertips as features used for hand gesture recognition. Through the observation and analysis of gesture contour and convex hull, we can get that with different gesture the tightness to its convex hull is also different, as shown in Figure 5, the convex hull of the fist gesture in (a) almost contains the whole gesture contour, but the gesture contour in (b) has big difference with its convex hull, with several depression existing between. The tightness of hand gesture contour to its convex hull is defined as the gesture convexity, which is denoted by δ, its value is the area ratio of gesture contour and convex hull.

$$\delta = contourArea/\mathrm{H}ullArea$$

*w*here, hullArea is the area of convex hull, contourArea is the area of gesture contour, we can get according to the image that hullArea>contourArea, so $\delta \in (0,1)$, and the bigger the value, the tighter the gesture contour to convex hull.

In addition, different gestures can be distinguished by the relative position of fingertips, which is composed of two values of α and β, α is the summation of angles with centroid as vertex, lines from centroid to the first fingertip and other fingertips as edges, and β is the value of the angle with centroid as vertex, lines from centroid to the first fingertip and last fingertip as edges, that is, $\alpha = \theta 1 + \theta 2 + \cdots + \theta N{-}1$, $\beta = \theta N{-}1$, where N is the number of fingertips and θ is the angle between the first fingertip and the other fingertip to the gesture centroid been considered as vertex, as shown in Figure 6, it has $N = 3$, $\alpha = \theta 1 + \theta 2$, $\beta = \theta 2$.

4. EXPERIMENTAL RESULTS

This section elaborates the implementation details of our hand gesture recognition system. In our gesture recognition system we have included a total of four gestures. Figure 5 shows the inputs taken from user for each gesture. The captions written at the top of each gesture i.e. “1.Google”, “2.Quora” denotes the number of fingers in each gesture and the website to which it is mapped. In gesture that do not have any defect i.e. palm, their name has been written as a caption above the gesture and is used to exit from the system.

Figure 6 shown above shows the gesture 2 which launches the facebook homepage, Figure 7 shows the gesture 3 which launches the youtube homepage, Figure 8 shows the gesture 4 that launches the twitter homepage and Figure 9 shows the gesture 5 which enables to close the terminal.

