INDIAN SIGN LANGUAGE CHARACTER DETECTION USING GESTURE RECOGNITION TECHNIQUES

A PROJECT REPORT

Submitted by

CB.EN.U4CSE09405	**AJITH.J**
CB.EN.U4CSE09415	**KARTHIK.KS**
CB.EN.U4CSE09427	**NIRANJAN.M**
CB.EN.U4CSE09443	**SAIPREETHY.MS**
CB.EN.U4CSE09447	**SHANGEETHA.RK**
CB.EN.U4CSE09454	**VALLIAMMAI.V**

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF ENGINEERING, COIMBATORE

AMRITA VISHWA VIDYAPEETHAM

COIMBATORE 641 105

May 2013

AMRITA VISHWA VIDYAPEETHAM
AMRITA SCHOOL OF ENGINEERING, COIMBATORE, 641105

श्रद्धावान् लभते ज्ञानम्

BONAFIDE CERTIFICATE

This is to certify that the project report entitled "**INDIAN SIGN LANGUAGE CHARACTER DETECTION USING GESTURE RECOGNITION TECHNIQUES**" *submitted by*

CB.EN.U4CSE09405	**AJITH.J**
CB.EN.U4CSE09415	**KARTHIK.KS**
CB.EN.U4CSE09427	**NIRANJAN.M**
CB.EN.U4CSE09443	**SAIPREETHY.MS**
CB.EN.U4CSE09447	**SHANGEETHA.RK**
CB.EN.U4CSE09454	**VALLIAMMAI.V**

In partial fulfillment of the requirements for the award of the **Degree of Bachelor of Technology in COMPUTER SCIENCE AND ENGINEERING** is a bonafide record of the work carried out under my (our) guidance and supervision at Amrita School of Engineering, Coimbatore.

Ms. S.Padmavathi
Project Guide
Asst. Professor, Information Technology

Prof. K. Gangadharan
Chairman
Dept. of Information Technology

This project report was evaluated by us on:

INTERNAL EXAMINER

EXTERNAL EXAMINER

Acknowledgement

We would like to express our deep gratitude to our beloved **Satguru Sri Mata Amritanandamayi Devi** for providing the bright academic climate at this college, which has made this entire task appreciable. We take this opportunity to thank our Pro Chancellor **Bramachari Abhayamrita Chaitanya,** Vice Chancellor **Dr. Venkat Rangan .P** and **Dr. M.P.Chandrasekaran**, Dean (Engineering) for providing the necessary infrastructure required for the successful completion of the project.

We take this opportunity to thank **Prof. K. Gangadharan**, Chairperson of the Department of Information Technology for providing an opportunity to do this project. We express our sincere thanks and whole hearted gratitude to our Guide **Ms. S.Padmavathi**, Assistant Professor of the Department of Information Technology and our co-guide, **Mr. Raghesh Krishnan K** Assistant Professor of the Department of Information Technology who have been a pillar behind every step in this project.

We extend our thanks to **Ms. R.Aarthi, Ms. Anisha, Mr. C. Arun Kumar** Assistant Professors of the Department of Information Technology. They have supported us in executing this project and helped us in meeting the objectives of this project.

We sincerely acknowledge our seniors AshokGautham. M, Arunachalam R.M, Nagarjunan and Ram Rajesh who helped us to get knowledge on our project.

We also extend our thanks to all the faculty members and non-teaching staff of our college, our parents and our friends whose timely help and co-operation is worth mentioning.

ABSTRACT

Communication is the exchange of thoughts, messages, or information, by speech, visual signals, writing, or behaviour. Deaf and dumb people communicate among themselves using sign languages, but they find it difficult to expose themselves to the outside world. This paper proposes a method for recognizing Indian sign language characters given as input by the user in the form of hand gestures. Unlike the conventional method, this method does not require any additional hardware and makes the user comfortable. The system takes input at real time through a webcam integrated in the laptop. The hand region is separated out from the background using skin segmentation and motion segmentation. The ISL characters are shown in a way that they resemble the character itself. There are 18 ISL characters which are shown using two hands and 8 characters which are shown using single hand. The single hand characters can be identified using the number of open fingers, the angle between the fingers and their state. But the two hand characters have both the hands overlapped hence making it difficult to segregate them and identify the state of fingers. So, two hand characters are identified using HOG features. Neural network is used as the learning algorithm to make the system adaptable for different users. The system has been tested by several people of varying skin complexions, in several environments and was found to have accuracy of about 90%. The accuracy mainly dropped due to the illumination of the environment and occlusion of hands involved in two hand gestures.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$\ominus$	:	Erosion
$\oplus$	:	Dilation

LIST OF ABBREVIATIONS

HOG: Histogram of Oriented Gradients

CV: Computer Vision

SVM: Support Vector Machine

HMM: Hidden Monrovian Model

FCM: Fuzzy C-means Algorithm

ISL: Indian Sign Language

ASL: American Sign Language

PCM: Principle Component Analysis

GMM: Gaussian Mixture Model

RGB: Red Green and Blue

HSI: Hue Saturation and Intensity

HCI: Human Computer Interface

BPN: Back Propagation Network

Chapter 1

INTRODUCTION

Chapter 1

INTRODUCTION

1.1 INTRODUCTION TO GESTURES

Gestures are defined by Iverson and Thal (1998) as "actions produced with the intent to communicate and are typically expressed using fingers, hands, and arms, but can also include facial features (e.g. lip smacking for "eating") and body motions. The applications of gesture recognition are manifold, ranging from sign language recognition to human computer interaction and virtual reality. In gesture recognition technology, a camera reads the movements of the human body and communicates the data to a computer. The data is used to achieve some goal.

There are two different terms called hand gesture and hand posture. The difference is that the hand posture is a static pose without any movement whereas the hand gesture is a sequence of hand postures that is connected with continuous movement in the hand and fingers in a short period of time.

1.1.1 Taxonomy of gestures

The most interesting part is how gestures can be used to communicate with systems. Rime and Schiaratura(1991) propose the following gesture taxonomy:

Symbolic gestures: These are gestures that, within each culture, have come to have a single meaning. An Emblem such as the "OK" gesture is one such example; however American Sign Language gestures also fall into this category.

Deictic gestures: These are the types of gestures most generally seen in HCI and are the gestures of pointing, or otherwise directing the listeners' attention to specific events or objects in the environment. They are the gestures made when someone says "Put that there".

Iconic gestures: As the name suggests, these gestures are used to convey information about the size, shape or orientation of the object of discourse. They are the gestures made when someone says "The plane flew like this", while moving their hand through the air like the flight path of the aircraft.

Pantomimic gestures: These are the gestures typically used in showing the use of movement of some invisible tool or object in the speaker's hand. When a speaker says "I turned the steering wheel hard to the left", while mimicking the action of turning a wheel with both hands, they are making a pantomimic gesture.

1.2 APPLICATIONS OF GESTURE RECONITION

Gesture recognition has found various applications in the current scenario like human computer interface, automatic inspection, modelling objects, and robotics.

- **Human Computer Interfaces:** The technology also has the potential to change the way users interact with computers by eliminating input devices such as joysticks, mice and keyboards and allowing the unencumbered body to give signals to the computer through gestures such as finger pointing.

- **Robot Control:** Hand movements can be tracked using computer vision and this motion is mimicked by a haptic robot arm. This arm is used to accomplish several applications. One most recent application being performing a medical operation being performed remotely.

- **Sign Language Detection:** One way gesture recognition is being used is to help the physically impaired to interact with computers, such as interpreting sign language.

Apart from gesture recognition, computer vision from which gesture recognition technology has been derived is also used in machine inspection, medical imaging, automotive safety, surveillance, finger print recognition and biometrics.

1.3 CHALLENGES OF INDIAN SIGN LANGUAGE

Deaf and dumb people find it extremely hard to communicate with normal people because most of the normal people are not aware of sign languages. Further, sign language is regional. Indian sign language is completely different from American Sign Language. While there are a lot of efforts going into American Sign Language very little research have gone into Indian Sign Language. Figure 1.1 shows the character set of Indian Sign Language. Figure 1.2 shows American Sign Language characters. If one looks at the ISL character set, it can be seen that many of the gestures requires both the hands. There are a lot of gestures which involve overlapping of hands. From a gesture recognition point of view, detection of these overlapping hand gestures is a very challenging issue. American Sign Language on the other hand is fairly simpler and requires one hand. Also existing systems are highly dependent on additional hardware support which makes it less affordable for common people.

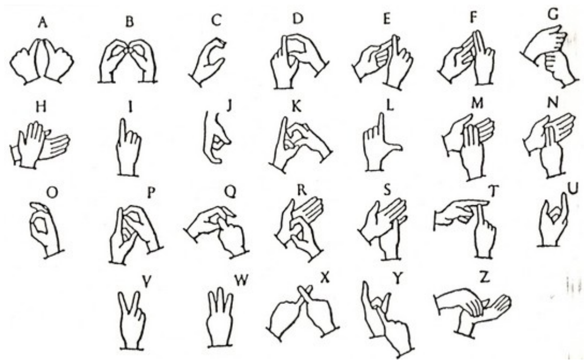


Fig 1.1: English character set using Indian Sign Language[19].

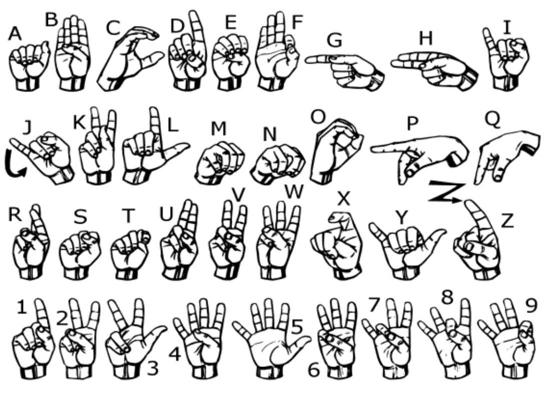


Fig1.2: English character set using American Sign Language[17]

To overcome these drawbacks, this project aim to build a webcam based, cost efficient and robust Indian Sign Language detector. Robustness is enhanced by using a combination of different algorithms together to extract the features instead of relying on a single algorithm.

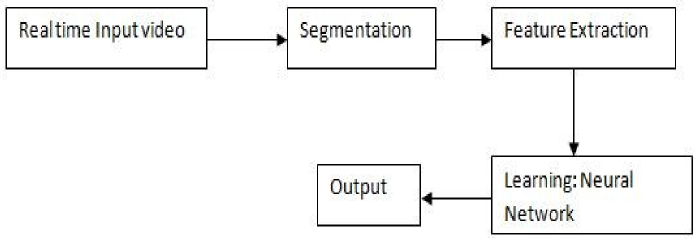


Fig. 1.3: General architecture of the proposed system

Figure 1.3 shows a general architecture of the proposed system. The input is taken in through a simple webcam and converted into frames. The input frame may also contain unnecessary background information from which the hand region has to be separated. The hand region in each frame is segmented out from the background using a hybrid segmentation technique. In feature extraction phase two separate methods are used, based on whether the signer uses two hands or a single hand to show the gesture. The decision among single hand or two hands is made based on the number of hand regions detected in the prior frames. Since single hand gestures do not have characters involving overlapping of hands, distance transform method is used. Some of the details that are extracted are the number of fingers that are opened, the angle between the fingers. The Histogram of Oriented Gradients is used in case of two handed gestures because it involves characters involving overlapping of hands. The extracted features are passed through the recognition phase to identify the gestures. The recognition phase employs a neural network that is trained by the developers so that the system responds properly to any user. The recognized gesture is thus given as output.

1.4 ORGANIZATION OF THE PROJECT REPORT

The project report is organized as follows

Chapter 2 surveys the available literatures in the field of gesture recognition. The chapter gives information about various methods available for image acquisition, skin region segmentation, feature extraction and learning algorithms.

Chapter 3 explains the system design and overall flow of the system. The chapter describes about the system requirements.

Chapter 4 explains the process of hand segmentation in detail. The chapter describes about the various methods for segmentation, its advantages, disadvantages and accuracy.

Chapter 5 and Chapter 6 explain the method of feature extraction techniques used for single hand gestures and two hand gestures respectively.

Chapter 7 explains the neural network algorithm used to classify single hand and two hand gestures.

Chapter 8 explains the analysis of results obtained from various methods.

Chapter 9 explains the conclusion and future work of the project.

Chapter 2

LITERATURE SURVEY

Chapter 2

LITERATURE SURVEY

The human hand is a highly deformable object with many degrees of freedom [1]. To recognise the sign language, the input image has to undergo many processes through which the required information will be collected to identify the gesture. There are many methods in each process. The image can be acquired by using a fixed camera or by using multiple cameras. After obtaining the image it has to be segmented where only the required information will be collected. After segmenting the image the features will be extracted from the segmented region in order to recognise the gesture. And the extracted image is studied in depth using the learning algorithms.

Each of these modules has several techniques associated with them. This chapter discusses the techniques that have been implemented in gesture recognition systems, their pros and cons and also presents an idea of how they could be used in ISL recognition.

2.1 SURVEY OF IMAGE ACQUISITION TECHNIQUES

The image acquisition is the first stage of any computer vision based technology. The processing can be done on the image only when the image has been obtained. If the image has not been acquired properly, the satisfactory output will not be obtained even after so many image processing techniques (i.e.) if the gesture has not been acquired properly from the input, it cannot be recognised. So the image acquisition plays a pivotal role in computer vision based system. The two dimensional image is usually described in terms of spatial co ordinates. An image is said to be digitized after spatial sampling and intensity quantization. The input can be obtained by many methods. It can be done be using:

2.1.1. Single Camera

An important factor in any computerized system is its viability to reach the common man, which boils down to the affordability of the built system. In computer vision system using a single camera to acquire input is an effective way to bring down the cost. But with this being said if the quality of the camera is too low then the detail that can be extracted from such a camera is very much lesser [2]. Also the camera must be able to compensate to a certain extent to the illumination variance of the environment. Thus a balance between the cost and efficiency of the device used to take in input.

