Image Based Pakistan Sign Language Recognition System

Developed by

Arzu Fatima
274-FAS/MSCS/F05
Kokab Huma
278-FAS/MSCS/F05

Supervised by:

Dr. Syed Afaq Hussain

Department of Computer Science
Faculty of Basic and Applied Sciences
International Islamic University, Islamabad

Department of Computer Sciences
International Islamic University, Islamabad

Final Approval

Dated: 22-09-07

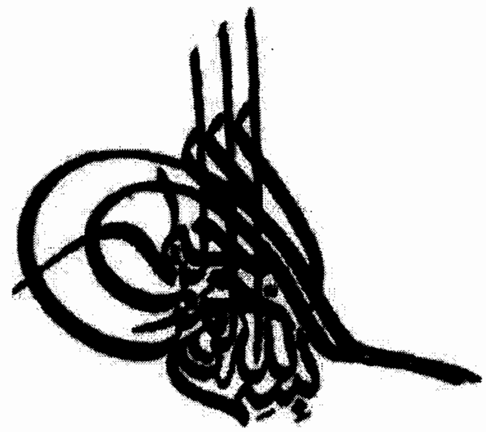
It is certified that we have read the thesis report submitted by Arzu Fatima and Kokab Huma, Registration Number 274-FAS/MSCS/F05 and 278-FAS/MSCS/F05, and it is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for the MS Degree in Computer Science.

COMMITTEE

External Examiner
Dr. Mahboob Yasin
Professor
Faculty of Information Science and Technology
COMSATS Institute of Information Technology (CIIT)
Islamabad Pakistan

Internal Examiner
Mr. Muhammad Nadeem
Assistant Professor
Faculty of Applied Sciences
International Islamic University,
Islamabad, Pakistan

Supervisor
Dr. Syed Afaq Hussain
Ex-Dean Faculty of Engineering & Technology
International Islamic University,
Islamabad, Pakistan



In The Name of

ALLAH ALMIGHTY

The Most Merciful The Most Beneficent

"Lo! In the creation of the heavens and the earth and the alternation of the night and the day, there are surely signs for men of understanding." (Al-Imran: 190-191)

**A dissertation submitted to the
Department of Computer Science,
International Islamic University, Islamabad
as a partial fulfillment of the requirements
for the award of the degree of**

MS in Computer Science

DEDICATION

We dedicate this work to our Dear

ALLAH T'AALAH

For all His blessings

and

for granting us with such loving relationships

who

encouraged and helped us in our entire Life.

Arzu Fatima 274-FAS/MSCS/F05

Kokab Huma 278-FAS/MSCS/F05

DECLARATION

We hereby declare and affirm that this research neither as a whole nor as a part thereof has been copied out from any source. It is further declared that we have developed this software and accompanied thesis entirely on the basis of our personal efforts, made under the sincere guidance of our teachers. If any part of this project is proved to be copied out or found to be a reproduction of some other project, we shall stand by the consequences.

No portion of the work presented in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or institute of learning.

Arzu Fatima 274-FAS/MSCS/F05

Kokab Huma 278-FAS/MSCS/F05

ACKNOWLEDGEMENT

All the praise of Almighty Allah, The Lord of Creation, the known and unknown universe, Who descended His last Prophet Hazrat Mohammad (SAW), the Benefactor of Humanity; to guide the mankind and enlighten the ones who believe Almighty Allah.

It is unimaginable that an academic effort of this magnitude could successfully come to fruition without the help of others. Expressing gratitude to those whom it is due is a highly regarded Islamic custom based upon the statement of Prophet Muhammad (S.A.W):

"*Whoever does not thank people does not thank Allah.*""

It is with genuine gratitude that we thank to our kind supervisor, **Dr. Syed Afaq Hussain** who kept our morale high by his suggestions and appreciation. If it had it not been for his constant supply of insight and encouragement, we are sure this project would not have been completed in good spirits nor would it have been completed in time.

Special thanks go to all our teachers. In particular, we would like to thank **Mr. Asim Munir** who has been our teacher since the 1st semester of MSCS and his help has always been available whenever we faced difficulties during this project. . We would like to pay thanks to our dear friends, especially all the memories of the **group of six in the Room 30** of the hostel is unforgettable. We would also love to acknowledge the support of our Family members. We would like to admit that we owe all our achievements to our truly, sincere and most loving parents, brothers, sisters, whose prayers are a source of determination for us.

Arzu Fatima 274-FAS/MSCS/F05

Kokab Huma 278-FAS/MSCS/F05

PROJECT IN BRIEF

Project Title:	Image Based Pakistan Sign Language Recognition System
Undertaken by:	Arzu Fatima 274-FAS/MSCS/F05 Kokab Huma 278-FAS/MSCS/F05
Supervised By:	Dr Syed Afaq Hussain Dean Faculty Of Engineering & Technology International Islamic University Islamabad
Starting Date:	October 2006
Ending Date:	June 2007
Tools Used	Matlab 7
System Used:	Pentium lV
Operating System:	Windows XP Professional 2005

ABSTRACT

Human Computer Interaction is the discipline that studies the existing and possible communication interfaces between man and machine. Sign language uses a system of manual, facial, and other body movements as the means of communication, especially deaf people.

This research work proposes a Human Computer Interface that uses hand gesture to input data. These gestures are based on the Urdu alphabets in Pakistan Sign language (PSL). The system analyzed these gestures and the corresponding text was displayed for ordinary people to understand. Since only single handed and static gestures have been considered in this project that why a subset of PSL was considered to be implementation. Its is probably the first work done for the recognition of Pakistan Sign Language without the use of any type of data glove. It's the first vision based system for the people, who uses or want to use PSL, which doesn't require them to have a lot of equipment and provide ease of use.

Table of Content

Chapter No	Contents	Page No

CHAPTER- 1

INTRODUCTION

1. Introduction

Humans know each other by conveying their ideas, thoughts, and experiences to the people around them. Among many ways, best one is through "Speech". Everyone can very convincingly transfer their thoughts and understand each other by the use of words. It will be injustice if we ignore those who are deprived of this invaluable gift i.e. power to communicate through common languages around them. That's the stage where Sign language comes in, as only means of communication available to the vocally disabled is the use of "Sign Language".

1.1Signs

Signs usually represent ideas and not single words. Many signs are iconic, that is, they use a visual image for signing the idea e.g. animals, elephant (the trunk), donkey (the ears), and the goat (the beard and horns). Signs are also represented by actions e.g. milk, coffee, love, grow. Other signs are arbitrary and although the originators may have reasons for forming or moving a sign in a particular way these reasons are unknown today. It is interesting to note that many of the older signs have remained as originally created even though the connection to the origin no longer exists.

Finger spelling, the use of hand positions to represent the letters of the alphabet, is considered a vital and historical element of manual communication. The positions of the fingers of the hand do, to some extent, resemble the printed letters of the alphabet.

1.1.1 Sign Language

Throughout the world, deaf people have developed visual languages known as sign languages.

A **sign language** is a language which uses manual communication instead of sound to convey meaning - simultaneously combining hand shapes, orientation and movement of the hands, arms or body, and facial expressions to express fluidly a speaker's thoughts.

Aristotle (384-322 BC) claimed that deaf people could not be educated. He thought that without hearing people could not learn. But in 16th century, Germino Carnado, a physician in North Italy, said that deaf people could learn and understand by the use of sign language.

Like spoken languages sign language is not universal.Although signs are used in many countries, each has developed it's own system, which has been standardized to some extent within that country.Thatswhy there are unique sign languages around world.The ISO639-2 has standerdised 3 letter code to identify sign language called sgn. as it is limitd to 3 letters,country code is also used with it.

In recent years an international sign language has been developed that crosses national barriers and permits communication between deaf persons of many countries. This language, sometimes called Gestuno, has been found useful for international events, such as conferences and Olympic Games for the Deaf.

1.1.2 Pakistan Sign Language (PSL)

Pakistan Sign Language (PSL) is a Deaf-Sign Language of Pakistan. It is a visual gestural language having its own vocabulary and syntax used in Pakistan.

According to earlier research Pakistan Sign Language contains approximately 4000 different gestures for common words. Alternate name of PSL is ISHARON KI ZUBANN. It is related to Nepalese Sign Language; may be the same language as Indian Sign Language. It is used in urban centers with some regional variation in vocabulary.

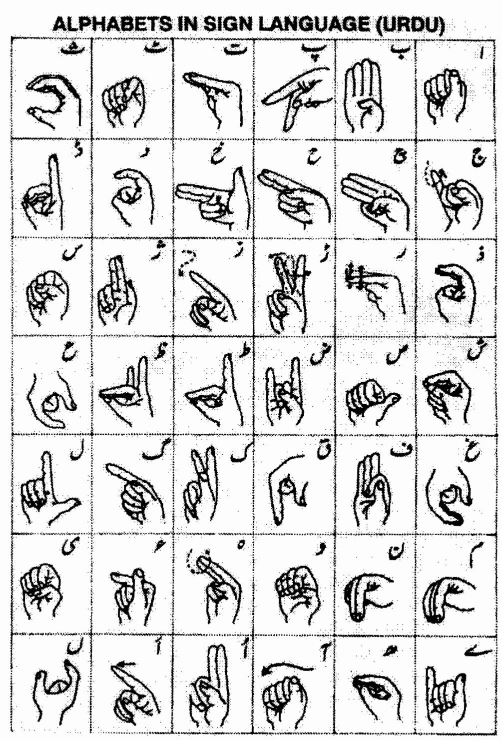


Figure1.1: Signs in PSL

Using sign language vocally disabled and hearing-impaired persons are limited to their own world. This limitation prevents them from interacting with the outer world to share their feelings, creative ideas and potentials. Another problem is that very few people who are not themselves deaf ever learn to sign. This therefore creates the gap between them. To overcome this gap, several attempts to design smart devices that can work as interpreters between the deaf people and others were made. These devices are categorized as Human-Computer Interaction (HCI) systems.

1.2 Human-Computer Interaction (HCI)

Human-Computer Interaction (HCI), alternatively Man-Machine interaction (MMI) or Computer-Human Interaction (CHI), is the study of interaction between people (users) and computers. It relates computer science with many other fields of study and research. Interaction between users and computers occurs at the user interface (or simply interface), which includes both software and hardware.Actually it is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.

As HCI studies a human and a machine in communication, it uses knowledge drawn from both the human and machine side. On the human side, communication theory, graphic and industrial design disciplines, linguistics, social sciences, cognitive psychology, and human performance are relevant.On the machine side, techniques in computer graphics, operating systems, programming languages, and development environments are relevant.

1.2.1 History of HCI

- HCI vs MMI

MMI [2] has been used to refer to any man–machine interaction, including, but not exclusively computers. The term was used early on in control room design for anything operated on or observed by an operator, e.g. dials, switches, knobs and gauges.

‡ HCI vs CHI

The acronym CHI (pronounced kai), for computer–human interaction, has been used to refer to this field, perhaps more frequently in the past than now.

‡ HCI

Researchers and practitioners now refer to their field of study as HCI (pronounced as an initialism), which perhaps rose in popularity partly because of the notion that the human, and the human's needs and time, should be considered first, and are more important than the machine's. This notion became increasingly relevant towards the end of the 20th century as computers became increasingly inexpensive small, and powerful. Human-computer interaction arose as a field from intertwined roots in computer graphics, operating systems, human factors, ergonomics, industrial engineering, cognitive psychology, and the systems part of computer science. Computer graphics was born from the use of CRT and pen devices very early in the history of computers. This led to the development of several human-computer interaction techniques.

1.2.2 Goals of HCI

The basic goal of HCI is to improve the interaction between users and computers by making computers more usable and receptive to the user's needs.A long term goal of HCI is to design systems that minimize the barrier between the human's cognitive model of what they want to accomplish and the computer's understanding of the user's task.

Researchers in HCI are interested in developing new design methodologies, experimenting with new hardware devices, prototyping new software systems, exploring new paradigms for interaction, and developing models and theories of interaction.

1.2.3 Design methodologies in HCI

Design methodologies in HCI aim to create user interfaces that are usable, i.e. that can be operated with ease and efficiency.

A number of diverse methodologies outlining techniques for human–computer interaction design have emerged since the rise of the field in the 1980s. Most design methodologies stem from a model for how users, designers, and technical systems interact.

Early methodologies treated users' cognitive processes as predictable and quantifiable and encouraged design practitioners to look to cognitive science results in areas such as memory and attention when designing user interfaces.

Modern methodologies tend to focus on a constant feedback and conversation between users, designers, and engineers and push for technical systems to be wrapped around the types of experiences users want to have, rather than wrapping user experience around a completed system.

Research in Human-Computer Interaction (HCI) has been spectacularly successful, and has fundamentally changed computing. Many of the most famous HCI successes developed by companies are deeply rooted in university research.

1.3Upcoming Areas of Human-Computer Interaction (HCI)

The means by which humans interact with machines continues to evolve rapidly. The reason for this swift development is because of the ever-changing demands of the public as well as the emergence of better technology.

For example, hardware costs have decreased and memory and speed of computer systems has increased greatly. This motivates the search for innovative methods of input such as voice, gesture, and pen. The fact that 'size does matter' indicates that smaller devices are preferred as they are less cumbersome and more portable. Some of the upcoming areas of HCI include Gesture Recognition, 3-D, and Virtual and Augmented Reality.

1.4 Gesture Recognition

Human gestures come in many forms such as hand gestures, body gestures and facial expressions, Hand Gesture Recognition in particular is an area in which effort and interest has been invested with the aim of developing techniques that reduce the complexity of interaction between humans and computers. In this paper human computer interactions through hand gestures will be considered.

1.4.1 The Nature of Gesture

Gestures are expressive, meaningful body motions - physical movements of the fingers, hands, arms, head, face, or body with the intent to convey information or interact with the environment.

Hand gestures are a means of non-verbal interaction for people. They range from interactions with objects (manipulative gestures) to more complex ones that express feelings or provide a form of communication amongst humans [3]. Hand gestures in particular are those movements made by a person's hands while he or she is communicating with others. Gestures in essence complement speech and may in some cases be more effective or may even replace speech.

Although the concept of gesture is loosely defined and depends on the context of the interaction, there are four basic common gesture types and can be defined as [4]:

- Iconic: Representational gestures depicting some feature of the object, action or event
- Metaphoric: Gestures that represent a common metaphor, rather than the object or event
- Beat: Small, formless gestures, often associated with word emphasis
- Deictic: Pointing gestures that refer to people, objects, or events in space or time [5]

The vast majority of automatic recognition systems are for deictic gestures (pointing), emblematic gestures (isolated signs) and sign languages (with a limited vocabulary and syntax).

As mentioned earlier, it is desirable to have a human computer interface that is more natural and that can take advantage of human traits. Hand gesture recognition certainly is a field that is very rapidly opening the doors to constructive interfaces between man and machine.

In terms of communicative gestures, sign language for the deaf (e.g. American Sign Language) is an example that has received significant attention in the gesture literature [6].

1.4.2 Representation of Gesture

Gestures can be static, where the user assumes a certain pose or configuration, or dynamic, defined by movement. When gestures are produced continuously, each gesture is affected by the gesture that preceded it, and possibly by the gesture that follows it. This co-articulation may be taken into account as a system is trained if the representation supports it. There are several aspects of a gesture, which may be relevant and therefore may need to be represented explicitly.

