A

Dissertation

On

A STUDY ON HAND GESTURE RECOGNITION TECHNIQUE

Submitted in Partial fulfilment of the requirement

For the award of the degree of

MASTER OF TECHNOLOGY

In

(Signal Processing and Digital Design)

Submitted by

Rashmi Gautam

Roll No. DTU/M.Tech/174

Under the Guidance of

Assistant Prof. D. K. Vishwakarma

DEPARTMENT OF ELECTRONICS and COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

BAWANA ROAD, DELHI- 110042

June 2012

DECLARATION BY THE CANDIDATE

June 2012

Date: ___________

I hereby declare that the work presented in this dissertation entitled **"A Study on Hand Gesture Recognition Technique"** has been carried out by me under the guidance of Mr. D. K. Vishwakarma, Assistant Professor, Department of Electronics and Communication Engineering, Delhi Technological University, Delhi and hereby submitted for the partial fulfillment for the award of degree of Master of Technology in Signal Processing and Digital Design at Electronics and Communication Department, Delhi Technological University, Delhi.

I further undertake that the matter embodied in this project work has not been submitted for the award of any other degree elsewhere.

Rashmi Gautam
DTU/M.Tech/174
M.Tech (SP&DD)

CERTIFICATE

It is to certify that the above statement made by the candidate is true to the best of my knowledge and belief.

Mr. D. K. Vishwakarma
Assistant Professor
Electronics and Communication Department
Delhi Technological University, Delhi-42

Dated:——————

ACKNOWLEDGEMENT

I feel honored in expressing my profound sense of gratitude and indebtedness to **Mr. D. K. Vishwakarma**, Assistant Professor for their guidance, meticulous efforts, constructive criticism, inspiring encouragement and invaluable co-operation which enabled me to enrich my knowledge and reproduce it in the present form.

I would like to extend my gratefulness to **Prof. Rajiv Kapoor**, Head of Electronics and communication Department, Delhi Technological University, Delhi for his perpetual encouragement, generous help and inspiring guidance.

I am also very thankful to the entire faculty and staff members of Electronic Engineering Department for their direct or indirect help, cooperation, love and affection, which made my stay at Delhi Technological University memorable.

I humbly extend my grateful appreciation to my friends Ms. Pallavi Mathur, Ms. Indu Sirohi and Ms. Jaya Chaudhary for their time to time suggestions and cooperation without which I would not have been able to complete my work.

I would like to thanks the **Almighty,** who has always guided me to work on the right path of the life, and greatest thanks to my family and my classmates who bestowed ability and strength in me to complete this work.

RASHMI GAUTAM

DTU/M.TECH/174

(06/SPD/2K10)

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

ABSTRACT

Gesture recognition is a growing field of research and among various human computer interactions; hand gesture recognition is very popular for interacting between human and machines. It is non verbal way of communication and this research area is full of innovative approaches. This project aims at recognising 40 basic hand gestures. The main features used are centroid in the hand, presence of thumb and number of peaks in the hand gesture. That is the algorithm is based on shape based features by keeping in the mind that shape of human hand is same for all human beings except in some situations. The main shape parameters are thumb and four fingers. The hand gesture of user is captured and stored in the disk. It is then converted to binary image; in this step pre-processing is performed to eliminate noise then features are extracted. On the basis of these features a five bit binary sequence is generated. In this project rule based classification is used. The algorithm is tested for 40 different hand gestures with the database of 200 images taken from a simple camera of 3.2 mega pixels.

Dedication

I dedicate this thesis

To my family, my teachers and my friends for

Supporting me all the way and doing all thee

Wonderful things for me.

Chapter 1
Introduction to Hand Gesture Recognition

1.1 Introduction

Hand gesture recognition is an important area of computer vision and pattern recognition field. Gestures are the way by which we can communicate non verbally. Now day's computers are playing an essential role in human life. So efforts are being given in the direction to make human – computer interaction more user friendly. Keyboards and mouse are the main inputs devices to the computer, with over the time , enhanced input and output devices have been designed in order to ease the interaction between computers and humans [1]

The idea is to make computers understand human language and develop a user friendly human computer interfaces (HCI). Making a computer to understand speech, facial expressions and human gestures are some steps towards it. Gestures are the non-verbally exchanged information. A person can perform innumerable gestures at a time. Since human gestures are perceived through vision, it is a subject of great interest for computer vision researchers.

Gesture recognition is a field, in which we have large number of innovations. We can define the gestures as a physical action, which is used to convey the information. As we know , there are various input – output devices for interacting with the computer, but now days emphasis is given ,how to make human – computer interaction more easy going, and for that purpose hand gesture recognition comes in light. We can use hand as an input device, by making its gesture understandable to computer, and for this purpose, this project aims at recognising the various hand gestures.

1.2 Motivation

Hand gesture recognition is done in this project by aiming forty basic shapes made by hand. Communication in our daily life is generally vocal, but body language has its own significance, like hand gestures, facial expressions and sometimes they play an important role in conveying the information. Hand gesture would be an ideal option for expressing the feelings, or in order to convey something, like representing a number. It has many areas of application like sign languages are used for various purposes, for praying and in case of people who are deaf and dumb, sign language plays an important role. Gestures are the very first form of communication. Even a small child conveys his needs, through making gestures. So this area influenced me very much to carry on the further work related to hand gesture recognition.

This area have vast scope for innovations; the hand gesture can represent different actions and thoughts by making the orientation of fingers in different directions and can further be interfaced with the computer. This type of interaction is the heart of immersive virtual environments. With the development of virtual environment, current user interaction approaches with the use of mouse, keyboard and pen are not sufficient. So gesture recognition has now become a very popular research direction in the field of human-computer interaction, sign language and computer vision. If for a second we consider interaction among human beings and remove ourselves from the world of computers, we can quickly realize that we are utilizing a broad range of gestures in our daily life.

The gestures vary greatly among cultures and context, but still are intimately used in communication. In fact, it is shown that people even gesticulate as much when they talk on telephone, and unable to see each other as in face to face interaction and communication.

This significant use of gestures as a mode of interaction in our daily life motivates the use of gestural interface in this modern era. It is the demand of available advanced technology to recognize, classify and interpret various different hand gestures and use them in a wide range of application through computer vision. So I have founded this area of research with wide variety of applications, each having the innovative areas of application, among them sign language recognition is very interesting.

1.3 Gesture Analysis

It is not easy to carry with particular useful definition of gestures because of its wide variety of applications and a proper statement can only specify a particular domain of gestures. Many people have defined the definition of gestures in their own terms, so the definition is still arbitrary. In a simple way we can define gestures as a physical activity that conveys some message, whether it can be facial expressions, your body language, hand movements etc. Humans are so inherited with all these things naturally, for example when we talk on the phone, one can easily observe the movement of the hand, and it's quite natural.

Gesture can be defined as the motion of the body in order to communicate with others [2], and in order to accomplish the communication between sender and receiver, they must possess same set of gestures. There are mainly two kinds of gestures static and dynamic. As the name suggest ,static gestures are static with respect to time, the stop sign signal is an example of static gesture, whereas a waving hand means goodbye, is an example of dynamic gesture. In order to interpret the full meaning of a message, we must correctly interpret the static and dynamic gesture within the time. This is what we called as gesture recognition process.

As per the context of my project, gestures (particularly hand gestures) are defined as the movement that conveys some information. Hand gesture recognition is very interesting area, as we can communicate with deaf and dumb people. Hand gesture analysis can be divided into two main streams, model based and view based approach [3].

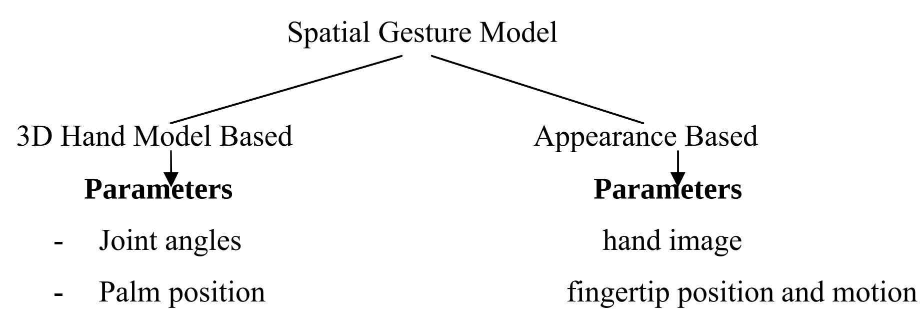


Figure 1.1 Spatial gesture model [3]

Model based approach uses the 3D hand model of the image, then this model is projected into the image and an error function is calculated. The aim is to minimize the error function, so that the best matching can be obtained. But in this type of approach model initialization is a challenge. So we opt for the view based approach.

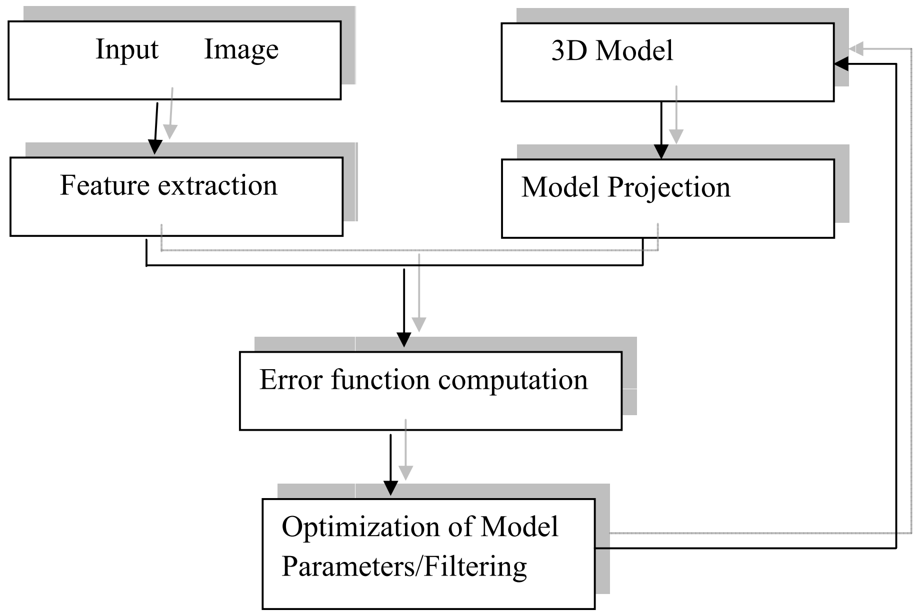


Figure 1.2 Components of Model Based System [4]

The gestures are modelled by relating the appearance of any gesture to the appearance of the set of predefined, template gestures. These methods are usually described as bottom-up, because low-level features are used to generate high level information. The main problem is segmenting the hand from the background scene and which features to extract from the segmented region. Generally, the classifier learns from a set of training images and assigns the input to one of the possible categories [5].

The majority of appearance-based models however, use parameters derived from images in the templates. Parameters can be: contours and edges, image moments, and image Eigen vectors to mention a few. Many of these parameters are also used as features in the analysis of gestures. Another group of models uses fingertip positions as parameters. This approach follows the assumption that the position of fingertips in the human hand, relative to the palm is almost always sufficient to differentiate a finite number of different gestures. Some of the restrictions that validate the assumption were noted by Lee and Kunii [6, 4]: the palm must be assumed to be rigid and the finger must have a limited number of Degree of Freedoms.