2.1.2. Multiple Cameras

Multiple cameras are used to acquire more than one perspective of an object. Usually multiple cameras are used in computer vision applications which require high precision [3]. The down side of using multiple cameras is that the cost multiplies. Also since multiple images have to be processed the execution time increases. The most important feature that is extracted using multiple cameras is the depth of the objects. Response time is very important in the system being developed because ISL recognition has to happen at real time. Hence going for multiple cameras is a disadvantage [3].

Apart from the using a single or multiple cameras for image acquisition, the user sometimes is required to use some additional hardware to felicitate proper image acquisition. There are 2 different approaches used in several gesture based technologies.

- Data glove or cyber glove.
- Computer vision based human computer interaction using hand gesture.

2.1.3. Glove based hand gesture recognition

The Cyber glove II (a product of Cyber Glove systems, USA) as shown in Fig.2.1 (a) and (b) is a fully instrumented glove that provides up to 22 high-accuracy joint-angle measurements. It uses proprietary resistive bend-sensing technology to accurately transform hand and finger motions into real-time digital joint-angle data.

(a) (b)

Fig 2.1: (a) Wireless cyber glove (b) Wired cyber glove

It is available in two models and for either hand. The 18-sensor model features two bend sensors on each finger, four abduction sensors [27], plus sensors measuring thumb crossover, palm arch, wrist flexion and wrist abduction. The 22-sensor model has three flexion sensors per finger, four abduction sensors, a palm-arch sensor, and sensors to measure flexion and abduction. Each sensor is extremely thin and flexible being virtually undetectable in the lightweight elastic glove [26].

The Cyber glove has a software programmable switch and LED on the wristband which allows the system software developer to provide the Cyber glove wearer with additional input/output capability.

The Cyber glove has been used in a wide variety of real-world applications, including digital prototype evaluation, virtual reality biomechanics, and animation. The Cyber glove has become the de facto standard for high-performance hand measurement and real-time motion capture.

2.1.4. Computer vision based hand gesture recognition

Unlike haptic interfaces, computer vision based gesture recognition does not require the user to wear any special equipment or attach any devices to the body. The gestures of the body are read by a camera instead of sensors attached to a device such as a data glove. In addition to hand and body movement, gesture recognition technology also can be used to read facial and speech expressions (i.e., lip reading), and eye movements [14].

Vision based hand gesture control involves various methods of hand gesture recognition. Hand gestures are identified easily when hand gloves which are either single coloured or multi-coloured are used as shown in Fig. 2.2. Colour paper caps on the tip of the fingers make tip detection much simpler than any other technique. Further gesture recognition processes are carried out after the detection of the fingertip [14]. Complex computation system involving database are also used for recognition of user inputs. They also have training phase which allows user to define their own gesture for communication with the desired system.

Fig 2.2: Multi coloured glove

Gesture recognition using colour gloves makes computation easy but reduces the comfort level of the user. In order to overcome the difficulty of using glove to recognize hand gestures a method is proposed to communicate with the system using bare hands. Also this method makes use of single webcam integrated with the laptop to acquire the user input hence there is no need for external device.

2.2. SURVEY OF SEGMENTATION TECHNIQUES

The various methods by which the skin segmentation can be performed are mentioned below

Skin colour segmentation based on histogram analysis: The part of the human body which is required for analysis is determined using pose estimation. In that region the maximum colour frequency and minimum colour frequency is determined using histogram analysis [9]. Based on the colour frequency threshold is fixed and skin region is segmented.

Adaptive skin colour segmentation based on GMM: In this method the parameters are modelled to cope up with the change in the illumination and noise in each frame [9].

Background subtraction: Initially image is captured without any hand gestures. Then the gestures are made. Now an image is captured again. When the initial image is subtracted from the image with hand gestures, the hand region alone is obtained. Then applying rules on the colour model selected skin region can be segmented.

Hybrid Image Segmentation using Watershed segmentation and Region Merging: There is a general segmentation problem as how to segment an image into homogeneous segments such that after combining two neighbors it gives a heterogeneous segment. There are many techniques for an error-free image partitions as histogram-based represents the simple probability distribution function of intensity values of any image. Edge-based technique used to detect using differential filter in order of image gradient or laplacian and then grouped them into contours represents the surface [23]. In the region-based segmentation technique segment the image into a set of homogeneous regions then merged them according to certain decision rules.

Active Contour Based Segmentation: Active contour based segmentation is a method where a random shape is drawn around the object of interest for all iteration it is deformed so that an energy function is minimized. When the contour overlaps with the entire object, the energy function becomes zero [21]. Though a very efficient segmentation method **ISL** gestures have a very complex shapes and hence the number of iterations needed to obtain a properly segmented image is very high.

2.3 SURVEY OF COLOUR MODELS

Various methods available for skin segmentation are discussed below. The various colour models available are [25]:

RGB colour model: In RGB colour model R represents red component, G represents green component, B represents the blue component. The RGB colour space do not separate luminance and chrominance hence R, G and B components are highly correlated. The luminance of the given RGB pixel is a linear combination of the R, G and B values. The skin segmentation performed on this colour model is very simple.

HSI colour model: The three components present in this model are hue, saturation and intensity represented as H, S and I respectively. Hue defines the dominant colour present. The colourfulness is measured using the Saturation component. The intensity component is used to measure the brightness of the colour [25]. Intuitiveness of the colour space component has made this colour model favourable for working with skin segmentation.

YCbCr colour model: YCbCr colour model contains two components namely the luma component (Y) and the chrominance component (Cb and Cr). The luma component is responsible for the brightness and the chrominance component is responsible for the colour. The location of skin colour in the chrominance channel is not affected by changing the intensity of the illumination [25]. This facilitates skin segmentation that is invariant to changes in the illumination.

2.4. A SURVEY OF FEATURES EXTRACTED IN GESTURE RECOGNITION

2.4.1. Features exclusive for the hand

Gesture recognition using finger count: The hand gestures are recognized based on the number of fingers used to represent particular gestures [13]. The gesture made may depend on the finger used or may be independent of the fingers used [13].

Gesture recognition using finger tip: In this method hand gestures are identified based on the finger tip of the finger used in a gesture. The number of finger tips present is used to recognize a particular hand gesture [15].

Hand gesture recognition using PCA: The hand gestures for a particular application are defined by the user themselves. This method contains a training phase in which the user registers the hand gestures and map them to corresponding functions. The gestures are identified by comparing the hand gestures with the images acquired during the training phase [24].

2.4.2. Shape based features

These are features that are unique for a particular shape.

Center of Gravity and Axis of Least Inertia: The center of gravity is also called centroid of the object. The position of shape centroid is fixed with different point's distribution on a contour. The axis of least inertia passes through the centroid. The axis of least inertia is unique to the shape. It serves as a unique reference line to preserve the orientation of the shape. The axis of least inertia (ALI) of a shape is defined as the line for which the integral of the square of the distances to points on the shape boundary is a minimum [11].

Convex Hull: Convex hull is the smallest polygon that covers all the points of the object. Convex hull alone cannot be used as a feature. But there is a lot of information given by the convex hull that can be used as features. For example the convexity, this is the ratio between the perimeter of the convex hull and the actual boundary of the object [23].

Area Function: When the boundary points change along the shape boundary, the area of the triangle formed by two successive boundary points and the center of gravity also changes [11]. This forms an area function which can be exploited as shape representation.

Polygon Evolution: The curve evolution method achieves the task of shape simplification, i.e., the process of evolution compares the significance of vertices of the contour based on a relevance measure [11]. Since any digital curve can be regarded as a polygon without loss of information (with possibly a large number of vertices), it is sufficient to study evolutions of polygonal shapes for shape feature extraction.

Moments: The concept of moment in mathematics evolved from the concept of moment in physics. It is an integrated theory system [23]. For both contour and region of a shape, one can use moment's theory to analyze the object.

2.4.3. Edge Based Features

Histogram of Oriented gradients: This concept was introduced by Dallal and Triggs in 2005 [24] primarily for object detection. The gradient angle and magnitude is measured and binned in a histogram. Localization of the features is a very important aspect in HOG [20].

Hand shape recognition using Distance transform and shape decomposition: In the work proposed by Junyeong Choi*et al* [21] the possible hand poses or hand shapes are stored in the database. The input hand pose is compared with the database using distance transform of the image and histogram of the image [23]. Moreover the hand regions are decomposed into fingers and palm regions and compared with the decomposed fingers and palm region stored in the database.

2.5. CLASSIFICATION TECHNIQUES

Computer vision systems have to respond to any user instantaneously. They cannot be personalised software. This being said, the complexity involved in extracting the features is large. As so, relying on hard bounded techniques is often seen as a disadvantage because different people may show the same gesture differently. To overcome this difficulty, the system is trained by the developers to adapt to any input the user may provide [16]. Hence learning algorithms are used.

2.5.1. Boosting

Boosting is one of a class of techniques which allows the combination of several statistical classifiers in order to generate a consensus classifier which attains high reliability and accuracy [4]. The well-known disadvantage of boosting is that as the hard sets get harder, the likelihood of finding a good classifier drops, so that training is notoriously slow. It is more suitable for hand postures and false negative occurs due to similar dark/bright patterns [5].

2.5.2. Support Vector Machines

Support vector machines are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked as belonging to either classification or regression analysis, an SVM training algorithm builds a model that assigns new examples into one category or the other [9].

An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. Multiclass SVM works in such a fashion that it aims to assign labels to instances by using support vector machines, where the labels are drawn from a finite set of several elements [16]. The dominant approach for doing so is to reduce the single multiclass problem into multiple binary classification problems**.**

2.5.3. Hidden Markov Model

A hidden Markov model (HMM) can be considered a generalization of a Markov chain without the Markov-chain restriction thereby making them non deterministic. The restriction is that, a state can have only one transition arc with a given output making the Markov chains deterministic. So, it is impossible to determine the sequence of states by simply looking at output [6].

In the context of hand gesture recognition, each state could represent a set of possible hand positions. The state transitions represent the probability that a certain hand position transitions into another; the corresponding output symbol represents a specific posture and a sequence of output symbols represents a hand gesture. One then uses a group of HMMs, one for each gesture, and runs a sequence of input data through each HMM [18]. The common way to represent input data is feature vector.

Starner used HMMs in a vision-based solution to recognize a forty word subset of American Sign Language. Instead of using a different model for each sign in the recognition set [6], Starner found the minimum and maximum number of states required for an individual HMM and then, using skip transitions (which give low weights to state transitions that are not needed) developed a general HMM topology for all models used in the system. With ample training of the HMMs (between 20 and 80 training 35 samples for each sign) the system was able to achieve an accuracy of over 90 percent.

2.5.4. Fuzzy C- Means Clustering

Mixture-of-experts models consist of a set of experts and a gating network which combines the decisions from experts. One motivation for the mixture-of-experts model is based in the divide-and-conquer principle, which is common to the field of computer science. Using this some of the complex problems were decomposed into a set of relatively simple sub-problems [16]. In the mixture-of-experts model, the assumption is that there are separate problems within the larger underlying problem.

Each expert deals with different features from a different perspective, thereby resolving the small separable problems. Through the features of the mixture-of-experts model, it has been applied to traditional complicated recognition problems, and it is also suitable for the hand gesture recognition.

One of the considerations when implementing ME models is the way to separate the decision surface and generate experts. In order to divide the decision surface into several subgroups, clustering techniques can be applied. Some hard clustering techniques, e.g., k-means clustering, have been widely used. However, it is not easy to define crisp boundaries between several hand gestures to generate experts since there are several similar ones, and some gestures should belong to two or more groups. Thus, hard clustering techniques are not fit to be applied to the domain of hand gesture recognition so the FCM was proposed [16].

2.5.5. Neural Network

Neural Network is a learning algorithm that mimics the learning capability of a human neuron. It consists of n nodes connected to each other [8]. The weighted sum of input is passed through an activation function, which decides if a particular node wins or loses. Initially more than one node may win for a set of input. The network is then mentored so that only one node win's for any input.

Feed back or Back Propagation of the output is what enables learning in a neural network. One of the first systems to use neural networks in hand posture and gesture recognition was developed by Murakami. Hand postures were recognized with a three-layered neural network that contained 13 input nodes, 100 hidden nodes, and 42 output nodes, one for each posture to be recognized [7]. The network used back propagation; a learning mechanism that minimizes the error between target output and the output produced by the network, and achieved 77% accuracy with an initial training set. Accuracy increased to 98% for participants in the original training set when the number of training patterns was increased from 42 to 206. Symeonidis trained a single perceptron back propagation model to recognize hand gestures of American Sign Language. An accuracy of 78% was achieved on an overall basis [12].