Figure 5. Input from user

Figure 6. Hand gesture 2- launches Facebook homepage

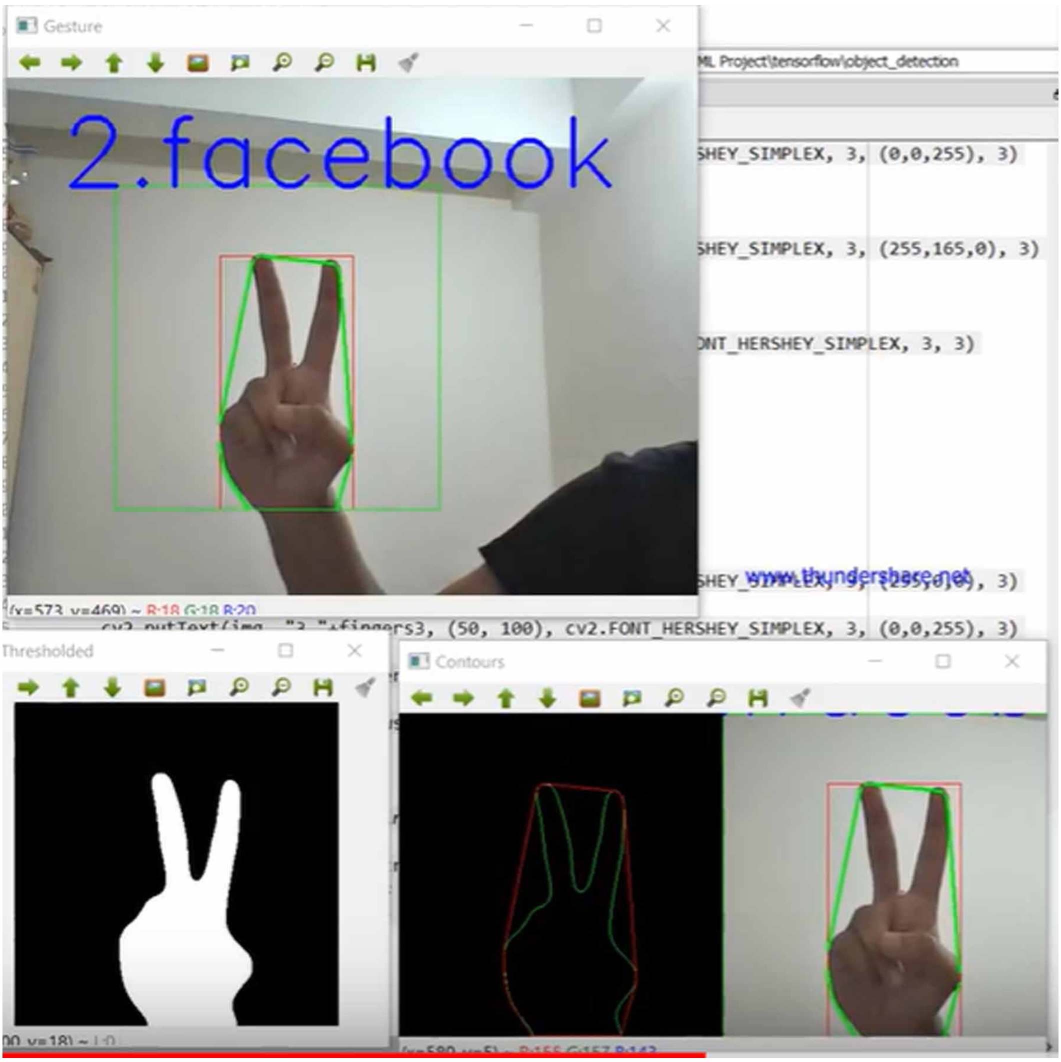
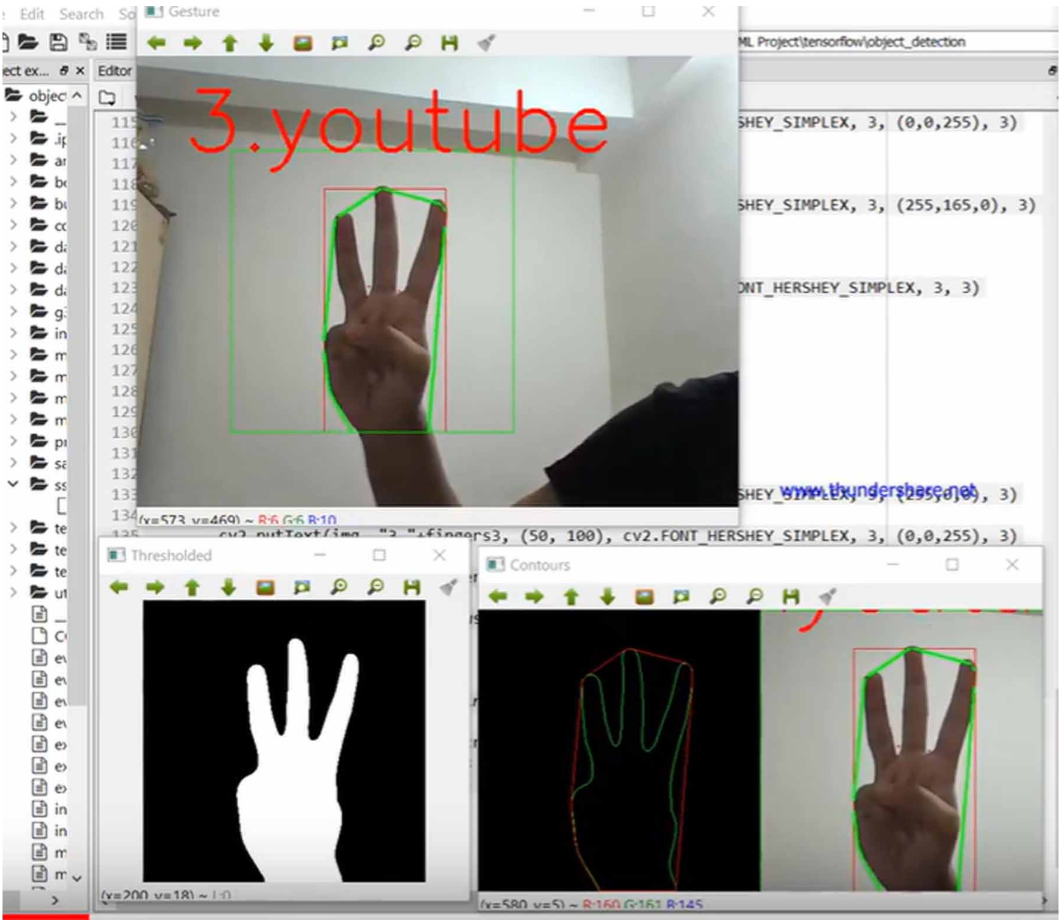