Hummels et al [7] describe four aspects of a gesture which may be important to its meaning:

- Spatial information: where it occurs, locations a gesture refers to
- Pathic information: the path which a gesture takes
- Symbolic information: the sign that a gesture makes
- Affective information: the emotional quality of a gesture

In order to infer these aspects of gesture, we first have to sense human position, configuration, and movement. This can be done directly with sensing devices such as magnetic field trackers, instrumented gloves, and data-suits (which are attached to the user) or indirectly using cameras and computer vision techniques. Each sensing technology differs along several dimensions, including accuracy, resolution, latency, and range of motion, user comfort, and cost.

1.4.3 Hand Gesture Input

With respect to hand gesture input, three main directions have been investigated [8]:

- Virtual Reality Systems: user interacts mainly by means of direct manipulation of the objects of the application.
- Multi-Modal Interfaces: provides natural and powerful interaction by using the natural human-to-human communication means such as speech combined with gesture and gaze.
- Recognition of Gestural Languages: for example deaf sign language.

1.5 Sign Language Recognition

Traditionally, the technology of sign language recognition is divided into two categories

1. Glove-based methods
2. Vision-based methods

1.5.1 Glove-based methods

People naturally use their hands for a wide variety of manipulation and communication tasks. Besides being quite convenient, hands are extremely dexterous and expressive, with approximately 29 degrees of freedom (including the wrist). In his comprehensive thesis on whole hand input, Sturman [9] showed that the hand could be used as a sophisticated input and control device in a wide variety of application domains, providing real-time control of complex tasks with many degrees of freedom.

There are a number of commercially available tracking systems which can be used during gesture recognition, primarily for tracking eye gaze, hand configuration, and overall body position. For several years, commercial devices have been available which measure, to various degrees of precision, accuracy, and completeness, the position and configuration of the hand. One of these devices is the data glove- an instrumental glove or exoskeleton device mounted on the hand and fingers. Several attempts to solve the hand gesture recognition problem have used mechanical glove devices that directly measure the hand pose and/or hand joint angles [10]. Many sign language recognition systems use data gloves to capture hand movements [11]. Data gloves are special gloves with sensors that directly measure the hand pose and/or hand joint angle.

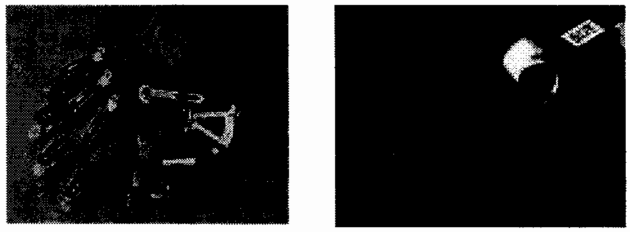


Figure1.2
a) Ultra Data Glove b) 5 DT Data Glove

Some advantages of data gloves include:

- Direct measurement of hand and finger parameters (joint angles, 3D spatial information, wrist rotation)
- Provides data at a high sampling frequency
- No line-of-sight occlusion problems
- Data is translation-independent (within the range of motion)

However, glove-based approaches are quite obtrusive due to the cables associated with the device that bind the user to small areas of the workspace. Another disadvantage is that the cost of the devices put them out of reach for general use.

Body suits are another method in which certain trackers or sensors are attached to the users clothing. The 3D position of these markers can be measured thereby allowing the perception of complex movement patterns.

Sensing technology has a long way to go to overcome the associated disadvantages. On the other hand however, passive sensing using computer vision techniques is beginning to make headway as a user-friendly interface technology.

1.5.2 Vision-based methods

Computer vision based methods provide relatively cost-effective methods to acquire and interpret human hand gestures while being minimally obtrusive to the participant. Vision-based methods generally have three types.

- The Stationary cameras (color and black/white cameras) that capture the hand gestures, without any colored gloves, for further processing and recognition.
- 2nd method includes use of camera to capture hand movements by distinguishing hands by colored gloves wore by signers
- Another type is Wearable cameras. Wearable camera method is that in which signer wear the camera e.g. on his cap and gesture are captured by the camera attached.

a) b) c)

Figure 1.3:a) Camera b) colored Gloves c) Wearable Camera

Vision-based interfaces use one or more cameras to capture images and understand those images to produce visual features that can be used to interpret human activity and recognize gestures.

For the past decade, there has been a significant amount of research in the computer vision community on detecting and recognizing faces, analyzing facial expression, extracting lip and facial motion to aid speech recognition, interpreting human activity, and recognizing particular gestures.

Vision-based methods provide relatively cost-effective methods to acquire and interpret human hand gestures while being minimally obtrusive to the participant. Vision-based systems for gesture recognition vary along a number of dimensions, most notably

- Number of cameras - How many cameras are used?
- Speed and latency - Is the system real-time?
- Structured environment - Are there restrictions on the background, movement, etc.?
- User requirements - Must the user (not) wear gloves, long sleeves, rings etc.?
- Primary features - What low-level features are computed (edges, regions, histograms)?
- Two or three-dimensional representation - Does the system construct a 2D or 3D model of the body?[1]

The answers to the questions of course mostly depend on the application that is to be developed. Certain constraints regarding background and/or foreground may be necessary to help with subsequent processing. During the development of a successful vision system certain decisions must be made as to which features must be extracted and which features can be ignored.

CHAPTER- 2

LITERATURE REVIEW

2. Literature Survey

A great deal of research and development has been done in the area of hand gesture recognition. Sign language recognition is said to be a form of gesture recognition. The literature survey for this field in general can be very comprehensive. However, in this context the specification is sign language recognition. This chapter therefore discusses that work which has been done in a context similar to the current one.

2.1 Pakistan Sign Language Recognition Using Statistical Template Matching [12]

The authors of this research paper were the first who have ventured into the area of Pakistan Sign Language (PSL). This aims at recognizing PSL gestures using Statistical Template Matching. This system is a computerized sign language recognition system for the vocally disabled (deaf and dumb) that uses sign language for communication. The basic concept involves the use of special gloves connected to a computer while a disabled person (who is wearing the gloves) makes the signs. The computer analyzes these gestures and synthesizes the sound for the corresponding word or letter for normal people to understand.

The primary input device is the DataGlove5 developed by 5DT [10]. A particular input sample in this system is defined by the combination of five sensors for fingers and one tilt sensor for roll and pitch which is stored in the Gesture Database during the data acquisition phase. The idea is to distinguish different gestures by calculating the mean (μ) and standard deviations (σ) of all the sensors for a gesture and then those input samples that are within limits bounded by an integral multiple of standard deviation are recognized to be correct. Output of the system is Converts word/ letters obtained after Gesture Recognition into corresponding sound.

2.1.1 Components of the System

These are given below:

Modules for Gesture Input – Get state of hand (position of fingers, orientation of hand) from glove and convey to the main software.

Gesture Preprocessing Module – Convert raw input into a process-able format for use in pattern matching. In this case, scaled integer values ranging from 0 to 255.

Gesture Recognition Engine – Examines the input gestures for match with a known gesture in the gesture database.

Gesture Database - Contains the necessary information required for pattern matching as well as a gesture-to-text dictionary.

Speech Synthesis Module – Converts word / letters obtained after gesture analysis into corresponding sound.

2.1.2 The Model

The statistical model used in Boltay Haath is the simplest approach to recognize postures (static gestures). The model used is known as "Template Matching" or "Prototype Matching". The idea is to demarcate different gestures by calculating the mean (μ) and standard deviations (σ) of all the sensors for a gesture and then those input samples that are within limits bounded by an integral multiple of standard deviation are recognized to be correct.

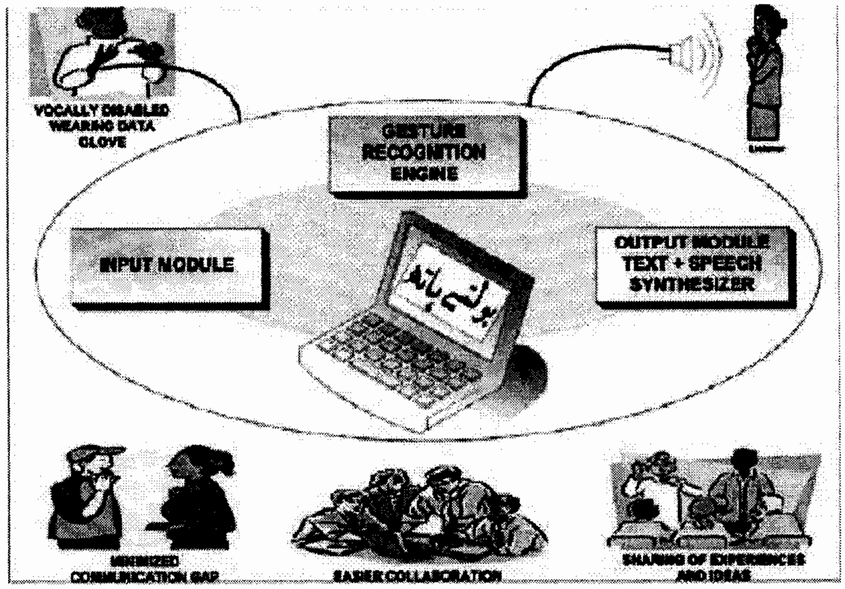


FIGURE : 2.1 THE BOLTAY HAATH SYSTEM [12]

The figure given above best describes the top level components and benefits of this research.

2.1.3 Training

A training sample consists of five values ranging from 0 to 255 each representing the state of the sensor on all five fingers of the glove. The sensors for roll and pitch have been ignored since their values do not uniquely identify an alphabet sign. This training data was then processed i.e., mean and standard deviation was calculated for all five sensors of each gesture in the training set. The resultant μ,σ pairs were stored in the gesture database for later use in gesture recognition.

2.1.4 Gesture Recognition Engine

After training, test samples are provided to the Gesture Recognition Engine which analyses them using the statistical model described previously.

Sometimes due to ambiguity between two gestures the system may produce two outputs. To cater to this problem the method of Least Mean Squares (LMS) is used.

Least Mean Squares (LMS) of all the candidate gestures and then selects the one with minimum LMS value.

Although this is first system, which recognizes Pakistan Sign Language (PSL), but being a glove based system it has its own limitations.

2.2 Recognition of Arabic Sign Language Alphabet Using Polynomial Classifiers [13]

In this paper, polynomial classifiers as a classification engine for the recognition of Arabic sign language (ArSL) alphabet is used. Being vision-based system the signers wears colored gloves, which were marked with six different colors at different six regions.

Figure:2.2 [13]

2.2.1 Stages of the recognition system

An image is taken of each signs using a camera. Each acquired image is fed to the image processing stage in which color representation and image segmentation are performed. RGB values are transformed to hue-saturation-intensity (HSI) representation.

In the image segmentation stage, the color information is used for segmenting the image into six regions representing the five fingertips and the wrist. Also the centroid for each region is identified.

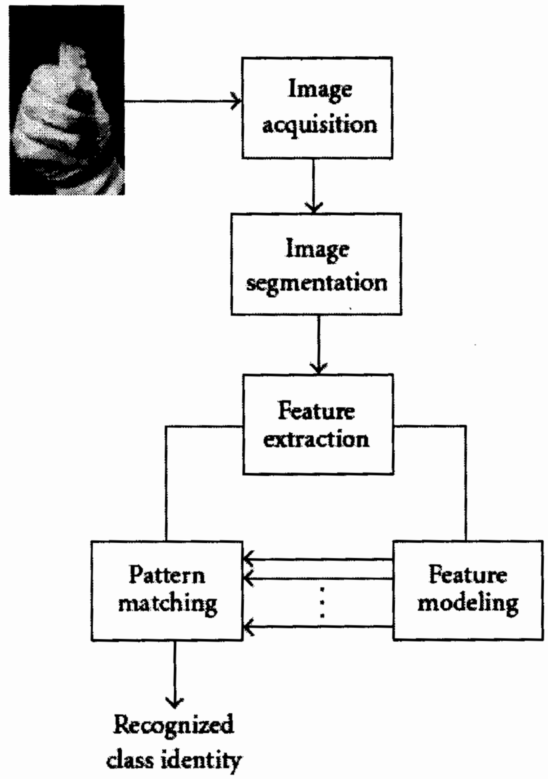


Figure: 2.3 [13]

Thirty features are extracted from the segmented color regions.

The training method of the polynomial classifier is applied by creating one 2nd-order polynomial classifier per class, resulting in a total of 42 networks. The feature vectors for the training data set are expanded into their polynomial terms, and the corresponding class labels are assigned accordingly before they were processed through the training algorithm. Consequently, each class is represented by the identification model. Therefore, multiple models represented alphabets with multiple gestures.

After creating all the identification models, two experiments were conducted to evaluate the performance of polynomial-based system. The first experiment was for evaluating the training data itself, and the second was for evaluating the test data set.

In the first experiment, the performance of the system is found to be superior as is usually expected when the same training data is used as test data. The polynomial-based system has produced superior recognition results for both training and test data.

The results of their polynomial-based recognition system are considered superior over previously published results in the field of ArSL More importantly; the polynomial-based recognition provides a major reduction in the number of misclassified patterns.

2.3 Image Based Arabic Sign Language Recognition [14]

In this paper an image based system for Arabic Sign Language recognition is proposed.

The recognition stage is performed using a Hidden Markov Model (HMM). A single Sonny video camera was used to capture the image sequences. The camera acquires 12 frames per second at a resolution of 352x576 pixels.

A Gaussian skin color model is used to detect the signer's face. The detected face region is then used as a reference to track the hands movement using region growing from the sequence of images comprising the signs. A number of features are then selected from the detected hand regions across the sequence of images. Such features are then used as input to the HMM.

The system discussed in this paper consists of three stages.

- A pre-processing stage to collect the signers' hands movement.
- A feature extraction stage in which a number of salient features are obtained.
- Finally, a recognition stage using Hidden Markov Models (HMMs) to identify the signs.

2.3.1 Hidden Markov Models

HMM is a probabilistic model representing a given process with a set of states (not directly observed) and transition probabilities among the states. Such model has been used in a number of applications.

Given a set of N states, $s_{i,}$ we describe the transitions from one state to another at each time step t as a stochastic process with state-transition probabilities a_{ij} from state s_i to state $s_{j.}$ The probability of the current state depends only on the previous state. Such process is called a Markov chain process of order one. Furthermore, we define a second stochastic process that produces at each time step t, a symbol vector $\mathbf{x}$ of dimension K. The probability density function (pdf) of the vector $\mathbf{x}$ depends only on the actual state, not on the way the state was reached. Such pdf at state i can either be discrete or continuous. This doubly stochastic process is what called an HMM.

In this paper the HMM is trained with seven samples from each sign. The three samples are used for testing purposes.