Chapter 3

SYSTEM DESIGN AND CONSTRAINTS

Chapter 3

SYSTEMDESIGN AND CONSTRAINTS

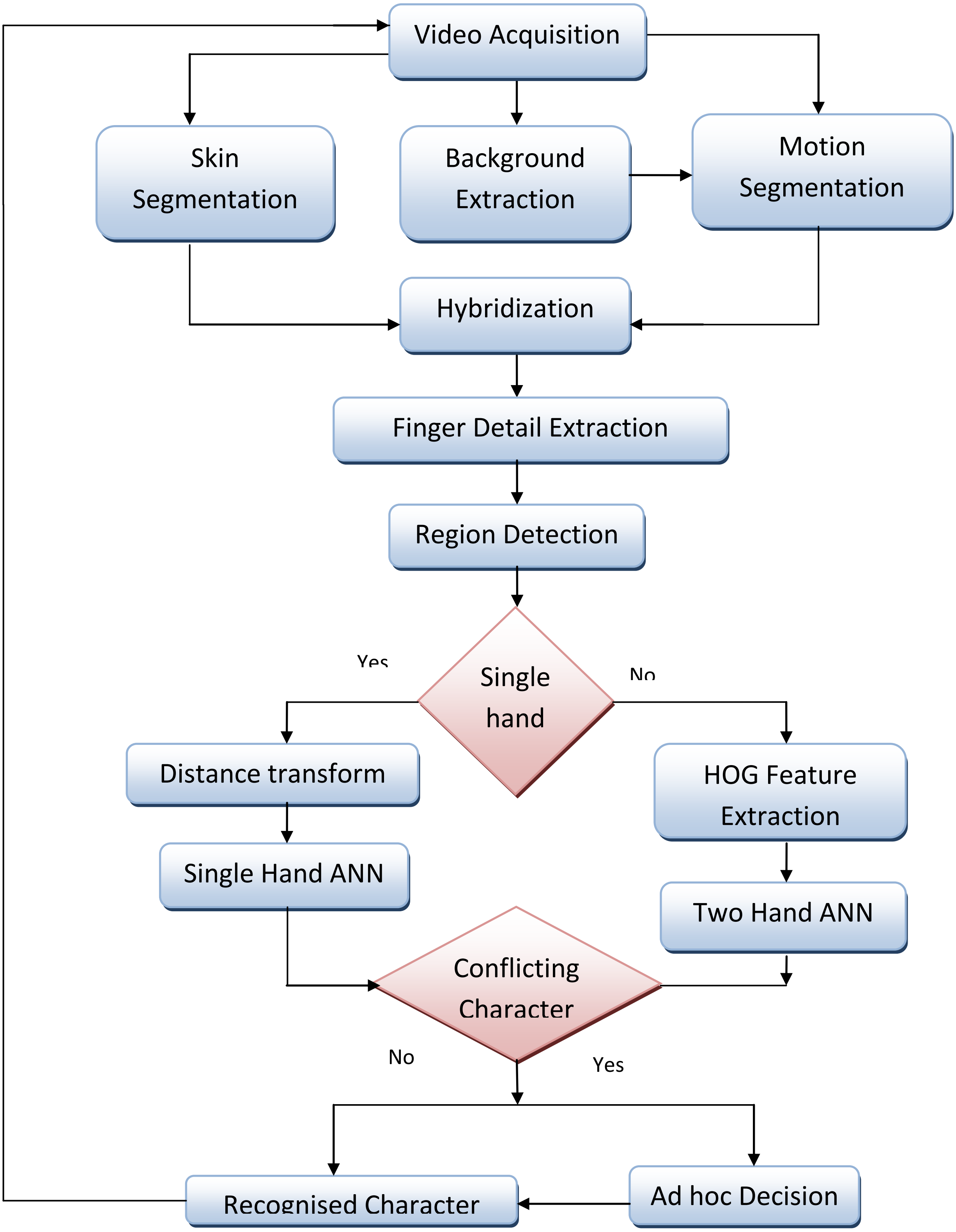


Fig 3.1 System Flow Diagram

The Figure 3.1 shows the finalized flow of the system. It shows the various activities that are performed. Indian Sign language contains both single hand gestures and two hand gestures. The single hand characters in Indian sign language are I, L, C, U, V, W, J, O. The two hand characters are A, B, D, E, F, G, H, K, M, N, P, Q, R, S, T, X, Y and Z. The input is taken from the webcam and segmentation is applied. From the segmented image, decision of whether the input is single hand or two hand is made. After making the decision corresponding neural network is applied. If it is a single hand gesture, the character will be recognized. The conflicting two hand characters like M, N, R appears similar after segmentation. If it is a two hand gesture, the non conflicting character is recognized and if it is a conflicting character, then it goes for an adhoc decision and the confusion is cleared out and the appropriate character will be recognized.

3.1 SYSTEM REQUIREMENTS

The input from the user is acquired with help of webcam and it is processed to recognize the hand gestures. The user makes the hand gestures by positioning the hand parallel to the plane webcam. The video is then processed to extract the hand region. The resolution of the webcam is kept at 640 x 480 pixels for better quality of video. The algorithm is implemented on a system with Intel Core i5 processor with speed of 2.3 GHz. The system has also been tested on an Intel dual core processor with a speed of 2.5 Ghz. Generally the performance was found to be better in the i5 processor. The preferred webcam configuration is a minimum of 2 mega pixels.

3.2 ASSUMPTIONS

Once the system starts, the user does not show any gesture initially for 30 seconds because the system assesses the background information. When the screen displays “Show the gesture” the user shows the gesture parallel to the webcam. The illumination of the environment must be even. When the user changes from a two hand gesture to a single hand gesture or vice versa, he has to separate the hands and then bring them back for the next gesture. For proper segmentation to take place the only moving skin coloured object in the frame must be the signer’s hands. The system has been implemented using Matlab R2011 version and is tested to be compatible with versions up to 2006.

3.3 SYSTEM FLOW

The following is the flow of the implemented system

- The user starts the system. The webcam is turned on automatically by the system.
- To aid the segmentation process, the background information is accessed in the first 30 seconds by taking the summation of the frames collected by the webcam.
- To customize the system for the signer's hand, 30 second frames are used when the user shows one of his hands, with all 5 fingers fully opened.
- The input frames that are acquired by the webcam may contain other objects in the background. So the hand region has to be separated from the background. This is done using the hybridized segmentation method that has been proposed.
- Initially the input frame is subjected to HSI based segmentation and then the input image is subtracted from the background to detect the moving objects in the image. The two resultant images are overlapped to obtain the segmented hand region.
- After segmentation, the details of the fingers like length of each finger, angle between the fingers and posture of the fingers are extracted for 30 frames.
- The user should show the gesture after the system displays the message "Show the gestures "to the user. The hand is segmented as specified above.
- The number of hand regions in the segmented frame is kept track of. If the number of regions is 1, the procedure for single hand is followed. Else if the number of regions is 2, the procedure for two hands is followed.

- Suppose the signer shows a 2 hand gesture, initially the number of regions will be 2. When the gesture is complete the number of regions becomes 1.
- At this point, the two hand feature extraction function is initialised and the HOG descriptor is extracted and fed into a trained neural network explained in chapter 6 which recognizes the character.
- If it is a single hand gesture, the number of regions is 1 and hence the distance transform function is executed for extracting the features like length of each finger, angle between the fingers and posture of the fingers. These features are fed into the corresponding single hand trained neural network explained in chapter 6.
- If the number of regions is neither one nor two, it means that either nothing is detected which means the illumination is insufficient or there is some other object that is being detected. In these cases the system is irresponsive.
- The looping stops when continuously 10 frames of blank screen is recognized.

Chapter 4

SEGMENTATION

Chapter 4

SEGMENTATION

The first step in any vision based application is the process of separating the region of interest from the background. This process is segmentation. In our problem, segmentation pertains to the segmentation of the hand region from the background. There are several constrains, which make segmentation a difficult problem in itself. There are several methods in which segmentation is carried out:

Colour based segmentation: Is useful when the object of interest is of a distinguishable colour to the background. It simply separates this particular colour from the background. One of the major decisions in this method is the selection of the colour space on which we operate.

Motion based segmentation: Here the object of interest is assumed to be moving and the background is assumed to be stationary.

Contour or Shape based segmentation: Here the object of interest is of a specific shape. Identifying and isolating the shape is the challenge that this method poses.

For our problem the region of interest is the hand region. Not using any sensors or colour gloves makes segmentation more challenging problem. This being said, the complexity of the problem should not hinder with the robustness and the efficiency (speed) of the system. This chapter explains the pros and cons of each of the above methods. The proposed segmentation technique that is used is also explained.

4.1. SKIN COLOUR BASED SEGMENTATION

Skin colour based segmentation is fairly simple to implement with the main decision being the colour space which is used. Several colour models have been proposed so far, a few predominant ones being, HSI, RGB, YCbCr colour models. For vision based applications one of the major factors in deciding the colour space is the dependence to brightness. Colour models with relatively less dependence on the brightness factor are favourable. In this respect, HSI colour model is identified as the best from the survey.

4.1.1. HSI Segmentation:

The Hue Saturation and Intensity colour model is used for skin colour segmentation. The Hue denotes the colour of the pixel, the Saturation denotes the depth of that particular colour and the Intensity denotes the brightness. It is sufficient to work with Hue and Saturation to identify the skin colour. The ranges for hue and saturation for the skin region in the Cartesian coordinate system was identified as given in Equation (4.1) and Equation (4.2).

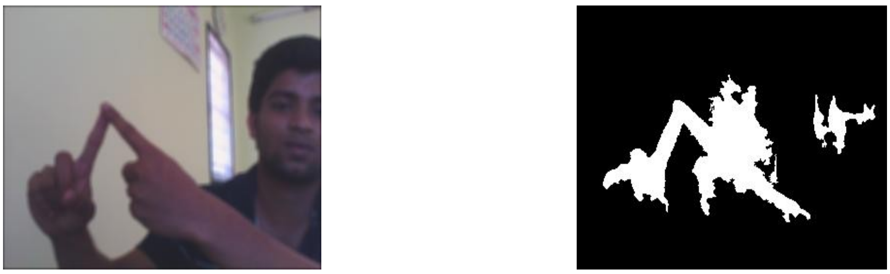
$$H<25 \text{ or } H>230 \qquad \text{Eq (4.1)}$$

$$S<25 \text{ or } S>230 \qquad \text{Eq (4.2)}$$

Where H is the Hue and S is the Saturation. The hue and saturation[15] values are normalized using Equation (4.3) and Equation (4.4) to a range between 0.0 to 1.0.Now the range of Hue and Saturation that has to be consider becomes 0.4 to 0.6. The normalisation is done to simplify the segmentation process.

$$H=H/255 \qquad \text{Eq (4.3)}$$

$$S=S/255 \qquad \text{Eq (4.4)}$$



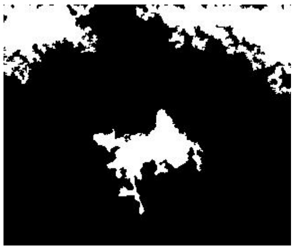
(a) (b)

Fig (4.1) A sample frame that has undergone skin segmentation

a) The input frame for skin segmentation b) Skin segmented image

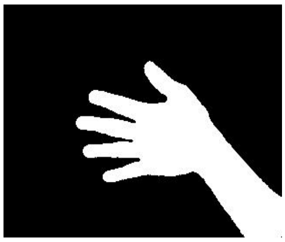
The Figure 4.1 shows a sample frame which has been subjected to skin colour segmentation using HSI colour model. The HSI model is found to be capable of segmenting the skin region when the illumination was even. But when the illumination of the environment was uneven it was found that the segmentation was rather erratic. Figure 4.2 shows the same gesture taken under different illumination conditions. It is observed that the segmented region is very irregular. This method identifies the skin region based on some colour range, so any region in the image whose hue and saturation values fall under this range, will be detected, for example the face region. This is not desirable because only the hand region has to be detected for proper processing to happen.

(a)



(b)

(c)



(d)

Fig (4.2) Effect of illumination on skin segmentation

(a) Sample frame under bad illumination (b) Skin segmented image under bad illumination (c) Sample frame under good illumination (d) Skin segmented image under good illumination

4.2. MOTION BASED SEGMENTATION

One factor that is both a boon and a bane in the problem at hand is that the system takes dynamic input through a webcam. This means that there is constant motion of the hands when giving input to the system. So, using motion based segmentation makes sense. So initially for the first 5 frames the background is assessed. Then the user shows the gesture.

The current frame is subtracted from the background frame, which gives the regions that are moving. Equation (4.5) shows the absolute difference D(i,j) between a pixel in the background image B(i,j) whose position in the image array is (i,j) and the pixel in the same position in the current frame C(i,j). The image size is taken as mxn. While this method has the advantage of being able to detect movement in any illumination condition, it detects all the moving objects from the image. The slightest difference is eliminated by thresholding as given in Equation 4.5.

$$D(i,j)=|C(i,j)-B(i,j)| \text{ where } i=1,2,3,\ldots m \text{ and } j=1,2,3,4,\ldots n \qquad \text{Eq (4.5)}$$

This threshold was determined after experimenting using several values. The objects other than the hand region exceeded the threshold in some cases. For example a fan that is switched on in the background would be detected as it would exceed the threshold. So it is not possible to use motion based segmentation as a standalone entity to detect the hand region. Figure 4.3 shows a sample frame subjected to motion based segmentation and the corresponding background frame respectively.

$$T>=36 \qquad \text{Eq (4.6)}$$

(a) (b)

Fig (4.3) Motion Segmentation

(a) Input Frame for Motion Segmentation (b) Motion segmented image

4.3. CONTOUR BASED SEGMENTATION

The main advantage of the contour based segmentation, proposed in this chapter is that it does not require an additional hardware. The Chan-Veese [28] method for segmentation is one among the most powerful and flexible. It can segment different kinds of images. It is based on Mumorf-shah [28] function for segmentation that is mostly used in medical field. It is an energy minimization method that will be reformulated to level set formulation to solve the problem in an easier way. The Chan and Vese model for image segmentation is one among the most succesful minimization problems that uses level set formulation and utilizing the image statistics inside and outside the curve to form the contour. Initially a curve is drawn aroung the object considereing the fact that the required object is in the middle of the frame.

The basic idea of the proposed method is to deform the contour that minimizes the given energy function to result in a desired segmentation. The local region-based framework is described in this section for the proposed method. The foreground and background are described in a local small regions because this method is not based on global region models.

The local regions are analysed it leads to a family of local energies at every point, so optimisation of the local energy at each point seperately which will minimize or maximize the computed energy in that particular local region. Let Ω be a bounded openset on R, and the given image be represented as I: $\Omega \rightarrow R^2$. And let C be the piesewise parametrised curve represented in the zero level set. Let the region inside C be denoted as ω and the reigion outside it be $\Omega \backslash \omega$. Let c_1 be the average of the pixels' intensity inside C, and c_2 be the average of the intensity outside C.

The Chan and Vese Model uses a case in the Mumford-Shah method to evolve the cruve. It finds the pair of (u,C),

where u is the piecewise smooth approximatin of the image I, and C will be the closed curve.

$$\text{Ems} = \int_{\Omega} |I(x,y) - u(x,y)|^2 + \mu \int_{\Omega/C} \text{Length(C)} + \nu.\text{Area(C)} \qquad \text{Eq(4.7)}$$

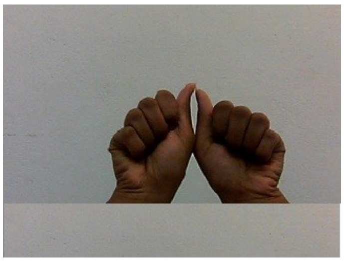
where (x,y) is the spatial co-ordinate of the pixels in the image and μ, is a positive integer value to smoothen the contour. Minimization of the energy function is the only way to solve this problem.

