

Comparison of Feature Based Fingerspelling Recognition Algorithms

Aman Ghasemzadeh

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
June 2012
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Assoc. Prof. Dr. Aykut Hocanın
Chair, Department of Electrical and Electronic
Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Assoc. Prof. Dr. Erhan A. İnce
Supervisor

Examining Committee

1. Prof. Dr. Hüseyin Özkaramanlı
2. Assoc. Prof. Dr. Erhan A. İnce
3. Assoc. Prof. Dr. Hasan Demirel

ABSTRACT

Sign language is a manual language which uses hand gestures instead of sounds. These gestures are produced by combining hand-shapes, orientation and movement of hands. Sign language is not international and it has been defined with the intension of communicating with deaf people. In sign language, two major types of communication are considered. The first one is based on word sign vocabulary, where common words are defined by body language. The second, which is also known as fingerspelling is a letter based vocabulary which uses the letters in a particular alphabet and involves the use of hands only. The manual alphabets created for fingerspelling are called finger alphabets. There are two main families of manual alphabets. The one-hand and the two-hand families.

American Sign Language (ASL) is used for deaf people in America and south of Canada and it belongs to the one-hand family. The work carried out in this thesis includes the analysis of recognition performance of ASL fingerspelling under four main methods. Mainly the prominent feature extraction, Principal Component Analysis (PCA), Discrete Cosine Transform (DCT) based code assignment and Singular Value Decomposition (SVD) are coupled with circularity.

In this work while developing the ASL fingerspelling recognition for the 26 letters of the English alphabet a custom database has been used. This database was generated by using four different signers and each person has signed a total of six times for each letter. Hence 156 images for each signer and a total of 624 images for the entire alphabet were acquired. Each image had 640×480 pixel resolution. Throughout the simulations the custom hand dataset and three randomly shuffled versions of this original set were obtained. Each one of the four methods mentioned

above had been applied to the individual sets and the results were compared referring to accuracy in determining the signed characters. The simulation results show that when DCT and SVD are applied locally (to sub-blocks instead of the global image) they both give very good performances. In the case of 4:2 training vs. testing the overall recognition rate for the SVD applied locally is 100% and for the DCT applied locally this value was 97.11%. When the SVD is applied globally under the same conditions the overall recognition rate was 92.3%. In fact, SVD using all the singular values has a better performance than the DCT using only the most important coefficients. If we are not concerned about complexity SVD would give the highest overall recognition rate whereas if reduction in complexity is a must DCT is the best contender. The third best result was obtained using the prominent features based method. In contrast, the poorest recognition rate was related to the PCA. The performance of PCA is degraded since hand patterns are not correlated and the mean hand image is quite dispersed. As the training to testing ratio is decreased the overall performance for all methods would gradually also go down.

Keywords: American Sign Language, fingerspelling, principal component analysis, discrete cosine transform and singular value decomposition.

ÖZ

İşaret dili, ses yerine el hareketlerini kullanan manuel bir dil olarak tanımlanmaktadır. Bu figürler, değişik el-şekilleri, oryantasyonu ve hareketlerinden oluşmaktadır. İşaret dili uluslararası bir dil olmayıp duyma yeteneği bulunmayan insanlar ile iletişimin güçlendirilmesi amacıyla tanımlanmıştır. İşaret dilinde iki ana iletişim şekli tanımlanmaktadır. Bunların birincisi, ortak kelimeler için vücut dilinin kullanıldığı kelime işaretlerine dayanmaktadır. Aynı zamanda Parmak Yazımı olarak da adlandırılan ikinci iletişim yöntemi ise karakterleri özel bir alfabede kullanan ve yalnızca ellerin kullanıldığı yazıma dayalı bir anlatım yöntemidir. Parmak yazımı için oluşturulan manuel alfabe, parmak alfabesi olarak adlandırılmaktadır.

Amerikan İşaret Dili (AİD) [ASL] duyma yeteneği bulunmayan insanlar için Amerika ve Güney Kanada'da kullanılmakta olup bir-el ailesine aittir. Bu tez çalışmasında AİD Parmak Yazımı tanıma performansı 4 ana yöntem altında analiz edilmiştir. Bu yöntemler esas olarak Önemli Özellik Çıkarma, Ana Bileşen Analizi (ABA) [PCA], Ayrık Kosinüs Dönüşümüne (AKD) [DCT] dayalı kod ataması ve dairesellikle bir araya getirilen Tekil Değer Ayrışması (TDA) [SVD] olarak sayılabilmektedir.