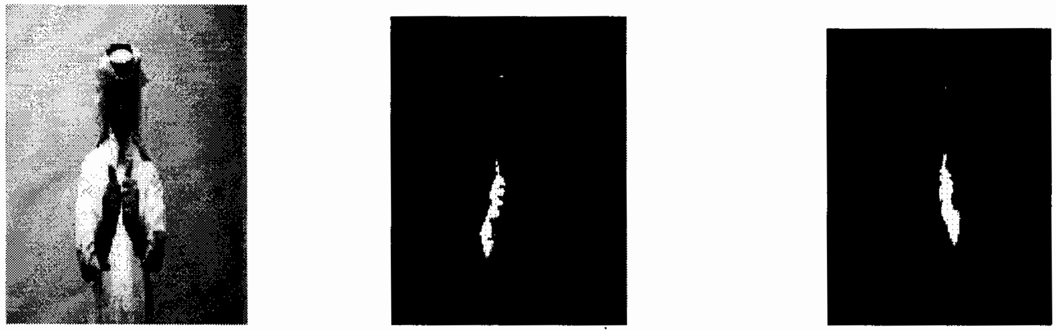
2.3.2 The Hand Tracking Model using Skin Color

Its aim is to recognize isolated Arabic Signs performed by a signer sitting in front of a single video camera wearing two differently colored gloves. A simple homogeneous background is used.

This paper uses the skin color model in the chromatic color space to track the hands from a sequence of images. A Gaussian model is used to characterize the color distribution. In developing the Gaussian model, a transformation from the RGB color space to the chromatic color space is performed. The skin likelihood image is then transformed into a binary image using an adaptive thresholding technique. The optimal threshold is then selected and based on this threshold, the binary image is obtained.

2.3.3 The Region Growing Method

As the signer wears a pair of colored gloves i.e. orange and yellow. The region growing approach starts by scanning the image and searching for the first pure yellow and orange pixels. Such pixels will then be used as seeds to grow the yellow and orange regions in the image.



A signer performing a sign. | Extracted right hand with some noise. | Extracted left hand with some noise.

Figure 2.4 [14]

The centroids of the hands from previous frames were used as seeds for future frames.

2.3.4 The Hybrid Approach

To use the Gaussian model to detect the face and use its centroid as a reference point . the skin color regions and the hands regions were combined. The authors centralize the body coordinates with respect to the centroid of the signer's face.

2.3.5 Features Extraction

The features considered in this paper are both hands' centroids (with respect to the centroid of the face), eccentricity of the bounded ellipse for both hands, the angle of the first principal component and the area of both hands. Using the data from the hand region, the Principal Component Analysis technique is performed to find the first principal component.

2.3.6 Analysis

Such approach is found to provide a reasonably accurate tracking of the hands. With the proposed system, authors achieved a recognition accuracy of 98% for a dataset of 50 words.

2.4 Recognition of Sign Language Gestures Using Neural Networks[15]

This paper describes the structure and performance of the SLARTI (sign language recognition) system developed at the University of Tasmania. The aim of this research is to develop a prototype system for the recognition of the hand gestures used in Australian Sign Language (Auslan).

2.4.1 Input Hardware

The specific input devices used in developing SLARTI were a CyberGlove with 18 sensors and a Polhemus IsoTrak. The Polhemus allows tracking of the spatial position and orientation of the hand with respect to a fixed electromagnetic source.

2.4.2 System Architecture

The approach taken within SLARTI is to initially process the input data so as to produce a description of this sequence. The sign can then be classified on the basis of this feature vector. The SLARTI system consists of four separate feature-extraction neural networks, each trained specifically to recognize one of the features of the sign.

The feature vector produced by these networks is then used to perform the overall classification of the input sequence.

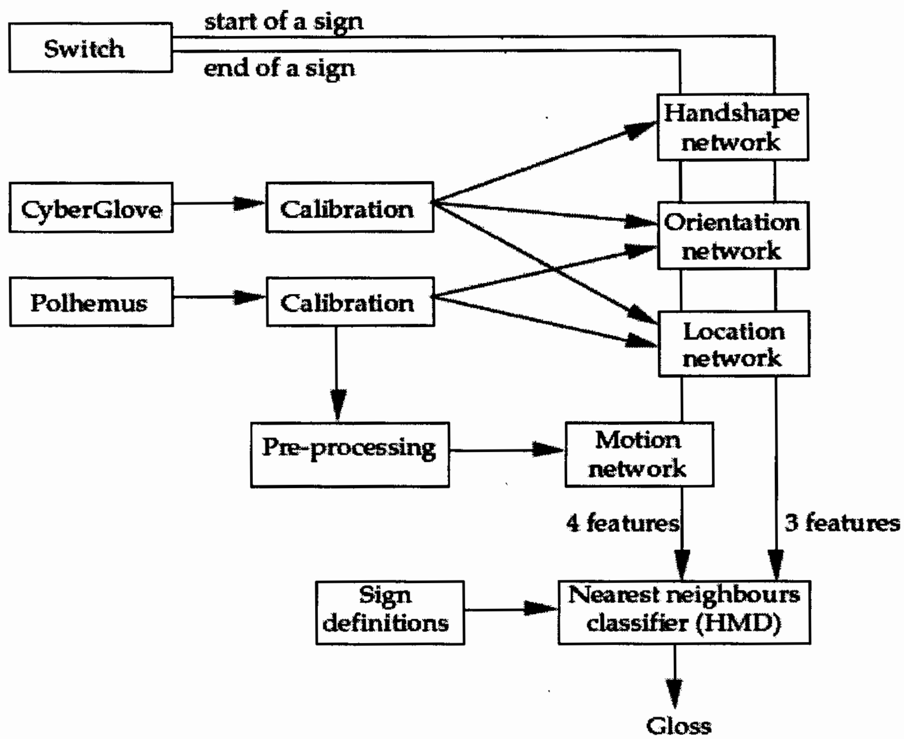


FIGURE: 2.5 Modular Architecture of the SLARTI system [15]

2.4.3 Feature Extraction Networks

All of the feature-extraction networks were trained on examples gathered from 7 signers. A fully-connected feed forward architecture with a single hidden layer was used for all four networks and back propagation without momentum was used as the training algorithm. All input data were scaled to lie in the range -1 to 1.

2.4.4 Handshape Recognition Network

Networks were trained on both the calibrated and uncalibrated data. In both cases the networks had an 18:40:30 architecture (18 inputs, 40 hidden nodes and 30 output nodes) and were trained for 1,000,000 pattern presentations with a learning rate of 0.2 These parameters were determined through a series of initial trials.

2.4.5 Orientation Recognition Network

The orientation values were affected by the degree to which the wrist was flexed. The orientation values returned by the Polhemus were cyclical in nature (ranging from 0 to 255, and then back to 0). To avoid the problems caused by this discontinuity in the input data the network was presented with the sine and cosine of the orientation values rather than the raw values.

2.4.6 Location Recognition Network

The SLARTI system considers only the 16 primary locations as well as neutral space.

Any signs made using the subordinate hand as a base were regarded as being performed in neutral space. The networks developed had an 11:19:19 architecture. The location network achieved a much lower level of accuracy than any of the other feature extraction networks. This is due primarily to the tracking technology used.

2.4.7 Motion Recognition Network

Motion differs from the other features in that it is inherently temporal in nature. Two approaches were taken to dealing with this aspect of the problem. The first was to use a recurrent network with 3 inputs per time frame, feeding into a layer of 30 recurrently interconnected nodes (13 of these were output nodes, the remainder served to store the network's internal state). The input values were the difference in location from the previous time-step. This recurrent network was trained using the back propagation-through-time algorithm. The second approach was to pre-process the input sequence to extract features for presentation to a standard feed-forward network.

2.4.8 Algorithms Used

Nearest-Neighbor Lookup was used for Classification of Signs. By using the simple distance measure the lookup algorithm, using the training examples, easily outperforms that using the sign definitions. However the heuristic distance measure successfully captures the extra information. The second classification algorithm trialed was the C4.5 inductive learning system developed by [11].

2.4.9 Analysis

SLARTI is capable of classifying Auslan signs with an accuracy of around 94% on the signers used in training, and about 85% for other signers.

2.5 Visual Recognition of American Sign Language Using Hidden Morkov Model [16]

In this paper an extensible system is described which uses a single camera and interprets American Sign Language (ASL) using Hidden Morkov Model (HMM).

First the subjects wear colored gloves on each of their hands. The system recognized the hand by scanning the area and finding the pixel of appropriate color. The region is grown by checking the pixels eight neighbors. Then, an eight elements feature vector consist of hands x and y co-ordinates, angle of axis of least inertia was chosen. A Viterbi algorithm was used to reduce the computational load. Then these features were fed in HMM which was trained on experimental data.

2.6 Problem Identified after Literature Survey

Like spoken languages sign language is not universal. There are unique sign languages around the world which vary from region to region. As sign language of one country is different from other so a great deal of research work is required to fulfill the requirements of each sign language.

Although much work has been done on other sign languages but unfortunately the recognition of PSL (Pakistan Sign Language) has received little attention from researchers. The only work done on PSL is Glove-based [12] which has its own limitations. Glove-based systems force the user to carry a load of cables and sensors, they are not completely natural the way an HCI should be. Still if a person wants to use it, sensory gloves are costly and are neither easy to use nor available in everyday life.

This negligence towards the sign language of Pakistan has become our motivation to do the research related to PSL. Our aim is to develop a vision based system to recognize Pakistan Sign Language (PSL), which has not been done in any other system yet.

The objective is to develop a computerized Pakistan Sign Language (PSL) recognition system, which is an application of Human Computer Interaction (HCI).

The basic concept involves the use of images of a person who makes the signs that are captured by camera. The computer will analyze these gestures, minimizes the variations and display the text for the corresponding word or letter for normal people to understand.

CHAPTER- 3

SYSTEM DESIGN

3. System Design

Once the goal has been defined and understood, it is possible to write up a description of what functions the system is to perform with respect to the application and how it is to carry out those functions. Figure 3.1 shows the block diagram of the system.

In this chapter, a detailed description of the system and its modules is given from the software point of view, along with the hardware requirements, configurations, and settings.

3.1 System Setup

The basic advantage of image base system is that it doesn't restrict the user like the glove-based system and it is easy to use.

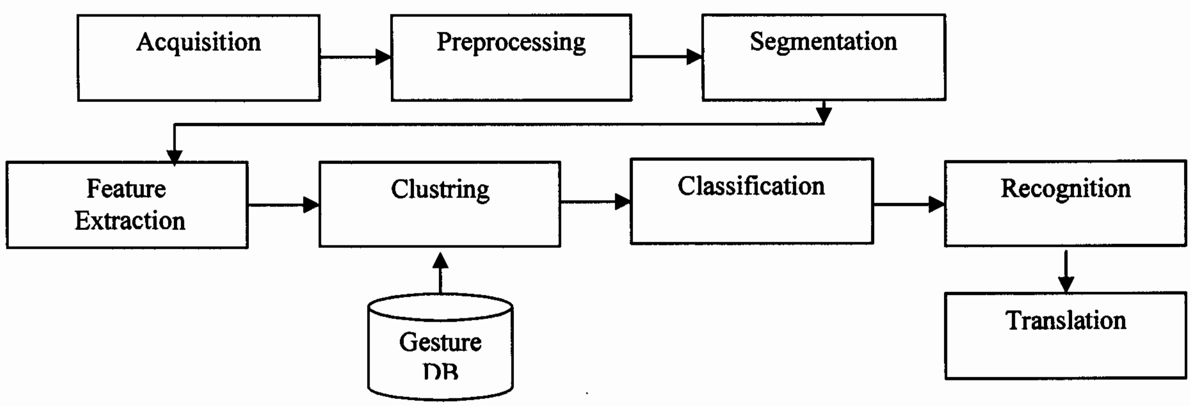
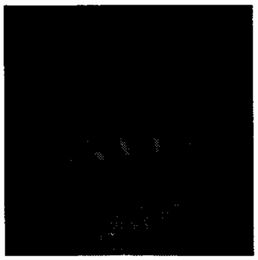


Figure3.1: System Architecture

3.2 Gesture Vocabulary

The sign language into Sub-domains that is English and Urdu. This is because of the similarity of some gestures. Moreover English and Urdu both contain gestures of words and letters. Gestures have been categorized into dynamic and Static. In Urdu there are 38 letters in which few are dynamic and of both types one-handed and two-handed.

As it is the 1st image based research on PSL, so we are going to use only static and single-handed gestures i.e. right-handed. Gesture vocabulary of this system would consist of 32 signs of Urdu alphabets.