The Chan and Vese is an alternative of the Mumford method. The objective of Chan and Vese algorithm is to minimize the energy funtional represented in $F(c_1,c_2,C)$ that is defined as:

$$\text{F(c1, c2, C)} = \mu.\text{Length(C)} + \nu.\text{Area(inside(C))} + \lambda 1 \int |I(x,y) - c_1|^2 dxdy + \lambda_2 /Ix,y-c1/2dxdy \qquad \text{Eq(4.8)}$$

where I(x,y) will be the spatial co-ordinate of the pixel. And $\mu \geq 0$, $\nu > 0$, λ_1, $\lambda_2 > 0$ are all fixed parameters that will set by the user according to the classs of the image, which also minimizes the Mumford shah method. Here μ is the penalty on the total length of the curve. It decides on the smoothening term of the curve. To get a smoother curve, the value has to increased. The ν is the penalty on the total area in the foreground of an image. The term $(I(x,y)-c_1)$ is propotional to the variance of the gray scale imag in the forground of an image and it measeures the uniformity in the terms of pixel intensity. The term $(I(x,y)-c_2)$ does the same for the background of an image. The sum of these two terms will result in an uniform foreground and background region. For eg, when λ_1 is given an higher value than λ_2 the final segmentation will be more unifrom in the foreground region than the background. Figure 4.4 shows the input frame and segmented frame for gesture A by applying contour based segmentation.

Even though it can be understood that the contour based segmentation is the most accurate segmentation it takes too many iteration to produce that result. So it is not possible to opt for contour based segmentation because for such dynamic systems.



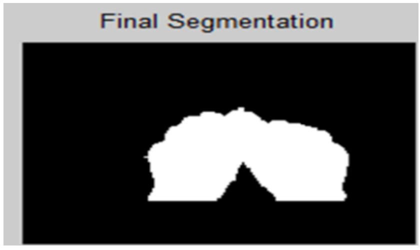


Fig 4.4 Contour based segmentation

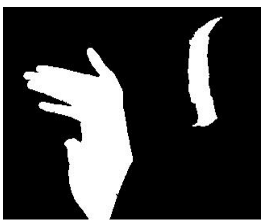
(a) Input frame for Contour Segmentation (b) Contour Segmented image

4.4. HYBRIDIZATION METHOD

To improve the accuracy of detection and to reduce the dependence on illumination the results obtained through skin segmentation and motion segmentation is combined and the intersecting regions are retained. Figure 4.5 (a) shows a sample frame. Figure 4.5 (b) shows the skin segmented image of the corresponding image. Figure 4.5 (c) shows the motion segmented image of the same. Figure 4.5 (d) shows the final hybridized image.

(a)

(b)



(c)

(d)

Fig 4.5 Segmentation Process

(a) Sample Input frame for hybrid segmentation (b) Skin Segmented Image in hybridization

(c) Motion Segmented Image in hybridization (d) Final segmentation

Table 4.1 shows the comparison between various segmentation techniques. The accuracy is calculated for 100 frames. A frame which has the hand region without any background objects is identified as the correctly segmented frame. Based upon this the accuracy is calculated.

TABLE 4.1 Comparison between the segmentation techniques

Method	Time for execution (100 frames) in secs	Accuracy (100 frames)
Skin Segmentation	14.270	84
Motion Segmentation	15.4717	93
Contour Based Segmentation	18756.3	100
Proposed Method	34.9714	98

After segmentation using hybridization method, in order to identify whether the user shows single hand or two hands, the number of hand regions in the frame is tracked. If a single region is detected, the single hand features are extracted as explained in chapter 5. If two regions are detected, the two hand features are extracted as explained in chapter 6.

Chapter 5

SINGLE HAND FEATURE EXTRACTION

Chapter 5

SINGLE HAND FEATURE EXTRACTION

Indian Sign language contains both single hand gestures and two hand gestures. The single hand characters in Indian sign language are I, L, C, U, V, W, J, O. As single hand characters are less complex they can be recognized with minimal features. This chapter explains the techniques implemented for extracting features like centroid, state of the fingers, length of the fingers and angle between them which helps in recognizing the single hand characters.

5.1. CIRCULAR PROFILING

Circular profiling is a method in which a circle is drawn on the hand. From the intersecting points of the circle and the hand region, the numbers of open fingers are found.

5.1.1 Centroid Identification

The first step in this method is the identification of the centroid of the hand region. The segmented binary image in which the white pixels represent the hand region and black pixels represent the background region is used to identify the centroid using the equation 5.1.

$$X=\frac{\sum_{i=0}^{k} x(i)}{k} \text{ and } Y=\frac{\sum_{i=0}^{k} y(i)}{k} \qquad \text{Eq (5.1)}$$

Where X and Y are the (x, y) coordinates of the centroid of the hand region, xi and yi are the (x, y) coordinates of all the white pixels in the binary image which represents the hand region and k is the number of white pixels in the hand region.

5.1.2 Active Finger Identification

With the centroid determined using the above method, the next step is to draw a circle of radius r such that it passes through all the fingers. The distance between the centroid and the farthest skin pixel d_f is found. A circle is drawn with centroid as centre and radius d_f. If the circle is too small or too big the fingers will not be detected. The radius of the circle was found using the trial and error method as in equation 5.2.

$$R=0.7 \times d_f \qquad \text{Eq (5.2)}$$

From the circle, the number of white to black transitions which occur along the circumference of the circle is determined. Figure 5.1 shows a sample frame after circular profiling.

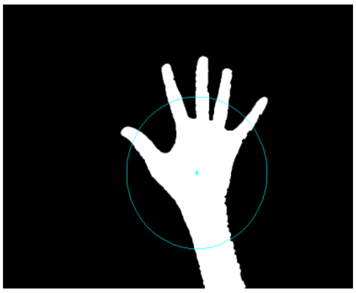


Fig 5.1: Sample frame after circular profiling

If the radius is calculated with a multiplying factor greater than 0.7 it does not intersect the thumb. This situation is illustrated in Figure 5.2 (a). If the multiplying factor is smaller than 0.7, say 0.5, the circle passes within the palm region. This situation is illustrated in Figure 5.2 (b).

(a) (b)

Fig. 5.2:(a) shows the circle drawn with r = 0.9 x d_f(b) show the circle drawn with r = 0.5 x d_f

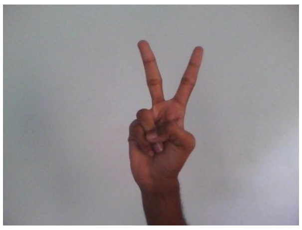
In Figure 5.1 it can be seen that the number of white to black transitions is 6 but the number of active fingers however is only 5. This error is due to the wrist region which accounts to one white to black transition. Therefore the number of active figures is one less than the number of transitions. The number of active figures is given by the Equation 5.3.

$$\text{No of active fingers} = T-1 \qquad \text{Eq (5.3)}$$

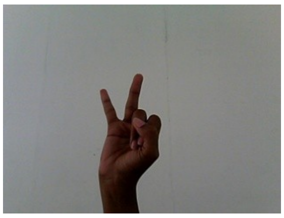
where T denotes the number of white to black transitions.

5.1.3. **Disadvantages**

The circular profiling helps only in identifying the number of fingers opened or closed. In Figure 5.3 it can be seen that if any two fingers are open they are recognized as V because circular profiling cannot identify which finger is open. The feature extracted from this technique alone is not sufficient for differentiating the 26 English characters of Indian Sign Language.



(a)



(b)

Fig 5.3: (a)Indian Sign language V (b)Gesture wrongly recognized as V

So a technique that could identify the state of the fingers was adapted.

5.2. THE DISTANCE TRANSFORM APPROACH

5.2.1. Centroid identification

Distance transform is performed on the segmented image. The segmented hand region is in white colour and the background is in black. The distance between each white pixel with the closest background pixel is identified. And the pixel intensity is made proportional to distance. The pixel with the maximum intensity represents the centroid of the hand. Figure 5.4 shows a sample frame after it undergoes distance transform operation. The centroid is the pixel in cyan color.

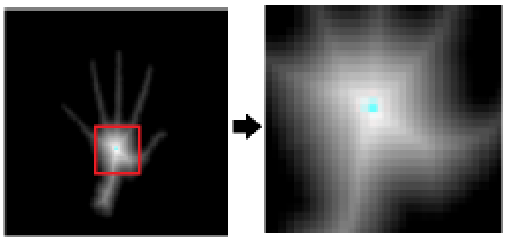


Fig 5.4: Centroid obtained using distance transform

5.2.2. Finger Identification

Depending on the centroid and the segmented hand region the appropriate size for the structuring element (a disc) is identified. The segmented image is eroded with the structuring element as given in equation (5.4). This results in palm region.

$$A \ominus B = \{z \in E | B_z \subseteq A\} \qquad \text{Eq(5.4)}$$

where A is the binary image in E, B is the structuring element and B_z is the translation vector of B at pixel z. When the image obtained after erosion as shown in Figure 5.5 (a) is subtracted from the original segmented image some regions of the palm will be found along with the finger region hence making it difficult for the individual finger regions to be identified so dilation is performed as specified in equation 5.5.

$$A \oplus B = \{z \in E | (B^s)_z \cap A \neq \emptyset\}. \qquad \text{Eq(5.5)}$$

where A is the binary image in E, B is the structuring element and $(B^S)_z$ is the symmetric of B at pixel z.

(a) (b) (c)

Fig 5.5: (a) Eroded palm region (b) Dilated palm region (c) Extracted fingers

The palm region so obtained as in Figure 5.5(b) is subtracted from the segmented hand. This leaves the fingers and the wrist portion as disconnected regions. The area of the wrist region is greater than the finger region. So if the area of the region lies within the threshold it is identified as finger region which is shown in Figure 5.5 (c). The number of such regions gives the number of fingers. The distance transform method as mentioned in Section 5.2.1 is applied for each finger region. This helps to identify the major axis in each finger. The farthest point and closest point to the palm region along the major axis is identified as the finger tip and the base point respectively. The base point is plotted in green color and the finger tip point is plotted in blue color as shown in Figure 5.6.

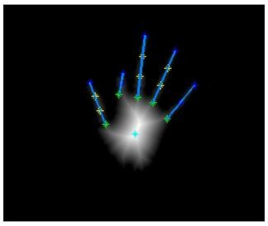


Fig 5.6 Image obtained after distance transform

5.2.3. Extraction of features

Once the fingers are separated the next step is to extract information from them to distinguish one gesture from another. There is a initial feature extraction phase in which the user has to keep all the five fingers open for the initial 30 frames to help extract the features like distance between the fingers and the length of the fingers. This is done because the length and distance between the fingers varies from person to person and is also affected by the closeness of hand to webcam.

The distance between each adjacent base point are calculated and stored in an array. The distance between the base point of thumb and index finger is maximum this criterion helps identify the thumb position. Figure 5.7 shows the user showing the right and the left hand. The thumb position from the origin of the screen for the left and the right hand is 1 and 5 respectively.

(a) (b)

Fig 5.7: (a) Thumb position Left hand (b) Thumb position Right hand

Using the position of the thumb, we identify whether the signer is showing the left hand or the right hand. Since the distance between two fingers satisfies the commutative property, it is enough to store it in an upper triangular matrix. This is a 5x5 constraint matrix containing distance between all pairs of fingers. The average length of each of the fingers is estimated from the first 30 frames as show in equation 5.6. The average is taken because the length cannot be decided from single frame as the illumination is not constant.

$$A(i) = \sum_{i=1}^{5} \sum_{j=1}^{30} L(i,j)/30 \qquad \text{Eq (5.6)}$$

During the testing phase which succeeds the feature extraction phase, the user shows the gesture. The centroid of the hand, the base point and the finger tip of each finger is identified using the distance transform procedure .The distance between every consecutive pair of the base points of finger is found. The Euclidean distance between this value and every element of the constraint matrix is taken. The row and column index of the value with the minimum Euclidean distance is noted. The index helps identify the finger which is open

The index value of 1,2,3,4,5 represent thumb , index , middle, ring and little finger respectively when left hand is shown. The order is reversed when right hand is shown. Figure 5.8 shows a screen shot of the results of the feature extraction phase. 'Allfing' is the variable that stores the constraint matrix for the distance between the fingers. 'tnp' represents the thumb position estimation. The lengths of the fingers are determined using the Euclidean distance between base point and the finger tip. 'arr' represents the length of the finger. The finger lengths obtained is compared with the lengths obtained during the feature extraction phase to determine if the fingers are fully open, half open or fully closed. If the length of the finger is about one third of the original length then it is considered as semi closed, if it is about two third of the original length then it is considered half closed. The fingers that are not detected are considered fully closed. Table 5.1 shows the states for single hand characters.