Figure 7. Hand gesture 3 launches Youtube homepage

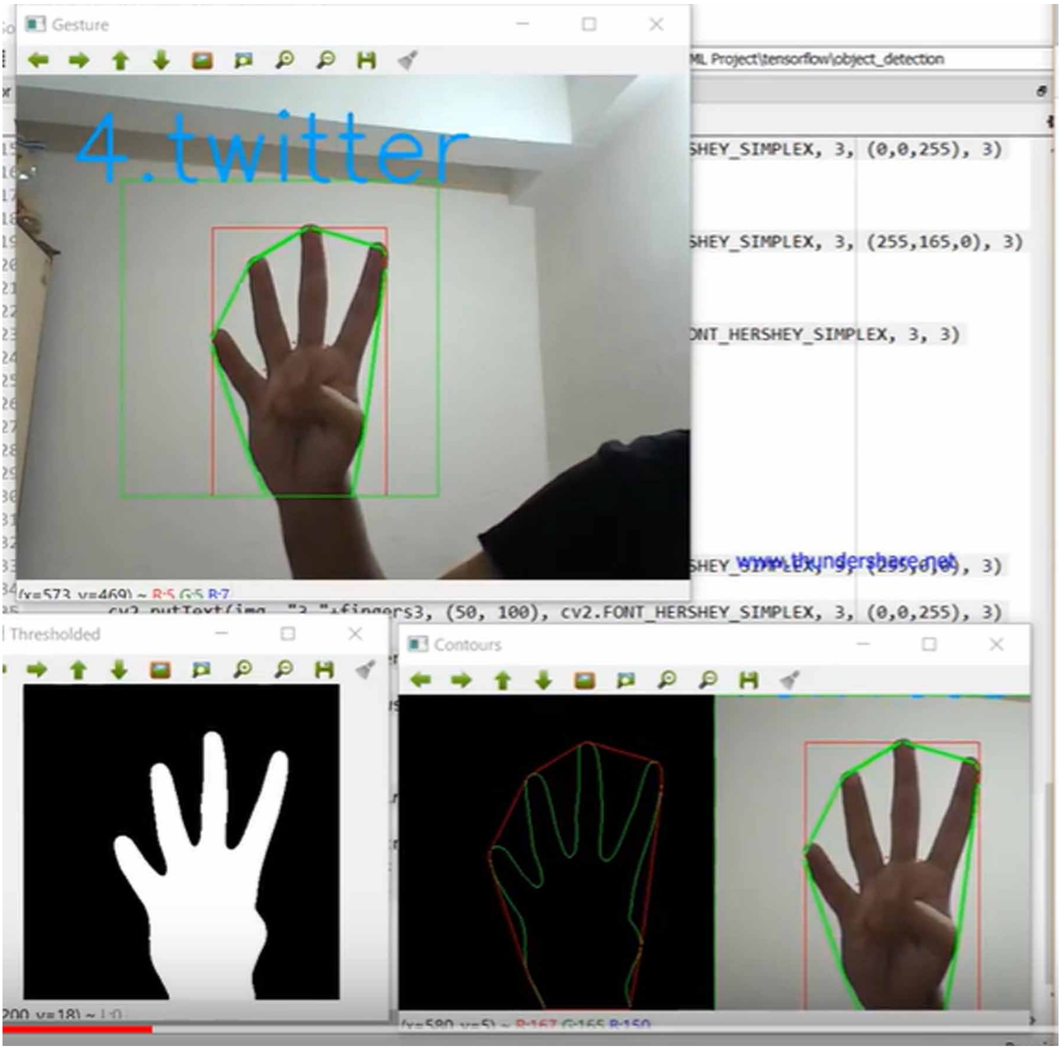


5. RESULT ANALYSIS

Our first approach to create a gesture recognition system was through the method of background subtraction using K-means clustering. Background subtraction, as the name suggests, is the process of separating foreground objects from the background in a sequence of video frames. It is a widely used approach for detecting moving objects from static cameras. When implementing the recognition system using background subtraction with K-means clustering, we encountered several drawbacks and accuracy issues. Background subtraction using K-means clustering cannot deal with sudden, drastic lighting changes leading to several inconsistencies. This method also requires relatively many parameters, which needs to be selected intelligently. Due to these complications faced, we made a decision to utilize contours, convexity defects to detect the object especially the hand. The combination of these methods enabled us to achieve a greater range of accuracy and overcome the challenges faced during the use of background subtraction using K-means.