Alf ▪

Bay ▪

Pay ▪

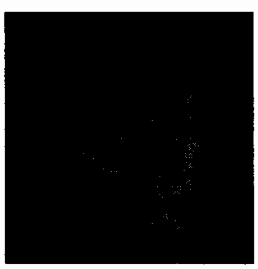
Tay ▪

Ttay ▪

Say ▪

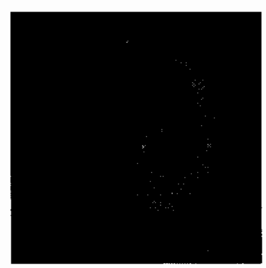
Chay ▪

Hay ▪

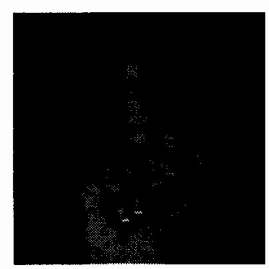


Daal ▪

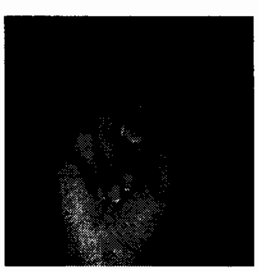
Ddaal ▪



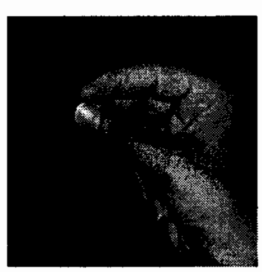
Zaal ▪



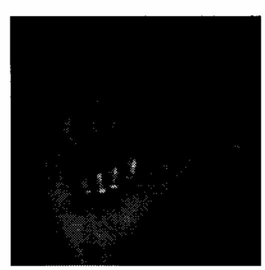
Sey ▪



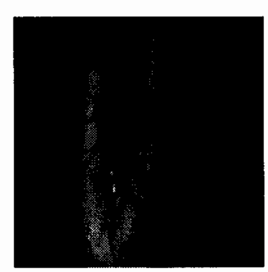
Seen ▪



Sheen ▪



Swad ▪



Dwad ▪

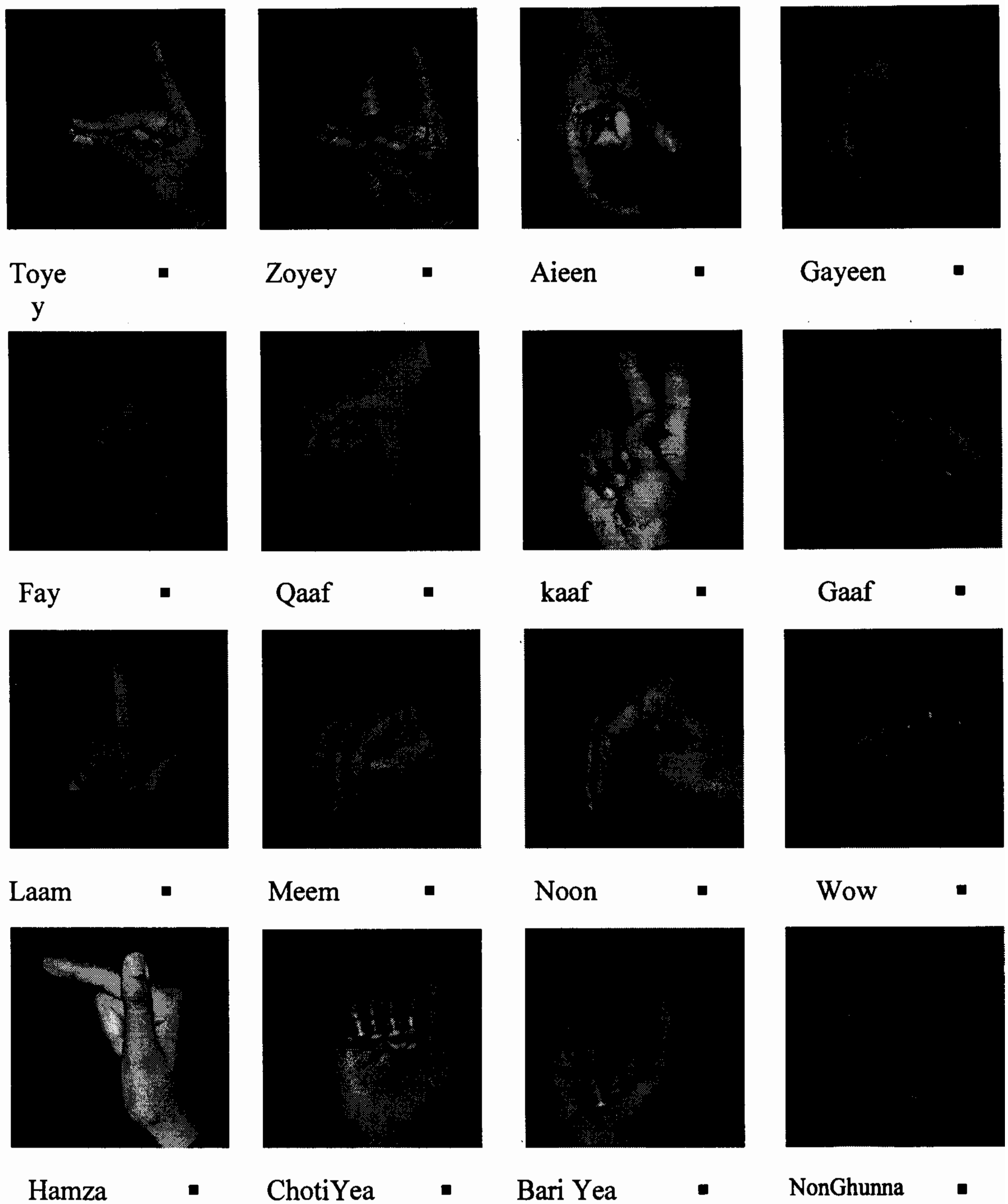


Figure3.2: Gestures used in this system

3.3 Image Acquisition

The acquisition of the images is always the first step. In this project the images are the hand gestures showing the alphabets of the PSL. The images would be acquired through a colored digital camera of 5MP.To make the images clearer the black background is used.

All the samples of one person's hand are acquired at the same time with the same light and same resolution of the camera.

3.4 Conversion of Colored Image

The input images are colored. They are converted to gray scale before further processing. The function of MATLAB `rgb2gray` converts the RGB values to NTSC coordinates, and sets the hue and saturation components to zero. It converts RGB values to grayscale values by forming a weighted sum of the R, G, and B components.

$$0.2989 * R + 0.5870 * G + 0.1140 * B$$

3.5 Hand Segmentation

Segmentation has to be done to obtain the binary form of the images. A binary image can be considered a special kind of intensity image, containing only black and white

The black background was selected to facilitate hand segmentation. It is expected that the only object that will appear in the working space is hand. In case other objects happen to appear in that area then the largest object will be taken as the hand.

3.6 Image Preprocessing

Once the hand object has been segmented from the background, it must be preprocessed. Several morphological operations are applied on the object to clean it and remove spurious pixels.

3.6.1 Image Cleaning

This is a morphological operation that removes isolated pixels i.e. individual 1's surrounded by 0's, such as the center pixel in the pattern below.

$$\begin{matrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix} \longrightarrow \begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{matrix}$$

Figure 3.3: Image Cleaning

3.6.2 Image Erosion

Erosion is a well-known morphological operation that erodes or shrinks an image from the outer boundaries. Erosion can be defined as the act of stripping the outer layer of pixels from a region. The amount of erosion depends on the structuring element. In this case, the structuring element is such that it erodes at a depth of one pixel.

An example is shown below. All foreground pixels that have at least 1 neighbor that belongs to the background will be eliminated.

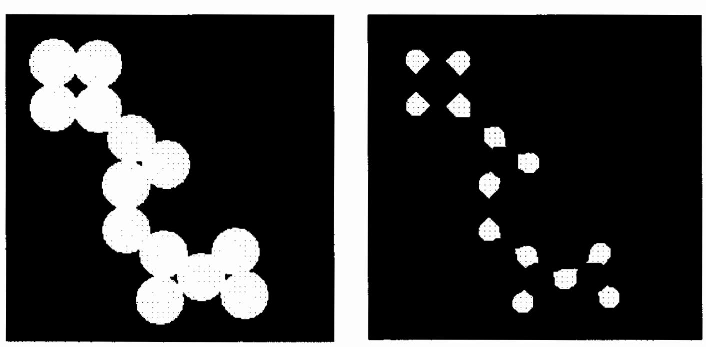


Figure 3.4: Image Erosion [20]

3.7 Connected Component Labeling

Once the necessary pixels have been removed, the next step is to distinguish one object from the other in the image. Labeling the connected components of the image does this. Two pixels are said to be 4-connected if they show horizontal or vertical adjacency. If this adjacency includes the diagonal neighbors as well, then the pixels are said to be 8-connected. In our system 4-connectivity has been used.

3.8 Feature Extraction

Once all the objects have been distinctly labeled, we extracted following features.

3.8.1 Area Of Object

The size of the object is calculated in terms of its area. The number of pixels composing that object represents the area of the object. It is expected that only hands will be in the work area, so the largest object will be that of the user's hand and all other objects are discarded, as there may be noise or redundant pixels belonging to some smaller object.

3.8.2 Minimum Enclosing Rectangle

The Minimum Enclosing Rectangle (MER) is the smallest rectangle that completely encloses the region. By finding the upper left coordinates and the lower right coordinates of the region a corresponding MER could be drawn around the object. The coordinates obtained will thereafter be used to determine the user's hand.

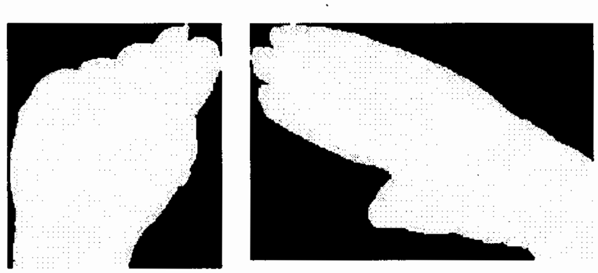


Figure 3.5 MER Of Hand Object

3.8.3 Length Of Major Axis

The principal axis of an object is a line passing through the object's center of mass having a minimum total distance from all pixels belonging to the object. It is the length (in pixels) of the major axis of the ellipse that has the same normalized second central moments as the region. We calculated major axis as it is used in several other features.

3.8.4 Eccentricity

The eccentricity can be defined as length/width of the given object and gives a measure of how elongated an object is. The value is between 0 and 1. (0 and 1 are degenerate cases) an ellipse whose eccentricity is 0 is actually a circle, while an ellipse whose eccentricity is 1 is a line segment.

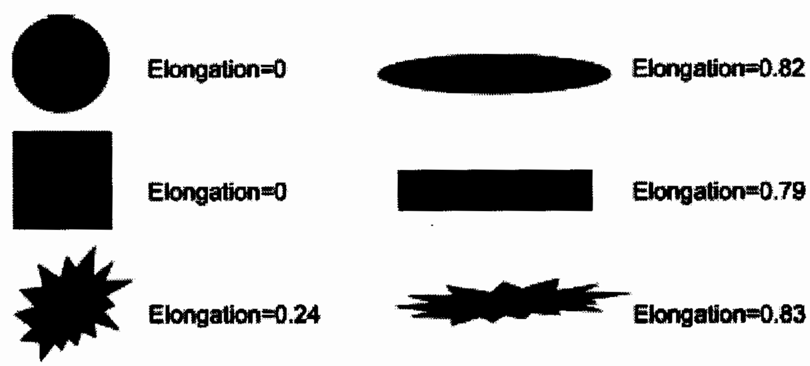


Figure 3.6: Eccentricity of different objects

3.8.5 Circularity

One measure of shape is to quantify the 'closeness' to a perfect circle. For this we used the parameter Circularity that is defined as follows:

$$Circularity = 4\pi A / P^2$$

Where A is the object area and P is its perimeter. Circularity is a ratio of the perimeter of a circle with the same area as the object divided by the perimeter of the actual object image. Circularity has values in the range 0-1. A perfect circle has circularity of 1 while a very 'spiky' or irregular object has a circularity value closer to 0. Circularity is sensitive to both overall form and surface roughness.

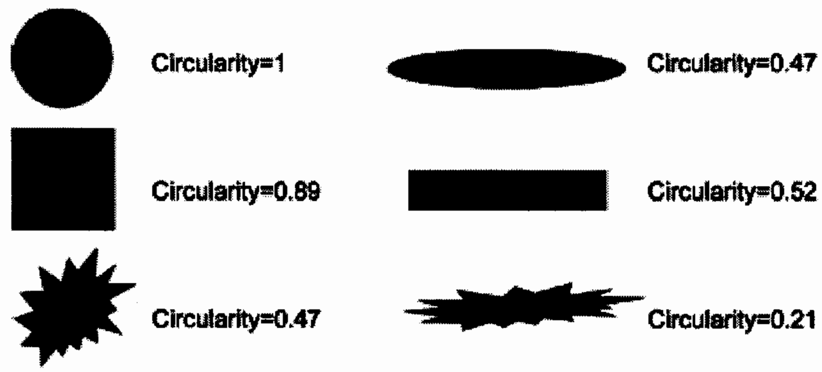


Figure 3.7: Circularity of different objects

3.8.6 Orientation

Its is the angle (in degrees) between the *x*-axis and the major axis of the ellipse that has the same second-moments as the region.

This figure illustrates the axes and orientation of the ellipse. The left side of the figure shows an image region and its corresponding ellipse. The right side shows the same ellipse, with features indicated graphically; the solid blue lines are the axes, the red dots are the foci, and the orientation is the angle between the horizontal dotted line and the major axis.

Figure 3.8: Orientation angle [20]

3.9 Gesture Classification

On the basis of some features we divided the gesture into classes. For these purpose neural nets was used

3.9.1 Neural Network

A neural network is a computational structure inspired by the study of biological neural processing. Neural networks are composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. As in nature, the network function is determined largely by the connections between elements. A neural network can be trained to perform a particular function by adjusting the values of the connections (weights) between elements. There are many different types of neural networks; from relatively simple to very complex, just as there are many theories on how biological neural processing works.

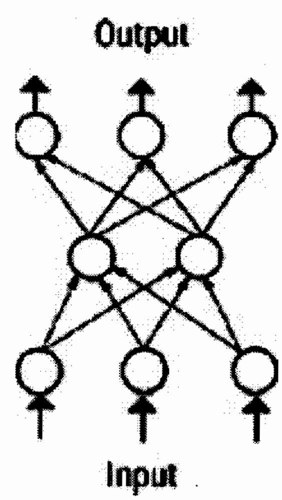


Figure 3.9: A Simple Neural Network

3.9.2 Supervised And Unsupervised Learning

A network can be subject to supervised or unsupervised learning.

The learning would be supervised if external criteria are used and matched by the network output, and if not, the learning is unsupervised. This is one broad way to divide different neural network approaches. Unsupervised approaches are also termed self-organizing. There is more interaction between neurons, typically with feedback and interlayer connections between neurons promoting self-organization.

3.9.3 Kohonen's Self Organizing Map

Self-organization means self-adaptation of a neural network. Without target outputs, the closest possible response to a given input signal is to be generated. Like inputs will cluster together. The connection weights are modified through different iterations of network operation, and the network capable of self-organizing creates on its own the closest possible set of outputs for the given inputs. This happens in the model in Kohonen's self-organizing map.

Self-organizing in networks is one of the most fascinating topics in the neural network field. Such networks can learn to detect regularities and correlations in their input and adapt their future responses to that input accordingly. The neurons of competitive networks learn to recognize groups of similar input vectors. Self-organizing maps learn to recognize groups of similar input vectors in such a way that neurons physically near each other in the neuron layer respond to similar input vectors. They are also called topology-preserving maps

There are m cluster units arranged in one or two-dimensional array; the input signals are n-tuples. The weight vector for a cluster unit serves as an exemplar of the input patterns associated with that cluster. During the self-organization process, the cluster unit whose weight vector matches the input pattern most closely is chosen as winner. The winning unit and its neighboring units (in terms of topology of cluster units) update their weight. [19]

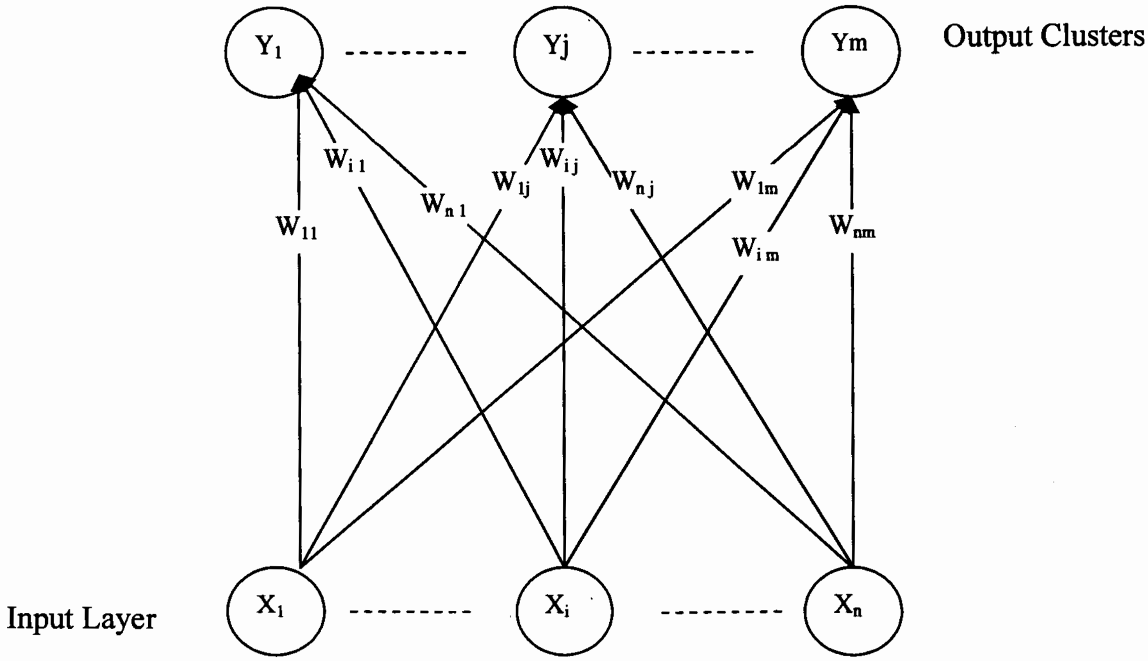


Figure 3.10: Kohonen's Self Organizing map architecture with n-inputs & m-clusters

We used Kohonen's self-organizing maps to group similar gestures into similar cluster. We used a feature vector consisting of four features of 132 images to be trained on net.