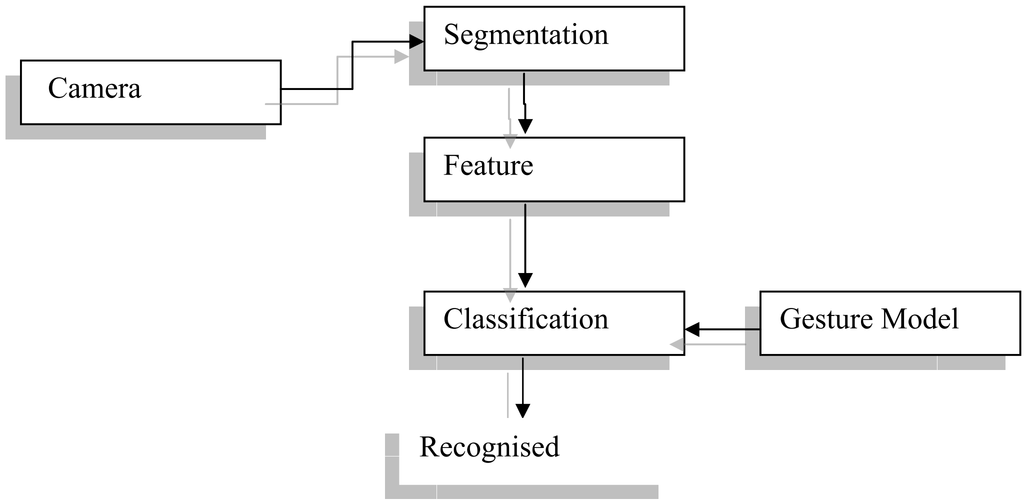


Figure 1.3 Appearances Based Modelling

In gesture analysis, there are mainly three steps to follow

- Hand localization
- Hand feature extraction
- Hand model parameter computation of features

There are two types of classification

- Rule Based Classification
- Learning Based Classification

Rule-based approaches use a set of manually encoded rules between feature inputs. Given an input gesture a set of features are extracted and compared to the encoded rules, the rule that matches the input is outputted as the gesture. A major problem with rule-based approaches is that they rely on the ability of a human to encode rules. In many cases the appropriate rules may not be intuitive especially when dealing with high-dimensional feature sets.

Given the limitations of humans to find relations between features in a high-dimensional feature space, many have turned to machine learning techniques to find mappings between gestures and high-dimensional sets of features.

A popular approach is to treat a gesture as an output of a stochastic process. Of this class of approaches, (Hidden Markov Models) [7] have received the most attention. Their introduction to the gesture recognition community was inspired by the success that HMM attained in speech recognition. An HMM consists of a number of hidden states, each with a probability of transitioning from itself to another state. The transitioning probabilities are modelled as nth order Markov processes (i.e. the probability to transitioning to a new state only depends on the n previous states visited).

For computational reasons most approaches assume a first-order Markov process. An important feature of the topology is that states are allowed to transition themselves.
Alternatively, models based on (finite state machines) have been used to capture the sequential nature of gestures by requiring a series of states estimated from visual data (i.e. features) to match in order, to a learned model of ordered states [8].

The advantage of this representation over HMMs is that it does not require a large set of data in order train the models. The disadvantage of the approach is that the associated training relies on finding distinct clusters in the feature space which has proven to be a non-trivial automatic task.

Finally, temporal extensions to neural networks, (time-delay neural networks), have been used to learn mappings between training exemplars (2D or 3D features extracted from video) and gestures. Like HMMs, a major difficulty of this approach is that the selection of the topology is based on educated guesses and intuition. Also, much care must be taken during the training stage; otherwise the network may over fit on the training gesture set and not generalizes well to variations of gestures outside the training set.

1.4 Organization of the thesis

The remainder part of this thesis is organized in the following chapters:

Chapter 2: Literature review

This section gives the review of the work done earlier in the field of hand gesture recognition and thus forms the basis to carry on the further work in the same area. Literature review provides us the useful database on previously attempted techniques and algorithm.

Chapter 3: The Proposed algorithm

This section provides the details of the proposed algorithm, its implementation and the results.

Chapter 4: Discussion and Results

In this section, results of simulation are discussed.

Chapter 5: Applications of hand gesture recognition

In this chapter, the applications of hand gesture recognition are discussed.

Chapter 6: Conclusion and future scope

In this section the conclusion of the thesis work and the future scope of the work are presented.

References: This section gives the reference details of the thesis.

APPENDIX A: Introduction to Image Processing in MATLAB

Chapter 2
Literature Review

Human hand is one of the promising methods to provide human computer interaction by using hand as an input in comparison to other input devices like mouse, keyboard. Human gesture recognition includes mainly three steps. The first one is image segmentation (pre-processing), the other two steps are feature extraction and classification. The most important thing in human gesture recognition is the selection of good features. In this chapter we present the review of hand gesture recognition methods.

New researches in the field of hand gesture recognition [9-12] proved its importance in human computer interaction (HCI). The whole survey on gesture recognition has been provided in [13]. The two main approaches for recognising hand gesture for the purpose of human computer interaction are glove based techniques and vision based techniques. The glove based techniques are quite cumbersome as it required many connecting cables to connect with the computer and on the other hand vision based techniques are quite feasible to handle with them [14].

The very first gestures were applied to the computer interaction by Ivan Sutherland[15] who demonstrated Sketchpad, an early form of stroke-based gestures using a light pen to manipulate graphical objects on a tablet display. This form of gesturing has since received widespread acceptance in the human-computer interaction (HCI) community [16].

A vision-based system able to recognize 14 gestures in real time to manipulate windows and objects within a graphical interface was developed by C.W. Ng in [17]. Abe [18] proposed a system which recognizes hand gestures through the detection of the bending of the hand's five fingers, based on image-property analysis [17]. Hasanuzzaman [19] presented a real-time hand gesture recognition system using skin colour segmentation and multiple-feature based template-matching techniques. In their method, the three largest skin-like regions are segmented from the input images by skin colour segmentation technique from YIQ colour space and they are compared for feature based template matching using a combination of two features: correlation coefficient and minimum (Manhattan distance) distance qualifier [16].

Ramamoorthy used HMM based real time dynamic gesture recognition system which uses both the temporal and shape characteristics of the gesture for recognition. Use of both hand shape and motion pattern is a novel feature of this work [16].

a **b**

Figure 2.1 Examples of data glove based and vision based approaches [20]

a. Vision based(image gallery); b. Glove based[21]

The following table shows the hand gesture recognition system classification.

Table 2.1 Hand gesture recognition system classification [20]

Category	Type
Application	Sign language , Robot control,Games,Tracking Gestures
Motion	Static , Dynamic
Image acquired Data	Camera , videos
Data dimensions	2D, 3D
No of hands used	One hand, Two hands
Input features	3D hand model, low level features, Appearance based
Gesture modality	Communicative, manipulative

2.1 Approaches for hand gesture recognition

There are three types of gesture recognition systems namely:

1. Glove Based
2. Vision Based
3. Low level features based

Model based approaches: Many models have already been applied to analysis, model and represent the hand shape, which gives us a description and make a wide range of human hand to be represented, and a large database for storing the extracted shape characteristics is needed as well[20]. But 3D hand model have many degrees of freedom, therefore feature extraction can be a difficult task

Appearance based approaches: Same as view based approach; Appearance based approaches use image features to model the visual appearance of the hand and compare these parameters with the extracted image features from the video input. These approaches have the advantage of real time performance due to the easier 2 D image features that are employed.

Low Level Features based Approaches: This approach is based on extracting the low level features which includes: the centroid of hand, the bounding box, edges, silhouettes and histograms. These features are easy to extract and easy to carry the further work with them

Other than the above discussed approaches, there are two main basic approaches in static gesture recognition process:

Top – down approach: where a previously created model of collected information's about hand configurations is rendered to some feature in the image co-ordinates. Comparing the likelihood of the rendered image with the real gesture image is then used to decide whether the gesture of the real image corresponds to the rendered one.

Bottom-down approach: which extracts features from an input image and uses them to query images from a database, where the result is based on a similarity measurement of the database image features and the input features. The disadvantage of the previous approach is that it seems to use a high computational effort in order to achieve robust recognition. The second approach however requires an adequate Pre-processing in order to achieve a reliable segmentation.

2.2 Literature review on Segmentation

Segmentation phase plays an important role in the system recognition process. Perfect segmentation effects on the accuracy of the recognition system [22]. For any segmentation process, some image processing operations are required for hand shape detection [23, 22].

There are two ways in which we can define the image segmentation algorithm as according to grey level properties [24].

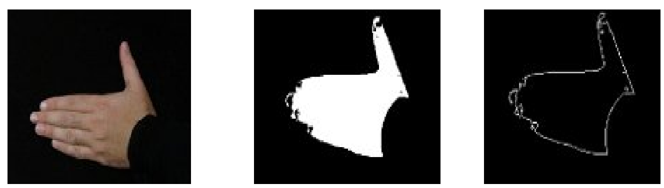
1. Discontinuity 2. Similarity

Discontinuity refers to the huge variation in contrast and whereas similarity refers the sameness in the neighbouring pixels. After capturing the image from the camera, the very first step is segmentation that is isolating the hand region from the captured image [22].

The methods for object segmentation mainly depends on the colour model that can be extracted from the existence RGB colour model which could be HSV colour model or YCbCr colour space [25]; which deals with the pigment of the skin of the human hand, the important property of this colour space is that the human different ethics group can be recognized according to their skin pigment concentration which can be differentiated according to some skin colour saturation [22].

Then, the hand area is isolated from the input hand gesture with some predefined value. Some normalization for the segmented image will be required for obtaining the hand gestures database which would be invariant against different variations like translation, scaling and rotation [22]. The number of samples and accuracy are directly proportional whereas speed and no of samples are inversely proportional [22]. So a compromise must be made between these two parameters.

The HSV colour model is used [22] to find out the skin coloured like hand region by estimating the parameter values for skin pigment. The Laplacian filter is used for the detection of edges. YCbCr colours model has been used to segment the hand from the rest of the image in [25]. Segmentation can also be done by the method of clustering in which the grouping of image pixels is done [24] .



a b c

Figure 2.2 Segmentation process [22]

a. Input image; b. segmented image; c. edge detection

In figure 2.2, the segmentation process described in [22] is shown, in this HSV colour model is used and laplacian filter is used for the detection of edges.

a b

Figure 2.3 Segmentation process [25]

a. Input image; b. segmented mage

a b

Figure 2.4 Hand segmentation [26]

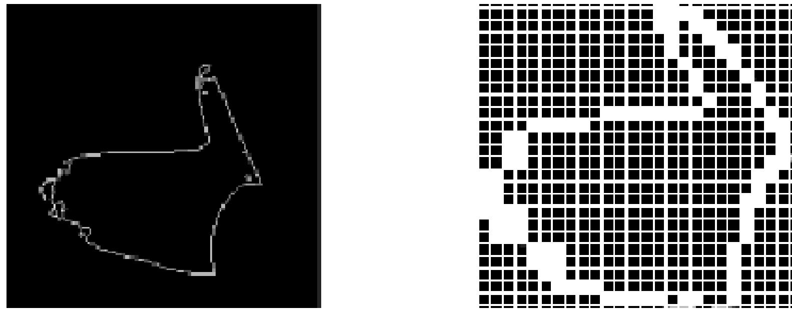
a. Input image ; b. segmented image

2.3 Literature Review on Feature Extraction

Feature extraction is the second step in the hand gesture recognition process. Features represent the useful information that can be extracted from the segmented hand image so that the machine can understand the significant meaning of that posture. The mathematical representation of the features will be obtained from the hand segmented image.

Feature vector plays an important role in detection, [22] has extracted the feature by taking segmented hand object and divided it into fixed block of size 5x5.This will produce 625 feature vectors. Self-Growing and Self-Organized Neural Gas (SGONG) network has been applied in [25].

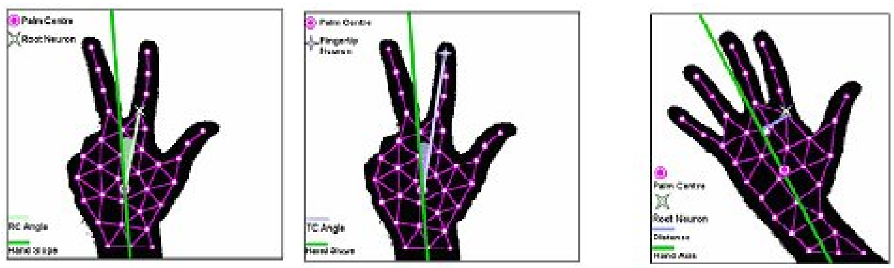
By this the hand shape is extracted and three features are determined namely Palm region, Palm center, and Hand slope. RC angle is the angle between the finger root and hand center. TC angle is the angle between the joints of finger tip and hand center. In [27] divided the segmented hand containing region into the 12 blocks with gray scale feature vector. The average value of the brightness of the pixels in the segmented hand image is represented in each grid cell by mean value.



a b

Figure 2.5 Feature vector representations [22]

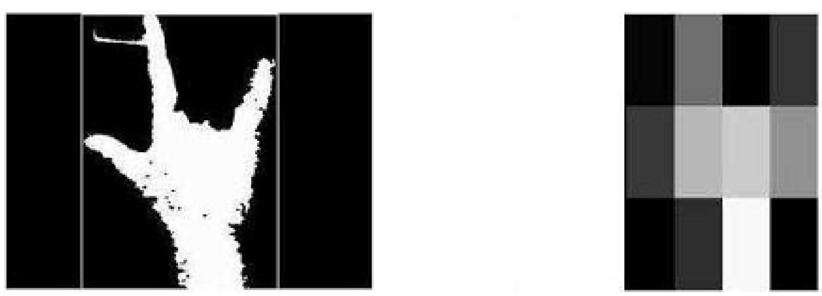
a. Segmented hand; b. feature vector representation



a b c

Figure 2.6 Feature vector representations [25]

a. RC angle ; b. TC angle ; c. Distance from the palm center



a b

Figure 2.7 Feature vector [27]

a. Normalized hand; b. gray scale image partitioned into 12 blocks feature vector

2.4 Literature Review on Recognition

After segmentation and feature extraction, recognition or classification is the last step in the process of hand gesture recognition. There are two ways of doing hand gesture classification as given in [23].

