# 3rd Sem Mini Project Report on

# PROJECT TITLE

**Submitted in partial fulfillment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Anshika Saklani** **University Roll No.-2318420**

*Under the Guidance of*

Mr. Rahul Chauhan
Assistant Professor

**Department of Computer Science and Engineering**
**Graphic Era Hill University**
**Dehradun, Uttarakhand**
**2024-25**

# CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the project report entitled **"HAND TRACKING SYSTEM"** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineering in the Department of Computer Science and Engineering of the Graphic Era Hill University, Dehradun shall be carried out by the undersigned under the supervision of **Mr. Rahul Chauhan, Assistant Professor**, Department of Computer Science and Engineering, Graphic Era Hill University, Dehradun.

Anshika Saklani Roll no:- 2318420

The above mentioned student shall be working under the supervision of the undersigned on the **"Hand Tracking System"**

**Supervisor** **Head of the Department**

## <u>Examination</u>

**Name of the Examiners:** **Signature with Date**

1.

2.

**TABLE OF CONTENTS**

# HAND GESTURE VOLUME CONTROL

**ABSTRACT**

Hand gesture recognition system received great attention in the recent few years because of its manifoldness applications and the ability to interact with machine efficiently through human computer interaction. In this paper a survey of recent hand gesture recognition systems is presented. Key issues of hand gesture recognition system are presented with challenges of gesture system. Review methods of recent postures and gestures recognition system presented as well. Summary of research results of hand gesture methods, databases, and comparison between main gesture recognition phases are also given. Advantages and drawbacks of the discussed systems are explained finally.

**KEYWORDS**

Hand Posture, Hand Gesture, Human Computer Interaction (HCI), Segmentation, Feature Extraction,

Classification Tools, Neural Networks.

1. **INTRODUCTION**

The essential aim of building hand gesture recognition system is to create a natural interaction between human and computer where the recognized gestures can be used for controlling a robot or conveying meaningful information . How to form the resulted hand gestures to be understood and well interpreted by the computer considered as the problem of gesture interaction Human computer interaction (HCI) also named Man-Machine Interaction (MMI) refers to the relation between the human and the computer or more precisely the machine, and since the machine is insignificant without suitable utilize by the human . There are two main characteristics should be deemed when designing a HCI system as mentioned in : functionality and usability. System functionality referred to the set of functions or services that the system equips to the users , while system usability referred to the level and scope that the system can operate and perform specific user purposes efficiently . The system that attains a suitable balance between these concepts considered as influential performance and powerful system . Gestures used for communicating between human and machines as well as between people using sign language . Gestures can be static (posture or certain pose) which require less computational complexity or dynamic (sequence of postures) which are more complex but suitable for real time environments . Different methods have been proposed for acquiring information necessary for recognition gestures system . Some methods used additional hardware devices such as data glove devices and color markers to easily extract comprehensive description of gesture features .

## 1. ISSUES TO HAND GESTURE RECOGNITION: EXTRACTION METHODS AND FEATURES EXTRACTION

Most of the researchers classified gesture recognition system into mainly three steps after acquiring the input image from camera(s), videos or even data glove instrumented device. These steps are: Extraction Method, features estimation and extraction, and classification or recognition as illustrated in Figure 1.

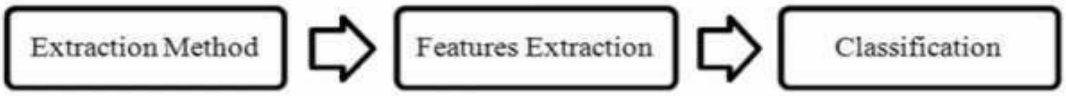



Figure 1. Gesture recognition system steps.

## Extraction Method and image pre-processing

Segmentation process is the first process for recognizing hand gestures. It is the process of dividing the input image (in this case hand gesture image) into regions separated by boundaries . The segmentation process depends on the type of gesture, if it is dynamic gesture then the hand gesture need to be located and tracked , if it is static gesture (posture) the input image have to be segmented only. The common helpful cue used for segmenting the hand is the skin color , since it is easy and invariant to scale, translation, and rotation changes . Different tools and methods used skin and non-skin pixels to model the hand. These methods are parametric and non-parametric techniques, Gaussian Model (GM) and Gaussian Mixture Model (GMM) are parametric techniques, and histogram based techniques are non- parametric. However it is affected with illumination condition changes abs different races. Some researches overcome this problem using data glove and colored markers which provide exact information about the orientation and position of palm and fingers . Others used infrared camera , and range information generated by special camera Time-of-Flight (ToF) camera , although these systems can detect different skin colors under cluttered background but it is affected with changing in temperature degrees besides their expensive cost . The segmentation considered as an open issue problem itself . The color space used in a specific application plays an essential role in the success of segmentation process, however color spaces are sensitive to lighting changes, for this reason, researches tend to use chrominance components only and neglect the luminance components such as r-g, and HS color spaces. However there are some factors that obstacle the segmentation process which is ; complex background, illumination changes, low video quality. Applied HSV color model which concentrates on the pigments of the pixel, used YCbCr color space. used normalized r-g color space. Some preprocessing operations are applied such as subtraction, edge detection, and normalization to enhance the segmented hand image .

Figure 2 shows some segmentation method examples.