Bu tez çalışmasında 26 statik İngiliz alfabesi karakteri için AİD tanıma sisteminin geliştirilmesi sırasında özel bir veritabanı kullanılmıştır. Bu veritabanı 4 değişik denek kullanılarak hazırlanmış olup her denek her bir karakter için toplam 6 kez işaret kullanmıştır. Dolayısıyla her bir denek için toplam 156 görüntü ve tüm alfabe için toplam 624 görüntü elde edilmiştir. Her görüntünün çözünürlüğü 640×480 pikseldir. Simülasyonlar boyunca özel el veritabanları ve bu orijinal setlerin gelişigüzel bir şekilde karıştırılan 3 versiyonu elde edilmiştir. Yukarıda açıklanan 4

yöntemin her biri bir kez olmak üzere her bir sete uygulanmış olup çıkan sonuçlar hassasiyet faktörü göz önünde bulundurularak karşılaştırılmıştır. Çıkan sonuçlar AKD ve TDA'nın yerel olarak kullanıldığında (global görüntü yerine alt-bloklara) bu yöntemlerin her ikisinin de çok iyi performans sergilediğini göstermektedir. 4:2'lik öğrenme – test durumunda genel tanıma oranı, yerel olarak uygulanan TDA için 100% olup bu oran yerel olarak uygulanan AKD için 97.11%'dir. TDA aynı koşullar altında global olarak uygulandığında genel tanıma oranı 92.3% olmuştur. Gerçek anlamda tüm tekil değerleri kullanan TDA yalnızca en önemli katsayıları kullanan AKD'den daha iyi bir performansa sahiptir. Karmaşıklık oranının azaltılmasının bir zorunluluk olduğu durumlarda AKD'nin en iyi performansı sunduğu gerçeğinin yanısıra, karmaşıklık oranının bizim açımızdan endişe yaratmadığı durumlarda TDA en yüksek genel tanıma oranını sunacaktır. En iyi üçüncü sonuç Önemli Özellik bazlı yöntem kullanıldığında elde edilmiştir. Buna karşın en zayıf tanıma oranı ABA'ya ait olmuştur. ABA'nın performansı el hareketleri arasında bir ilişki bulunmadığından ve ortalama el görüntüsünün oldukça dağınık olduğundan dolayı düşmüştür. Öğrenme – test oranının düşmesi ile birlikte tüm yöntemler için genel performans değerleri de giderek azalacaktır.

Anahtar Kelimeler: Amerikan İşaret Dili, Parmak Yazımı, Ana Bileşen Analizi, Ayrık Kosinüs Dönüştürme ve Tekil Değer Ayrışması

Dedicated to

My parents and brother

ACKNOWLEDGMENT

First and foremost I thank my supervisor Assoc. Prof. Dr. Erhan A. İnce for guiding and helping me in my master study, for his patience and sharing kindly his knowledge with me.

Besides my supervisor, I would like to thank Prof. Dr. Hüseyin Özkaramanlı and Assoc. Prof. Dr. Hasan Demirel, for giving their time in the contribution of the thesis as Jury members.

Many thanks to my friends who enhanced my motivation by supporting me with their presence.

I also wish to thank all the faculty members at the department of Electrical and Electronic Engineering, and specially the chairman, Assoc. Prof. Dr. Aykut Hocann.

Last but not least, I would like to express my appreciation to my parents and my brother who support me in all of my life.

TABLE OF CONTENTS

ABSTRACT.....	iii
ÖZ.....	v
DEDICATION.....	vii
ACKNOWLEDGMENT.....	viii
LIST OF TABLES.....	xii
LIST OF FIGURES.....	xiv
LIST OF SYMBOLS AND ABBREVIATIONS.....	xv
1 INTRODUCTION.....	1
1.1 Overview.....	1
1.2 Outline.....	2
1.3 Organization.....	4
2 PROMINENT FEATURES BASED RECOGNITION.....	5
2.1 Introduction.....	5
2.2 Feature extraction process.....	5
2.2.1 Detection of Skin Regions using <i>YCbCr</i> Color Space.....	6
2.2.2 Maximum correlation based template matching by using fast alignment ..	7
2.2.3 Additional object features.....	9
3 PRINCIPAL COMPONENT ANALYSIS.....	11
3.1 Introduction.....	11
3.2 Procedure.....	11
3.2.1 Finding the basis.....	12
3.2.1.1 Subtracting the mean.....	13
3.2.1.2 Calculating the covariance matrix.....	14