1. Rule based approach
2. Machine learning based approach

In rule based approaches, the rule is manually created on the basis of feature vectors, and those who get matched with the rule, can be recognised as the final result. But the disadvantage of this approach is human handling capacity of creating so many rules. That is, manually we can't create a vast variety of rules in order to get perfect recognition [23].

Some of the machine based learning classification methods are discussed below:

Neural networks: This method is based on modelling of the human nervous system element called neuron and its interaction with the other neurons to transfer the information. Each node consists of an input function which computes the weighted sum and the activation function to generate the response based on the weighted sum. There are two types of NN, feed forward and recurrent. The methods based on this approach deal with the problem of heavy training and computation cost involved offline for training. Also for complex systems, such a model could be very complex. Also addition of each new gesture/posture requires complete retraining of the network.

Hidden markov model: The Hidden Markov Model (HMM) classifiers belong to the class of trainable classifiers. It represents a statistical model, in which the most probable matching gesture class is determined for a given feature vector, based on the training data. This method has been widely exploited for temporal gesture recognition. An HMM consists of states and state transitions with observation probabilities. For watch gesture a separate HMM is trained and the recognition of the gesture is based on the generation of maximum probability by a particular HMM. This method also suffers from training time involved and complex working nature as the results are unpredicted because of the hidden nature. In order to train the HMM, a Baum-Welch re-estimation algorithm, which adapts the internal states of the HMM according to some feedback concerning the accuracy, was used.

K – Nearest neighbours: This classification method uses the feature-vectors gathered in the training to find the k nearest neighbours in an n-dimensional space. The training mainly consists of the extraction of (possible good discriminable) features from training images, which are then stored for later classification. Due to the use of distance measuring such as the Euclidian or Manhattan distance, the algorithm performs relatively slowly in higher dimensional spaces or if there are many reference features.

Chapter 3

The Proposed Algorithm

The proposed algorithm has three main steps:

1. Image segmentation
2. Feature extraction
3. Classification

The flow chart of the proposed algorithm shown below:

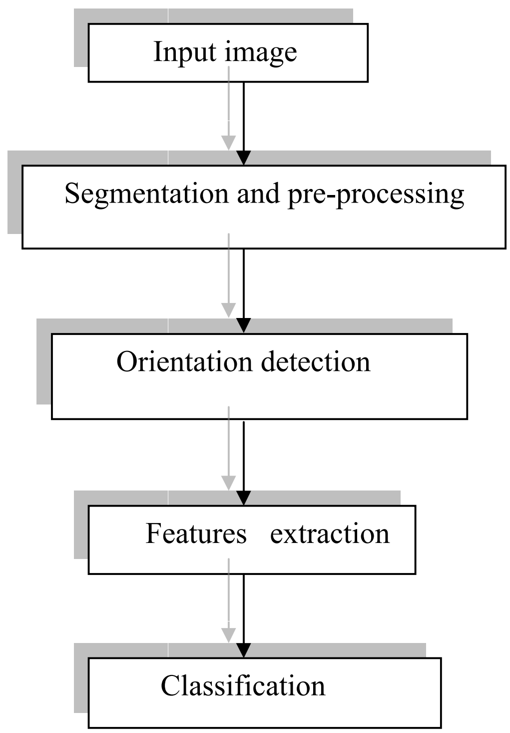


Figure 3.1 Flowchart of algorithm

3.1 Pre-processing and segmentation

Image pre-processing is the very first step, pre-processing of image is done for enhancement of image and also for getting results with minimum error. In the proposed algorithm the image is pre-processed using YCbCr colour model. The various colour models that can be used for segmentation purpose are RGB, HSV (hue, saturation, and value), and CIE-Lab.

3.1.1 Colour Models:

RGB: Three primary colours red(R), green (G), and blue (B) are used. The main advantage of this colour space is its simplicity. However, it is not perceptually uniform. It does not separate luminance and chrominance, and the R, G, and B components are highly correlated.

HSV: Hue, saturation, value. It expresses Hue with dominant colour (such as red, green, purple and yellow) of an area. Saturation measures the colourfulness of an area in proportion to its brightness. The “intensity”, “lightness”, or “Values” is related to the colour luminance. This model discriminates luminance from chrominance. This is a more intuitive method for describing colours, and because the intensity is independent of the colour information this is very useful model for computer vision. This model gives poor result where the brightness is very low.

CIE- Lab: It defined by the International Commission on Illumination. It separates a luminance variable L from two perceptually uniform chromaticity variable (a,b).

YCbCr colour space: In YCbCr colour space □ Y represents brightness, Cb and Cr represent chromaticity respectively. The YCbCr colour space acts as the mapping space of the skin colour distribution statistics. The main advantage of using this colour space is that it is invariant by the brightness variation. Our eyes are very sensitive to the changes of brightness in comparison to chromaticness, therefore, the colour loss caused by reducing the bandwidth will have very minor effect on human eyes. On the other hand, the YCbCr colour space is two-dimensional. It can limit the skin colour distribution area more effectively. YCbCr colour space reduces the storage and bandwidth. Another big advantage of using YCbCr colour space is that it can separate luminance component from the colour space in HIS format. The YCbCr colour space can be obtained from the RGB linear transformation**.** In the proposed algorithm the input image is converted into YCbCr image, because RGB colour space is very sensitive to lighting conditions. So the RGB image is converted into the YCbCr colour format .The transformation formula for RGB to YCbCr is given below:

$$Y = 0.2990 \times R + 0.5870 \times G + 0.1140 \times B \qquad 3.1$$

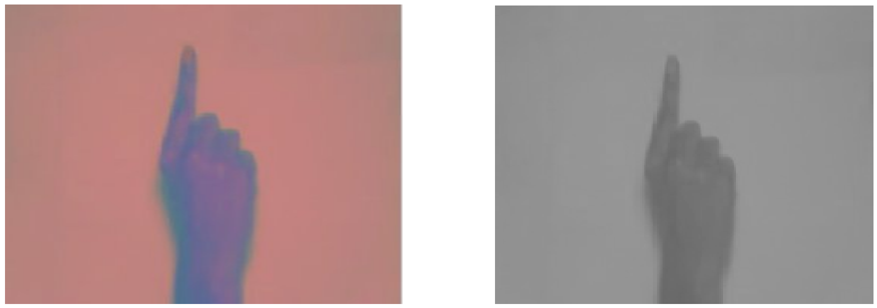
$$Cb = -0.1687 \times R - 0.3313 \times 0.5000 \times B + 128 \qquad 3.2$$

$$Cr = 0.5000 \times R - 0.4187 \times G - 0.0813 \times B + 128 \qquad 3.3$$

After converting the input image into YCbCr, now it is being converted into binary image. The thresholding is done on the basis of otsu's method [41] using the inbuilt function graythresh in matlab.

The background noise is removed, that is all small connected components are being removed, small components are those who has fewer than p pixel. After this step also one more is needed to be done that is the filling of holes in the binary image. After getting converted into binary image, the small objects were being removed, so now we are having the finest image that is morphologically open binary image. After performing the hand segmentation, now we will locate the boundary of the hand region in the image, this boundary will further help in calculating the area of the hand region in the image. The bounding box is determined by scanning the image from top to bottom and from left to right.

Figure 3.2 Input Image

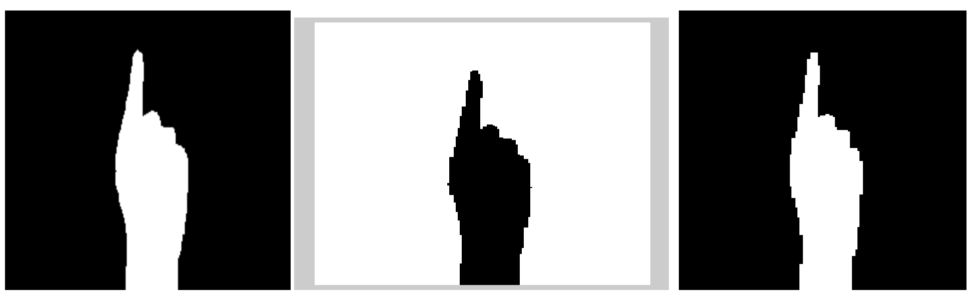


a b

Figure 3.3 Segmentation Process

a. YCbCr image; b. Gray scale image

Comments: In figure 3.3 (a) the input image (RGB) is converted into YCbCr colour space. The RGB colour image is more sensitive to the variations due to light, so we need to convert it into YCbCr colour space.



a b c

Figure 3.4 Removing noise

a. Binary image with noise; b. inverted binary image with noise; c. without noise

Comments: After converting into gray scale image, now it is being converted into binary image as shown in figure 3.4. But still there is some background noise in the image as it clear from the figure 3.4(b), so noise is removed in figure 3.4(c), by removing the connected components that have fewer than P pixels.

In our experiment, to remove background noise P is taken as 50 and to remove foreground noise P is taken as 5000 pixels

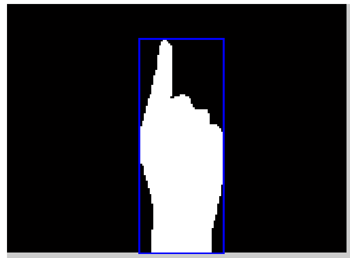


Figure 3.5 Hand containing box

Comment: In figure 3.5, the bounding box is determined by scanning the image from left to right and from top to bottom. The left uppermost corner is the starting point, from where the scanning will start.

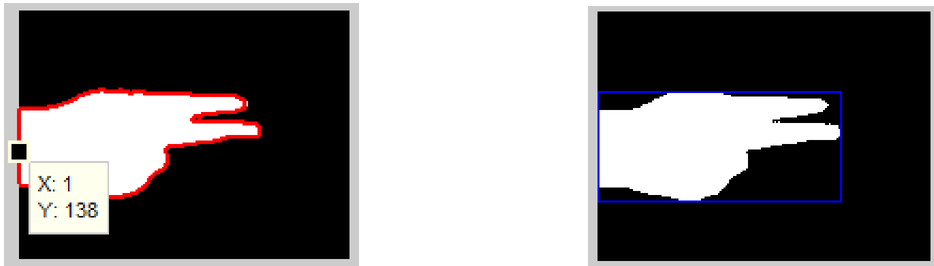
3.2 Orientation of image

This is the second step in the implemented algorithm, as in our experiment we have considered only two types of orientation, horizontal and vertical. So initially we have to find out the orientation of the image, either the image is vertical or horizontal.

In order to proceed we will find out the length and width ratio of the bounding box containing the hand image, in the previous step. If the image is horizontal then its width will be more than the length, i.e. the length – width ratio will be less than unity. On the other hand if the width is less than the length, then automatically the image is vertical that is length – width ratio is greater than unity.

This is the one way in which we can check the orientation of the image. The another way in which we can check for orientation is , tracing the boundary of the image which we obtained in the previous step and check for the values , if X=1 for many values of Y(in increasing order) then the image is said to be in horizontal position. On the other side if we get the maximum value of Y with increasing value of X, then it is said to be in vertical position. It is shown in figure 3.6 and 3.7

.



a b

Figure 3.6 Horizontal image

a. Boundary contour; b. bounding box containing hand

Comments: In figure 3.6, showing the horizontal orientation as it has the value X=1 with increasing values of Y, and also the width is more than the length that is length – width ratio is less than unity. So both conditions are satisfied.