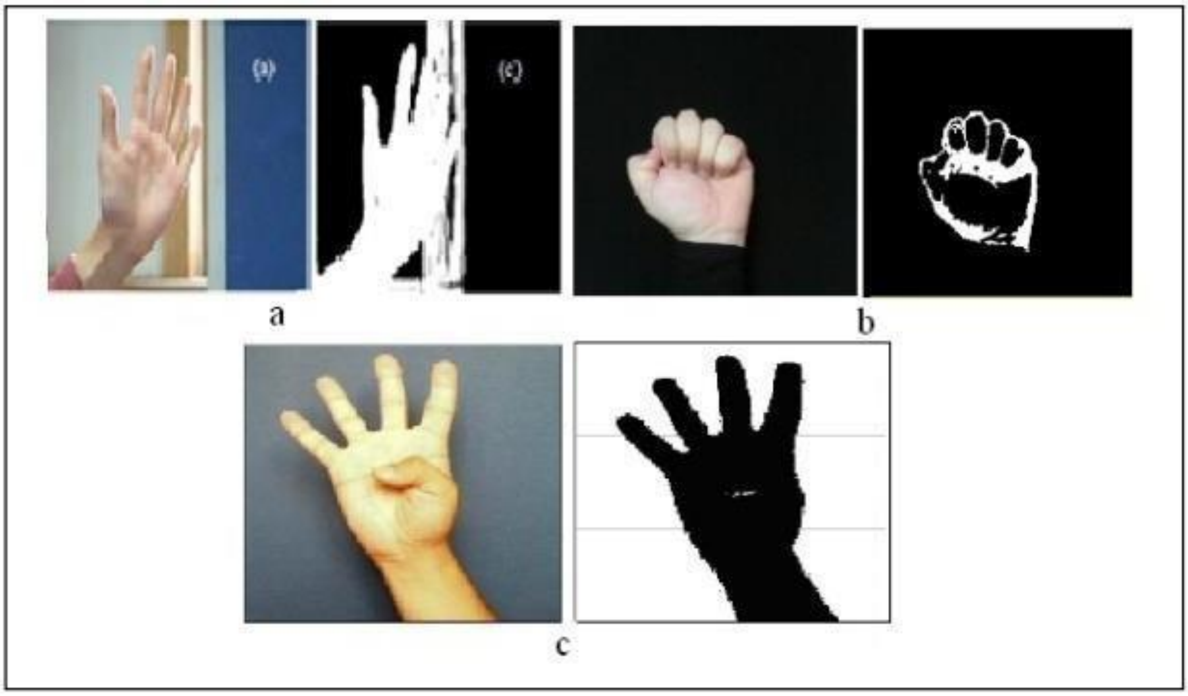



Figure 2. segmentation method

**Features Extraction**

Good segmentation process leads to perfect features extraction process and the latter play an important role in a successful recognition process . Features vector of the segmented image can be extracted in different ways according to particular application. Various methods have beenapplied for representing the features can be extracted. Some methods used the shape of the hand such as hand contour and silhouette while others utilized fingertips position, palm center, etc. created 13 parameters as a feature vector, the first parameters represents the ratio aspect of the bounding box of the hand and the rest 12 parameters are mean values of brightness pixels in the image. used Self- Growing and Self-Organized Neural Gas (SGONG) neural algorithm to capture the shape of the hand, then three features are obtained; Palm region, Palm center, and Hand slope. calculated the Center Of Gravity (COG) of the segmented hand and the distance from the COG to the farthest point in the fingers, and extracted one binary signal (1D) to estimate the number of fingers in the hand region. divided the segmented image into different blocks size and each block represents the brightness measurements in the image. Many experiments were applied to decide the right block size that can achieve good recognition rate . used Gaussian pdf to extract geometric central moment as local and global features.
Figure 3 shows some applications of feature extraction methods.