To compute the accuracy of the proposed optimized hand gesture recognition system, we conducted two sets of evaluations. In the first set of evaluation, we used K-means clustering algorithm for segmentation of the image. In the second evaluation, we used the Otsu's image binarization method for segmentation. Each gesture was performed 10 times in both the setups. The average of the number

Figure 8. Hand gesture 4 launches Twitter homepage



of times a particular gesture was recognized correctly was taken as its accuracy in percentage and the accuracy obtained is shown in Table 1 for K-means clustering and Table 2 for Otsu's method.

When implemented using Otsu's method, the gesture recognition system was robust and performed with good accuracy. This accuracy was maintained irrespective of the colour of the background, provided it is a plain, solid colour background devoid of any inconsistencies. In cases where the background was not plain, the objects in the background proved to be inconsistencies to the image capture process, resulting in faulty outputs. However, the accuracy was not as good as the Otsu's method when implemented using K-means clustering. After observing the results produced by both the K- means and Otsu's method gesture recognition systems, it is recommended that system using Otsu's method and Convex Hull, be used with a plain background to produce the best possible results and great accuracy.

Table 1 presents the accuracy values for each gesture mentioned. These accuracy values are obtained using K-means clustering based segmentation method. Table 2 presents the accuracy values for each gesture mentioned using the combined method of otsu and convex hull. From the