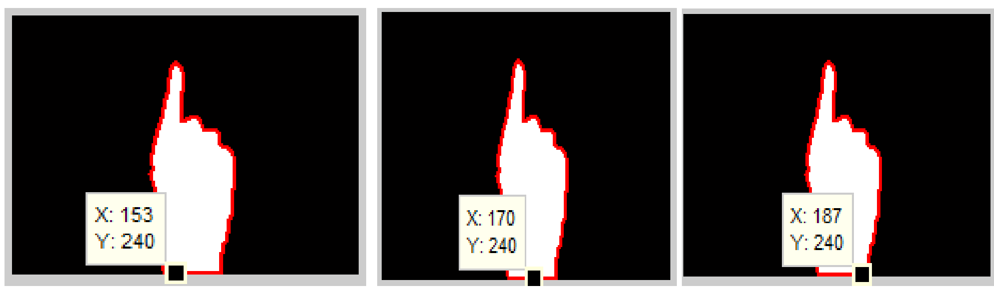


Figure 3.7 Vertical orientation

Comments: in figure 3.7, the vertical orientation is shown, as it is clear from the figure that X is increasing with maximum value of Y. In our experiment the size of the image is 320 x 240. Also we can notice that the length – width ratio is greater than unity as length is more than width. So both conditions are satisfied to get the correct orientation.

3.3 Features Extractions

This is the third step in the implemented algorithm; features extraction plays an important role in the whole process. It is only the features which decide the accuracy of the algorithm. Initially many other techniques were used for the feature extraction, which includes colour, texture etc. but these feature may vary from person to person as each person can have different tone of colour , and it may also be affected by varying lightning conditions.

In our experiment we are working on vision based hand gesture recognition techniques, which mainly focus on the shape of the hand. The shape of hand for every person will remain almost same, including one thumb and four fingers.

We mainly focus on the presence of thumb, and number of peaks, the gesture has. In [41], three features namely compactness, area and radial distance are extracted; Compactness is defined as the ratio of squared perimeter to area of the hand shape. So this feature can be same for two hands, hence they will be recognised as same gesture. That is the limitation of using compactness as a feature. But we are using a five bit pattern for the recognition of hand gesture, which uniquely identifies the hand gesture.

The database includes 40 images with different hand gestures, as shown in figure 3.8, 3.9. All the images are with uniform background and only horizontal and vertical orientations are considered, for other orientations it will give error. To take images we have used a simple camera with 3.2 mega pixels.

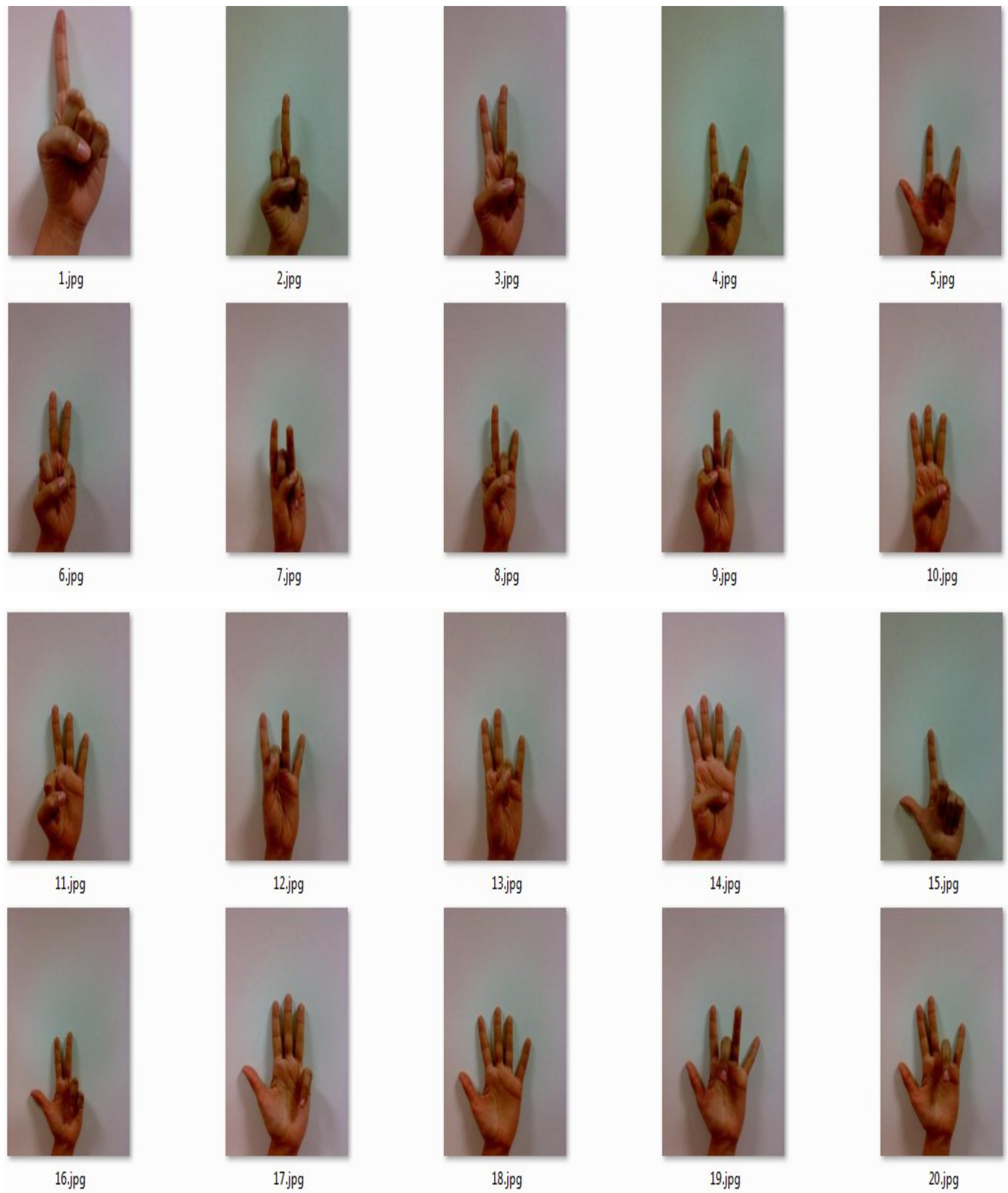


Figure 3.8 Sample hand gestures

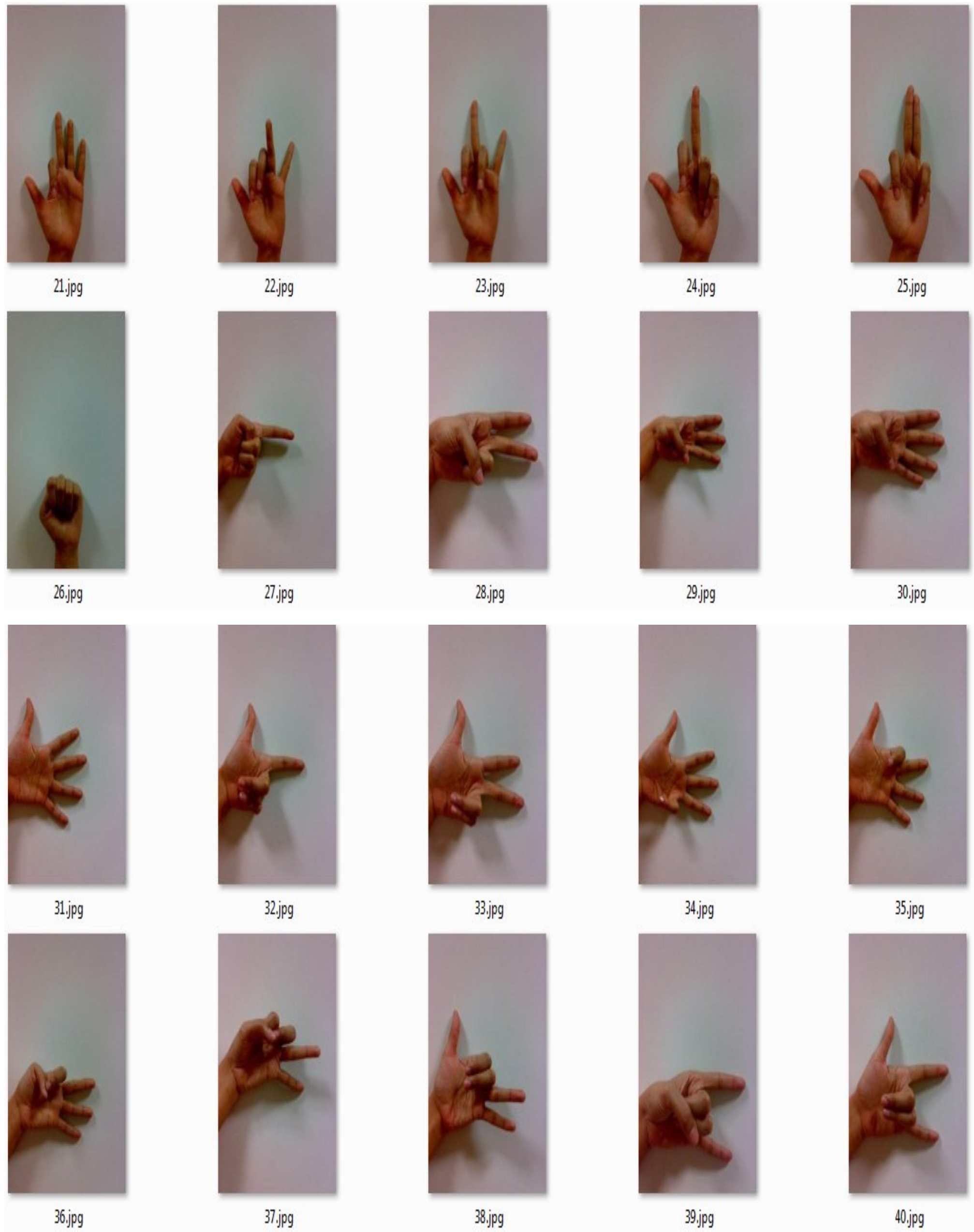
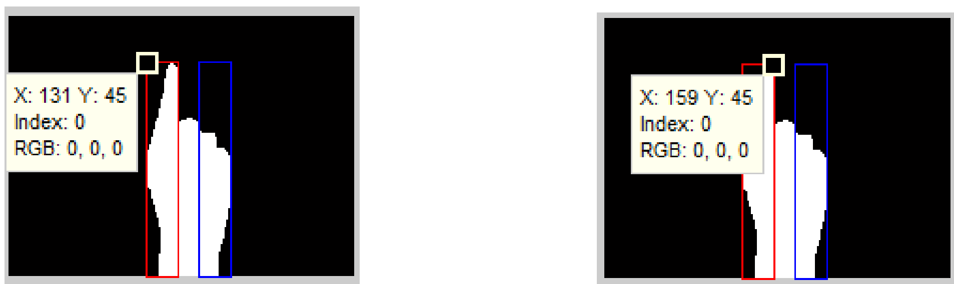


Figure 3.9 Sample hand gestures

Now we will find out features namely : presence of thumb , centroid of the image , finding peaks in the hand image.

3.3.1 Presence of Thumb

Finding the presence of thumb is an important feature. It can be on either side that is either on left side or on right side of the hand. So in order to detect the presence of the thumb, we will take the bounding box, which was calculated in the pre-processing step. Firstly we will check for the orientation of the image as discussed previously, then according to the orientation, two bounding box will be drawn on both side of the image as shown below.



a b

Figure 3.10 Thumb Detection

Comment: In figure 3.10 two bounding box are created on both sides of the image. From our experiment the width came out to be 28 pixels from both sides.

Now the area of hand image is calculated, which is the sum of all white pixels in that area. This is obvious that sum count of white pixels for thumb region will be very less when compared with finger region. By observing, we get that thumb percent is approximately 6.9% of total area of the hand (white pixels). So we apply the condition that if sum of white pixels in both boxes (red and blue box) individually less than 6.9%, then thumb will be present in that particular box. If the percent increases from 6.9%, then thumb is not present. This approximation of 6.9% is estimated after testing many images.