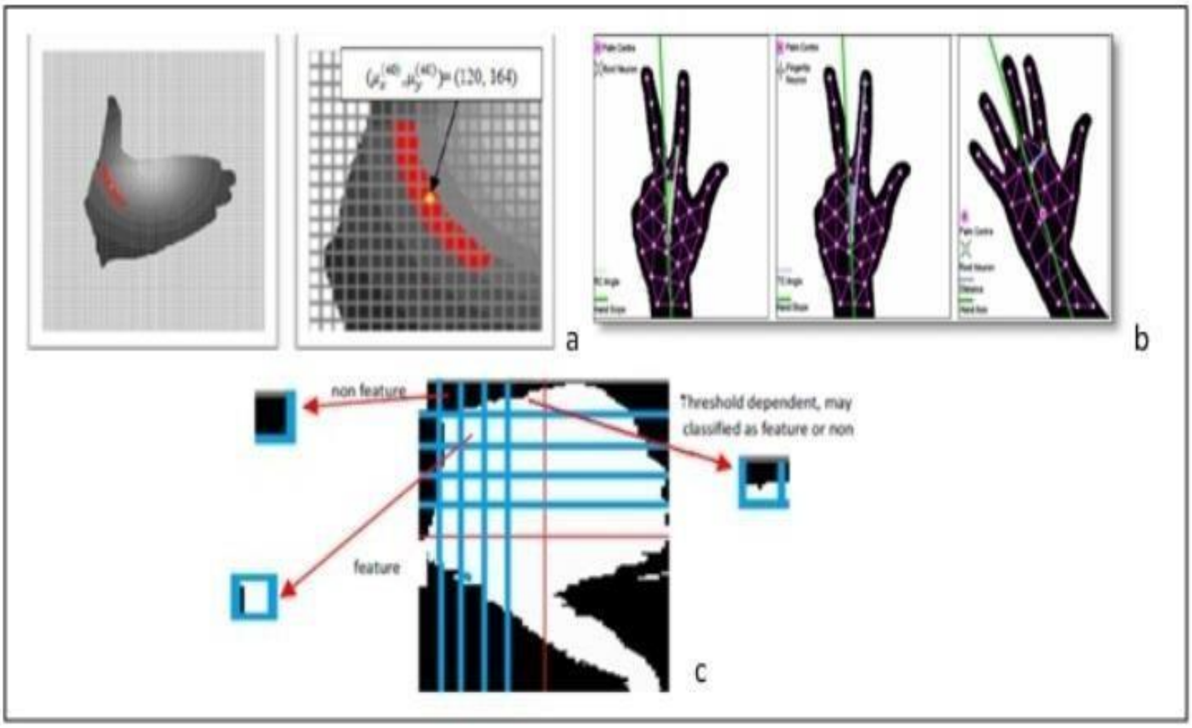



Figure 3. features representation. a) The segmented image is partitioned into 11 terraces with 8 regions per terrace to extract local and global geometric central moment . b) Three angles are extracted: RC angle, TC angle, and distance from the palm center . Segmented hand divided into blocks and the brightness factor for each block represents the feature vector (blocks with black area are discarded) .

## Gestures Classification

After modeling 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 . For example edge detection or contour operators cannot be used for gesture recognition since many hand postures are generated and could produce misclassification . Euclidean distance metric used to classify the gestures . Statistical tools used for gesture classification, HMM tool has shown its ability to recognize dynamic gestures besides, Finite State Machine (FSM) , Learning Vector Quantization , and Principal Component Analysis (PCA) . Neural network has been widely applied in the field of extracted the hand shape , and for hand gesture recognition . Other soft computing tools are effective in this field as well, such as Fuzzy C-Means clustering (FCM), and Genetic Algorithms GAs .

Figure 4 explain the architecture of classification system.

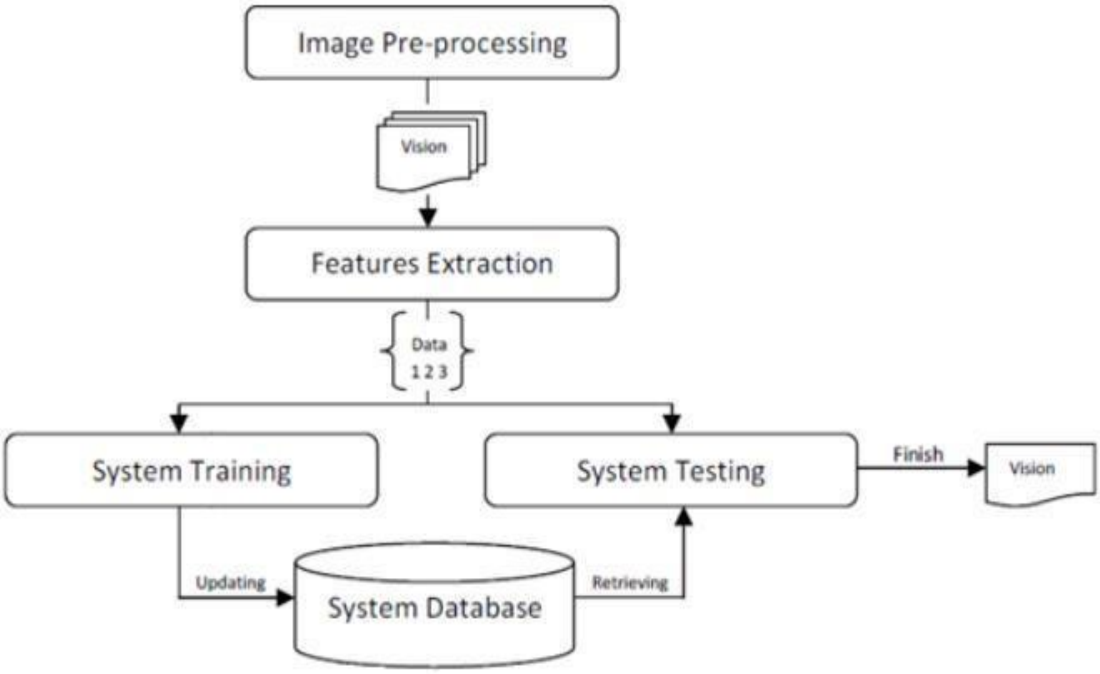



Figure 4. Architecture of gesture recognition system

## 1. APPLICATION AREAS OF HAND GESTURES SYSTEM

Hand gestures recognition system has been applied for different applications on different domains, as mentioned in including; sign language translation, virtual environments, smart surveillance, robot control, medical systems etc. overview of some hand gesture application areas are listed below.

### A. Sign Language Recognition:

Since the sign language is used for interpreting and explanations of a certain subject during the conversation, it has received special attention . A lot of systems have been proposed to recognize gestures using different types of sign languages . For example recognized American Sign Language ASL using boundary histogram, MLP neural network and dynamic programming matching. recognized Japanese sign language JSL using Recurrent Neural Network, 42 alphabet and 10 words. recognized Arabic Sign language ArSL using two different types of Neural Network, Partially and Fully Recurrent neural Network.

### B. Robot Control:

Controlling the robot using gestures considered as one of the interesting applications in this field . proposed a system that uses the numbering to count the five fingers for controlling a robot using hand pose signs. The orders are given to the robot to perform a particular task , where each sign has a specific meaning and represents different function for example, “one” means “move forward”, “five” means “stop”, and so on.

### C. Graphic Editor Control:

Graphic editor control system requires the hand gesture to be tracked and located as a preprocessing operation . used 12 dynamic gestures for drawing and editing graphic system. Shapes for drawing are; triangle, rectangular, circle, arc, horizontal and vertical line for drawing, and commands for editing graphic system are; copy, delete, move, swap, undo, and close .