3.2.1.3 Calculating the eigenvectors and eigenvalues of the covariance matrix	15
.....	15
3.3 Eigenhand classification of hand images	17
4 DCT COEFFICIENTS BASED CODE ASSIGNMENT	20
4.1 Introduction	20
4.2 The DCT Process	21
4.2.1 One-dimensional DCT	21
4.2.2 Two-dimensional DCT	22
4.2.3 Reorganization of DCT Coefficients	24
4.2.4 Recognition	24
5 SINGULAR VALUE DECOMPOSITION COUPLED WITH CIRCULARITY	
MEASURE	25
5.1 Introduction	25
5.2 Significance of SVD	25
5.3 Circularity measure	26
5.4 Euclidean distance for selecting the best match	26
6 CUSTOM DATABASE	27
6.1 Custom Database	27
6.1.1 Camera and Programming Platform	27
6.1.2 Lighting	27
6.1.3 Background	27
6.1.4 Signers	27
7 SIMULATION RESULTS	39
7.1 Simulation approach	39
7.1.1 Prominent Features based Recognition Results	39

7.1.2 PCA Results	41
7.1.3 Performance Analysis for DCT Coefficients based Code Assignment	45
7.1.4 Recognition rates for SVD Coupled with circularity	47
8 CONCLUSION AND FUTURE WORK	53
8.1 Future Work	54
REFERENCES	55

LIST OF TABLES

Table 7.1: Prominent features based recognition results using 2:4 training/testing ratio.	40
Table 7.2: Prominent features based recognition results using 3:3 training/testing ratio.	40
Table 7.3: Prominent features based recognition results using 4:2 training/testing ratio.	41
Table 7.4: PCA performance using 4:2 training/testing ratios while using 128×128 resized images	42
Table 7.5: PCA performance using 3:3 training/testing ratios while using 128×128 resized images	42
Table 7.6: PCA performance using 2:4 training/testing ratios while using 128×128 resized images	43
Table 7.7: PCA performance using 4:2 training/testing ratios while using 256×256 resized images	43
Table 7.8: PCA performance using 3:3 training/testing ratios while using 256×256 resized images	44
Table 7.9: PCA performance using 2:4 training/testing ratios while using 256×256 resized images	44
Table 7.10: DCT coefficients based code assignment results using 4:2 training/testing ratio.	46
Table 7.11: DCT coefficients based code assignment results using 3:3 training/testing ratio.	46

Table 7.12: DCT coefficients based code assignment results using 2:4 training/testing ratio.	47
Table 7.13: Recognition rates for SVD Coupled with Circularity method applied to global images under 4:2 training/testing ratio.....	48
Table 7.14: Recognition rates for SVD method applied to global images under 4:2 training/testing ratio	48
Table 7.15: Recognition rates for SVD Coupled with Circularity method applied to global images under 3:3 training/testing ratio.....	49
Table 7.16: Recognition rates for SVD Coupled with Circularity method applied to global images under 2:4 training/testing ratio.....	49
Table 7.17: Recognition rates for SVD Coupled with Circularity method applied to 8×8 sub-blocks under 4:2 training/testing ratio	510
Table 7.18: Recognition rates for SVD method applied to 8×8 sub-blocks under 4:2 training/testing ratio	510
Table 7.19: Recognition rates for SVD Coupled with Circularity method applied to 8×8 sub-blocks under 3:3 training/testing ratio.....	51
Table 7.20: Recognition rates for SVD Coupled with Circularity method applied to 8×8 sub-blocks under 2:4 training/testing ratio.....	51