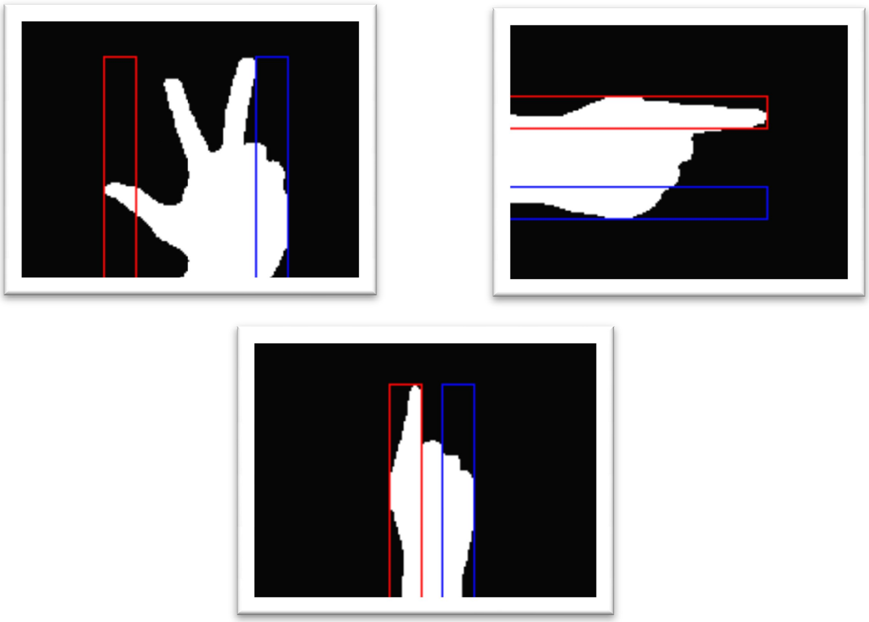


Figure 3. 11 Thumb detection

Comments: In figure 3.11, thumb detection box is shown for horizontally and vertically oriented images; from the figure it is clear that the white pixels count for the thumb region is very less than the white pixels count for finger region.

Now we proceed to find the second feature that is, the centroid of the hand image.

3.3.2 Centroid of the Hand Image

We are dividing the hand region into two half, depending upon the orientation of the hand image, the first half consist only thumb and the second half consist of fingers region (four fingers). The centroid will lie within the hand image therefore by finding centroid we can easily separate fingers and thumb region within the hand image.

The centroid [47] of an image can be found by calculating the image moment using the formula written below,

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y) \qquad 3.4$$

Where I(x,y) defines the intensity at coordinate (x,y), The coordinate of the centroid $(\bar{x}, \bar{y})$ is found by using[41]

$$\bar{x} = \frac{M_{10}}{M_{00}} \qquad 3.5$$

$$\bar{Y} = \frac{M_{01}}{M_{00}} \qquad 3.6$$

3.3.3 Peaks in Hand image

In this step we denote tip of the finger as peak. For getting the total number of finger raised in hand gesture we need to process only finger region of the hand that we have got in previous step by computing centroid.

To precede this task we trace the entire boundary matrices of hand. Vertical hand image and horizontal hand image have been processed in different manner. For vertical hand image, we only consider the y coordinates of the boundary matrices. When we get the values of y coordinates of boundaries starts increasing after the sharp decrement in the y-boundaries value.

We consider this indication as tip of the finger and we fix it as a peak value or a peak. Similarly for horizontal image, we consider the x coordinate of the boundary matrices. This time only the x coordinates of the boundary matrices is traced. When we get the x coordinate of boundaries starts decreasing after the continuous increment we mark this point as a tip of the finger in horizontal hand and set it as peak. In this way we found the tip of all raised and folded fingers in the image, but we need to classify significant peaks and insignificant peaks among them. For this we need to proceed to the next step to calculate the Euclidean distance.

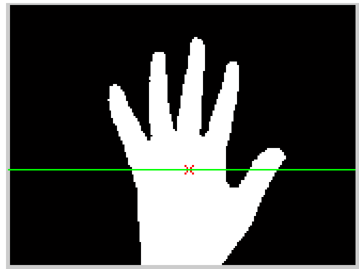
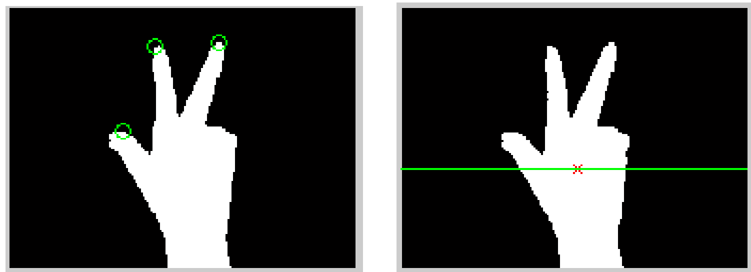


Figure 3.12 Centroid of image

Now in order to get the proper peaks, we need to calculate the distance between peaks and centroid of the image. This distance is calculated using Euclidian distance [41] formula given by,

$$E.D(p,q) = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2} \qquad 3.7$$

Where p includes all the boundary points and q is the reference point, in this version, the reference point is taken at the centroid, and any boundary points left of the centroid are not included in the profile. The focus is now on the right half of the hand, where we want to extract the number of fingers found in the image. In this step we also make sure that any discontinuous regions are discarded. To extract the number of fingers, we define a threshold line at seventy five percent of the maximum distance away from the centroid of the hand. We count the number of points that this line intersects with the radial distance function. The number of intersections determines the number of fingers.
There may be some peaks detected which do not actually represent the tip of the raised fingers, but the tip of folded fingers. We can get rid of these kinds of insignificant peaks by computing the maximum peak. Putting the threshold at 75% of the maximum peak value, we can choose only those significant peaks whose values are more than this threshold value as these peaks represents the raised finger in hand gesture. Other peaks that are detected but do not intersect or fall above this threshold line would be treated as insignificant peak or folded fingers.



a b

Figure 3.13 Showing numbers of peaks for vertical orientation

Comments: In figure 3.13 (a), it is showing the number of peaks the gesture is having, but on the same side it is also showing the insignificant peaks. In our experiment insignificant peaks are defined as those peaks which are not at a distance greater than 75% from the centroid. This distance is Euclidian distance. The centroid is shown in figure 3.13(b).

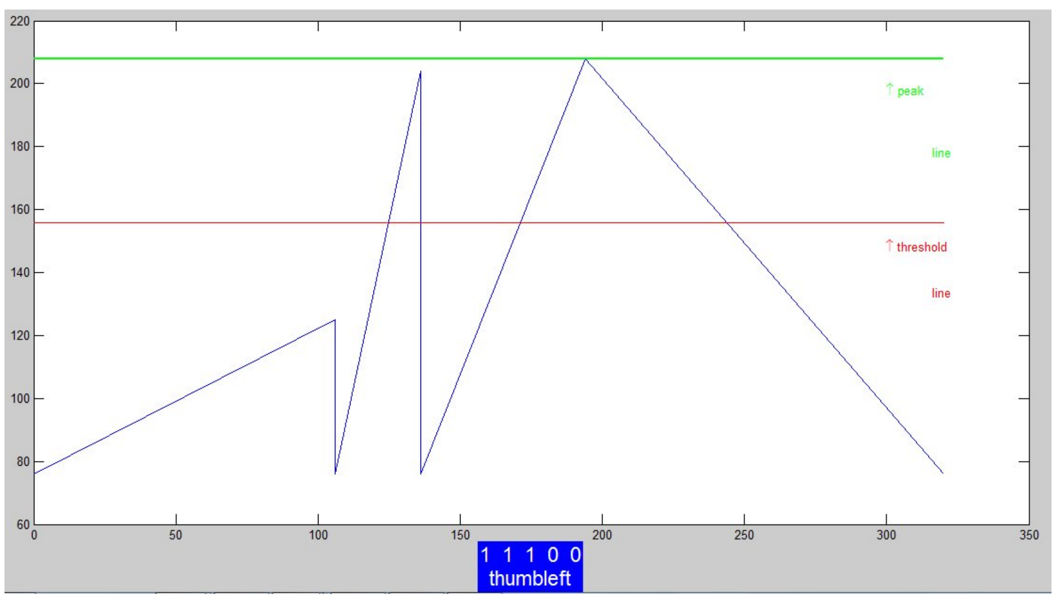


Figure 3.14 Peaks Histogram

Comment: In figure 3.14, plot of peaks histogram is shown, the green line signifies the maximum distance of the peak from the centroid, whereas red line indicates the threshold line, which is taken as the 75% of the maximum peak distance. So instead of having three peaks, only two out of them are significant which are above the threshold line (red line). And the five bit pattern will be discussed in the next section.

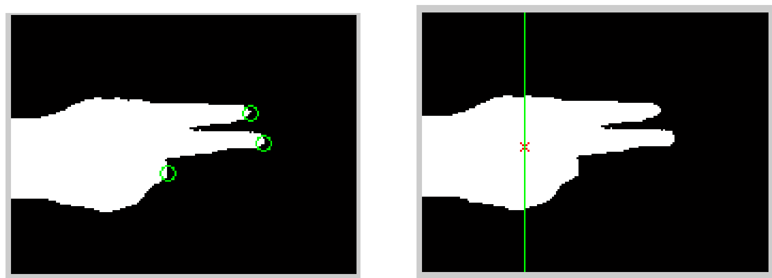


Figure 3.15 Showing peaks for horizontal orientation

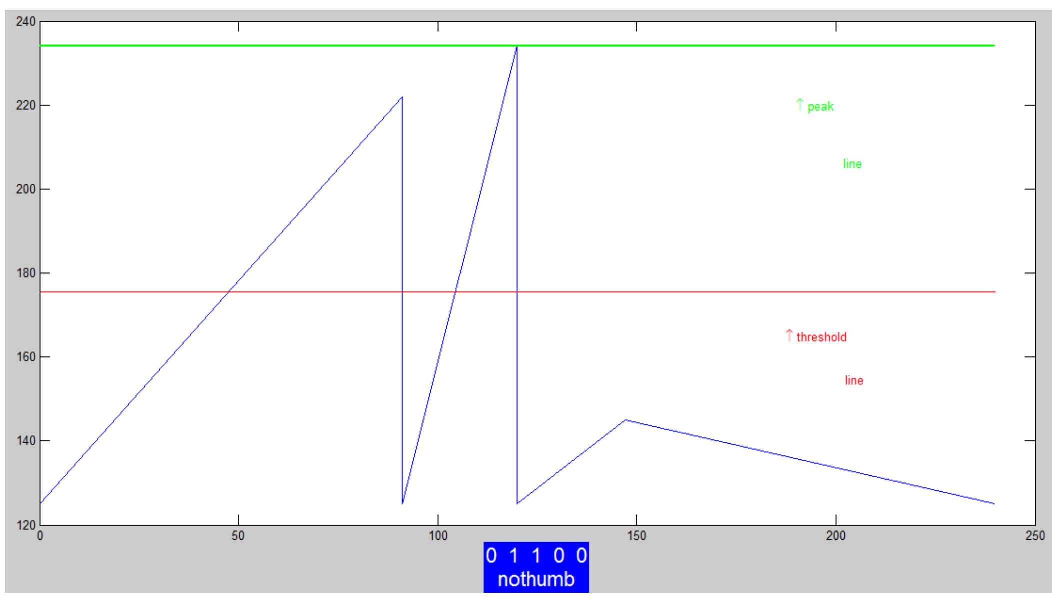


Figure 3.16 Peak Histogram

Comment: In figure 3.15, the peaks are shown for horizontal orientation of image, again it is showing three peaks, but only two of them are significant. This can be deduced from figure 3.16, because only two peaks are there above red line that is threshold line, and one peak is below the red line.