### D. Virtual Environments ( VEs):

One of the popular applications in gesture recognition system is virtual environments VEs, especially for communication media systems . provided 3D pointing gesture recognition for natural human computer Interaction HCI in a real-time from binocular views. The proposed system is accurate and independent of user characteristics and environmental changes .

### E. Numbers Recognition:

Another recent application of hand gesture is recognizing numbers. proposed an automatic system that could isolate and recognize a meaningful gesture from hand motion of Arabic numbers from 0 to 9 in a real time system using HMM.

### F. Television Control:

Hand postures and gestures are used for controlling the Television device . In a set of hand gesture are used to control the TV activities, such as turning the TV on and off, increasing and decreasing the volume, muting the sound, and changing the channel using open and close hand .

### G. 3D Modeling:

To build 3D modeling, a determination of hand shapes are needed to create, built and view 3D shape of the hand . Some systems built the 2D and 3D objects using hand silhouette . 3D hand modeling can be used for this purpose also which still a promising field of research .

## 2. LITERATURE REVIEW OF GESTURE RECOGNITION SYSTEMS

The applied multivariate Gaussian distribution to recognize hand gestures using non geometric features. The input hand image is segmented using two different methods ; skin color based segmentation by applying HSV color model and clustering based thresholding techniques . Some operations are performed to capture the shape of the hand to extract hand feature; the modified Direction Analysis Algorithm are adopted to find a relationship between statistical parameters (variance and covariance) from the data, and used to compute object (hand) slope and trend by finding the direction of the hand gesture , As shown in Figure 5.

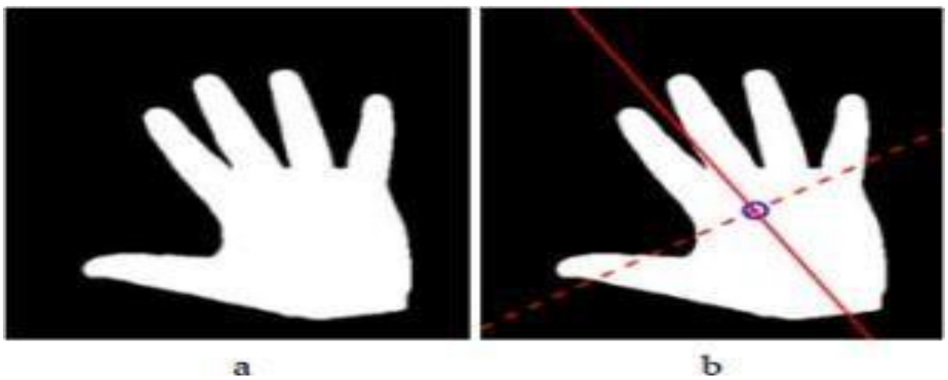



Figure5. computing hand direction

Then Gaussian distinction is applied on the segmented image, and it takes the direction of the hand as shown in figure 6.

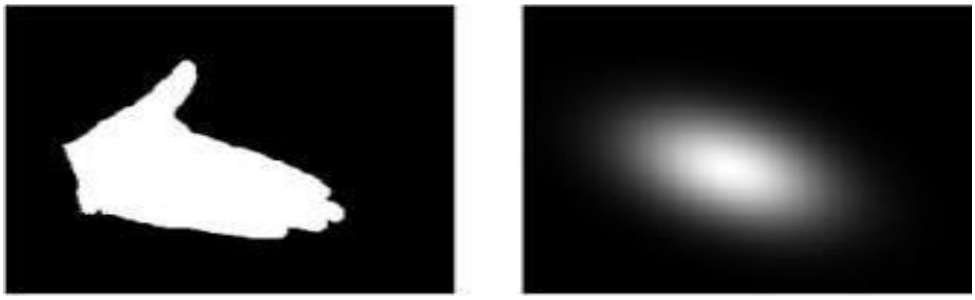


Figure 6. Gaussian distribution applied on the segmented image .

Form the resultant Gaussian function the image has been divided into circular regions in other words that regions are formed in a terrace shape so that to eliminate the rotation affect . The shape is divided into 11 terraces with a 0.1 width for each terrace . 9 terraces are resultant from the 0.1 width division which are; (1-0.9, 0.9-0.8, 0.8-0.7, 0.7-0.6, 0.6, 0.5, 0.5-0.4, 0.4-0.3, 0.3-0.2, 0.2-0.1), and one terrace for the terrace that has value smaller than 0.1 and the last one for the external area that extended out of the outer terrace [17][18]. An explanation of this division is demonstrated in Figure 7.

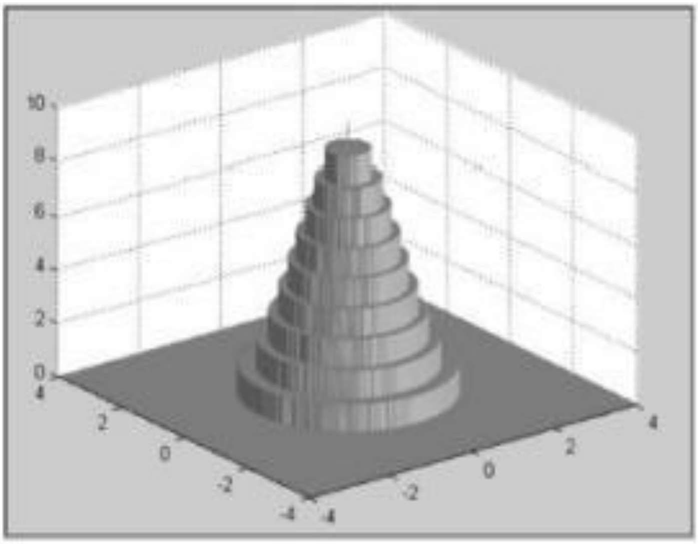



Figure 7. Terraces division with 0.1 likelihood .