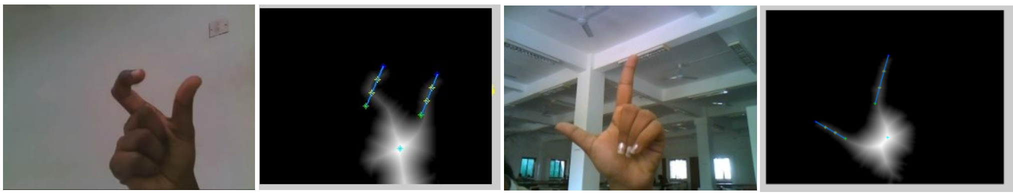
```
Starting Video Device
arr =

   56.8832   66.7736   75.9451   72.8670   51.1239


tnp =

     4


fingdis =

   26.9590   20.0460   23.7162   46.8378

Thumb is finger 5
allfing =

  1.0e+003 *

    1.0000    0.0270    0.0441    0.0611    0.0925
    1.0000    1.0000    0.0200    0.0422    0.0835
    1.0000    1.0000    1.0000    0.0237    0.0689
    1.0000    1.0000    1.0000    1.0000    0.0468
    1.0000    1.0000    1.0000    1.0000    1.0000
```

Fig 5.8 Feature extraction phase results

TABLE 5.1 States of the fingers for each gesture

Characters	Little finger	Ring finger	Middle Finger	Index finger	Thumb finger
C	Fully Closed	Fully Closed	Fully Closed	Half open	Fully open
I	Fully Closed	Fully Closed	Fully Closed	Fully open	Fully closed
J	Fully Closed	Fully Closed	Fully Closed	Half open	Fully open
L	Fully Closed	Fully Closed	Fully Closed	Fully open	Fully open
O	Fully open	Fully open	Fully Open	Half open	Half closed
U	Fully Closed	Fully Closed	Fully Closed	half open	Fully open
V	Fully Closed	Fully Closed	Fully open	Fully open	Fully Closed
W	Fully Closed	Fully open	Fully open	Fully open	Fully Closed

When alphabets like C, J, L and U were segmented and distance transform was applied, there was no much difference in the extracted features due to improper segmentation which happens due to illumination problems. So an additional feature which was extracted was the angle between the fingers. The angle is calculated between the major axis of each finger and a horizontal line drawn through the centroid of the hand. This angle between the fingers is used to further differentiate between the gestures. Figure 5.9 (c) and figure 5.9 (d) shows the gestures U and L which have the same two fingers open. . Table 5.2 shows the angle range between the fingers for the different gestures.



(a) (b) (c) (d)

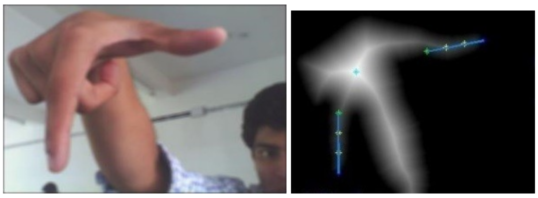
Fig 5.9: (a) Alphabet U (b) Alphabet L(c)Segmented U after distance transform(d)Segmented L after distance transform

TABLE 5.2 Range of angle between pairs of fingers

Characters	Little-Ring	Ring-Middle	Middle- Index	Index-Thumb
C	0	0	0	150
I	0	0	0	0
J	0	0	0	150
L	0	0	0	90
O	0	0	0	0
U	0	0	0	150
V	0	0	45	0
W	0	30	30	0

A 10-tuple feature vector is extracted at the end of this phase and is used as the input for the artificial neural network as explained in Chapter 7. First 5-tuples represents the posture of the fingers, next 4-tuples represents the angle between the five fingers and the last tuple is to say whether the hand is right or left. The value of each posture is indicated as 0, 1, or 2 depending on whether the finger is fully closed or semi-closed or fully open respectively. A value 1 or 0 representing hand in the feature vector indicates the right hand and the left hand respectively.

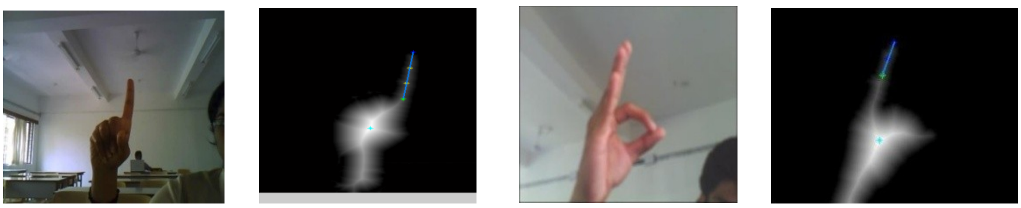
Angle feature resolved most of the conflict except for alphabets C and J because their angles were similar. The segmented image of J after distance transform is shown in Figure 5.10 (b). So the ad hoc decision method was done as explained in Chapter 7.



(a) (b)

Fig5.10: (a) Alphabet J (b) Segmented J after distance

During the segmentation of alphabet O, only one finger is detected as shown in Figure 5.11 (d) so it is difficult to differentiate it from I as shown in Figure 5.10(b) based on the state of the finger. Therefore the ad hoc decision is used to distinguish them as mentioned in Chapter 7.



(a) (b) (c) (d)

Fig 5.11: (a) Alphabet I (b) Segmented I after distance transform (c) Alphabet O (d) Segnmented O after distance transform

Chapter 6

TWO HAND FEATURE EXTRACTION

Chapter 6

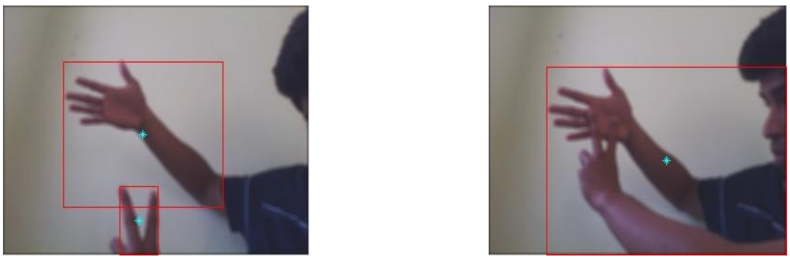
TWO HAND FEATURE EXTRACTION

The major challenge with extracting features from two handed characters is that they involve overlapping gestures. To resolve this issue, two hands are separated based on skin colour segmentation and histogram. But since the area of the region involving the two hands is large, the difference between the two hands in the gesture is less. So the skin colour segmentation and histogram based methods failed.

6.1. COLOUR BASED SEGMENTATION

6.1.1 Tracking hand motion

For separating the two hands, we have to detect the two hands. To do this, the movement of the hand is detected by detecting the regions in the image. A bounding box is drawn around the regions and only the overlapping hand is considered to be moving where as the other hand is considered to be static. The left bottom and top right coordinates of the bounding box which is moving are stored in each frame as shown in Figure 5.2. When the hands overlap the number of regions extracted from that frame will be one. Using the coordinates of the bounding boxes collected in the direction of movement of the overlapping hand, the current position of the overlapping hand is predicted by extrapolating the bounding boxes. This region is then cropped out for further processing. Figure 6.1 shows the various stages of the region detection phase.



(a) (b)

Fig 6.1: (a) Region detection with two bounding box (b) Region detection with one bounding box

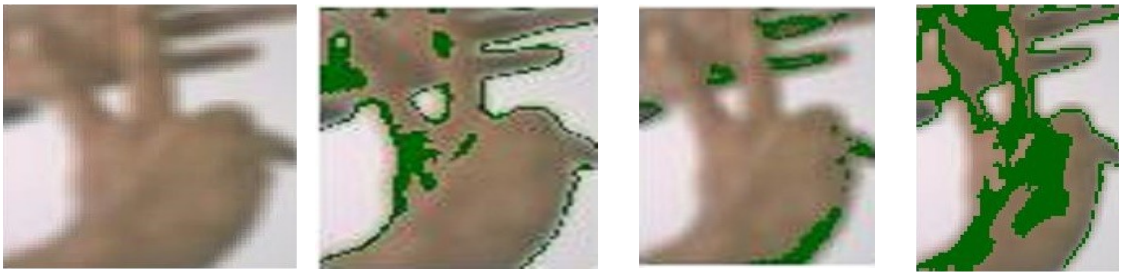
The linear extrapolation equation is given in equation 6.1.

$$y(x_*) = y_{k-1} + (((x_*-x_{k-1})/(x_k-x_{k-1}))*(y_k-y_{k-1}) \qquad \text{Eq(6.1)}$$

Where (x_*,y_*) is the point which is extrapolated.(x_k,y_k) and (x_{k-1},y_{k-1}) are the points of the overlapping hands bounding box in the previous frames.

6.1.2. Contrast Stretching

The cropped image obtained after extrapolation was subjected to selective contrast stretching. This is done to find the range that the overlapping hands lie in. Figure 6.2(a) shows the cropped image. Figure 6.2 (b), (c), (d) shows the extracted regions for different contrast ranges shaded in green after contrast stretching the cropped image.



(a) (b) (c) (d)

Figure 6.2: (a) Cropped region (b) Contrast stretched image under the range: 0.5-0.7(c) Contrast stretched image under the range: 0.75-0.9(d) Contrast stretched image under the range: 0.3 to 0.5

From the images it is clear that we could not obtain a clear range to separate the two hands using contrast stretching. The reasons for this being that for some people, the fore hand and back hand are of almost the same complexion and hence segmenting them based on colour was not possible. However when tested with people whose forehand and backhand had a significantly different skin tone, segmentation yielded positive results. Figure 6.3 shows a sample frame of a person with contrasting fore hand and back hand skin tone.

(a) (b)

Fig 6.3: (a)Input image (b) Contart stretched Image for Success Frame

But since the system is being developed for any user and that robustness is an element that cannot be compromised, this technique cannot be used. So an alternate approach had to be adopted.

6.2. SHAPE BASED SEGMENTATION

6.2.1 Convex Hull

Shape based features were a tempting venture because each of the gestures was found to have distinct shapes. So after detecting the region, the smallest polygon that encompasses the entire region is drawn. The polygon is called the convex hull of the hand region that is detected.

After determining the convex hull, the point having maximum convexity defects along the boundary of the hand region between each vertex of the convex hull is determined. Simply put, the points with maximum convexity defects are the concave regions in the hand, for example the ridges between fingers. Figure 6.4 shows a sample frame after detecting the convex hull. The red circles represent the points of maximum convexity defects. The yellow lines represent the edges of the convex hull.

Fig 6.4 Convex hull and convexity defects

The convex hull drawn was inconsistent based on the result of segmentation. There were a lot of faulty points when the segmentation of the hand region was slightly varied or when the user shows the wrist region. But the segmentation is subject to irregularities because, of the illumination criteria as shown in Figure 6.5. So using shape based features from the convex hull would not yield consistent and distinct features. So an alternative approach had to be adopted. Figure 6.5 shows inconsistent convex hull detection.

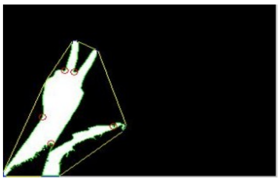


Fig 6.5 Inconsistent convex hull and convexity defects

The convex hull obtained is completely different. The number of convexity defects detected points varies as shown in Figure 6.5. So this method does not give robust and unique features.

6.3 HISTOGRAM OF ORIENTATION GRADIENT

6.3.1 Introduction

Histograms of Oriented Gradients (HOG) are feature descriptors used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

6.3.2. Extraction Of HOG Descriptor

From the segmented hand region, the information required to distinguish one gesture from another has to be extracted as feature vector. The feature vector should be of uniform size for any input image and should be independent of the location of the hand in the segmented image. To ensure this, the smallest bounding box enclosing the hand region is cropped and resized to fixed size.

The orientation gradient gives the direction and magnitude of the gradients present in the image locally. HOG divides the direction into n radial bins and forms a histogram of magnitudes in them. The gradients Gx and Gy are calculated using the filters X and Y as given in Equation 6.2.

$$X= [-1\ 0\ 1] \text{ and } Y= [-1\ 0\ 1]^T \qquad \text{Eq(6.2)}$$

Where T represents Transpose, the resultant magnitude is calculated using the Equation 6.3.

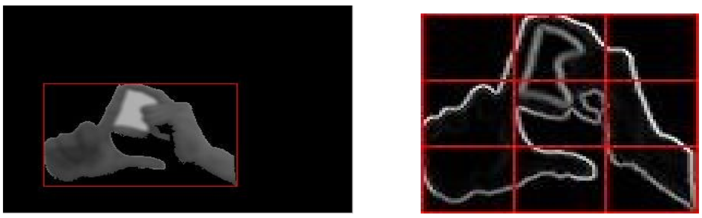
$$\text{Magnitude} = ((Gx)^2+(Gy^2))^{1/2} \qquad \text{Eq(6.3)}$$

$$\text{Gradient angle} = \text{atan2}\left(\frac{\partial f}{\partial y}, \frac{\partial f}{\partial x}\right) \qquad \text{Eq(6.4)}$$

Where $\frac{\partial f}{\partial y}$ is the gradient in x direction and $\frac{\partial f}{\partial x}$ is the gradient in y direction.

The hand region is divided into blocks and HOGs are extracted from each block as a feature vector. In order to reduce the effect of illumination and slight variation in the gesture, the feature vectors are normalized considering a small neighborhood.

After the hybridized segmentation technique Figure 6.6 shows the cropped hand.



(a) (b)

Fig 6.6: (a) Bounding box (b) Extracted Image with blocks

The cropped image is then resized to 90x90 pixels. In this system, each image is divided into 3x3 blocks as shown in Fig 6.6(b), each block containing 30x30 pixels in them. The gradient angle obtained using Equation 6.4 from each block is divided into 12 bins each denoting 30 degrees. Histogram of each angular bin is taken as the sum of the gradient magnitudes that fall in that angular range. Thus the resultant descriptor is a 9x12 matrix that contains the 12 bin values of each of the 9 cells. The 9x12 descriptor is normalized to a value between 0 and 1 as shown in Equation 6.4.

$$H2=H2/\ (\text{norm}\ (H2) + 0.01) \qquad \text{Eq}(6.5)$$

Where H2 is the HOG descriptor.

Figure 6.7 shows the orientation histogram for the image in Figure 6.6(b) considering it as a single cell. Figure 6.8 shows the quiver plot of the input image obtained after being subjected to the HOG normalization. The quiver plot gives the magnitude and the direction of each vector in the image.