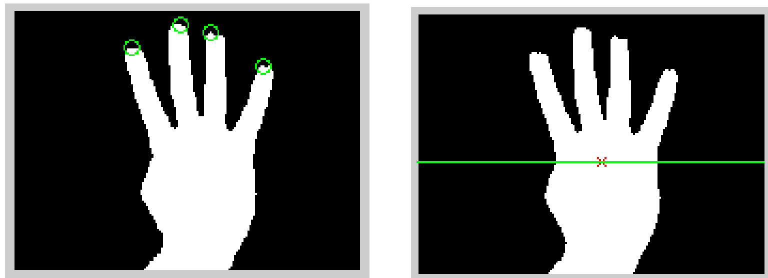


Figure 3.17 Showing Peaks

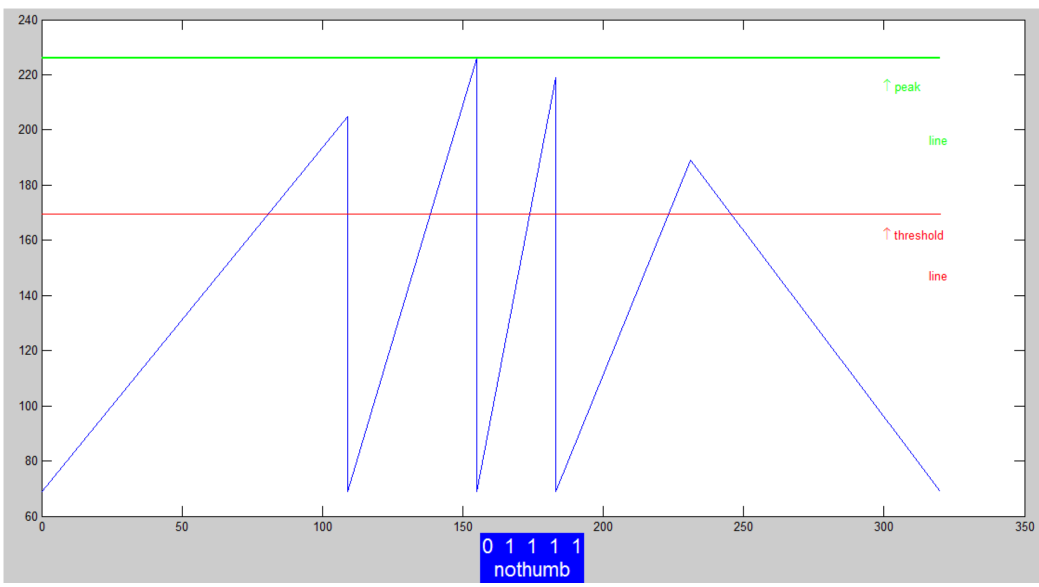


Figure 3.18 Peaks Histogram

Comment: In figure 3.17, the total number of peaks is four, and all of them are significant because all four peaks are lying above the threshold line, this is clear from figure 3.18.

3.3 Classification and Bit Pattern generation

There are many ways for classification, the two main are: Machine Based Learning and Rule Based learning. In machine based learning, most of them are based on statistical modelling such as principal component analysis, finite state machine, and hidden markov models. Some other techniques rely on neural network, in which back propagation algorithm is used. Kalman filtering and Gaussian distribution is also used for gesture recognition.

In our project, rule based classification is used. The classification is based on the features extracted earlier, on the basis of these features five bit coded sequence is generated, which uniquely identifies the gesture and support for human computer interaction.. Peak-Centroid plots are shown in Figures 3.12, 3.14, 3.16.The significant peaks we identified in previous step is encoded as '1' and insignificant peaks is encoded as '0' based on the intersection status of various finger tips to threshold line. Leftmost bit in the 5 bit binary sequence is reserved for status of thumb in hand image. If thumb is present, leftmost bit will be 1 otherwise 0.

The advantages of using rule based classification is its simplicity but on the other hand its disadvantage lies in the fact that manually it is impossible to create many rules for recognition.

Chapter 4
Discussion and Results

The proposed algorithm is applied on 200 images with 40 different gestures. With the help of defined features and five bit pattern encoded sequence, we can successfully recognize the 40 different hand gesture patterns. The leftmost bit in the encoded five bit sequence, represent thumb, the special feature that we have used in our proposed algorithm. Furthermore these recognized gestures can be used for human computer interaction.

Figure 4.1 shows the graphical user interface of the final output, in which the test image is compared with the reference image. In table 4.1, the experimental data is shown, starting with the input hand gesture image, along with the centroid plot, number of peaks detected and the peak histogram. Also the encoded five bit sequence is also shown there. Under single orientation, either vertical or horizontal, the bit sequence is unique.

This may be able to happen that two hand gestures can have the same encoded bit of sequence, but having different orientation. This can be solved internally in the code on the basis of their orientation category so that these two same encoded bits which belong to different class of orientation can be assigned to perform different key press events. So finding the orientation prior to features extraction is an important step. Only vertical and horizontal orientation is considered in our proposed algorithm, with uniform background images.

Table 4.2 shows the result of 200 images tested through this algorithm. Out of 200 images, it has correctly identified 187 images and falsely identified the remaining 13 cases, gives the success rate of 93.5% approximately with average computation time of 2 second for recognizing single image in image sequence. The algorithm is based on simple shape based feature calculation which provides us with the comfort of implementation.

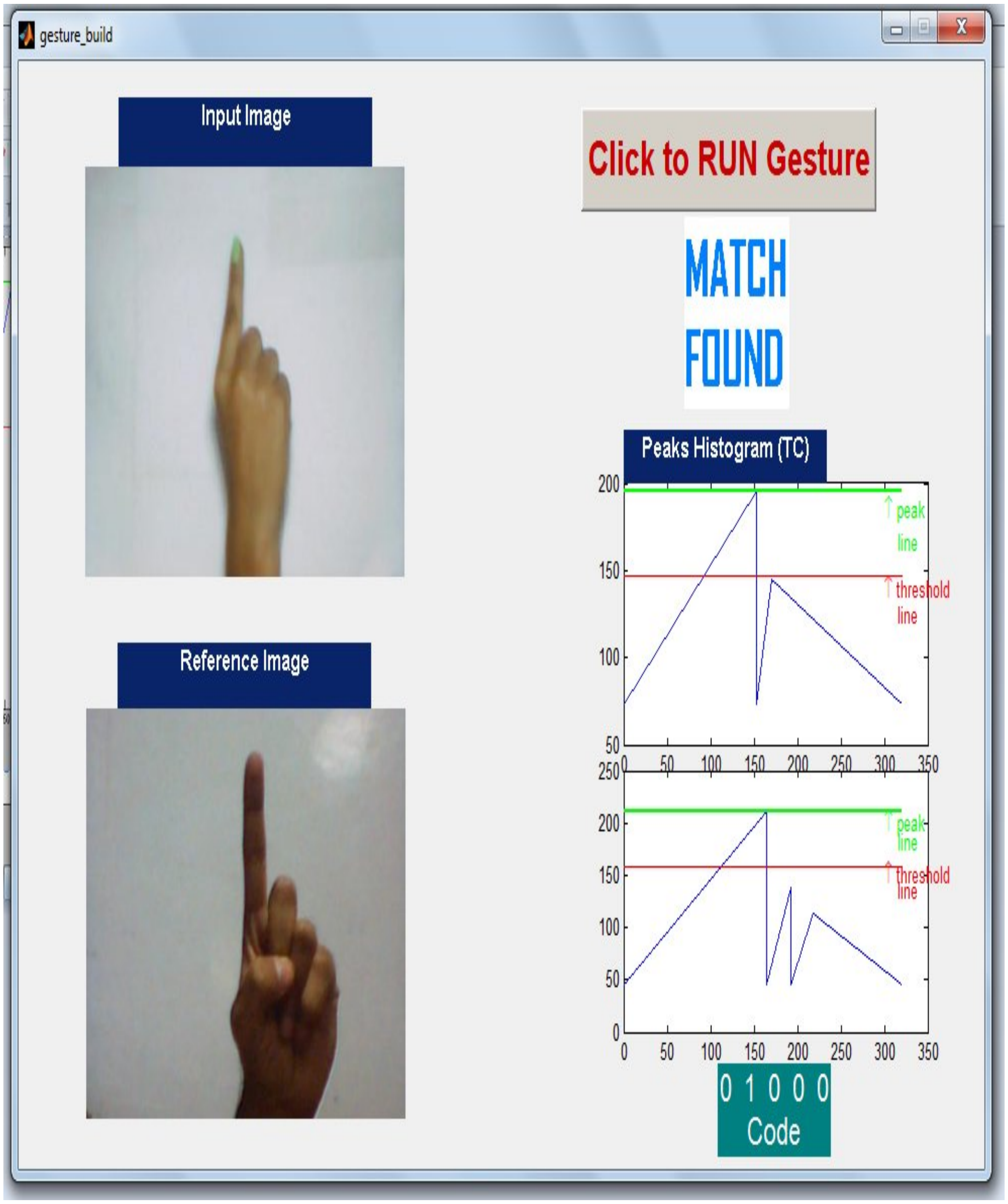
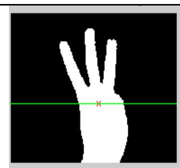
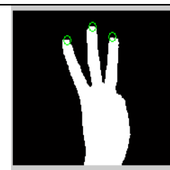
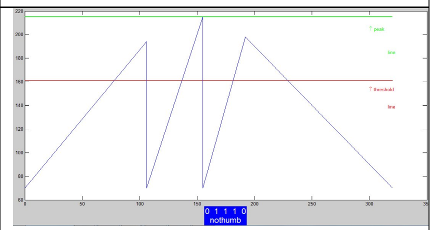
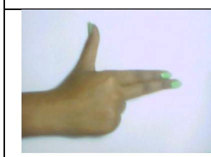
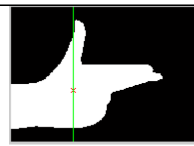
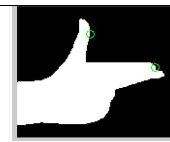
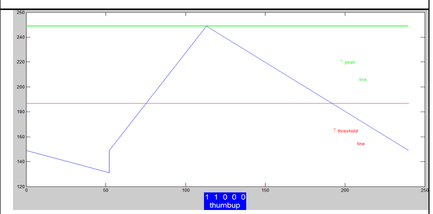
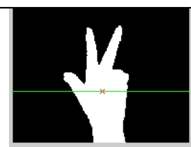
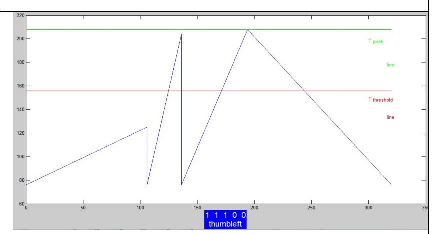
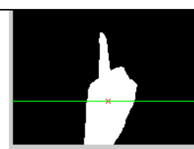
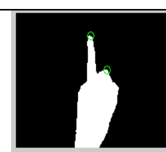
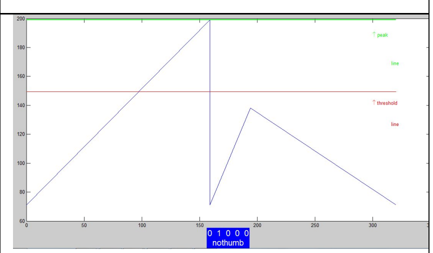
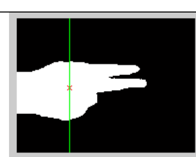
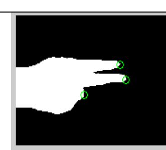
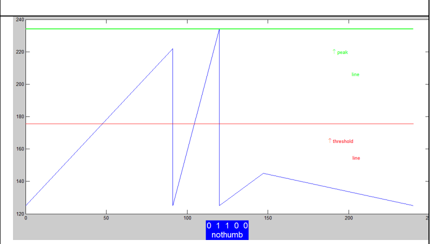
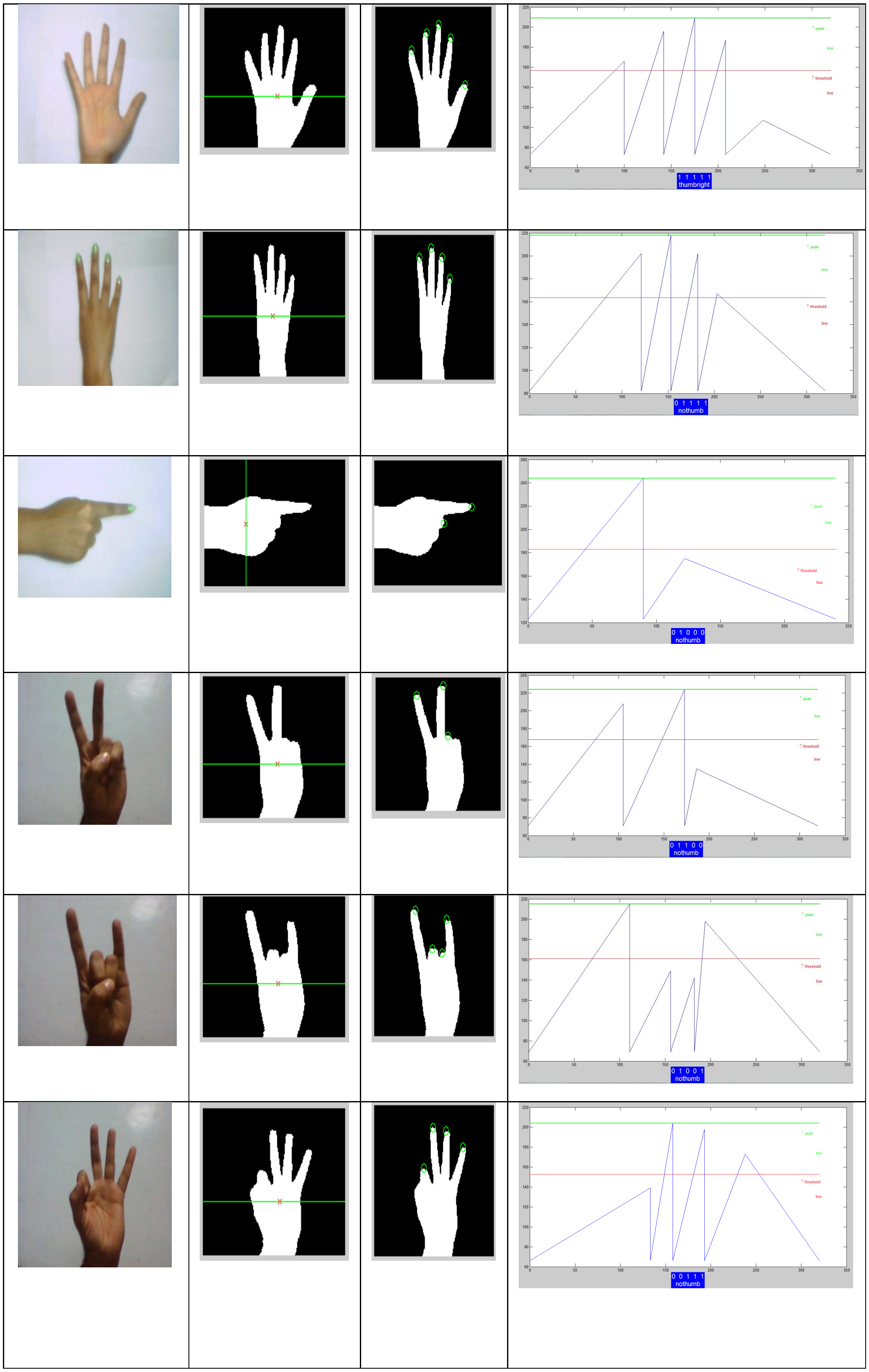