Each terrace is divided into 8 sectors which named as the feature areas, empirically discovered that number 8 is suitable for features divisions , To attain best capturing of the Gaussian to fit the segmented hand, re-estimation are performed on the shape to fit capturing the hand object , then the Gaussian shape are matched on the segmented hand to prepare the final hand shape for extracting the features, Figure 8 shown this process .

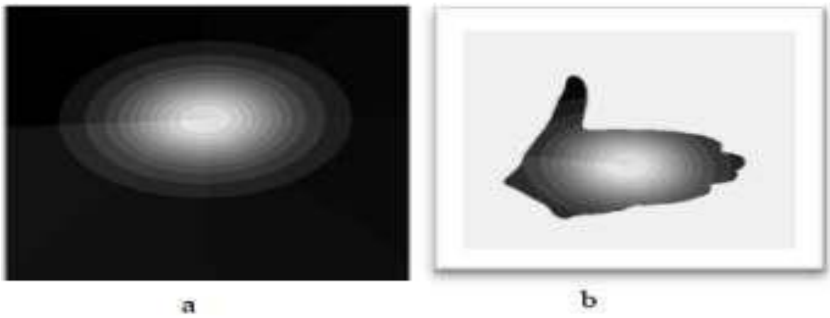



Figure 8. Features divisions . a) Terrace area in Gaussian. b) Terrace area in hand image

After capturing the hand shape, two types of features are extracted to form the feature vector ; local feature, and global features. Local features using geometric central moments which provide two different moments $\mu 00$ , $\mu 11$ as shown by equation (1):

$$\mu_{pp} = \sum_{x}\sum_{y}(x-\mu_x)^p(y-\mu_y)^p f(x,y) \tag{1}$$

$$\mu_{pp}^{(k)} = \sum_{y}\sum_{x}(x^{(k)}-\mu_x^{(k)})^p(y^{(k)}-\mu_y^{(k)})^p f(x^{(k)},y^{(k)}) \tag{2}$$

$$\forall k \in \{1, 2, 3, \ldots, 88\} \ \& \ \forall p \in \{0, 1\}$$

Where $\mu x$ and $\mu y$ is the mean value for the input feature area , x and y are the coordinated, and for this, the input image is represented by 88*2 features, as explained in detail in equation (2). While the global features are two features the first and second moments that are the computed for the whole hand features area . These feature areas are computed by multiplying feature area intensity plus feature area's map location . In this case, any input image is represented with 178 features . The system carried out using 20 different gestures , 10 samples for each gesture, 5 samples for training and 5 for testing, with 100% recognition percentage and it decreased when the number of gestures are more than 14 gestures . In 6 gestures are recognized with 10 samples for each gesture. Euclidian distance used for the classification of the feature . Kulkarni recognize static posture of American Sign Language using neural networks algorithm. The input image are converted into HSV color model, resized into 80x64 and some image preprocessing operations are applied to segment the hand from a uniform background , features are extracted using histogram technique and Hough algorithm. Feed forward Neural Networks with three layers are used for gesture classification. 8 samples are used for each 26 characters in sign language, for each gesture, 5 samples are used for training and 3samples for testing, the system achieved 92.78% recognition rate using MATLAB language.

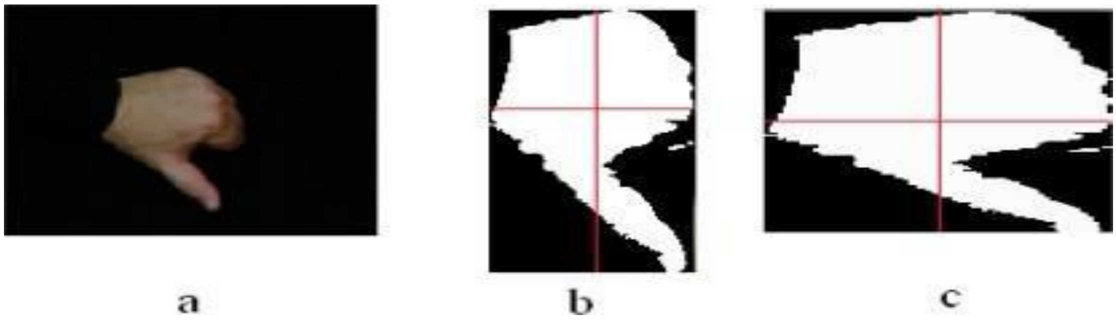


Figure 9. applying trimming process on the input image, followed by scaling normalization process

Wysoski et al presented rotation invariant postures using boundary histogram. Camera used for acquire the input image, filter for skin color detection has been used followed by clustering process to find the boundary for each group in the clustered image using ordinary contourtracking algorithm. The image was divided into grids and the boundaries have been normalized. The boundary was represented as chord's size chain which has been used as histograms, by dividing the image into number of regions N in a radial form, according to specific angle. For classification process Neural Networks MLP and Dynamic Programming DP matching were used. Many experiments have implemented on different features format in addition to use different chord's size histogram, chord's size FFT. 26 static postures from American Sign Language used in the experiments. Homogeneous background was applied in the work. Stergiopoulou suggested a new Self-Growing and Self-Organized Neural Gas (SGONG) network for hand gesture recognition. For hand region detection a color segmentation technique based on skin color filter in the YCbCr color space was used, an approximation of hand shape morphology has been detected using (SGONG) network; Three features were extracted using finger identification process which determines the number of the raised fingers and characteristics of hand shape, and Gaussian distribution model used for recognition.