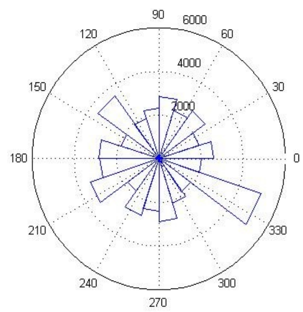
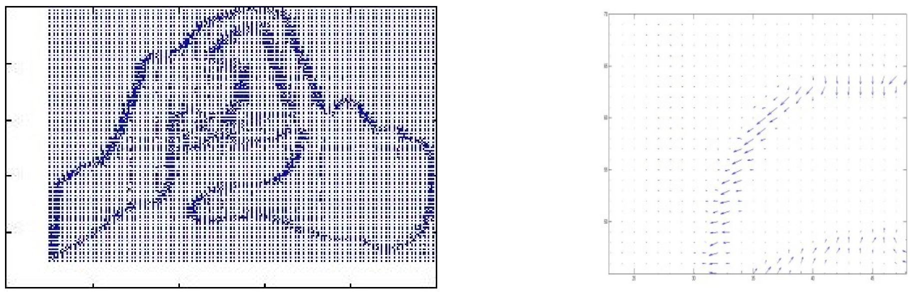


Fig 6.7: Orientation histogram



(a) (b)

Fig 6.8:(a) Plot in actual size (b) Enlarged view of the plot

The 9 x12 HOG vector that is obtained fed as input to the neural network explained in chapter 7. The reason behind using the HOG descriptor is that is has a unique pattern for each of the gestures and slight variations in showing the gestures or inconsistencies had almost the same features. Different gestures had enough variation in them to be classified as different classes.

Figure 6.9 shows the 9x12 feature descriptor that is extracted for the letter A.

-1.290485e+002	-6.444891e+001	-2.754259e+001	-6.696289e+001	-2.657149e+001	-1.061060e+001	9.549584e+000	2.476393e+001	1.366472e+001	6.019176e+001	9.651806e+001	6.881630e+001
-2.059866e+002	-5.819506e+001	-2.138844e+001	-4.578381e+001	-3.073031e+001	-2.385448e+001	7.080111e+000	6.510621e+000	1.107148e+000	2.402559e+001	3.088625e+001	2.043961e+002
-1.095778e+002	-5.727517e+001	-7.839276e+000	-2.396046e+001	-1.973893e+001	-1.188623e+001	7.497909e+000	1.441812e+001	1.801687e+001	6.075655e+001	1.166662e+002	4.476621e+002
-1.100576e+002	-1.108560e+002	-5.065157e+001	-1.049731e+002	-2.889958e+001	-9.561935e+000	1.398922e+001	4.815188e+001	5.710115e+001	1.212528e+002	1.168023e+002	1.451036e+002
-3.203235e+001	-1.649993e+001	-1.768191e+000	-1.099557e+001	-3.332348e+000	-5.188740e-001	6.639355e+000	6.135799e+000	0.000000e+000	2.042035e+001	4.712389e+000	7.408272e+001
0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000	0.000000e+000
-2.082771e+002	-4.096316e+001	-2.288007e+001	-6.375879e+001	-2.027886e+001	-1.139659e+001	9.071213e+000	2.666631e+001	2.024546e+001	6.479285e+001	6.730778e+001	1.832497e+002
-2.612103e+002	-7.021190e+001	-2.682994e+001	-3.606385e+001	-1.939338e+001	-1.159864e+001	1.942468e+001	2.148570e+001	1.155640e+001	3.713430e+001	2.134764e+001	1.028471e+002
-2.423448e+002	-4.759049e+001	-1.962344e+001	-3.240443e+001	-9.088742e+000	-6.204512e+000	3.273170e+001	4.102674e+001	1.739720e+001	1.002947e+002	5.896755e+001	8.736597e+001

Fig 6.9: Feature descriptor

Chapter 7

NEURAL NETWORK

Chapter 7

NEURAL NETWORK

Even though a brute force method could be used to classify single handed characters there are some inconsistencies in detection because determining a fixed range for angles and for the extent of closure is difficult. Also, different people show a particular gesture differently. Two different back propagation neural networks are used for recognizing the single hand and two hand gestures. Its performance is compared with other architectures such as cascade feed forward and feed forward neural networks.

7.1. NEURAL NETWORK ARCHITECTURE

The general architecture of the back propagation neural network used in this project consists of one input layer with n neurons, two hidden layers with m and o neurons respectively and one output layer with p neurons as shown in Figure 7.1.

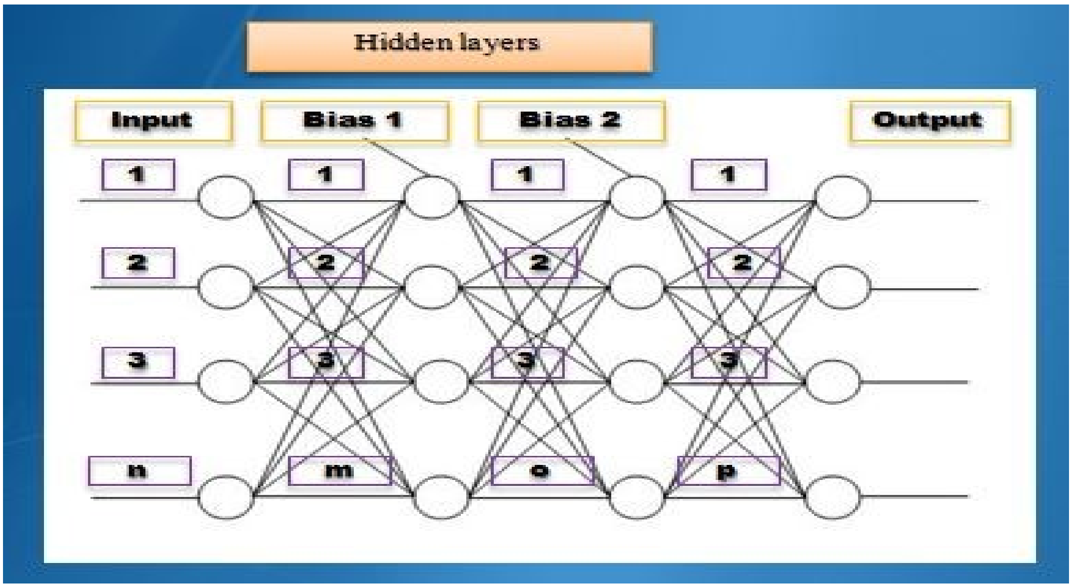


Fig 7.1 Generalised neural network architecture

The activation function used is tansig which is reperesented in Figure 7.2 and the equation is shown in Equation 7.1. it is continuous with values ranging from -1 to +1.

$$\tan(x) = \frac{2}{(1+e^{-2x})-1} \qquad \text{Eq(7.1)}$$

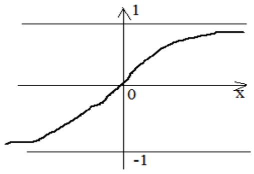


Fig 7.2 tansig function

The output of the input layer is calculated as the dot product (I) of the input vector X and the weight vector W as shown in Equation 7.2 and on applying the activation function F on it as given in Equation 7.3 where Y_j represents the output of the 'jth' neuron in that layer.

$$I = X.W \qquad \text{Eq (7.2)}$$

$$Y_j = F(I_j) \qquad \text{Eq (7.3)}$$

The output vector Y_j becomes the input vector for the next layer. In presence of bias 'b_j' the output of a neuron j is calculated as equation 7.4.

$$Y_j = F(I_j + b_j) \qquad \text{Eq(7.4)}$$

Hence the output of hidden layer 1 becomes the input of hidden layer 2 and ouput of hidden layer 2 becomes the input of the output layer neurons.

During the training phase, the weights are randomly initialized to some values. These weights have to be adjusted according to the desired output 'D_{ij}'. The error in the output is calculated using Equation 7.5 where Y specifies the computed output of the output layer neurons.

$$E_j = Y_j.\left(1 - Y_j\right).(D_j - Y_j) \qquad \text{Eq (7.5)}$$

The error term for the other layers is calculated using Equation 7.6.

$$e_j = Y_i.(1 - Y_i).\sum e_k.w'_{jk} \quad \text{Eq (7.6)}$$

Where e_k is the error term of each neuron k in the layer that is successive to the layer getting processed. The weights are updated using Equation 7.7.

$$W_{ij} = W'_{ij} + R.E_j X_j \quad \text{Eq (7.7)}$$

Where R is the Levenberg-Marquardt Learning Rule and E_j is the error term which is calculated using Equation 7.5.

In the training of the neural network, the features of the training characters are given individually till the desired output is obtained for that particular character. If that gesture is identified correctly, then weight updating is done for the next character. The training of the neural network stops when the error term tends to be negligible till the input layer. In case of neural network testing phase, the output Y_j is used to classify the gestures.

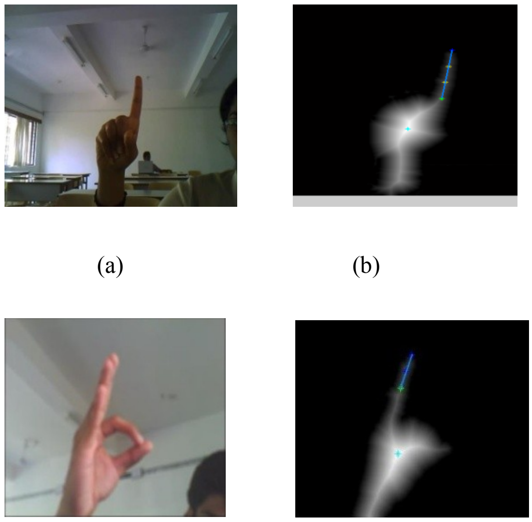
The minimum gradient of 0.0087 was reached during the 100th epoch and this was the stopping criteria.

7.2 SINGLE HAND NEURAL NETWORK

The back propagation network has 1 input layer with 10 neurons which is the 10-tuple feature vector as explained in chapter 5. The hidden layer has 10 neurons each and the output layer has 7 neurons which consist of L, C, U, V, W, I and a class which does not belong to any of the above classes. The class that does not belong to any of the above classes includes gestures with four fingers fully opened; five fingers fully opened and so on. As mentioned in chapter 5, since O has the same feature as I and J has the same feature as C and L after applying distance transform, it is said that O conflicts with I and J conflicts with C and L. Hence separate output neurons are not specified for O and J.

7.2.1 Resolving Conflicts

When the neural network outputs the character I there is an adhoc decision made based on whether the gesture has a region enclosing a hole or not. Figure 7.3 shows the images of gesture O and I.



(a) (b) (c) (d)

Fig 7.3: (a) Alphabet I (b) Distance transform image of I (c) Alphabet O (d) Distance transform image of O

The letter O consists of an inner region inside the segmented hand region and only one finger is identified in the feature extraction process as shown in Figure 7.3(d). It gets confused with Letter I as Letter I also has only one finger fully open. So the inner boundary hole is identified in the segmented hand region and its area is calculated. The inner region is plotted in green colour as shown in Figure 7.4. If the area of the hole is greater than the fixed threshold which is 30, it is recognized as Letter O. The threshold is used in order to avoid small regions inside the hand region, which is considered to as noise.

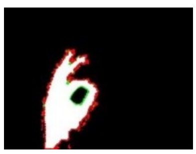
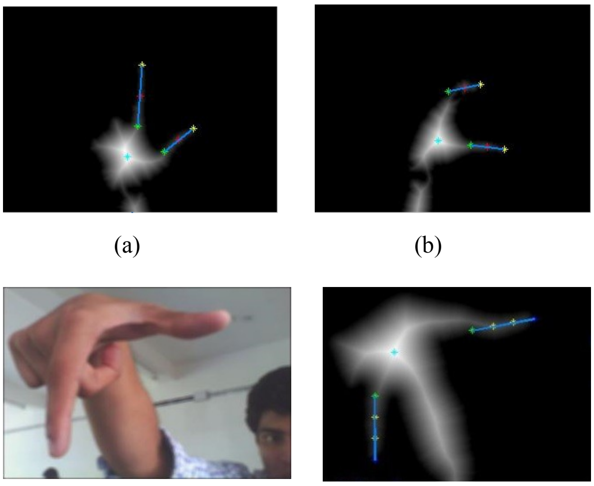


Fig 7.4: Segmented O after hole detection

The letter J consists of the same fingers as letter L and C therefore it is difficult to differentiate J with other characters. But the index finger of letter J lies below the centroid of the hand region. The centroid, base point and finger tip point are found using distance transform method in chapter 5. So if both the base point and the finger tip point of the index finger lie below the centroid of the hand region, the distance in the vertical axes which approximates the finger length is identified. Thresholding is done because the thumb finger of letter C and L are detected below the centroid of the hand region in some frames. Figure 7.5 shows the distance transform image for gesture L,C and J.



(a) (b)

(c) (d)

Fig 7.5: (a) Distance transform image of L (b) Distance transform image of C

(c) Alphabet J (d) Distance transform image of J

7.3 TWO HAND NEURAL NETWORK

The proposed back propagation neural network has an input layer consisting of 108 neurons, representing the 9x12 HOG descriptor. It has 2 hidden layers of 58 and 21 neurons respectively. It has an output layer consisting of 18 neurons representing the 18 two hand characters.

7.3.1 Classification problems

Characters like M and N have close HOG features and could not be distinguished. The reason for this inability is that the difficulty in identifying the overlapping hand region and the variations in overlapping gesture. Figure 7.6 (a) and 7.6 (b) shows M and N segmentation and it is clear from the figures that the two gestures are not similar. Similarly ISL has a lot of gestures that are different but that difference is negligible after segmentation. So even when the neural network is trained to draw a plane between these gestures, the neural network ends up intersecting other classes and hence causing misclassification of other characters.

(a)

(b)

Fig 7.6: (a) classification problem on The Letter M (b) The Letter N

7.3.2 Conflict Resolving Methods

After detailed experimentation, it was found that the conflicting alphabets form three different groups. These characters get confused within the characters of the same group. Table 7.1 shows the characters involved.

TABLE 7.1: Conflicting Groups

Group	Characters
1	X, E, F, T
2	H,G,Z
3	R,M,N,S

Apart from these characters, if the neural network detected any other character, the output was directly taken. The results for this network are discussed in chapter 8.To classify the alphabets in group 1 and 2, the binary image of the gesture is obtained. From that the tip of the hand is identified by scanning from top to bottom. From the tip, a distance of 25 pixels is traversed down. The value of 25 is defined based on the trial and error method.