- Eccentricity
- Orientation
- Circularity
- Area

We used 7 by 6 layers of neurons to classify the vectors. We created a layer of 42 neurons spread out in a 7 by 6 grid. The learning rate was reduced from 0.9 to 0.02. And the neighborhood perimeter was reduced to 1 in 2000 epochs.

3.10 Gesture Recognition

There is no standard way to perform gesture recognition - a variety of representations and classification schemes are used. However, most gesture recognition systems share some common structure. Recognizing gestures is a pattern recognition task, which typically involves transforming the input into the appropriate representation and then classifying it from a database of predefined gesture representations. As gestures are highly variable from one person to another and from one example to another with a single person, it is essential to capture the essence of the gesture and use this as a representation.

Static gesture, or pose recognition can be accomplished by a straightforward implementation using template matching, geometric feature classification, neural networks, or other standard pattern recognition techniques to classify pose.

Once we have grouped the similar gestures in groups, we recognize the gestures among groups using feature "number of concavity"

3.10.1 Number Of Concavities

Number of concavities in a gesture can be used to recognize the gesture as it can give us the number fingers in hand, the shape of fingers. We calculated the number of concavities of two different sizes according the area

The number of concavities was calculated by taking convex image of gesture, n the subtracting it from original one. The areas left were counted according to specified threshold of area

a) **b)** **c)**
Figure3.11a) Hand object b) Convex Imagec) a-b

3.11 Translation

After recognition the appropriate text for that gesture was displayed.

3.12 Testing

After completion of all other phases, testing against the samples other than that which are used for training can check the working of the system. The system will then be tested to check its performance.

CHAPTER- 4

IMPLEMENTATION

4. Implementation

Implementation includes all the details that were required to make the system operational. The development tools and technologies to implement the system and also reasons for selecting particular tool is discussed. Then the modules being translated into the implementation tool will be described. This chapter introduces the tool used (MATLAB 7.0) and describes the functions used during the implementation phase.

4.1 MATLAB 7.0

The research work was implemented in MATLAB 7.0. MATLAB integrates mathematical computing, visualization, and a powerful language to provide a flexible environment for technical computing. The open architecture makes it easy to use MATLAB and its companion products to explore data, create algorithms, and create custom tools that provide early insights and competitive advantages.

MATLAB has evolved over a period of years with input from many users. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of add-on application-specific solutions called *toolboxes*. Moreover toolboxes in MATLAB, allow one to *learn* and *apply* specialized technology. These are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, image processing, image acquisition, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

4.2 Image Acquisition

The application is based on the static images i.e. signs. It is necessary to configure certain input device parameters. The input device in this case was a HP's colored digital camera of 5 MegaPixel. The resolution of camera was fixed.

The acquired images were stored in the .JPEG format. These images became the input for the system. For their processing certain functions are called and executed. Given below is the sequence of functions that are called along with their descriptions.

4.3 Image Preprocessing

The colored image is converted to gray scale. The gray scale image is binarized and then cleaned using a variety of morphological filters applied in succession. The filters clean the image by removing isolated pixels and spurious pixels.

```
HAND SEGMENTATION

    1- Convert grayscale image to binarized image:
          a. If pixel's value is below 0.5, set to 0.
          b. If the pixel's value is above 0.5, set to 1.

IMAGE PREPROCESSING

    1- Convert colored image to gray scale image.
          a. Set the hue and saturation components to zero.
          b. Retain the luminance
    2- Clean image.
          a. Remove each pixel whose value is 1 and is
             surrounded by 0's.
    3- Erode image:I
          a. If a foreground pixel has at least one
             neighbor belonging to the background, delete the
             pixel.
```

4.4 Region Labeling

Before an object can be described or recognized, its position in the image must be known, and all pixels belonging to the object must be identified. This function first segments the image into regions and then labels them. Since the pixels in a connected region are all ultimately connected to each other, it is necessary to find only one pixel belonging to the region. This pixel is known as the seed pixel. All other pixels connected to the seed pixel are then found. The area and bounding box are calculated. The object inside the bounding box is extracted and further processed.

```
REGION LABELING

Start Loop through image

   1-  Get seed pixel and set  its value  to  1
   2-  Set all 4-neighbors of seed pixel  to  1

End Loop
```

4.5 Feature Extraction

Following features were extracted.

4.5.1 Area Calculation

Region labeling may result in a number of regions being produced. It is of interest to use only that region which contains the hand object. The area property of the hand will be used to determine if an object is a hand. Although it is assumed that the only object that appears in the cameras field of view is the hand, other small objects may appear due to noise. The object with the largest area is taken to be the hand.

```
CALCULATE AREA

Start Loop through Number of Regions

   1-  Get area of each region

      a. Count the number of pixels belonging to that region

   2-  If the area satisfies a threshold, it is taken as hand
       Object else it is noise.
```

End Loop

4.5.2 Minimum Enclosing Rectangle

The system waits until a hand object is found. After it is known that a region contains a hand, the minimum enclosing rectangle (MER) of the hand object must be found. The MER is the smallest rectangle that will completely enclose the object.

```
MINIMUM ENCLOSING RECTANGLE

   1-  Find x-coordinates of the uppermost left corner of
       object

   2-  Find coordinates of the bottommost right corner of
       object

   3-  H = height of object by counting rows

   4-  W = width of object by counting columns
```

4.5.3 Length Of Major Axis

The Length of Major Axis is found to know the minimum total distance from all pixels belonging to the object to the centre of mass of an object.

```
MAJOR AXIS's LENGTH

Start loop

   1-  locate the centre of mass
   2-  locate all the boundary pixels that passes through
       the centre of mass
   3-  compute the distance between each line and all of the
       pixel in the object
   4-  select the line having the smallest total distance
```

```
End loop
```

4.5.4 Eccentricity

To recognize a certain sign we need to know all the dimensions of an image and eccentricity is measured to determine that how elongated an object is.

```
ECCENTRICITY

    1-  Calculate the length of the major axis
    2-  Calculate the length of the minor axis
    3-  Divide the length of minor axis by the length of
        minor axis.
```

4.5.5 Circularity

One measure of shape is to quantify the 'closeness' to a perfect circle.

```
CIRCULARITY

    1-  Calculate the area of the object.
    2-  Multiply that area with 4π.
    3-  Calculate parameter of the object.
    4-  Take the square of the parameter.
    5-  Divide the square of the parameter by 4πA.
    6-  If the answer is 1, the object is taken as circular
    7-  If the answer is 0 or near to 0, the object is not
        circular
```

4.5.6 Orientation

Orientation is the necessary feature to know the direction and angle of the hand gesture.

```
ORIENTATION

   1-  Calculate the major axis.
   2-  Calculate the angle between the major axis and x-
       axis.
```

4.6 Gesture Classification

On the basis of features, similar gestures were grouped into same cluster using SOM.

4.6.1 Self Organizing Maps

Following algorithm was used to cluster the input feature vector.

```
SOM

   1-  Initialization
         a. Initialize Weights
         b. Set neighborhood Parameter
         c. Set learning rate
   2-  While stopping condition is false do
         a. For each input vector do
               i.  Compute Euclidean distance
              ii.  Select the minimum
             iii.  Update weight with specified neighborhood
         b. Update learning rate
         c. Reduce neighborhood parameter
         d. Test stopping condition
```

4.7 Gesture Recognition

Gesture were recognized by counting the number of concavities in image

4.7.1 Number of Concavities

```
NUMBER OF CONCAVITIES

   1- Calculate convex image of gesture
   2- Subtract the original from convex image
   3- Count different areas left according to specified
      threshold
```

CHAPTER- 5

RESULTS AND CONCLUSION

5. Results and Conclusion

After analyzing the problem definition, gathering requirements, proposing a solution, and implementing the system at hand, the next step is to derive results that describe the performance of the system. This chapter describes the results of the application developed. Screenshots are provided to show the processing of the system.

5.1 Image Segmentation

We have used gray thresh function of Matlab for segmentation. This function calculates the threshold value of every image individually.

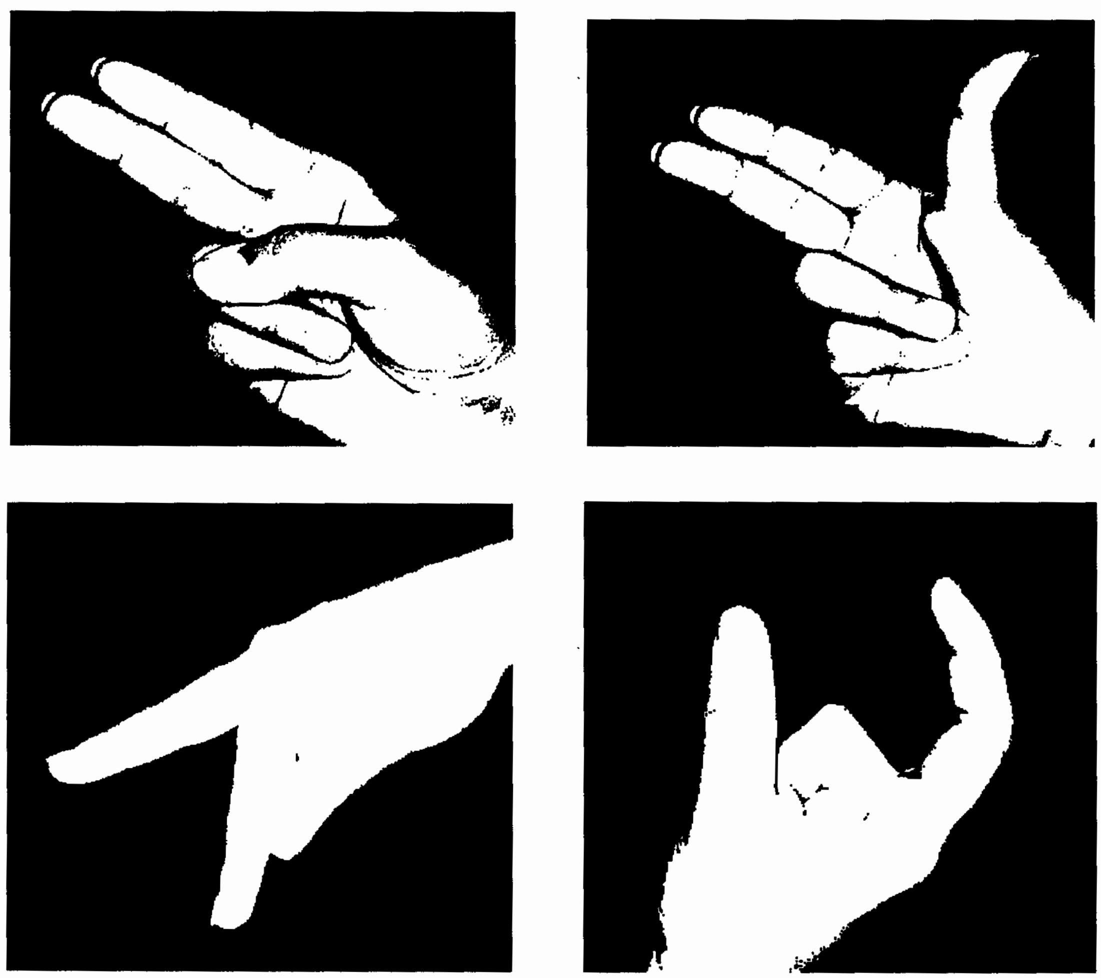


Figure 5.1: Image Segmentation

5.2 Feature Extraction

5.2.1 Minimum Enclosing Rectangle

Minimum Enclosing Rectangle (MER) is the smallest rectangle that completely encloses the region. By finding the upper left coordinates and the lower right coordinates of the region a corresponding MER was drawn around the object.

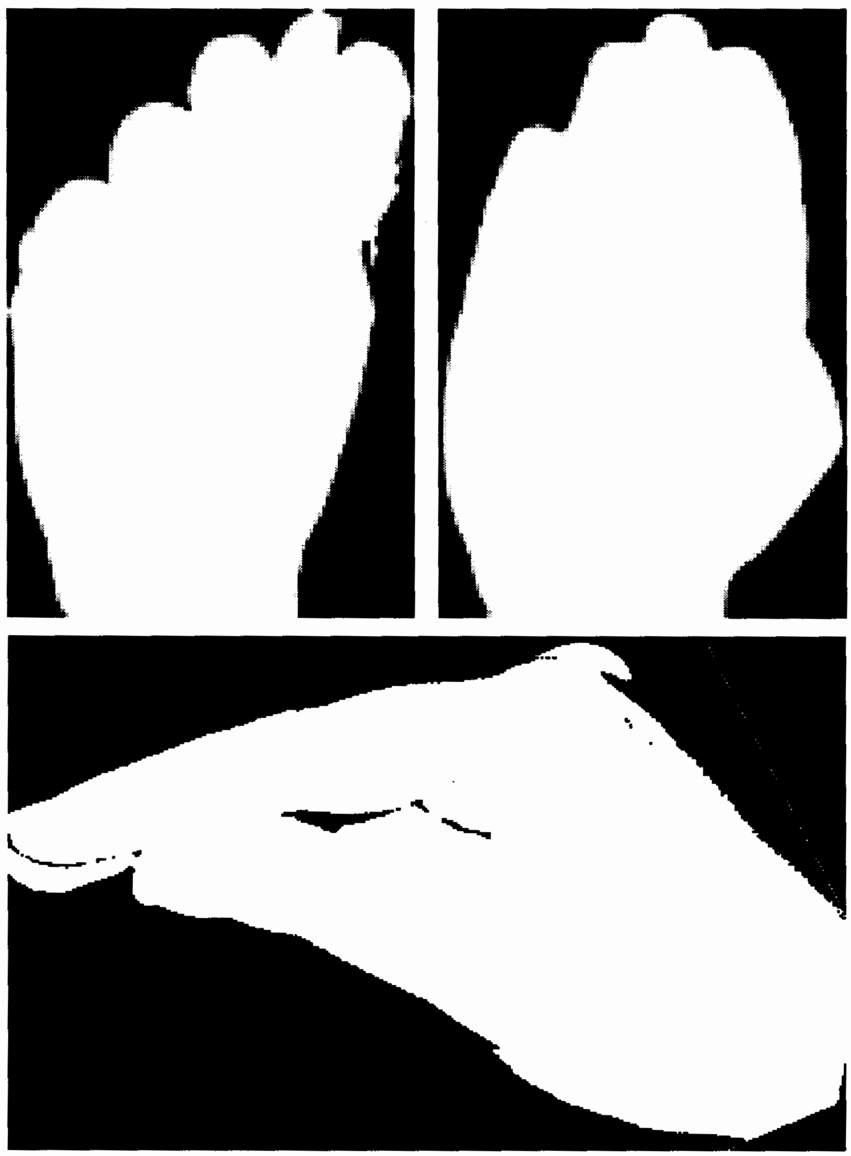


Figure5.2: MER of different hand gestures

5.2.2 Circularity

Circularity is a ratio of the perimeter of a circle with the same area as the object divided by the perimeter of the actual object image. Circularity has values in the range 0-1.