## 3. DRAWBACKS

In this section, drawbacks of some discussed methods are explained: Orientation histogram method applied in have some problems which are; similar gestures might have different orientation histograms and different gestures could have similar orientation histograms, besides that, the proposed method achieved well for any objects that dominate the image even if it is not the hand gesture . Neural Network classifier has been applied for gestures classification but it is time consuming and when the number of training data increase, the time needed for classification are increased too . In the NN required several hours for learning 42 characters and four days to learn ten words . Fuzzy c-means clustering algorithm applied in has some disadvantages; wrong object extraction problem raised if the objects larger than the hand. The performance of recognition algorithm decreases when the distance greater than 1.5 meters between the user and the camera. Besides that, its variation to lighting condition changes and unwanted objects might overlap with the hand gesture.

## 4. SUMMARY OF RESEARCH RESULTS

The following tables show summaries of some hand gesture recognition systems. In Table 1 a comparison between recognition methods in hand gesture recognition methods used. Table 2 provides a summary of application areas and invariant vector of some hand gesture recognition systems. Table 3 displays Summary of extraction method, features representation, and recognition of hand gesture recognition systems which are; hand extraction technique, features vector representation, and recognition used in the selected hand gesture recognition systems.

Table 1. Comparison between recognition methods in hand gesture recognition methods used

| Method | # Recognized Gestures | # Total Gestures used For Training And Testing | Recognition Percentage | Database used |
|---|---|---|---|---|
| [8] | 26 | 1040 | DP 98.8%<br>MLP 98.7% | American Sign Language (ASL) |
| [5] | 6 | 60 | normal method 84%<br>Scaling normalization method 95% | Own Database |
| [31] | 26 | 208 | 92.78% | American Sign Language (ASL) |
| [032] | 0-9 numbers | 298 video sequence for isolated gestures/ 270 video sequence for continuous gestures | 90.45% | Recognize Arabic numbers from 0 to 9. |
| [20] | 5 static/ 12 dynamic gestures | Totally 240 data are trained and then the trained are tested | 98.3% | 5 static gestures and 12 dynamic gestures. |
| [14] | 31 | 130 for testing | 90.45% | Own Database |
| [17] | 6 | 60 | 100% for more than 4 gestures | Own Database |
| [18] | 20 | 200 | 100% for 14 gestures, and >90 for 15-20 gestures | Own Database |

Table 2. Summary of application areas and invariant vector of some hand gesture recognition systems.

| Method | Application Area | Invariant factor |
|---|---|---|
| [19] | Real time system / control a computer graphic crane by hand gestures/ play games such as scissors/paper/stone | Lighting conditions / Translation |
| [14] | Sign Recognition | Lighting conditions / Translation |
| [8] | Sign Recognition | Rotation |
| [5] | Sign language | Rotation/ Translation/ Scaling |
| [16] | Robot control application | Translation/ Rotation / Scaling |
| [22] | Real-time system/ moderate computational resources devices e.g. netbooks | Rotation / Translation |
| [17] | Sign Recognition | Rotation/Translation/ Scaling |
| [18] | Sign Recognition | Rotation/Translation/ Scaling |
| [20] | Drawing graphical elements such as triangle, rectangular/ Editing graphical elements such as copy, paste, undo/ Mobile robot control/Virtual Reality. | - |

Table 3. Summary of extraction method, features representation, and recognition of hand gesture recognition systems

| Method | Extraction method | Features Vector Representation | Classifier |
|---|---|---|---|
| [6] | HSV color space | 13 parameters as a feature vector, the first parameters represents the ratio aspect of the bounding hand box and the rest 12 parameters are the mean values of brightness pixels in the image | Fuzzy C-Means (FCM) algorithm |
| [8] | - | Boundary Chord's size FFT<br>Boundary Chord's size<br>Boundary Chord's size histogram | MLP Neural Network/ Dynamic Programming (DP) matching |
| [14] | YCbCr color space | From Self-Growing and Self-Organized Neural obtained hand shape, then, three angles of the hand shape calculated; RC Angle, TC Angle, Distance from the palm center. | Gaussian distribution |
| [22] | Colored glove/ HSV and threshold based method | Nine numerical features formed by distances from palm to all fingers and four angles between the distances | Learning Vector Quantization (LVQ) |
| [032] | GMM for skin | Orientation quantization | HMM |

## 5. CODE & IMPLEMENTATION

### 7.1 HandTracking.py

```
import cv2
import mediapipe as mp
import time

cap = cv2.VideoCapture(0)

mpHands = mp.solutions.hands
hands = mpHands.Hands()
mpDraw = mp.solutions.drawing_utils

pTime = 0
cTime = 0

while True:
    success, img = cap.read()
    imgRGB = cv2.cvtColor(img,cv2.COLOR_BGR2RGB)
    results = hands.process(imgRGB)
    # print(results.multi_hand_landmarks)

    if results.multi_hand_landmarks:
        for handLms in results.multi_hand_landmarks:
            for id, lm in enumerate(handLms.landmark):
                # print(id, lm)
                h, w, c = img.shape
                cx, cy = int(lm.x*w), int(lm.y*h)
                print(id, cx, cy)
                if id == 0:
                    cv2.circle(img, (cx, cy), 25, (255, 0, 255), cv2.FILLED)

            mpDraw.draw_landmarks(img, handLms, mpHands.HAND_CONNECTIONS)

    cTime = time.time()
    fps = 1/(cTime-pTime)
    pTime = cTime

    cv2.putText(img, str(int(fps)), (10,70), cv2.FONT_HERSHEY_PLAIN, 3
                , (255, 0, 255), 3)

    cv2.imshow("Image", img)
    cv2.waitKey(1)
```