Figure 9. Hand gesture 5 closes the terminal

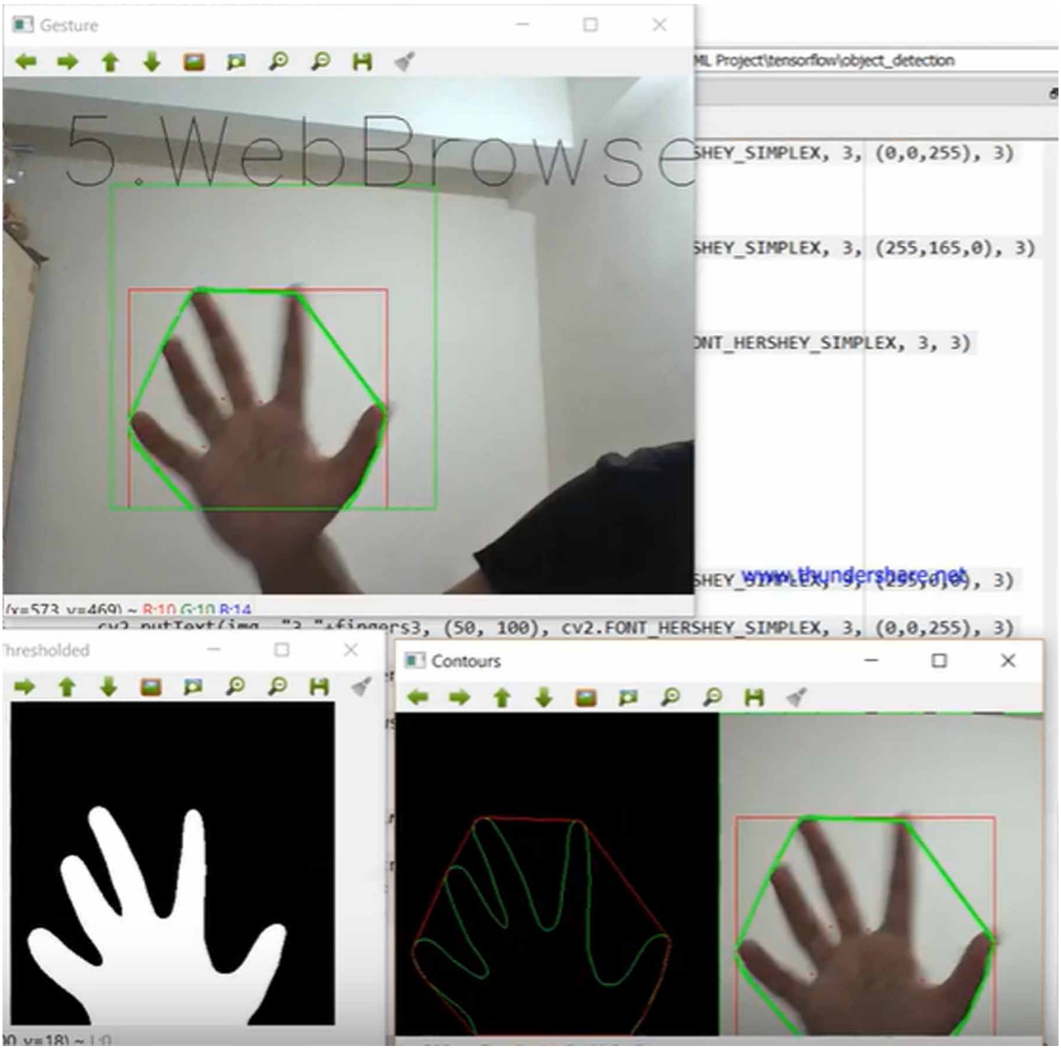


Table 1. Accuracy of each gesture with K-means clustering

Hand Gestures performed	Accuracy (%)
2 finger gesture (1 convex defect)	94
3 finger gesture (2 convex defects)	93
4 finger gesture (3 convex defects)	92
5 finger gesture (4 convex defects)	92

Table 2. Accuracy of each gesture with Otsu's method and Convex Hull

Hand Gestures performed	Accuracy(%)
2 finger gesture (1 convex defect)	85
3 finger gesture (2 convex defects)	89
4 finger gesture (3 convex defects)	90
5 finger gesture (4 convex defects)	87

results achieved and the accuracy values obtained, we could conclude that the proposed hand gesture recognition for possible website launching application works well. Accuracy of 90% has been achieved.

6. CONCLUSION

In this paper, a robust gesture recognition system is proposed that did not utilize any markers, hence making it more user friendly and low cost. This optimized hand gesture recognition system gives good accuracy percentages for the gesture recognition on par with the K-means clustering based segmentation. In this hand gesture recognition system, it is aimed to provide gestures, covering some aspects of human computer interface such as launching of applications and opening some popular websites. The human computer interaction was excellent that in the experiments conducted, every test turned out to be successful. This work used otsu's method, convex hull and convexity defects. In future, this could be further improvised with added gestures to incorporate more functions.

REFERENCES

Chaikhumpha, T., & Chomphuwiset, P. (2018, January). Real-time two hand gesture recognition with condensation and hidden Markov models. In Advanced *Image Technology (IWAIT), 2018 International Workshop on (pp. 1-4*). IEEE doi:10.1109/IWAIT.2018.8369811

Chanda, K., Ahmed, W., & Mitra, S. (2015). A new hand gesture recognition scheme for similarity measurement in a vision based barehanded approach. In Image *Information Processing (ICIIP); 2015 Third International Conference on. IEEE*. doi:10.1109/ICIIP.2015.7414712

Chen, W. L., Wu, C. H., & Lin, C. H. (2015). Depth-based hand gesture recognition using hand movements and defects. In Nex*t-Generation Electronics (ISNE); 2015 International Symposium on. I*EEE. doi:10.1109/ISNE.2015.7132005

Chen, Y., Ding, Z., Chen, Y. L., & Wu, X. (2015). Rapid recognition of dynamic hand gestures using leap motion. *Information and Automation; 2015 IEEE International Conference on*, 1419-1424 doi:10.1109/ICInfA.2015.7279509

Chen, Y., Luo, B., Chen, Y. L., Liang, G., & Wu, X. (2015). A real-time dynamic hand gesture recognition system using kinect sensor. In *Robotics and Biomimetics (ROBIO); 2015 IEEE International Conference on.* IEEE. doi:10.1109/ROBIO.2015.7419071

Dulayatrakul, J., Prasertsakul, P., Kondo, T., & Nilkhamhang, I. (2015). Robust implementation of hand gesture recognition for remote human-machine interaction. *Information Technology and Electrical Engineering (ICITEE); 2015 7th International Conference on*, 247-252.

Hu, B. & Wang, J. (2019). *Deep Learning Based Hand Gesture Recognition and UAV Flight Controls*. .10.1007/s11633-019-1194-7

Hung, C. H., Bai, Y. W., & Wu, H. Y. (2015). Home appliance control by a hand gesture recognition belt in LED array lamp case. In *Consumer Electronics (GCCE); 2015 IEEE 4th Global Conference on*. IEEE. doi:10.1109/GCCE.2015.7398611

Hung, C. H., Bai, Y. W., & Wu, H. Y. (2016). Home outlet and LED array lamp controlled by a smartphone with a hand gesture recognition. In *Consumer Electronics (ICCE); 2016 IEEE International Conference on*. IEEE. doi:10.1109/ICCE.2016.7430502

Huong, T. N., Huu, T. V., & Le Xuan, T. (2015). Static hand gesture recognition for vietnamese sign language (VSL) using principle components analysis. *Communications, Management and Telecommunications (ComManTel); 2015 International Conference on*, 138-141.