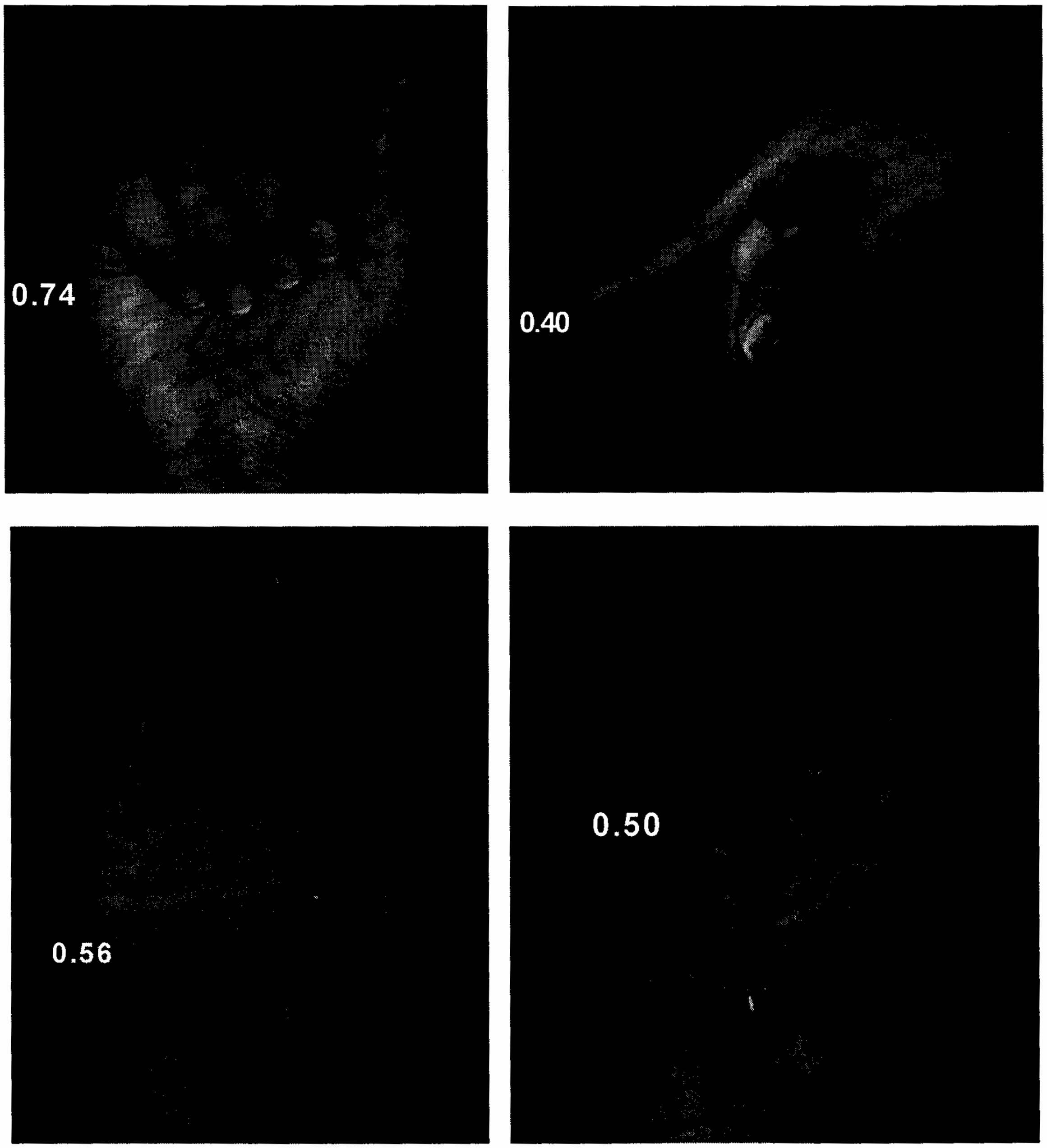


Figure5.3 Circularity of different hand gestures

5.2.3 Eccentricity

The eccentricity is the length/width of the given object and gives a measure of how elongated an object is. Its value is also between 0 and 1.

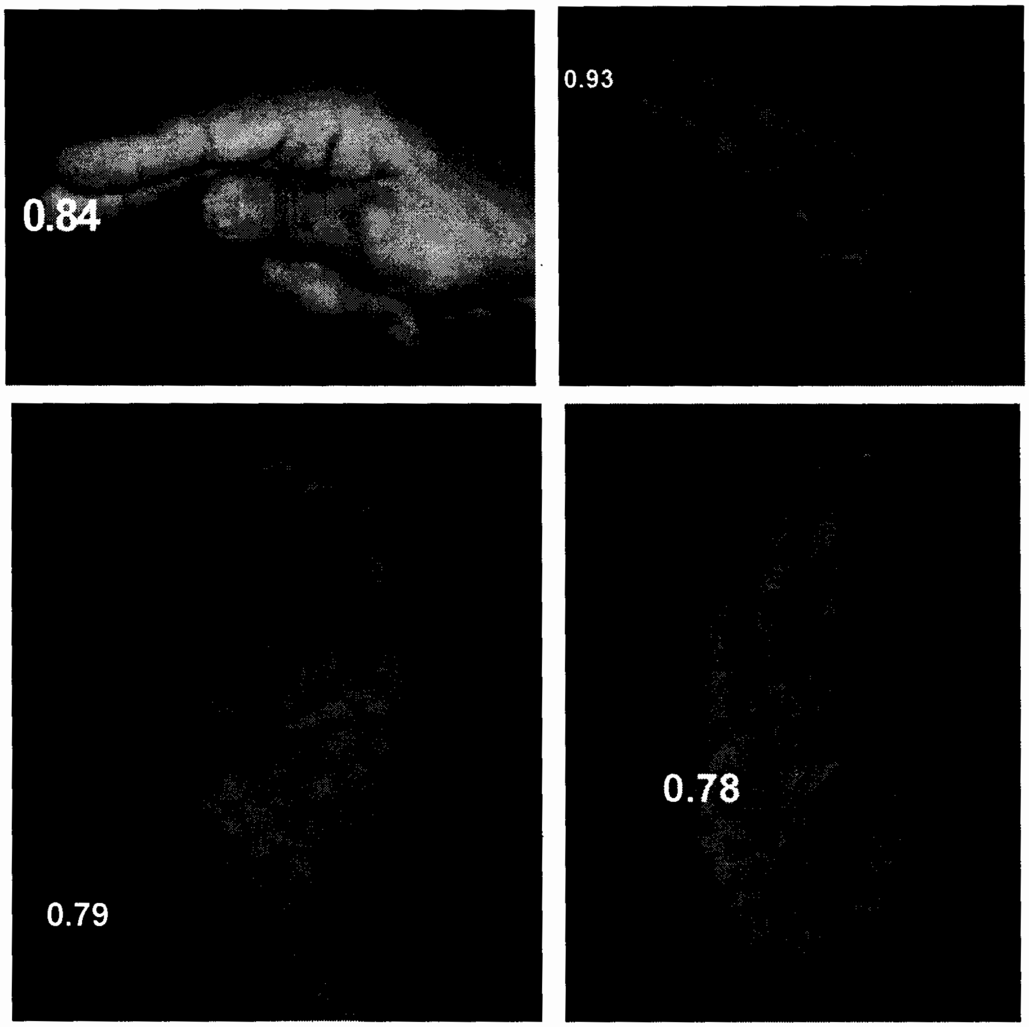


Figure5.4 Eccentricity of different hand gestures

5.3 Gesture Classification

5.3.1 Self Organizing Map

We used a feature vector consisting of four features i.e eccentricity orientation circularity and area of 132 images to be trained on net. All feature values were first normalized. We used 7 by 6 layers of neurons to classify the vectors. We created layer of 42 neurons spread out in a 7 by 6 grid. The learning rate was reduced from 0.9 to 0.02. And the neighborhood perimeter was reduced to 1 in 2000 epochs. The result was

Cluster	Gestures	Cluster	Gestures	Cluster	Gestures
1	ث$_1$ ث$_2$ ط$_2$	15	ء$_1$	29	
2	ط$_3$ ط$_4$ ظ$_2$ ظ$_3$ ظ$_4$ ل$_1$ ل$_2$ ل$_3$ ل$_4$	16	گ$_1$ گ$_2$ گ$_3$ گ$_4$ ء$_3$	30	چ$_3$ چ$_4$
3		17	ث$_3$ د$_1$ ء$_2$	31	
4	ع$_1$ ع$_2$ ع$_3$ ع$_4$ غ$_2$ غ$_3$	18	غ$_1$	32	ء$_4$ ذ$_4$
5		19	ض$_1$ ے$_3$	33	ف$_2$ ے$_1$
6	ق$_1$ ق$_2$ ق$_3$ ق$_4$	20	ڈ$_1$ ڈ$_2$ ض$_4$	34	ٹ$_4$
7	پ$_1$ پ$_2$ پ$_3$ پ$_4$	21	ص$_4$ ص$_1$ ص$_3$	35	ا$_2$ ا$_3$ ٹ$_1$ ٹ$_2$ س$_2$
8	ط$_1$	22	چ$_2$ چ$_1$ ح$_1$ ح$_2$ ح$_3$ ح$_4$ خ$_1$ خ$_2$ خ$_3$ خ$_4$	36	ت$_1$ ت$_2$ ت$_3$ ت$_4$ م$_3$ م$_4$ ن$_3$ ن$_4$
9	ڈ$_1$ ڈ$_2$ ڈ$_3$ ڈ$_4$ ظ$_1$	23	ث$_4$	37	ش$_2$ ش$_3$ ش$_4$
10	غ$_4$	24	د$_2$ د$_3$ ذ$_1$ ذ$_2$ ذ$_3$	38	ش$_1$
11		25	د$_4$ ک$_4$	39	ف$_3$
12	ض$_2$ ک$_1$ ک$_2$ ں$_1$ ں$_2$ ں$_3$ ں$_4$	26	ک$_3$ ے$_2$ ے$_4$ ض$_3$	40	ب$_3$ ڈ$_4$ ف$_1$ ف$_4$
13		27	ڈ$_3$	41	ب$_1$ ب$_2$ ب$_4$ و$_2$ ی$_3$
14	ص$_2$ ن$_1$ ن$_2$ م$_1$ م$_2$	28	ٹ$_3$	42	ا$_1$ ا$_4$ س$_1$ س$_3$ س$_4$ و$_1$ و$_3$ و$_4$ ی$_1$ ی$_2$ ی$_4$

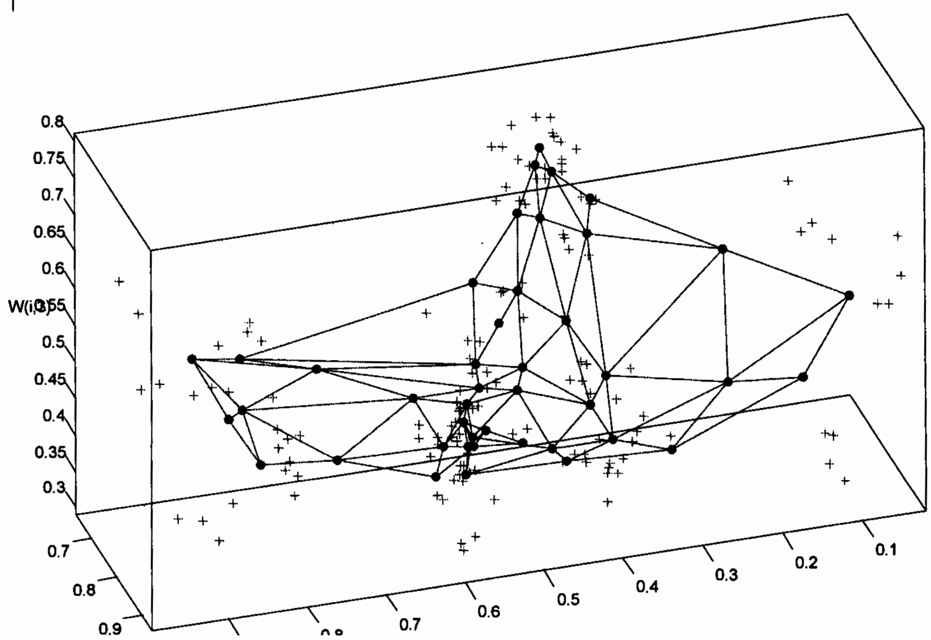


Figure 5.5 Neural Network results

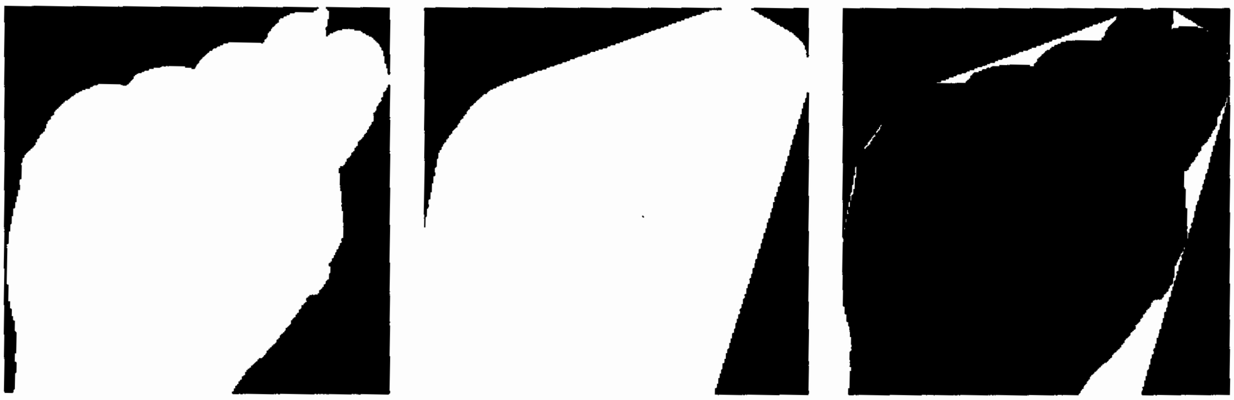
After clustering similar gestures in clusters were grouped in classes for final recognition.

We also tried to make classes on bases of feature, orientation, and final classification was feature based and also done without clustering, but result was not satisfactory. The recognition rate was not that high and it was difficult to combine similar gestures in similar classes on bass of only one feature.