### 7.2 HandTrackingModule.py

```
import cv2
```

```
import mediapipe as mp
import time

cap = cv2.VideoCapture(0)


class handDetector():
    def __init__(self, mode=False, maxHands = 2, detectionCon = 0.5, trackCon =
0.5):
        self.mode = mode
        self.maxHands = maxHands
        self.detectionCon = detectionCon
        self.trackCon = trackCon

        self.mpHands = mp.solutions.hands
        self.hands = self.mpHands.Hands(self.mode, self.maxHands, 1,
self.detectionCon, self.trackCon)
        self.mpDraw = mp.solutions.drawing_utils

    def findHands(self, img, draw=True):
        imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        self.results = self.hands.process(imgRGB)
        # print(results.multi_hand_landmarks)

        if self.results.multi_hand_landmarks:
            for handLms in self.results.multi_hand_landmarks:
                if draw:
                    self.mpDraw.draw_landmarks(img, handLms,
                                               self.mpHands.HAND_CONNECTIONS)
        return img

    def findPos(self, img, handNo=0, draw=True):

        lmList = []
        if self.results.multi_hand_landmarks:
            myHand = self.results.multi_hand_landmarks[handNo]

            for id, lm in enumerate(myHand.landmark):
                # print(id, lm)
                h, w, c = img.shape
                cx, cy = int(lm.x * w), int(lm.y * h)
                # print(id, cx, cy)
                lmList.append([id, cx, cy])
                if draw:
                    cv2.circle(img, (cx, cy), 7, (255, 0, 255), cv2.FILLED)

        return lmList
```

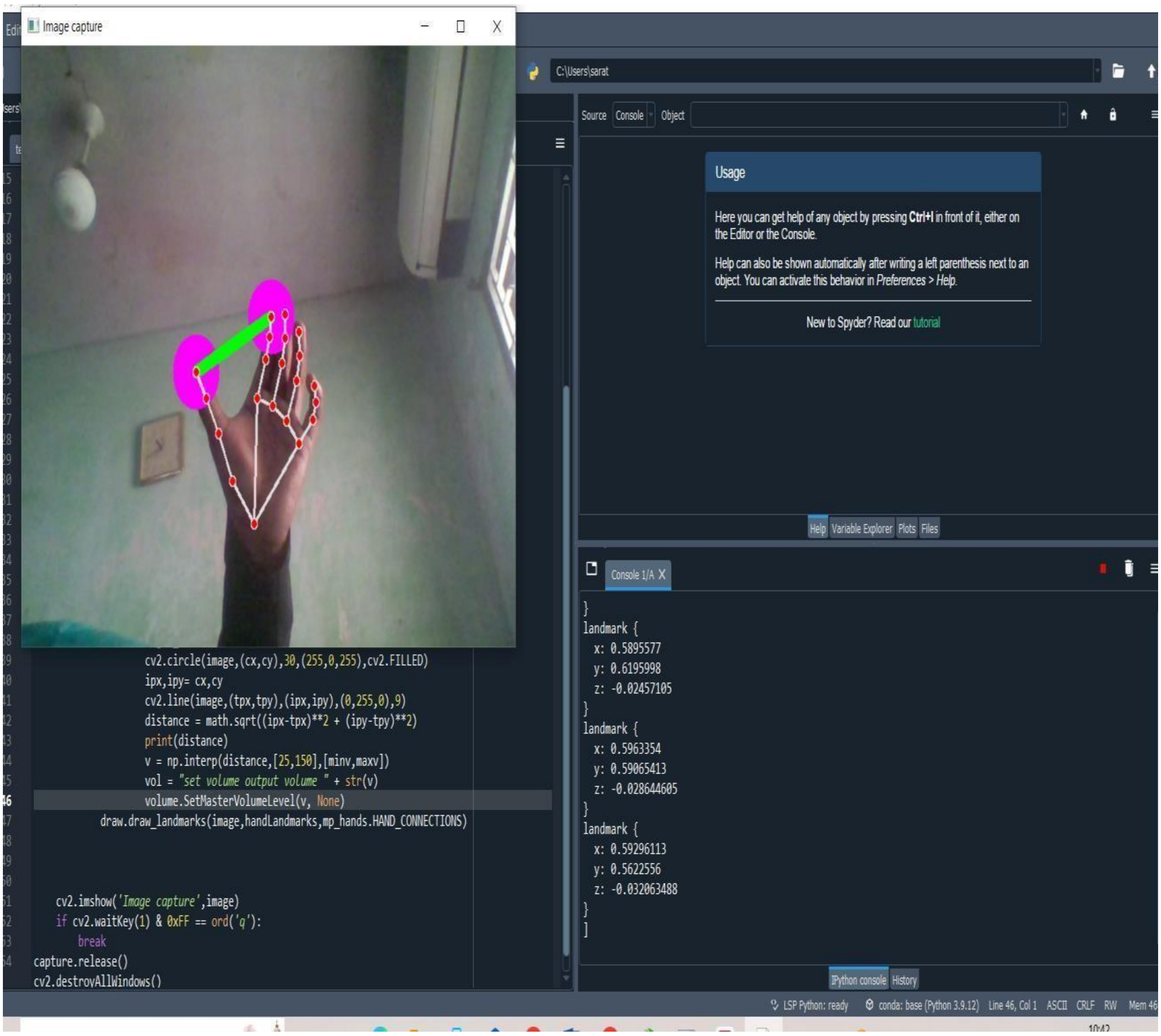
```
def main():
    pTime = 0
    cTime = 0
    cap = cv2.VideoCapture(0)
    detector = handDetector()
    while True:
        success, img = cap.read()
        img = detector.findHands(img, draw=False)
        lmList = detector.findPos(img)
        if len(lmList) != 0:
            print(lmList[0])
        cTime = time.time()
        fps = 1 / (cTime - pTime)
        pTime = cTime

        cv2.putText(img, str(int(fps)), (10, 70), cv2.FONT_HERSHEY_PLAIN, 3
                    , (255, 0, 255), 3)

        cv2.imshow("Image", img)
        cv2.waitKey(1)



if __name__ == "__main__":
    main()
```