LIST OF FIGURES

Figure 2.1: RGB image and skin regions	7
Figure 2.2: Angle of orientation.....	8
Figure 2.3: Stage of sign segmentation. (a) RGB frame. (b) Skin regions. (c) Interested region. (d) Rotated of interested region. (e) Wrist hand cut (f) Resized a frame with an object in the center as bounding square.....	8
Figure 3.1: A collection of hand vectors in the image space	12
Figure 3.2: Basis of the Image Space.....	13
Figure 3.3: The average hand.....	16
Figure 3.4: The difference of some selected hands and the average hand.....	13
Figure 3.5: Eigenhands generation process.....	16
Figure 3.6: Visualization of eigenhands	16
Figure 3.7: Reconstruction of hand in the hand space	17
Figure 3.8: Three hand images and their projections onto the hand space	18
Figure 3.9: Possible projection into hand space.....	19
Figure 4.1: 1-D DCT process.....	22
Figure 4.2: 2-D DCT process.....	23
Figure 6.1: Alphabet for signer-1 with 6 copies of each letter.	30
Figure 6.2: Alphabet for signer-2 with 6 copies of each letter.	33
Figure 6.3: Alphabet for signer-3 with 6 copies of each letter.	35
Figure 6.4: Alphabet for signer-4 with 6 copies of each letter.	38

LIST OF SYMBOLS AND ABBREVIATIONS

(\bar{x}, \bar{y})	The center of the area
c	Circularity
C	Covariance matrix
$F(u)$	The function for 1-D discrete cosine transform
$f(u)$	The function for 1-D inverse of discrete cosine transform
$F(u, v)$	The function for 2-D discrete cosine transform
$f(u, v)$	The function for 2-D inverse of discrete cosine transform
S	Matrix of singular values
w_k	Projections onto the hand space
θ	The angle of orientation
μ_R	Mean radial distance
σ	Singular value
σ_R	Standard deviation of radial distance
ψ	Mean of hands
Γ_i	Selected training set of hand images
Φ_i	Each hand differs from the mean hands
ASL	American Sign Language
DCT	Discrete Cosine Transform
IDCT	Inverse of Discrete Cosine Transform
MI	Mutual Information
PC	Principal Component
PCA	Principal Component Analysis
SV	Singular Value

SVD	Singular Value Decomposition
SVM	Support Vector Machine

Chapter 1

INTRODUCTION

1.1 Overview

The most structured sets of gestures belong to the sign languages. Sign language is a manual language which uses hand gestures instead of sound. These gestures are produced by combining hand-shapes, orientation and movement of hands. Sign language is not international. As spoken languages vary, sign languages also differ from region to region. Sign language is defined as a language for deaf people. It shortens the time for communication and also allows the possibility of sharing feelings much faster.

American Sign Language (ASL) is the language that is used for deaf people in America and south of Canada. ASL includes approximately 6000 gestures for common words and finger spelling for communicating unclear words or specific nouns that there is no gesture defined for. Developing sign language applications are extremely beneficial, since some people in the society are not able to speak, read, heard or writes. In sign language, two major types of communication are considered. The first one is based on word sign vocabulary, where the common words are defined by body language. The second one is a letter based vocabulary which is the representation of letters and numbers in an alphabet and a number system using only hands that is called fingerspelling. The manual alphabets created for fingerspelling are called finger alphabets. There are two families of manual alphabets which are the one-hand and the two-hand versions. The one-hand alphabet is more common and

dates back to 15th century. The two-hand manual alphabets have been developed by England, Scotland, Wales and Turkey. The early work on gesture analysis and fingerspelling started with the use of gloves equipped with sensors and trackers. Hollar used a sensing glove with six embedded accelerometers. This scheme would recognize 1 character/second out of 28 static hand gestures [1]. Lamar and Bhuiyant [2] have developed another method by using color gloves and neural networks. The success rate of their new method was stoked to be 23% better than the rate of what is reported in [1]. These methods are not convenient for daily applications because they require users to wear cumbersome devices on their hand to take accurate data for recognizing the hand position and finger configuration. Rebollar et al used more complicated gloves to classify 21 out of 26 letters with a remarkably high success rate [3]. In [4] Isaaca and Foo developed a two layer feed-forward neural network that could recognize 24 static letters in ASL alphabet.

The more autonomous methods based on prominent feature extraction, principal component analysis, codeword assignment, single value decomposition, support vector machine (SVM), discrete cosine transform, wavelet, etc. They had too many advantages to compare with other methods that four of them will be mentioned in this work.

1.2 Outline

The work presented herein covers concepts like prominent feature extraction, principal component analysis (PCA), discrete cosine transform (DCT) based code assignment and singular value decomposition (SVD) coupled with circularity.

Prominent Features based Recognition: in image processing and pattern recognition one of the forms that can reduce dimensionality is feature extraction. When the input data is too large for processing in real-time, the dimensionality

reduction is applied to extract some features related to the global image and then these features instead of the global image are used for recognition of the letters.