Figure 4.1 Graphical user interface of the final output

Table 4.1: Result Data

Gesture	Centroid	Peaks	Coded bits
			
			
			
			
			


1 1 1 1 1
thumbright
0 1 1 1 1
nothumb
0 1 0 0 0
nothumb
0 1 1 0 0
nothumb
0 1 0 0 1
nothumb
0 0 1 1 1
nothumb

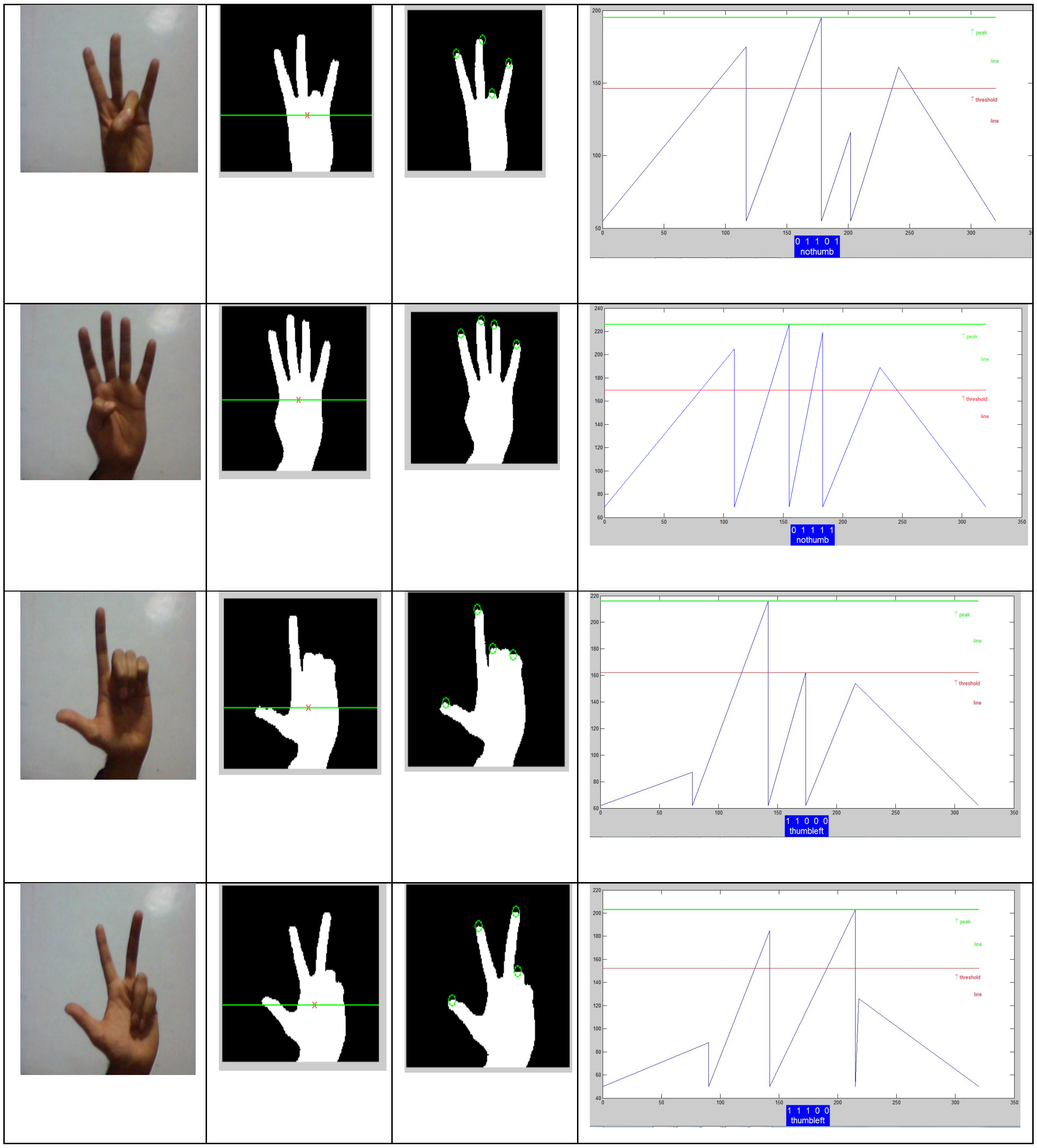

0 1 1 0 1
nothumb
0 1 1 1 1
nothumb
1 1 0 0 0
thumbleft
1 1 1 0 0
thumbleft

Table 4.2: Hand gesture Recognition results

Input gestures	No of input images	Successful cases	Percentage of recognition
1	5	5	100
2	5	4	80
3	5	5	100
4	5	5	100
5	5	5	100
6	5	5	100
7	5	5	100
8	5	5	100
9	5	5	100
10	5	5	100
11	5	4	80
12	5	3	60
13	5	4	80
14	5	5	100
15	5	5	100
16	5	5	100
17	5	5	100
18	5	4	80
19	5	5	100
20	5	4	80
21	5	5	100
22	5	4	80
23	5	5	100
24	5	5	100
25	5	5	100
26	5	5	100
27	5	5	100
28	5	5	100
29	5	5	100
30	5	5	100
31	5	4	80
32	5	4	80
33	5	3	60
34	5	4	80
35	5	5	100
36	5	5	100
37	5	5	100
38	5	5	100
39	5	5	100
40	5	5	100
ALL	200	187	93.5%

Chapter 5

Applications

In this section we will discuss the applications of hand gesture in various fields. Various applications have been used for hand postures and gestures as alternative level of interaction in different application domains, as mentioned in [28]: including virtual environments, smart surveillance, sign language translation, medical systems etc. This section gives a brief overview of some gesture recognition application areas. Gestures are so important in everyday's life, that sometimes we only make gestures to communicate with other, non verbal interaction. Application area is very vast; among those applications sign language and gaming are the most important one, because they are used in normal routine. Here we have some light on gesture recognition application area.

5.1 Sign language recognition

Sign languages are the most raw and natural form of languages could be dated back to as early as the advent of the human civilization, when the first theories of sign languages appeared in history. It has started even before the emergence of spoken languages. Since then the sign language has evolved and been adopted as an integral part of our day to day communication process. Sign language considered as an important and interesting application fields of hand posture and gesture recognition system, where many systems have been applied for this purpose [29]. Sign language has special importance for communications since the gestures are the way used for interpretation and explanation of specific subject [23]. It can be used for disabled peopled when communicating with the other people, and with the computer as well [23]. American Sign Language in [31, 03] is one example that has received utmost attention in the gesture literature. [32] Define a new gesture recognition algorithm for Korean scripts. Taguchi [33] recognized Japanese sign language alphabets and words; they could recognize 42 alphabets and 10 words using two types of Neural Networks. Gestures are one of the first forms of communication when a child learns to express its need for food, warmth and comfort. It enhances the emphasis of spoken language and helps in expressing thoughts and feelings effectively.

5.2 Robotics, human manipulation

One of the effective applications that can utilize hand postures and gestures is robot tele-manipulation [29]. Tele-robotic applications are typically classified under space exploration and military research domain. Using gestures for controlling robots is corresponding to virtual reality interaction system [23]. Recent researches used postures and gestures to learn the robot some interaction commands by explaining its appropriate meaning for the robot as an action [29].

5.3 Gesture to speech

Gesture-to-speech application which converts hand gestures into speech, this system enables hearing-impaired people to communicate with their surrounding environments through computers and interacts easily with other people even without the knowing for the sign language [29]. Hinton and Fels [34,35] introduced Glove Talk system interface between speech synthesizer using data glove device which mapping hand-gestures to speech using neural networks.

5.4 Games

For computer games, [36] applied gesture recognition on virtual game applications. [37] Used hierarchical recognition of human gestures for sports video annotation. [38] implemented computer vision and gesture recognition techniques, and developed a vision based low cost input device for controlling the VLC player through gestures.

5.6 Television control

Last application for hand postures and gestures is controlling Television devices [29]. [39] Developed a system to control a television set by hand gestures. Using an open hand and the user can change the channel, turn the television on and off, increase and decrease the volume, and mute the sound.

5.7 Tele Presence

There may raise the need of manual operations in some cases such as system failure or emergency hostile conditions or inaccessible remote areas. Often it is impossible for human operators to be physically present near the machines. Tele presence is that area of technical intelligence which aims to provide physical operation support that maps the operator arm to the robotic arm to carry out the necessary task. The prospects of tele presence includes space, undersea mission, medicine manufacturing and in maintenance of nuclear power reactors.

Chapter 6

Conclusion and Future Scope

We proposed a simple hand gesture recognition algorithm, followed by various steps like pre-processing, RGB image is converted into YCbCr , so that varying lightening conditions will not cause any problem. Then smudge elimination is done in order to get the finest image. These pre-processing steps are as important as any other step. After performing the pre-processing on the image, the second step is to determine the orientation of the image, only horizontal and vertical orientation is considered here and images with uniform background is taken.

Thumb presence, centroid and peaks in the hand are the main features, on the basis of these features a five bit binary code is generated , the leftmost bit represent the presence of thumb. '1' corresponds to the presence of significant peak and '0' corresponds to no peaks or insignificant peaks depending on the location of the peak whether it is below or above the threshold line as discussed in chapter 3

The strength of this approach includes its simplicity, ease of implementation, and it does not required any significant amount of training or post processing as rule based learning is used. It provides the higher recognition rate with minimum computational time.

The weakness of this method is that certain parameters and threshold values are taken experimentally that is it does not follow any systematic approach for gesture recognition, and many parameters taken in this algorithm are based on assumption made after testing number of images.