For E and F, the width of the finger is identified. From that point, the image is scanned from left to right. By scanning, the distance between the two red points as shown in Figure 7.7 and Figure 7.8 is calculated which is the length of the continuous white pixel region. This length is equal to one finger region for E and four finger region for F. Hence, this is used to differentiate E and F.

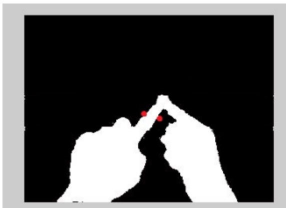


Fig 7.7 Gesture E

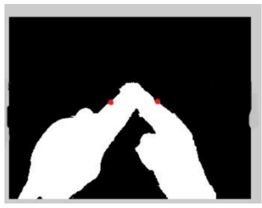


Fig 7.8 Gesture F

For X, from the tip 5 pixels is traversed down. From the left 2 white runs would be present with a black run in between as shown in Figure 7.9. The two white runs are shown using 4 red points in Figure 7.9.

Fig 7.9: Gesture X

For T, once a white pixel is identified when scanning the image from top to bottom, two black runs and one continuous white pixel region is identified and the length of the white run is calculated which is the distance between the red points in Figure 7.10.

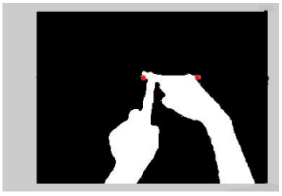


Fig 7.10: Gesture T

For G, after identifying the tip, from that point, a vertical line is drawn which covers the full gesture since for G both the hands will be covered. From that line, the height of the gesture is measured which is the distance between the two red points as shown in Figure 7.11.

Fig 7.11: Gesture G

For H, the same procedure for G is applied. But the line covers single hand alone and hence the height of G and H are different. If the height is large it is G and if it is small then the gesture shown is H. The height for gesture H is the distance between the two red points in Figure 7.12.

Fig 7.12: Gesture H

For Z, the first white pixel from left to right scanning is identifies and from that point, the width of the white run is obtained. This width identifies the distance between the two red points in Figure 7.13

Fig 7.13 Gesture Z

For the characters in group 3, adhoc decision is not possible due to the overlapping characteristic of those alphabets. It is possible to identify the gesture for certain people whose palm region and hand region are of different skin colour complexion. Hence there is an accuracy drop in identifying gestures M, N, R and S.

Chapter 8
RESULTS OBTAINED

Chapter 8

RESULTS OBTAINED

The results are obtained for all the methods and they are analysed to find out the best method.

8.1 PARAMETERS USED TO MEASURE PERFORMANCE

Four different statistical measures were used to analyze the performance. The statistical measures are:

True Positive: The signer shows the correct gesture and the system identifies the gesture correctly.

False Positive: The signer shows the correct gesture and the system identifies the gesture incorrectly.

True Negative: The signer shows the wrong gesture and the system correctly identifies it as the wrong gesture.

False Negative: The signer shows the correct gesture and the system identifies the gesture incorrectly.

Figure 8.1 better explains the above statistical measures.

	actual class (observation)	
predicted class (expectation)	**tp** (true positive) Correct result	**fp** (false positive) Unexpected result
	fn (false negative) Missing result	**tn** (true negative) Correct absence of result

Fig 8.1 Matrix explaining the four statistical measures

Using the above statistical features, we calculate 4 statistical measures of performance with which we analyze the performance of the system that has been developed.

Accuracy: The accuracy is the proportion of true results (both true positives and true negatives) in the population. Mathematically, accuracy is measured using Equation 8.1.

$$\text{Accuracy}=(tp+tn)/(tp+tn+fp+fn) \qquad \text{Eq (8.1)}$$

Precision: Precision or positive predictive value is defined as the proportion of the true positives against all the positive results (both true positives and false positives). Mathematically, precision is measured using Equation 8.2.

$$\text{Precision}=tp/(tp+fp) \qquad \text{Eq (8.2)}$$

Sensitivity or Recall: Sensitivity relates to the test's ability to identify positive results. Sensitivity is measured using Equation 8.3.

$$\text{Sensitivity}=tp/(tp+fn) \qquad \text{Eq (8.3)}$$

Specificity: Specificity relates to the test's ability to identify negative results. This is mathematically represented as in Equation 8.4.

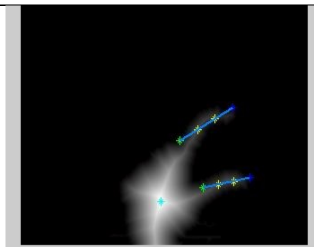
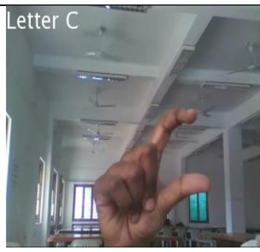
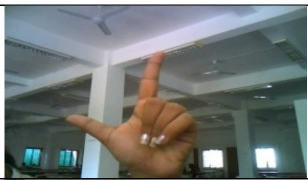
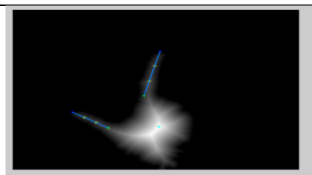
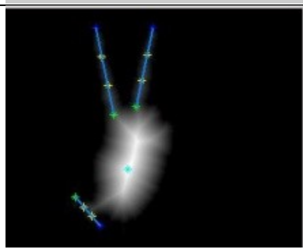
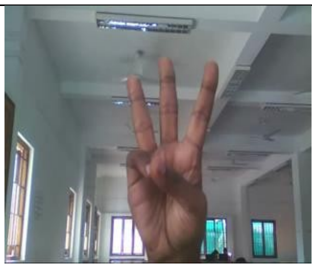
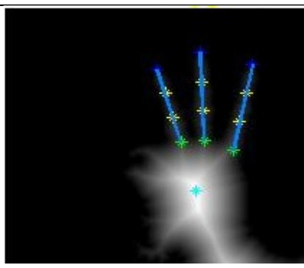
$$\text{Specificity}=tn/(tn+fp) \qquad \text{Eq (8.4)}$$

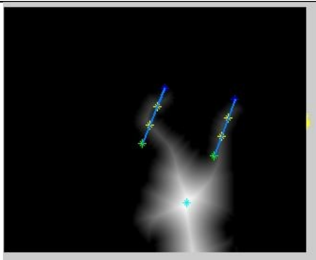
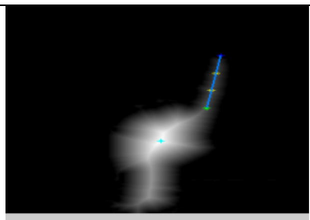
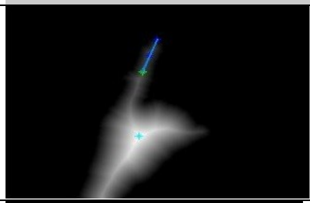
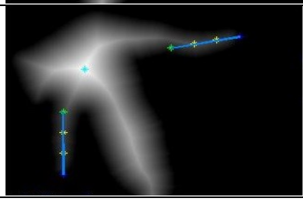
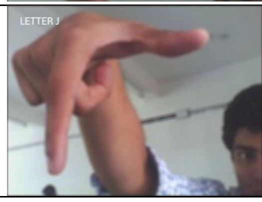
In the Equations 8.1, 8.2, 8.3 and 8.4 the term tn denotes number of true negatives, tp denotes number of true positives, fp denotes the number of false positives and fn denotes false negatives.

8.2 RESULTS OBTAINED FOR SINGLE HAND CHARACTERS

The ISL single hand characters include L, C, I, U, V, and W .The single hand features extracted using the distance transform method explained in chapter 5 is fed into back propagation neural network. Totally 80 datasets are used to train the neural network. The dataset comprise of 80 frames, which include approximately 10 frames for each alphabet and another 10 frames for recognizing gestures which do not belong to the ISL character. Gestures made by 2 different persons are used for training and gestures made by 10 persons are used for testing. The gesture is tested for 100 frames. Table 8.1 shows the sample frame of intermediate results obtained for single hand characters. The table include the image that is given as input, the image obtained after applying distance transform and the output image.

TABLE 8.1: Single Hand Characters

Character	Input Image	Distance Transform	Output Image
C			
L			
V			
W			

U			
I			
O			
J			

The experiment is repeated for different architectures and their outputs are tabulated. Table 8.2 shows the confusion matrix obtained for the feed forward neural network.

TABLE 8.2: Confusion Matrix for feed forward neural network

	L	C	I	U	V	O	W	J	Nothing
L	100								
C	77								23
I		2	65	1	3				29
U		4		92					4
V					2		85		13
O	2		19			60			19
J	32	20						36	12
W					4		96		
Nothing		3	82	2					13

From the above table it is seen that C is misclassified as L and V is misclassified as W. Table 8.3 shows the confusion matrix for a cascade feed forward network.

TABLE 8.3: Confusion Matrix for cascade feed forward neural network

	L	C	I	U	V	O	W	J	Nothing
L	100								
C		65							35
I	4	3	72	2	3				16
U		4		92					4
V					87		13		
O						100			
W							100		
J		10						83	7
Nothing		2	3						95

Though the recognition of C and V has improved greatly, most of I, V and O were not identified as characters. The confusion matrix for the back propagation neural network with 2 hidden layers is given in Table 8.4.

TABLE 8.4: ConfusionMatrix for back propagation neural network

	L	C	I	U	V	O	W	J	Nothing
L	100								
C		100							
I		1	62	6			2		29
U		5		87					8
V					77				23
O			7			63			30
W							100		
J								100	
Nothing		3	4						93

Apart from neural network, the features where just compared with the predefined values in order to indentify the gestures. But since the value varies from person to person, it does not give much accuracy compared to the neural network method. Table 8.5 shows the comparison in the accuracy obtained in several neural networks implemented along with the comparison method. From the table it can be concluded that the back propagation neural network gives better results.

TABLE 8.5: Comparison of accuracies between several architectures

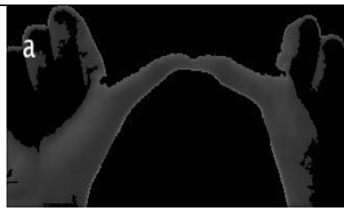
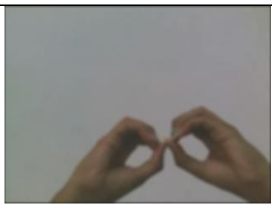
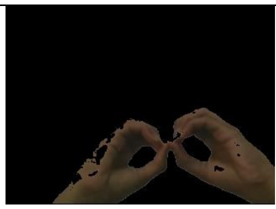
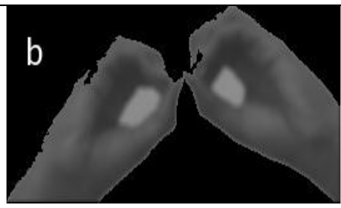
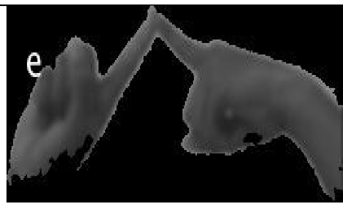
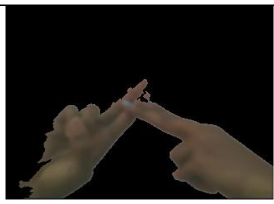
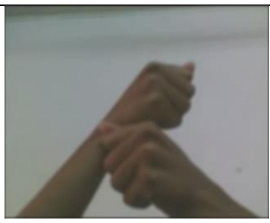
LETTER	COMPARISON METHOD	FEED FORWARD	CASCADE FEED FORWARD	PROPOSED BACKPROPAGATION NEURAL NET
L	80	97	95	100
C	42	93	95	97
I	88	100	100	100
U	60	87	82	96
W	71	100	100	100
V	85	100	100	100
O	61	72	74	96
J	37	47	43	97
Nothing	72	100	100	100

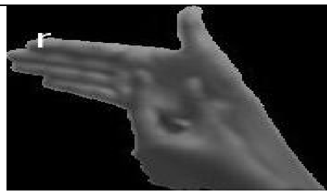
In all the confusion matrices given above, the classes are sometimes classified as nothing which is due to minimal illumination. At this point of time some other skin coloured objects in the background is taken into account and hence results in class nothing. The precision, recall and specificity are 89.47%, 89.78% and 97.54% respectively for the back propagation neural network.

8.3 RESULTS OBTAINED FOR TWO HAND CHARECTERS

The two hand gestures given as input to the webcam are tested using the neural network and the results are tabulated. The tests were performed for various neural networks with the back propagation architectures by varying the number of neurons in the hidden layers. The test data set and training data set are independent of each other. The training data set consists of 200 frames which includes 10 frames approximately for each alphabet. The training data set includes gestures shown by two different persons. The test dataset consists of 20 frames for each alphabet. Using the neural network with two hidden layers of 58 neurons in first layer and 21 neurons in second hidden layer, the two hand gestures are tested. The experimental results obtained for two hand gestures is tabulated below in table 8.6

TABLE 8.6: Two Hand Characters

Alphabet	Original Image	Segmented Image	Result
A			
B			
D			
E			
F			
G			

H			
K			
M			
N			
P			
Q			
R			

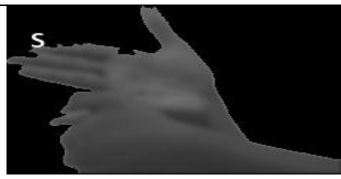
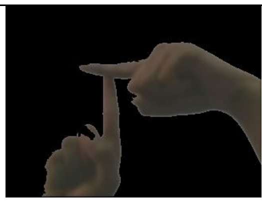
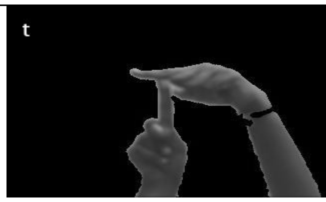
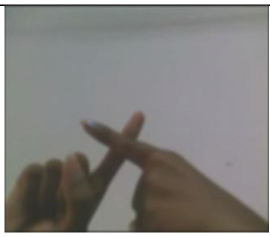
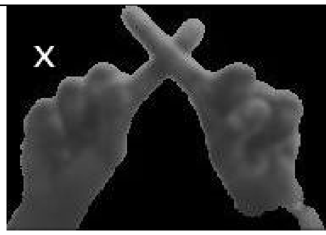
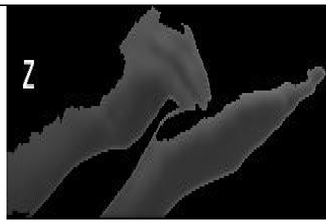
S			
T			
X			
Y			
Z			

Table 8.7 shows the confusion matrix obtained for the network with 36 neurons in hidden layer 1 and 21 neurons in hidden layer 2. Table 8.8 denotes the confusion matrix obtained for the network with 45 neurons in hidden layer 1 and 21 neurons in hidden layer 2. Table 8.9 shows the confusion matrix obtained for the neural network with 52 neurons in hidden layer 1 and 35 neurons in hidden layer 2. Table 8.10 shows the confusion matrix obtained for the neural network with 52 neurons in hidden layer 1 and 45 neurons in hidden layer 2. Table 8.11 shows the confusion

matrix for the neural network with 58 neurons in hidden layer 1 and 21 neurons in hidden layer 2.