## 7.3 mygame.py

```
import time

import cv2

import HandTrackingModule as htm


pTime = 0
cTime = 0
cap = cv2.VideoCapture(0)
detector = htm.handDetector()
while True:
    success, img = cap.read()
    img = detector.findHands(img)
    lmList = detector.findPos(img)
    if len(lmList) != 0:
        print(lmList[0])
    cTime = time.time()
    fps = 1 / (cTime - pTime)
    pTime = cTime
```

```
cv2.putText(img, str(int(fps)), (10, 70), cv2.FONT_HERSHEY_PLAIN, 3
            , (255, 0, 255), 3)

cv2.imshow("Image", img)
cv2.waitKey(1)
```

**OUTPUT**




## 6. CONCLUSIONS

In this paper various methods are discussed for gesture recognition, these methods include from

Neural Network, HMM, fuzzy c-means clustering, besides using orientation histogram for features representation. For dynamic gestures HMM tools are perfect and have shown its efficiency especially for robot control . NNs are used as classifier and for capturing hand shape in . For features extraction, some methods and algorithms are required even to capture the shape of the hand as in applied Gaussian bivariate function for fitting the segmented hand which used to minimize the rotation affection . The selection of specific algorithm for recognition depends on the application needed. In this work application areas for the gestures system are presented. Explanation of gesture recognition issues, detail discussion of recent recognition systems are given as well. Summary of some selected systems are listed as well.

## 7. REFERENCES


G. R. S. Murthy, R. S. Jadon. (2009). "A Review of Vision Based Hand Gestures Recognition,"International Journal of Information Technology and Knowledge Management, vol. 2(2), pp.

405-410.

P. Garg, N. Aggarwal and S. Sofat. (2009). "Vision Based Hand Gesture Recognition," World Academy of Science, Engineering and Technology, Vol. 49, pp. 972-977.

Fakhreddine Karray, Milad Alemzadeh, Jamil Abou Saleh, Mo Nours Arab, (2008) "HumanComputer Interaction: Overview on State of the Art", International Journal on Smart Sensing and Intelligent Systems, Vol. 1(1).

[1] Wikipedia Website.

Mokhtar M. Hasan, Pramoud K. Misra, (2011). "Brightness Factor Matching For Gesture Recognition System Using Scaled Normalization", International Journal of Computer Science & Information

Technology (IJCSIT), Vol. 3(2).

Xingyan Li. (2003). "Gesture Recognition Based on Fuzzy C-Means Clustering Algorithm", Department of Computer Science. The University of Tennessee Knoxville.
S. Mitra, and T. Acharya. (2007). "Gesture Recognition: A Survey" IEEE Transactions on systems, Man and Cybernetics, Part C: Applications and reviews, vol. 37 (3), pp. 311- 324, doi: 10.1109/TSMCC.2007.893280.

Simei G. Wysoski, Marcus V. Lamar, Susumu Kuroyanagi, Akira Iwata, (2002). "A Rotation Invariant Approach On Static-Gesture Recognition Using Boundary Histograms And Neural International Journal of Artificial Intelligence & Applications (IJAIA), Vol.3, No.4, July 2012 173
Networks," IEEE Proceedings of the 9th International Conference on Neural Information Processing,

Singapura.

Joseph J. LaViola Jr., (1999). "A Survey of Hand Posture and Gesture Recognition Techniques and Technology", Master Thesis, Science and Technology Center for Computer Graphics and Scientific Visualization, USA. [10] Rafiqul Z. Khan, Noor A. Ibraheem, (2012). "Survey on Gesture Recognition for Hand Image Postures", International Journal of Computer And Information Science, Vol. 5(3), Doi: 10.5539/cis.v5n3p110

Thomas B. Moeslund and Erik Granum, (2001). "A Survey of Computer Vision-Based Human Motion Capture," Elsevier, Computer Vision and Image Understanding, Vol. 81, pp. 231–268.

[11] N. Ibraheem, M. Hasan, R. Khan, P. Mishra, (2012). “comparative study of skin color based segmentation techniques”, Aligarh Muslim University, A.M.U., Aligarh, India.

[12] Mahmoud E., Ayoub A., J¨org A., and Bernd M., (2008). “Hidden Markov Model-Based Isolated and Meaningful Hand Gesture Recognition”, World Academy of Science, Engineering and Technology 41.

[13] E. Stergiopoulou, N. Papamarkos. (2009). “Hand gesture recognition using a neural network shape fitting technique,” Elsevier Engineering Applications of Artificial Intelligence, vol. 22(8), pp.1141–1158, doi: 10.1016/j.engappai.2009.03.008