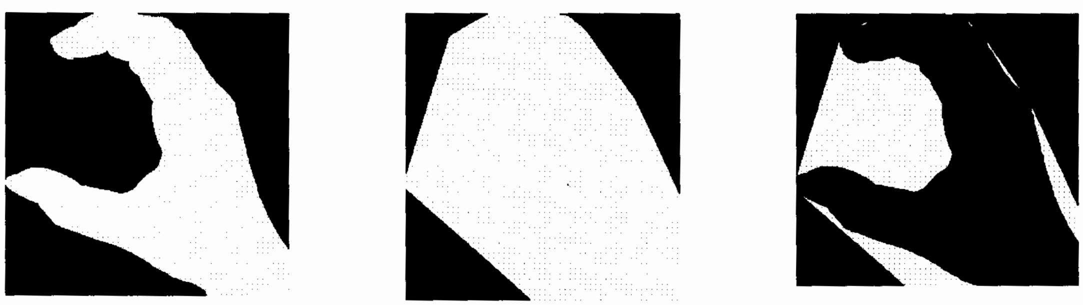
5.4 Gesture Recognition

5.4.1 Number Of Concavities

The gesture were then recognized by counting number of concavities in cluster



a) **b)** **c)**
Figure5.6 a) Hand object b) Convex Image c) a-b



a) **b)** **c)**
Figure5.7 a) Hand object b) Convex Image c) a-b

5.5 Results

5.5.1 Interface

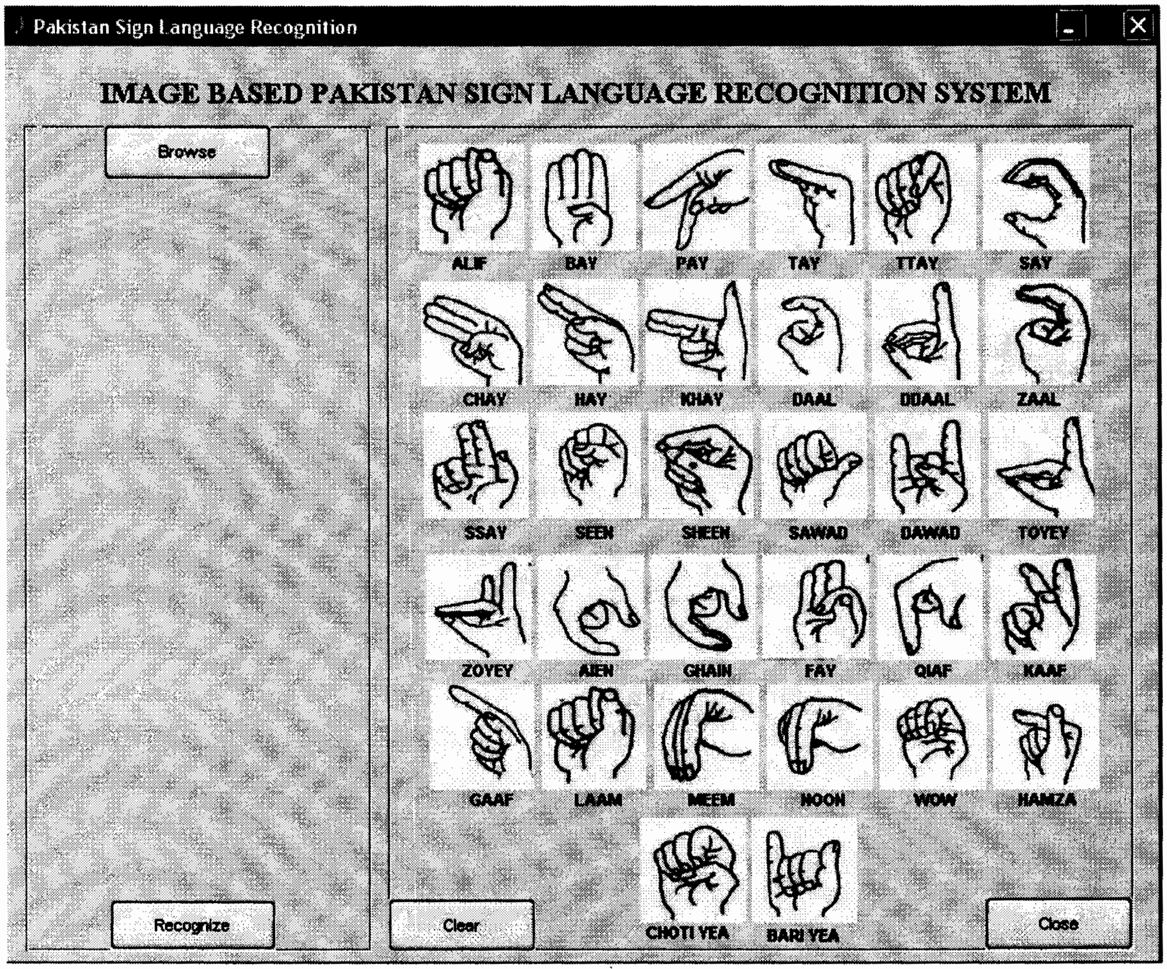


Figure5.8: System Interface

From browse button user can browse the images he wants to recognize.

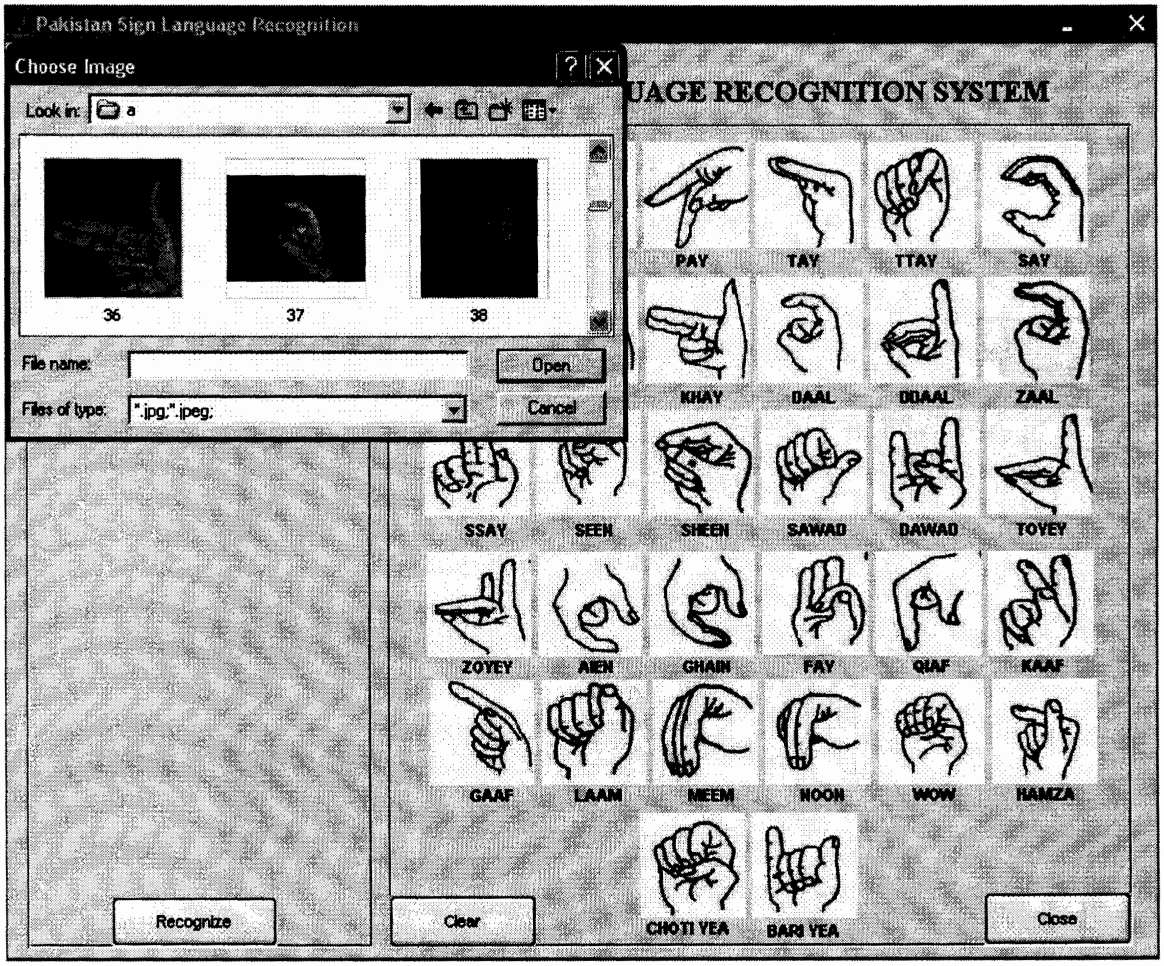


Figure5.9: System Interface

When user press recognize button, system takes the image perform following steps

- Segment image
- Extract Features
- Choose the cluster best matches the feature vector according to trained network
- Calculate number of concavities
- Display corresponding text

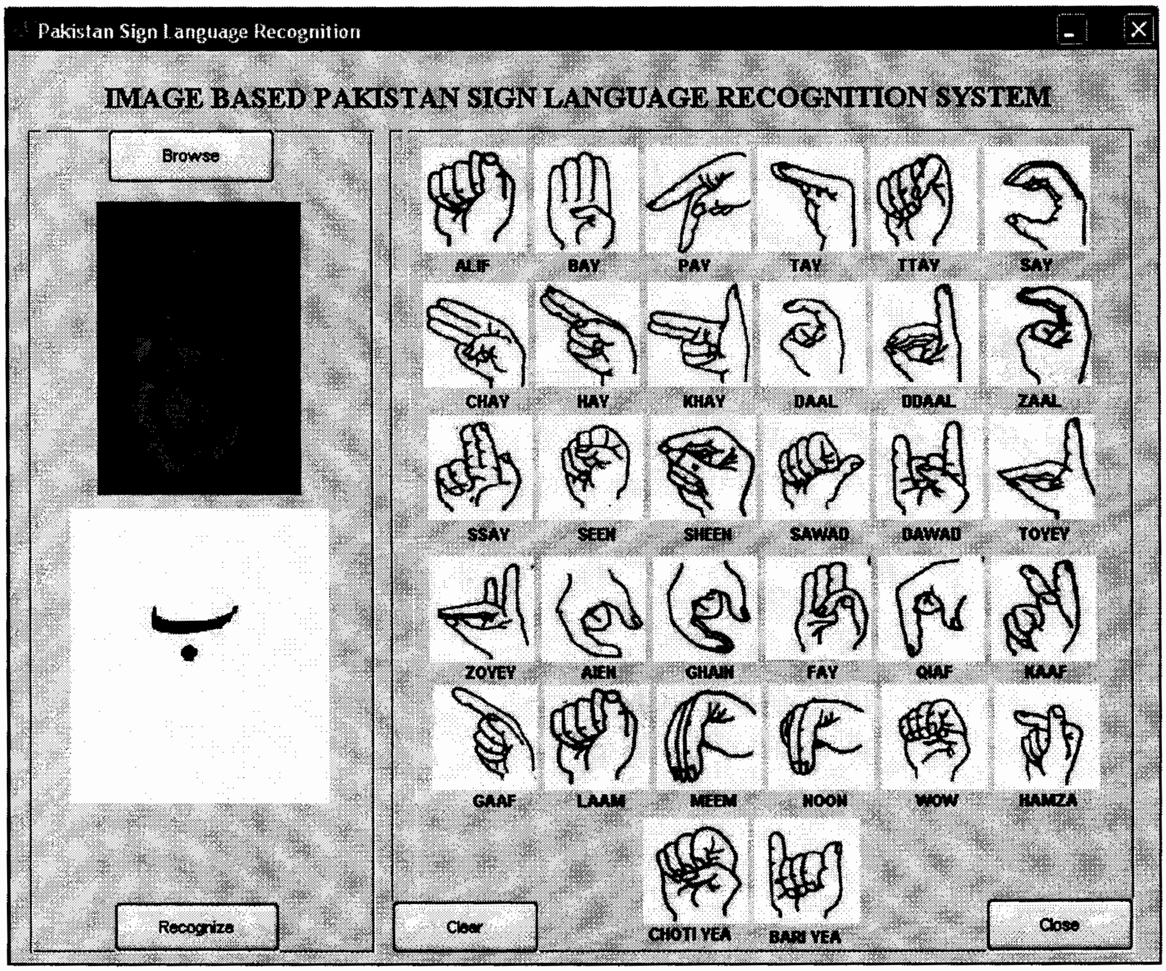
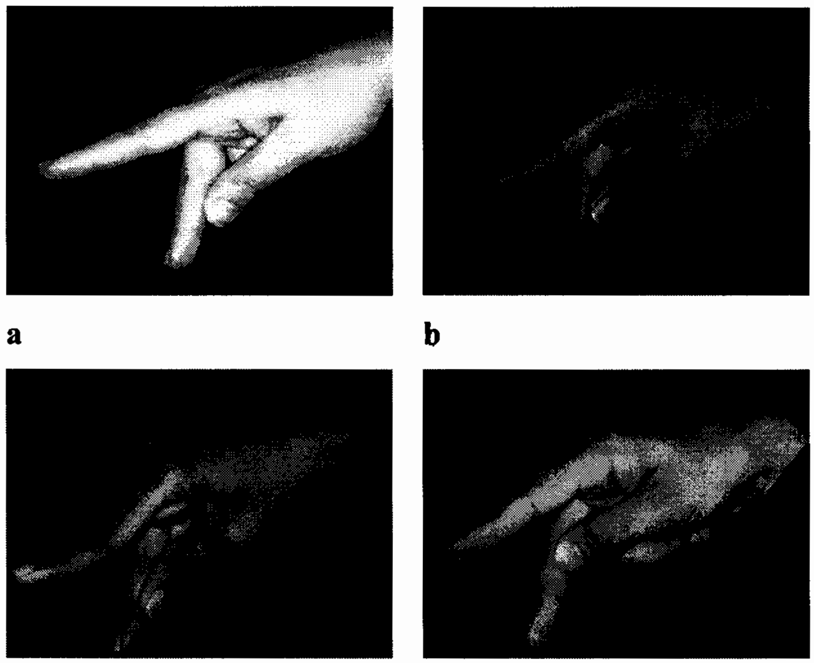


Figure5.10: System Interface

Results of some gesture are

PAY پ

All gestures were recognized.



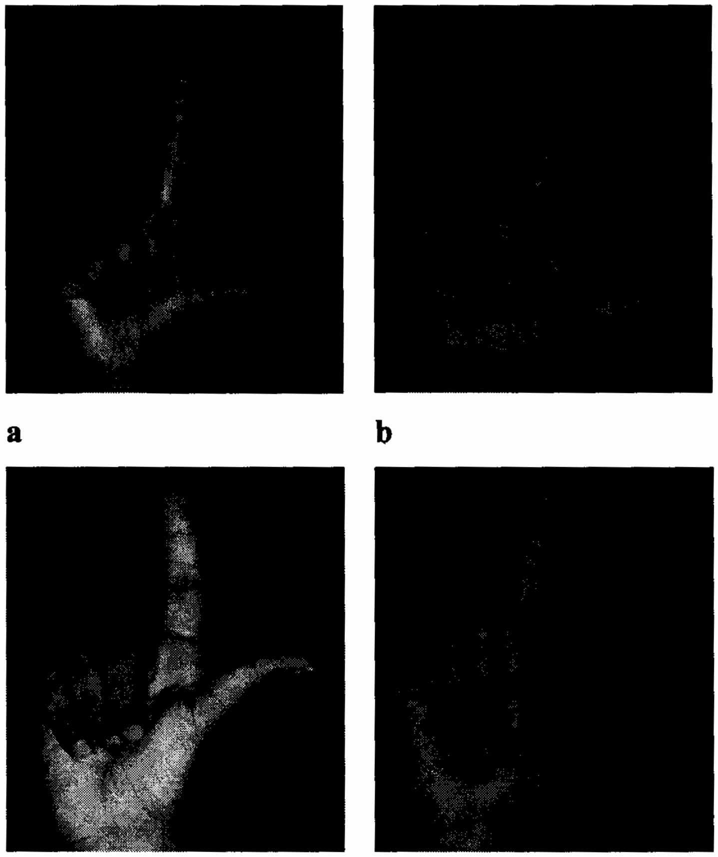
a b c d

Figure5.11: Test images for pay

Image	Classified
a	✔
b	✔
c	✔
d	✔

LAAM ل

All gestures were recognized.