PCA: the basis of PCA method is minimizing the dimension of the classification problem by reducing the variable number of datasets. In the case of hand recognition, there is a fact that is not necessary to analyze the whole hand image, but it is enough to store and analyze only difference between individual hand and it is used for hand recognition in this chapter. Each hand is uniquely described by a one-dimensional vector.

DCT Coefficients based Code Assignment: as DCT is a linear transform it can represent a finite sequence of data points in terms of a sum of cosine functions which are oscillating at different frequency [5]. The direct DCT transform apply to the original signal for converting it to the frequency domain. After transforming, the importance of the frequencies that are present in it will be reflected by DCT coefficients. The signal's lowest frequency is related to the very first coefficient and it carries the most meaningful and essential information from the original signal. The signal's higher frequencies are related to the last coefficient and they represent more detailed of signal [6]. The coefficients which are between the first and the last coefficients carry various information of the original signal. After extracting DCT coefficients for the train and test sets, by using Euclidean distance, minimum distance will be found. In image recognition and processing for reducing image information redundancy DCT is one of the best methods, since it just keeps transform coefficients which are essential to conserve the most important features [7].

SVD coupled with Circularity: SVD method can transform matrix of an image into product USV^T , which is refactoring a digital image in three matrices. This

refactoring can conserve useful features of the original image and represents the image with a smaller set of values instead of using less storage space in the memory. The singular values $\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n$ are unique; however the matrices U and V are not unique, so each image has its own singular values [8]. After extracting SVD for the train and test sets, by using Euclidean distance, minimum distance will be found.

1.3 Organization

Chapter 2 provides details about prominent feature extraction and explains how features are extracted and used for recognition. This is followed by an explanation of PCA in Chapter 3. Next in Chapter 4, the DCT coefficient based code assignment method is discussed. In Chapter 5 information about SVD and its applications for hand gesture recognition is represented. Discussion of custom database used is provided in Chapter 6. The results that obtained via Monte Carlo simulation are presented in Chapter 7. Finally, Chapter 8 makes some conclusion and provides directions for future work.

Chapter 2

PROMINENT FEATURES BASED RECOGNITION

2.1 Introduction

For the fields of pattern recognition and image processing feature extraction is a dimensionality reduction technique. Through the application of feature extraction the amount of resources that are required to describe a large set of data accurately will be reduced. When the input data is large and perhaps also redundant, the requirement for memory and computation power will increase and the need for a classification algorithm that transforms the input data into a feature vector (reduced representation of the global data) becomes clear. This generally relaxes the time considerations related to the training data. The transformation technique should be chosen such that extraction of the feature sets from the large collection of data should be as quick and as easy as possible and yet the feature sets should contain all the relevant information required for either identifying or differentiating.

2.2 Feature extraction process

Feature extraction identifies characteristic features by forming a good representation of an object; in this work, the objects are the skin regions which correspond to the hands of people who are using sign language to communicate. After the segmentation of skin regions a binary mask corresponding to each hand sign is obtained and features related to geometry etc. are then extracted from these masks and stored for each sign.

The skin region in an image is detected by color segmentation in YC_bC_r Space. The angle of orientation between the axis of least second moment and y axis applies to hand regions. This angle is used in a fast alignment method which helps to return all the images in a same condition. As all the images are in the same condition, the signs which are similar in shape but different in orientation can be easily differentiated. There is a bounding which is called a minimum bounding square that is used for resizing without breaking the alignment. As the image pixels are not enough for recognition, additional object features of this bounding are extracted and used as feature values, which are allowed to the algorithm. These features could be, an area of the object, center of the area, perimeter of the object, angle of orientation, mean radial distance, standard deviation of radial distance and circularity [9]. The extraction of these entire features is going to be represented by the following part.

2.2.1 Detection of Skin Regions using YC_bC_r Color Space

YC_bC_r in color space is not an absolute color space; it is a way of encoding RGB information so it is superior to RGB color space for skin detection. The difference between YC_bC_r and RGB is that YC_bC_r represents the color as brightness and two color difference signals, Y is lumulus which represents the brightness, C_b is blue minus Y and C_r is red minus Y , while RGB represents the color as red, green and blue. Hence the pixel values of images are converted from RGB color space to YC_bC_r as follow:

$$Y = 0.299R + 0.587G + 0.114B$$

$$C_r = B - Y \tag{2.1}$$

$$C_b = R - Y$$

D. Chai and K. N. Ngan [10] reported that C_b and C_r have the close values for the pixels that belong to skin region. Consequently, Y, C_b and C_r values of range $Y > 115, 135 < C_r < 180, 85 < C_b < 135$ belong to the skin region (Fig. 2.1).