Hussain, I., Talukdar, A. K., & Sarma, K. K. (2014). Hand gesture recognition system with real-time palm tracking. In India Conference (INDICON); 2014 Annual IEEE. IEEE. doi:10.1109/INDICON.2014.7030571

Ishiyama, H., & Kurabayashi, S. (2016). Monochrome glove: A robust real-time hand gesture recognition method by using a fabric glove with design of structured markers. In Virtual Reality (VR); 2016 IEEE. IEEE. doi:10.1109/VR.2016.7504716

Kim, Thang, & Kim. (2009). 3-D hand Motion Tracking and Gesture Recognition Using a Data Glove. In *Industrial Electronics; 2009 IEEE International Symposium on*. IEEE.

Lee, D. H., & Hong, K. S. (2010). A Hand gesture recognition system based on difference image entropy. In *Advanced Information Management and Service (IMS), 2010 6th International Conference on*. IEEE.

Li, D.J., Li, Y.Y., & Li, J.X. (2018). *Gesture Recognition Based on BP Neural Network Improved by Chaotic Genetic Algorithm*. .10.1007/s11633-017-1107-6

Luzhnica, Lex, & Pammer. (2016). A Sliding Window Approach to Natural Hand Gesture Recognition using a Custom Data Glove. *3D User Interfaces (3DUI); 2016 IEEE Symposium on*, 81-90.

Luzhnica, G., Simon, J., Lex, E., & Pammer, V. (2016). A sliding window approach to natural hand gesture recognition using a custom data glove. In *3D User Interfaces (3DUI); 2016 IEEE Symposium on*. IEEE. doi:10.1109/3DUI.2016.7460035

Panwar, M. (2012). Hand gesture recognition based on shape parameters. *2012 International Conference on Computing, Communication and Applications,* 1-6. doi:10.1109/ICCCA.2012.6179213

Patil, A. R., & Subbaraman, S. (2018). A spatiotemporal approach for vision-based hand gesture recognition using Hough transform and neural network. *Signal, Image and Video Processing*, 1–9.

She, Y., Wang, Q., Jia, Y., Gu, T., He, Q., & Yang, B. (2014). A real-time hand gesture recognition approach based on motion features of feature points. In *Computational Science and Engineering (CSE); 2014 IEEE 17th International Conference on*. IEEE. doi:10.1109/CSE.2014.216

Soe, H. M., & Naing, T. M. (2018, May). Real-Time Hand Pose Recognition Using Faster Region-Based Convolutional Neural Network. In *International Conference on Big Data Analysis and Deep Learning Applications* (pp. 104-112). Springer.

Suriya, R., & Vijayachamundeeswari, V. (2014). A survey on hand gesture recognition for simple mouse control. In *Information Communication and Embedded Systems (ICICES); 2014 International Conference on*. IEEE. doi:10.1109/ICICES.2014.7033762

Tsai, T. H., Huang, C. C., & Zhang, K. L. (2015). Embedded virtual mouse system by using hand gesture recognition. In *Consumer Electronics-Taiwan (ICCE-TW); 2015 IEEE International Conference on*. IEEE. doi:10.1109/ICCE-TW.2015.7216939

Wang, C., Liu, Z., & Chan, S. C. (2015). Superpixel-Based Hand Gesture Recognition with Kinect Depth Camera. *IEEE Transactions on Multimedia*, *17*(1), 29–39. doi:10.1109/TMM.2014.2374357

Wong, W. S., Hsu, S. C., & Huang, C. L. (2015). Virtual touchpad: Hand gesture recognition for smartphone with depth camera. In *Consumer Electronics-Taiwan (ICCE-TW);2015 IEEE International Conference on*. IEEE.

Priyanka R. is an undergraduate student in the Department of Computer Science and Engineering, Anna University, Chennai. Her research interests include human computer interaction and data mining.

Prahanya Sriram is an undergraduate student in the Department of Computer Science and Engineering, Anna University, Chennai. Her research interests include human computer interaction and image processing.

Jayasree L. N. is an undergraduate student in the Department of Computer Science and Engineering, Anna University, Chennai. Her research interests include image processing and data mining.

Angelin Gladston is working as Associate Professor at the Department of Computer Science and Engineering, Anna University, Chennai. Her research interests include software engineering, software testing, image processing, social network analysis, human computer interaction and data mining.