TABLE 8.7: Confusion matrix I

	A	B	D	G	M	Z	H	P	R	X	Y	E	F	N	K	S	Q	T
A	20																	
B		20																
D			20															
G				20														
M					15											5		
Z						20												
H							20											
P			5					15										
R					12									8				
X										20								
Y	7									5							8	
E										20								
F										20								
N					20													
K			20															
S																20		
Q		10									10							
T																		20

TABLE 8.8: Confusion matrix II

	A	B	D	G	M	Z	H	P	R	X	Y	E	F	N	K	S	Q	T
A	20																	
B		20																
D			20															
G				20														
M					8									7		5		
Z	20																	
H							20											
P			5					15										
R					14									6				
X												9						11
Y	7									5							8	
E										20								
F										12		8						
N					20													
K			20															
S																20		
Q		10									10							
T																		20

TABLE 8.9: Confusion matrix III

	A	B	D	G	M	Z	H	P	R	X	Y	E	F	N	K	S	Q	T
A	20																	
B		20																
D			20															
G				20														
M					20													
Z	17						3											
H							20											
P			5					15										
R					12									8				
X										13			7					
Y	2									9							9	
E										20								
F													20					
N					20													
K			20															
S																20		
Q		14									6							
T																		20

TABLE 8.10: Confusion matrix IV

	A	B	D	G	M	Z	H	P	R	X	Y	E	F	N	K	S	Q	T
A	20																	
B		20																
D			20															
G				20														
M					20													
Z						14	6											
H							20											
P			2					18										
R																20		
X										20								
Y					12	8												
E										20								
F										20								
N					10											10		
K															20			
S																20		
Q		5									15							
T																		20

TABLE 8.11: Confusion matrix V

	A	B	D	G	M	Z	H	P	R	X	Y	E	F	N	K	S	Q	T
A	20																	
B		20																
D			20															
G				20														
M					19													
Z						15					5							
H							20											
P								20										
R					4				16									
X										20								
Y						2					18						8	
E										3		17						
F										4			16					
N														20				
K															20			
S																20		
Q		10															10	
T																		20

From the confusion matrices given above, the true positives, true negatives, false positives and false negatives are found and these values are used to calculate the accuracy, sensitivity and specificity and precision.

TABLE 8.12: Performance Table for different BPN

No. of neurons	Accuracy	Specificity	Sensitivity	Precision
52, 35	70	50	68	68
52, 45	51	50	57	57
58, 21	86.5	50	92.2	92.2

From Table 8.12, it is clear that neural network with 58 neurons in first layer and 21 neurons in second layer are efficient according to the measure of all four parameters.

CHAPTER 9

CONCLUSION AND FUTURE WORK

CHAPTER 9

CONCLUSION AND FUTURE WORK

9.1. CONCLUSION

A real time system that is capable of recognizing the 26 English characters of the Indian Sign Language is implemented. The system enables hearing or speech impaired people to communicate their thoughts with a higher degree of freedom. The system, takes in real time input through a single camera webcam that is integrated with a laptop. Thus it does not require multiple cameras or a camera with high resolution, thus making the system a very cost effective solution for the problem at hand. Further the system is not a personalized product. It can be used by any random person. The system does not require any initial training by the user and can be used on the fly. The hybridized segmentation technique that has been implemented as part of this system is a novel approach. The advantage of this segmentation technique is that it is able to segment out the hand region properly in spite of marginal variations in the illumination of the environment. It also does not require any large databases to store the extracted features. The system is capable of handling dynamic environments where the user keeps on changing the gestures continuously. Each user has his/her way of showing the gestures, in spite of this the system is able to recognize the gestures with some accuracy.

One of the major advantages of the system is that the ISL gestures for which the system has been developed is predefined and is accepted internationally. This method was experimented with several individuals and the overall performance rate of the system turned out to be about 90%. The response time and the performance rate of the system is fair enough for real time use.

9.2. FUTURE WORK

Even though the system is capable of handling respectable amounts of illumination variance, it still fails at certain times when the illumination is uneven. Establishing a threshold for illumination variance can be a challenging problem. A effort can then be made to make the system more independent of illumination factors of the environment.

Other procedures for segmenting and segregating the overlapping hands could be done to improve the accuracy of the system.

ISL has a large vocabulary of almost 3000 words which involve not only the hand regions. They also involve several parts of the body. Detecting and classifying these body gestures pose a major challenge for the future.

Though the system has been developed for ISL recognition, its applications are limitless. With the features that have been extracted, the system can also be modified as a tool for human computer interaction, robot control and several other applications.

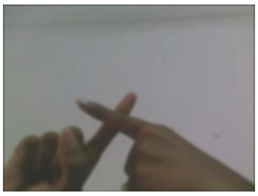
APPENDICES

CONFLICTING CHARACTERS

Group 1

Gesture E

Gesture F

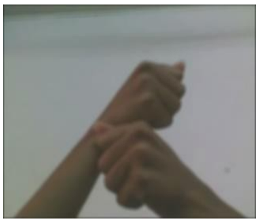


Gesture X

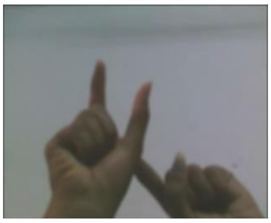
Gesture T

Group 2

Gesture Z



Gesture G



Gesture Y

Gesture H

Group 3

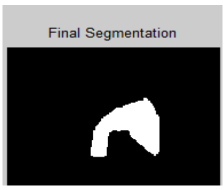
Gesture R

Gesture M

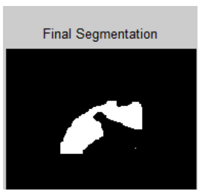
Gesture N

Gesture S

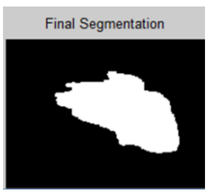
CONTOUR BASED SEGMENTATION



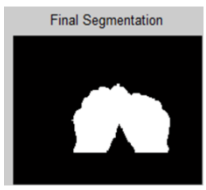
Gesture K



Gesture Q



Gesture M



Gesture A

REFERENCES

[1] Lars Bretzner, Ivan Laptev and Tony Lindeberg, “Hand Gesture Recognition using Multi-Scale Color Features, Hierarchical Models and Particle Filtering” Shortened version in Proc. Face and Gesture 2002, Washington DC, 423–428.

[2] Siddharth S. Rautaray1, Anupam Agrawal, “Real Time Hand Gesture Recognition for Dynamic Applications”, International Journal of ubicomp (IJU), Vol.3, No.1, January 2012.

[3] Wexelblat, Alan. “A Feature-Based Approach to Continuous-Gesture Analysis.”1994.

[4] Y. Freund and R. E. Schapire, “Experiments with a new boosting algorithm”, Machine Learning: Proceedings , 1996.

[5] Raymond Lockton and Andrew W. Fitzgibbon, ” Real-time gesture recognition using deterministic boosting”, University of Oxford, 1997.

[6] Starner, Thad, and Alex Pentland, ”Real-Time American Sign Language Recognition from Video Using Hidden Markov Models” MIT Media Laboratory Perceptual Computing Section Technical Report No. 375, 1996.

[7] Murakami, Kouichi, and Hitomi Taguchi,“Gesture Recognition Using Recurrent Neural Networks ”, Proceedings of CHI’91 Human Factors in Computing Systems,237-242, 1991.

[8] Russell, Stuart, and Peter Norvig, “Artificial Intelligence: A Modern Approach”, Prentice Hall,NJ, 1995.

[9] Ashwin Thangali and Stan Sclaroff, “An alignment based similarity measure for hand detection in cluttered sign language video”, Boston University, Computer Science technical report BUCS-TR-2009-009.

[10] Sue Twelves1, Pierre Payeur2, Monique Frize1 ,“Toward the Detection of Overlapping Body Parts in Infrared Images”,2005.

[11] Yang Mingqiang, Kpalma Kidiyo, Ronsin Joseph,“A survey of shape feature extraction techniques”, Pattern Recognition, Peng-Yeng Yin (Ed.) (2008) 43-90.

[12] Klimis Symeonidis, “Hand Gesture Recognition Using Neural Networks”, August 23, 2000.

[13] Ravikiran J, Kavi Mahesh, Suhas Mahishi, Dheeraj R,Sudheender S, Nitin V Pujari, “Finger Detection for Sign Language Recognition”, International Multi Conference of Engineers and Computer Scientists, Hong Kong, IIMECS 2009,March 2009.

[14] Mahmoud Elmezain, Ayoub Al-Hamadi, Bernd Michaelis “A Real-Time Capable System for Hand Gesture Recognition Using Hidden Markov Models in Stereo Colours Image Sequences”, (IESK) Otto-von-Guericke-University Magdeburg Germany 2009.

[15] R.M. Arunachalam, M. Ashok Gowtham, and R. Aarthi ,"Identifying Fingertips for Human Computer Interaction", ISSN: 2079-2557 Science Academy Publisher, United Kingdom www.sciacademypublisher.com 1375, International Journal of Research and Reviews in Computer Science (IJRRCS) ,[Vol. 3, No. 1],.February 2012.

[16] Justin K. Chen, Debabrata Sengupta, Rukmani Ravi Sundaram Vaishali.S.Kulkarni et al., "Sign Language Gesture Recognition with Unsupervised Feature Learning", (IJCSE) International Journal on Computer Science and Engineering[Vol.02,No. 03,560-565] ,2010.

[17] http://www.deaftravel.co.uk/signprint.php?id=27-AmericanSign Language Images

[18] Christian Vogler and Dimitris Metaxas, "A Framework for Recognizing the Simultaneous Aspects of American Sign Language", Computer Vision and Image Understanding **81,** 358–384 (2001).

[19] http://www.indian signlanguage.org- Indian Sign Language Images

[20] Jonathan Alon, Vassilis Athitsos, Quan Yuan and StanSclaroff," A Unified Framework for Gesture Recognition and Spatiotemporal Gesture Segmentation", IEEE Transactions of Pattern Analysis and Machine Intelligence (PAMI), September 2009.

[21] Navneet Dalal and Bill Triggs, "Histograms of Oriented Gradients for Human Detection", 2005.

[22] Junyeong Choi, Hanhoon Park and Jong-Il Park, "Hand shape recognition using distance transform and shape decomposition",

proceedings of 18^{th} IEEE international conference on image processing, 11-14 Sept. 2011, Brussels, Belgium.

[23] Muthukrishnan.R and M.Radha, "Edge Detection Techniques for Gesture Recognition", International Journal of Computer Science & Information Technology (IJCSIT) Vol 3, No 6, Dec 2011.

[24] J.J. Charles, L.I. Kuncheva, B. Wells, I.S. Lima," Object segmentation within microscope images of palynofacies", Computers & Geosciences 34 (2008) 688–698.

[25] Vladimir Vezhnevets Vassili, Sazonov Alla Andreeva, "*A Survey on Pixel-Based Skin Color Detection Techniques*", Graphics and Media Laboratory, Faculty of Computational, Mathematics and Cybernetics, Moscow State University, Moscow, Russia, In Proceedings of the GraphiCon 2003, pp. 85-92.

[26] Thomas ALLEVARD , Eric BENOIT, Laurent FOULLOY, "Fuzzy Glove for Gesture Recognition", XVII IMEKO World Congress Metrology, June 2003.

[27] Mario Ganzeboom, "How hand gestures are recognized using a data glove", 2005.

[28] Cootes, T.F., and C.J. Taylor. Active Shape Models – 'Smart Snakes'. In Proceedings of the British Machine Vision Conference, Springer-Verlag, 266-275,1992.

ACCOLADES

CONTESTS

1. Second runner up in the “Tech2Share”, technical video contest in the IBM Collaborative Academia Research Exchange (ICARE-2013) held at IISc Bangalore.
2. Runner up in the Paper presentation contest in Vidyut-2013, a technical fest organized by Amrita School of Engineering, Amritapuri Campus.
3. Winner in the Paper presentation contest held as part of the CSI National Student Convention 2013, held at Amrita School of Engineering, Coimbatore Campus.
4. Winner in the Paper Presentation contest held as part of Abacus-2013, the computer science departmental technical fest of Anna University, Guindy.

PAPERS AND JOURNALS

1. Sent paper “Indian Sign Language Character Recognition Using Neural Networks, Saipreethy M.S, Valliammai.V, Padmavathi.S, Amrita School of Engineering, Coimbatore, TamilNadu” to ICJS special edition for Image Processing and Pattern Recognition journal and waiting for the response.
2. Published paper “Computer Vision based approach for Indian Sign Language character recognition, Shangeetha.RK, Valliammai.V, Padmavathi.S, Amrita School of Engineering, Coimbatore, TamilNadu” in Machine Vision Image Processing IEEE Conference.
3. Paper “Indian Sign Language Gesture Segmentation using Active Contour , Shangeetha.RK , KS Karthik, Padmavathi.S” ,has been accepted to be published in the journal “International Journal of Electronics Communication and Computer Engineering”