Figure 2.1: RGB image and skin regions

2.2.2 Maximum correlation based template matching by using fast alignment

Size and orientation have too much effect on template matching. As all the images have not taken in the same condition (all of them not straight or do not have a same size), it will be generated an orientation problem which needs to be solved by finding an angle of orientation. First of all the center of the mass is going to be found (Eqn. 2.2) then the orientation method will be applied to the center [11]. Angle of orientation is the angle between the *axis of least second moment* and y axis (shown in Fig. 2.2) which can be found by equation 2.3:

$$\bar{x} = \frac{1}{A} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} xI(i,j) \quad (2.2)$$

$$\bar{y} = \frac{1}{A} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} yI(i,j)$$

$$2\theta = \arctan\left(\frac{2M_{11}}{M_{20} - M_{02}}\right) \quad (2.3)$$

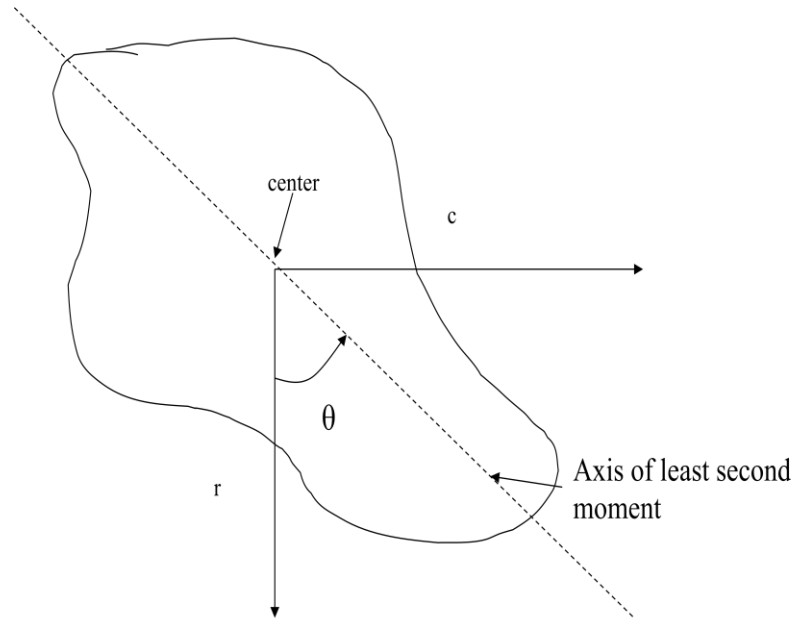


Figure 2.2: Angle of orientation [9]

Where $(I(x, y) = 1$ for pixels on the object, otherwise 0), A is the area of mass, $M_{11} = \sum_x \sum_y xyI(x, y)$, $M_{20} = \sum_x \sum_y x^2I(x, y)$ and $M_{02} = \sum_x \sum_y y^2I(x, y)$ [9].

All the pixels of the object can be completely surrounded by the smallest square; this square is defined as a bounding square which is used for resizing an image without breaking the alignment. Fast alignment process is going to be completed by finding the center of bounding square and then put the image in that center (Fig. 2.3).

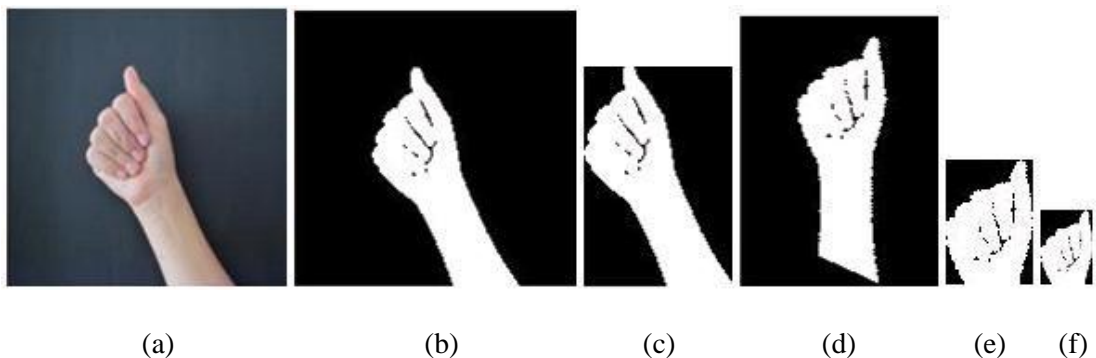


Figure 2.3: Phases of sign segmentation. (a) RGB frame. (b) Skin regions. (c) Interested region. (d) Rotated of interested region. (e) Wrist hand cut (f) Resized a frame with an object in the center of bounding square.