If we compare our approach with our previous approach described in paper [41].The success rate has been improved from 92.3% to 93.5%, the computation time decreased up to fraction of seconds. Also to make the system more robust, we have eliminated some of the constraints needed to be followed in our previous approach which makes it simpler. The proposed algorithm is simple and independent of user characteristics.

The future scope lies in making this algorithm applicable for various orientations of hand gestures, also different classification scheme can be applied. The algorithm can be improved so that images with non uniform background can also be used, this will enhance the human computer interaction. Visually impaired people can make use of hand gestures for human computer interaction like controlling television , in games and also in gesture to speech conversion.

REFERENCES

[1] Henrik Birk and Thomas Baltzer Moeslund, "Recognizing Gestures From the Hand Alphabet Using Principal Component Analysis", Master's Thesis, Laboratory of Image Analysis, Aalborg University, Denmark, 1996.

[2] Andrew Wilson and Aaron Bobick, "Learning visual behavior for gesture analysis", In Proceedings of the IEEE Symposium on Computer Vision, Coral Gables, Florida, pp. 19-21, November 1995.

[3] V. Pavlovic, R. Sharma, and T. Huang, "Visual interpretation of hand gestures for human-computer interaction: A review", IEEE Trans. Pattern Analysis and Machine Intelligence, 19(7):677–695, July 1997.

[4] J. Lee and T. L. Kunii, "Model-based analysis of hand posture", IEEE Computer Graphics and Applications, pp. 77-86, September 1995

[5] J. J. Kuch and T. S. Huang, "Vision based hand modelling and tracking", In Proceedings of International Conference on Computer Vision, (Cambridge, MA), June 1995

[6] J. Lee and T. L. Kunii, "Constraint-based hand animation", In Models and techniques in computer animation, pp. 110-127, Tokyo: Springer-Verlag, 1993

[7] L. R. Rabiner , "A tutorial on hidden Markov models and selected applications in speech recognition", In Proceedings of the IEEE, 77(2), 257–286,1989.

[8] A. Wilson and A. Bobick, " Parametric hidden markov models for gesture recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence,21(9), 884–900, 1999.

[9] V. Pavlovic, R. Sharma and T.S. Huang, "Visual interpretation of hand gestures for human-computer interaction: A review", IEEE Transaction on Pattern Analysis and Machine Intelligence, 19(7), pp 677–695,1997.

[10] Y. Wu and T.S. Huang, "Vision-based gesture recognition: A review", In Lecture Notes in Computer Science, Gesture Workshop,1999.

[11] Konstantinos G. Derpanis, "A Review of Vision-Based Hand Gestures", Internal Report, Department of Computer Science. York University,2004.

[12] Richard. Watson , "A Survey of Gesture Recognition Techniques", Technical Report TCD-CS-93-11, Department of Computer Science, Trinity College Dublin,1993.

[13] Sushmita Mitra and Tinku Acharya, "Gesture Recognition: A Survey", IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews, Vol. 37, No. 3,2007.

[14] T.S. Hunang and V.I. Pavloic, "Hand Gesture Modeling, Analysis, and Synthesis",Proc. of International Workshop on Automatic Face and gesture recognition, Zurich, pp.73-79, 1995

[15] I. E. Sutherland, "Sketchpad: A man-machine graphical communication system", In Proceedings of the AFIPS Spring Joint Computer Conference 23. pp. 329–346, 1963.

[16] G.R.S. Murthy, R.S. Jadon, "Hand Gesture Recognition using Neural Networks", In IEEE 2nd International Advance Computing Conference,2010.

[17] C. W. Ng and S. Ranganath, "Real-time gesture recognition system and application" ,Image Vis. Comput., vol. 20, no. 13–14, pp. 993–1007,2002 .

[18] K. Abe, H. Saito and S. Ozawa, "Virtual 3-D interface system via hand motion recognition from two cameras" , IEEE Trans. Syst., Man, Cybern. A, vol. 32, no. 4, pp. 536–540,2002

[19] Md. Hasanuzzaman, V. Ampornaramveth, Tao Zhang, M.A. Bhuiyan , Y. Shirai and H. Ueno, " Real-time Vision-based Gesture Recognition for Human Robot Interaction", In the Proceedings of the IEEE International Conference on Robotics and Biomimetics, Shenyang China ,2004.

[20] Rafiqul Zaman Khan and Noor Adnan Ibraheem, " Survey on Gesture Recognition for Hand Image Postures",Computer and Information Science ,Vol. 5, No. 3,pp.110-121, May 2012.

[21] L. Dipietro, A. M. Sabatini, and Dario , "Survey of glove-based systems and their applications" , IEEE Transactions on systems, Man and Cybernetics, Part C: Applications and reviews, 38(4), 461-482, 2008.

[22] M. M. Hasan and P. K. Mishra, " HSV brightness factor matching for gesture recognition system", International Journal of Image Processing (IJIP), vol. 4(5), 2010.

[23] G. R. S. Murthy and R. S. Jadon, " A review of vision based hand gestures recognition", International Journal of Information Technology and Knowledge Management, 2(2), 405-410,2009.

[24] H. P. Peter, " Image Segmentation through Clustering Based on Natural Computing Techniques", Image Segmentation Methods ,India,2011.

[25] E. Stergiopoulou and N. Papamarkos , "Hand gesture recognition using a neural network shape fitting technique", Elsevier Engineering Applications of Artificial Intelligence 22, 1141-1158,2009.

[26] S. E. Ghobadi, O. E. Loepprich, F. Ahmadov, J. Bernshausen, K. Hartmann,, and O. Lofield, " Real time hand based robot control using multimodal images", International Journal of Copmuter Science IJCS.Vol 35(4),2008.

[27] X. Li, "Gesture recognition based on fuzzy C-Means clustering algorithm",2003.

[28] S. Mitra, and T. Acharya, " Gesture recognition: A survey", IEEE Transactions on systems. Man and Cybernetics,Part C: Applications and reviews, 37(3), 311-324,2007.

[29] J. Joseph and J. LaViola, " A survey of hand posture and gesture recognition techniques and technology", Master Thesis, NSF Science and Technology Center for Computer Graphics and Scientific Visualization, USA,1999.

[30] T. Starner, J. Weaver, and A. Pentland, " Real-time American Sign Language recognition using desk and wearable computer based video", IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(12), 1371 – 1375,2002

[31] C. Vogler and D. Metaxas, "A framework for recognizing the simultaneous aspects of American Sign Language", Ideal Computer Vision and Image Understanding 81, pp. 358-384,2001.

[32] M. G. Cho., " New gesture recognition algorithm and segmentation method of Korean scripts for gesture-allowed Ink editor", Elsevier an International Journal Information Sciences, 176(9), 1290-1303,2006.

[33] K. Murakami and H. Taguchi, "Gesture recognition using recurrent neural networks", ACM, Proceedings of the SIGCHI conference on Human factors in computing systems: Reaching through technology CHI '91, pp. 237-242, 1999.

[34] S. S. Fels and G. E. Hinton, "Glove-talk: A neural network interface between a data-glove and a speech synthesizer", IEEE transaction on Neural Networks, 4(1), 2-8, 1993.

[35] S. S. Fels and G. E. Hinton, "Glove-talk II—A neural-network interface which maps gestures to parallel formant speech synthesizer controls", IEEE transactions on neural networks, 9(1), 205-212, 1998.

[36] Z. Xu, C. Xiang, V. Lantz and W. Kong-qiao, " Hand gesture recognition and virtual game control based on 3D accelerometer and EMG sensors",ACM Proceedings of the 14th international conference on Intelligent user interfaces IUI '09,2000.

[37] G. S. Chambers, S. Venkatesh, G. A. West and H. H. Bui, "Hierarchical Recognition of Intentional Human Gestures for Sports Video Annotation",IEEE 16th International Conference on Pattern Recognition Proceedings, 2, pp. 1082- 1085.2002.

[38] S. S. Rautaray and A. Agrawal, " A vision based hand gesture interface for controlling VLC media player" , International Journal of Computer Applications, 10(7),2010.

[39] W. T. Freeman and C. D. Weissman, "Television control by hand gestures", IEEE International Workshop on Automatic Face and Gesture Recognition. Zurich,1995.

[40] N.Otsu, "A Threshold Selection Method from Gray-Level Histograms", IEEE transactions on systems, man, and cybernetics, vol. smc-9, no. 1, January 1979.

[41] A. Jinda-Apiraksa, W. Pongstiensak and T. Kondo, "Shape-Based Finger Pattern Recognition using Compactness and Radial Distance," The 3rd International Conference on Embedded Systems and Intelligent Technology(ICESIT 2010), Chiang Mai, Thailand, February 2010

APPENDIX A

An Introduction to Image Processing in MATLAB

MATLAB is a high performance language for technical computing. It integrates computation, visualization and programming in an easy-to-use environment where problems and solutions are expresses in familiar mathematical notation. It is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations in a fraction of the time it would take to write a program in a scalar non interactive language such as C , C++ or JAVA.

Image formats supported by Matlab

The following image formats are supported by Matlab:

- BMP
- HDF
- JPEG
- PCX
- TIFF
- XWB

Working formats in Matlab

If an image is stored as a JPEG-image on your disc we first read it into Matlab. However, in order to start working with an image, for example perform a wavelet transform on the image, we must convert it into a different format.

Intensity image (gray scale image)

It represents an image as a matrix where every element has a value corresponding to howbright/dark the pixel at the corresponding position should be colored. There are two ways to represent the number that represents the brightness of the pixel: The double class (or data type). This assigns a floating number ("a number with decimals") between 0 and 1 to each pixel. The value 0 corresponds to black and the value 1 corresponds to white. The other class is called uint8 which assigns an integer between 0 and 255 to represent the brightness of a pixel.

Binary image

This image format also stores an image as a matrix but can only color a pixel black or white (and nothing in between). It assigns a 0 for black and a 1 for white.

Indexed image

This is a practical way of representing color images. An indexed image stores an image as two matrices. The first matrix has the same size as the image and one number for each pixel. The second matrix is called the *color map* and its size may be different from the image. The numbers in the first matrix is an instruction of what number to use in the color map matrix.

RGB image

This is another format for color images. It represents an image with three matrices of sizes matching the image format. Each matrix corresponds to one of the colors red, green or blue and gives an instruction of how much of each of these colors a certain pixel should use.

Reading Image Files

The command to read an image from file *filename* and store it in matrix variable p is:

p = imread('*filename*');

The filename may be an absolute pathname or a relative pathname from the current working directory. Omitting the ';' at the end of the command causes the value to be printed to the command window.

Writing Image Files

The command to write an image from variable p and store it in file *filename* is:

imwrite (p, '*filename*');

The filename may be an absolute pathname or a relative pathname from the current working directory.

Displaying Image Files

The command to display the image from variable p to a figure window is:

imshow(p);

An additional parameter may be used to set the number of display levels or set the range of display levels. Various controls, such as dynamic display of index and value for the cursor position, are available in the image display tool.

The figure command can be used to create a new current figure for the display:

figure, imshow (p);

Standard Arrays

MATLAB has standard arrays for creating arrays of a defined size with zeros, ones, true, false, or random values. For example: p = zeros(M,N);

Built-in Functions (used in this work)

MATLAB has many useful builtin functions:

rgb2ycbcr : Convert RGB color values to YCbCr color space

rgb2gray : Converts the truecolor image RGB to the grayscale intensity image I. rgb2gray converts RGB images to grayscale by eliminating the hue and saturation information while retaining the luminance.

graythresh : computes a global threshold (level) that can be used to convert an intensity image to a binary image with im2bw. level is a normalized intensity value that lies in the range [0, 1]. The graythresh function uses Otsu's method, which chooses the threshold to minimize the intra class variance of the black and white pixels.

Bwareaopen : Morphologically open binary image (remove small objects), removes from a binary image all connected components (objects) that have fewer than P pixels, producing another binary image.

Imfill : Fill image regions and holes.

Bwboundaries : Trace region boundaries in binary image, it traces the exterior boundaries of objects, as well as boundaries of holes inside these objects, in the binary image BW

Bwlabel : Label connected components in binary image.

Regionprops : regionprops(L, properties) measures a set of properties for each labeled region L. L can be a label matrix or a multidimensional array. When L is a label matrix, positive integer elements of L correspond to different regions

Imcrop : imcrop(X, map, rect) crops the indexed image X. Map specifies the colormap used with X. Rect is a four-element position vector [xmin ymin width height] that specifies the size and position of the cropping rectangle.