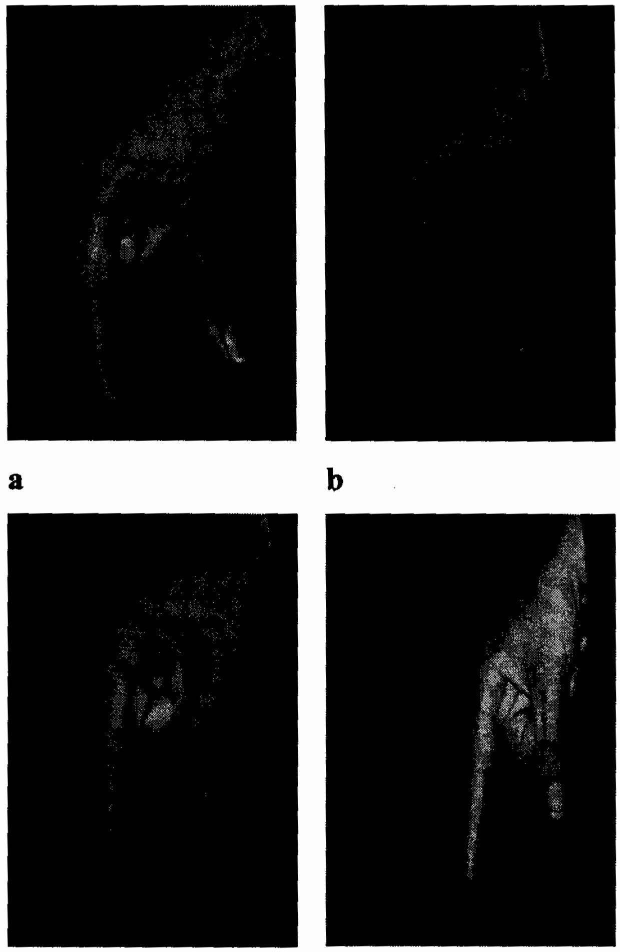
a b c d

Figure5.12: Test images for Laam

Image	Classified
a	✓
b	✓
c	✓
d	✓

QAAF ق

All gestures were recognized



a b

c d

Figure5.13: Test images for Qaaf

Image	Classified
a	✓
b	✓
c	✓
d	✓

NOON GUNNA ں

All gestures were recognized

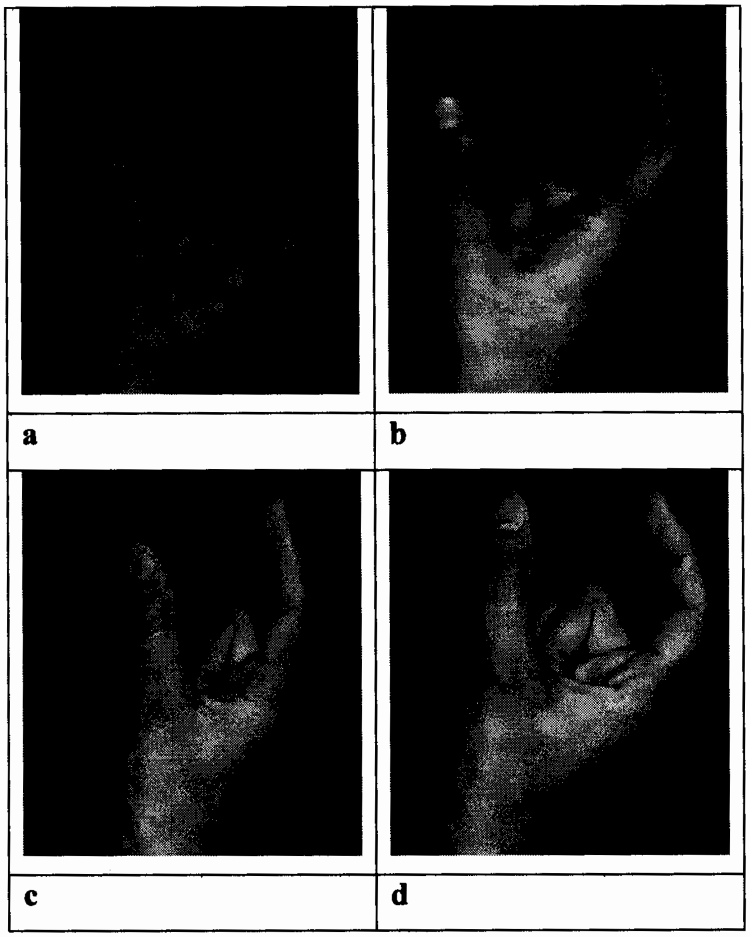


Figure5.14: Test images for Noon Gunna

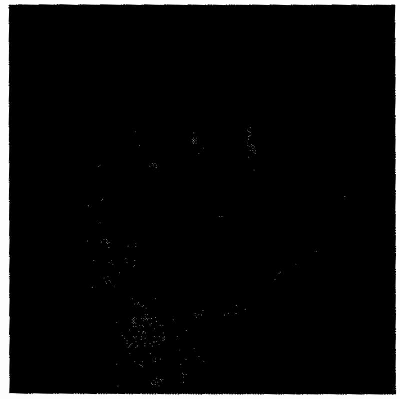
Image	Classified
a	✔
b	✔
c	✔
d	✔

The table gives below shows some results of the recognition rate.

We used two sets of signs to test data. First set consist of four signers signs that were used for training and second set was of test data which also comprised of four signers signs.

Gesture	Train Data	Test Data
Alf ا	3/4	2/4
Bayب	4/4	3/4
Payپ	4/4	4/4
Tayت	4/4	3/4
Ttayٹ	3/4	3/4
Sayث	4/4	3/4
Chayچ	4/4	3/4
Hayح	4/4	3/4
Daalد	2/4	2/4
Ddaalڈ	4/4	4/4
Zaalذ	3/4	2/4
Seyژ	2/4	2/4
Seenس	2/4	2/4
Sheenش	4/4	4/4
Swadص	4/4	4/4
Dwadض	4/4	3/4
Toyeyط	3/4	3/4
Zoyeyظ	4/4	4/4
Aieenع	2/4	2/4
Gayeenغ	2/4	2/4
Fayف	4/4	4/4
Qaafق	4/4	4/4
kaafک	4/4	4/4
Gaafگ	4/4	4/4
Laamل	4/4	4/4
Meemم	3/4	3/4
Noonن	3/4	2/4
Wowو	3/4	3/4
Hamzaء	4/4	4/4
ChotiYeaی	3/4	2/4
Bari Yeaے	3/4	2/4
NonGhunnaں	4/4	4/4
Success Rate	82%	76%

There are some gestures which were similar i.e, they contained ambiguity because of which the success rate was low. e.g, Alif and Seen.

a)Alif b)Seen

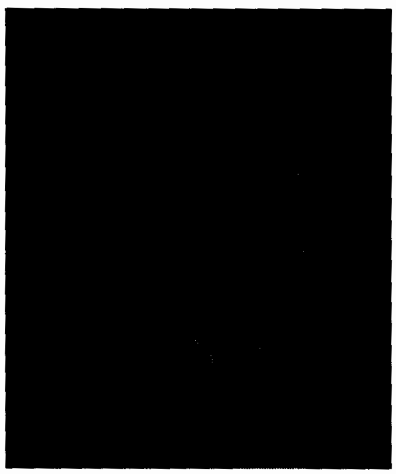
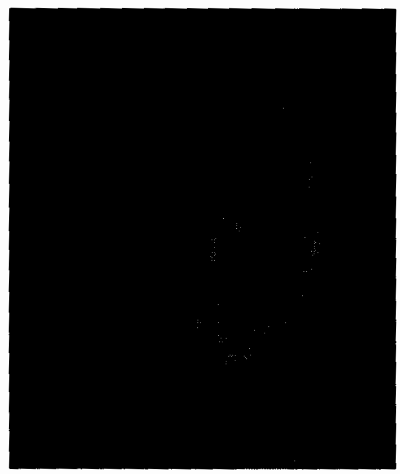



Figure a)Daal b)Zaal

Figure5.15: System Interface

5.6 Conclusion

This system was developed to allow deaf and dumb community to interact with normal people. This research work has several different features.

- It is vision based system that does not employ the use of data gloves

- It is second system which has recognized a subset of PSL and first system that is vision based.
- This system uses 32 gestures of Urdu alphabets.

5.7 Future Enhancement

Some suggestions for future enhancement include

- Remaining static gestures of PSL
- The addition of dynamic gestures i.e real time based application
- Conversion of recognition gestures of PSL into speech

APPENDIX

A.1 Reading The Image

A.1.1 Reading Train Data

```
v=132;
for i=1:v
        string = ['train\a\' num2str(i) '.jpg'];
        Rimages{i} = imread(string);
        Rimages{i}=imresize(Rimages{i},0.5);
        i
end
save Rimages Rimages
```

A.1.2 Browse image

```
[F,P]=uigetfile('*.jpg;*.jpeg;','Choose Image');
 if F==0
        return;
 else
         PF=[P,F];
        ext=PF(findstr(PF,'.')+1:end);
        img=imread(PF);
 end
```

A.2 Image Preprocessing

```
BWimages=bwmorph(Bm,'clean');
BWimages=bwmorph(BWimages,'spur', 10);
BWimages=bwmorph(BWimages,'majority');
BWimages=bwmorph(BWimages,'erode');
```

A.3 Region Labeling and Minimum Enclosing Rectangle

```
[L, N]=bwlabel(bw3);
stats=regionprops(L, 'Area', 'FilledImage', 'BoundingBox', 'Centroid')
for i=1:N
        areaa=[stats(i).Area];
        if(areaa>=8000);
                areaaaa(i)=[stats(i).Area];
                Bbox=floor([stats(i).BoundingBox]);
                ypos=Bbox(1);
                xpos=Bbox(2);
                h=Bbox(3);
                w=Bbox(4);
                centre=floor([stats(i).Centroid]);
```

```
                rectangle('position',[ypos xpos h w],'EdgeColor','r','LineWidth',2);
        end
end
```

A.4 Feature Extraction

A.4.1 Circularity & Area

```
function [c aree]=circularity(Bm)
L=0;
N=0;
[N L] = bwboundaries(Bm,'noholes');

% Display the label matrix and draw each boundary

for m = 1:length(N)
        boundary = N{m};
end

stats = regionprops(L,'Area','Centroid');

% loop over the boundaries

for m = 1:length(N)

        % obtain (X,Y) boundary coordinates corresponding to label 'k'
         boundary = N{m};

        % compute a simple estimate of the object's perimeter
         delta_sq = diff(boundary).^2;
         perimeter = sum(sqrt(sum(delta_sq,2)));

        % obtain the area calculation corresponding to label 'k'
        aree = stats(m).Area;

        % compute the roundness metric
        c = 4*pi*aree/perimeter^2;


end
```

A.4.2 Eccentricity & Orientation

```
[L, N]=bwlabel(Bm);
stats=regionprops(L,'MajorAxisLength','MinorAxisLength','Eccentricity','Centroi
d', 'Orientation');
```

```
for k=1:N
        angle=[stats(k).Orientation];
        if angle<0
                angle=angle+180;
        end
        ecc=[stats(k).Eccentricity];
end
```

A.5 Gesture Classification

A.5.1 SOM

```
load readimages
plot3(v(1,:),v(2,:),v(3,:), '+r');
hold on
net=newsom(minmax(v),[7 6],'hextop','linkdist',0.9,1500,0.02,1);
plotsom(net.iw{1,1},net.layers{1}.distances);
hold on
net.trainParam.epochs =2000;
net = train(net,v);
save net net
plotsom(net.iw{1,1},net.layers{1}.distances);
hold off
k=1;
b=1;
c=1;
d=2;
a = sim(net,v)
Cte = full(a);
[m n]=size(Cte);
y=zeros(2, n);

for i=1:n
        for j=1:m
                if Cte(j, i)==1
                        y(1,k)=i;
                        y(2,b)=j;
                        k=k+1;
                        b=b+1;
                end
        end
end
```

REFERENCES

[1] K. G. Derpanis, "*A Review of Vision-Based Hand Gestures,*" Unpublished. Feb. 2004 http://www.cvr.yorku.ca/members/gradstudents/kosta/publications/

[2] *Wikipedia An Electronic encyclopedia* Website http://www.wikipedia.com

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

[4] D.McNeill, "*Hand and Mind: What Gestures Reveal about thoughts*", Chicago: University of Chiago Press, 1992.

[5] McNeill, D.(1985). "*So you think gestures are nonverbal?*" Psychological Review. Vol 92(3) 350-371, July 1992.

[6] Starner, T., Weaver, J. & Pentland, A. (1998). "*Real-time American Sign Language recognition using desk and wearable computer based video*". IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(12), 1371–1375.

[7] C. Hummels , P. J. Stappers, "*Meaningful Gestures for Human Computer Interaction: Beyond Hand Postures*", Proceedings of the 3rd. International conference on Face & Gesture Recognition, p.591, April 14-16, 1998

[8] Thomas Baudel, Michel Beaudouin-Lafon, Annelies Braffort,Daniel Teil, *"An Interaction Model Designed For Hand Gesture Input"*

[9] J.Sturman, *"Whole-hand Input,"* Ph.D. Thesis, MIT Media Laboratory, Cambridge.MA,Feburary 1992.

[10] *The 5DT* Website, www.5dt.com

[11] Quinlan, J.R.: *C4.5: Programs for Machine Learning,* San Francisco: Morgan Kaufmann Publishers, San Mateo, CA, 1993

[12] Aleem Khalid Alvi, M. Yousuf Bin Azhar, Mehmood Usman, Suleman Mumtaz, Sameer Rafiq, RaziUr Rehman, Israr Ahmed. "*Pakistan Sign Language Recognition Using Statistical Template Matching*" Transactions On Engineering, Computing And Technology V3 December 2004

[13] Khaled Assaleh and M. Al-Rousan, "*Recognition of Arabic Sign Language Alphabet Using Polynomial Classifiers,*" EURASIP Journal on Applied Signal Processing, vol. 2005, no. 13, pp. 2136-2145, 2005.

[14] Mohamed Mohandes and Mohamed Deriche. "*Image Based Arabic Sign Language Recognition*" Proceedings of the Eighth International Symposium on Signal Processing and Its Applications, 2005. Volume 1, Issue , August 28-31, 2005 Page(s): 86 – 89

[15] Peter Vamplew and Anthony Adams "*Recognition of sign language gestures using neural networks*". Sign Language Recognition, Proceedings of the Conference on Technology and Persons with Disabilities, California State University, Northridge, 18-21 March, 1992

[16] Thad Starner, Alex Pentland "*Visual Recognition of American Sign Language Using Hidden Morkov Model*". Proc. International Workshop on Automatic Face and Gesture Recognition (AFGR '95), pp. 189–194, Zurich, Switzerland, June 1995

[17] J. Davis and M. Shah, "*Visual gesture recognition*", IEE Proceedings:Vision, Image and Signal Processing, vol. 141, no. 2, pp. 101–106, 1994.

[18] "*The Practical computer vision using C*" by J. R. Parker Univ. of Calgary, Calgary, Alta., Canada publishers John Wiley & Sons, Inc. New York, NY, USA

[19] Kohonen, T. "*Self-Organized and Associative Memory*", 2nd Edition, Berlin: Springer-Verlag, 1997

[20] *Matlab 7.0 Image Processing Toolbox* by The MathWorks, Inc