2.2.3 Additional object features

As pixel values of the bounding square are not enough for a decision processes, additional features are needed to improve the decision process.

These features are the area, center of the area, circularity and perimeter. In addition, mean radial distance μ_R and standard deviation of radial distance σ_R are extracted.

- Area: whose value corresponds to the total number of on pixels in the image, because of different patterns of pixels are weighted differently it may not be exactly the same as a real amount.
- Center of area (centroid): defined as the center of mass of the region. The first element of the centroid is x -coordinate, and the second element is y -coordinate of the center of mass.
- Perimeter: The distance around the boundary of the region which is computed by calculating the distance between each adjoining pair of pixels around the border of the region.
- Circularity: It is defined by $(perimeter^2/area)$ but practically it is calculated by (μ_R/σ_R) .

$$\mu_R = \frac{1}{N} \sum_n \|(x_n, y_n) - (\bar{x}, \bar{y})\| \quad (2.4)$$

$$\sigma_R = \left(\frac{1}{N} \sum_n [\|(x_n, y_n) - (\bar{x}, \bar{y})\| - \mu_R]^2 \right)^{1/2} \quad (2.5)$$

N is the number of pixels, n iterates over all pixels, (\bar{x}, \bar{y}) is the center of area, (x_n, y_n) is the n^{th} pixel coordinate, and $\| \cdot \|$ denotes the Euclidean distance [9].

At the end circularity is computed by

$$c = \mu_R / \sigma_R \quad (2.6)$$

At the end, these 6 additional features are added to the boundary square size of 40×40 .

After extracting features, the algorithm will be applied to the features extracted.
The scenarios of training/testing ratios and simulation will be discussed in Chapter 7.

Chapter 3

PRINCIPAL COMPONENT ANALYSIS

3.1 Introduction

PCA was introduced in 1901 by Karl Pearson [12]. PCA is a statistical technique which is a useful technique to classify and identify patterns in high dimensional data. The advantage of finding these patterns is that PCA reduces the number of dimensions by compressing the data without losing too much information [13]. This reduction of data is called principal component (PC) that will account for most of the variance in the high dimensional data [14]. In other word, PCA is a mathematical procedure which is in the definition; an orthogonal transformation is going to be used to convert a set of high dimension data of possible correlated variables into a set of values of reduced data. They are linearly uncorrelated variables which are called PCs. The number of PCs is less than or equal to the number of original variables which the largest possible variance is related to the first PC [15]. If the data set is jointly normally distributed then, the PCs are guaranteed to be independent.

3.2 Procedure

Principle component analysis aims to catch the total variation in the set of the training set, and to explain the variation by a few variables. Usually after normalizing the data matrix for each attribute, eigenvalue decomposition of a data covariance (or correlation) matrix or SVD of a data matrix can calculate PCA [15]. The PCA results are usually discussed in terms of component scores and loadings which is the weight, and each standardized original variable should be multiplied to get the component

score, sometimes defined as a transformed variable value that correspond to a particular data point which is called factor scores [16]. The most famous method based on PCA is definitely an Eigenface method [17], and it is going to explain in this work just by changing eigenface to eigenhand.

In the case of hand recognition, there is a fact that is not necessary to analyze the whole hand image, but it is enough to store and analyze only difference between individual hand and it is used for hand recognition in this chapter. Each hand is uniquely described by a one-dimensional vector.

The PCA steps to approach eigenhands are going to be presented by the following sections.

3.2.1 Finding the basis

A basis is a set of linearly independent vectors that represents every vector in a given vector space which are located in a coordinate system (Figure 3.1 & Figure 3.2). The goal of PCA is to identify the most meaningful basis to re-express a data set [18]. To find a new basis there are some steps which are going to be explained by the following sections.

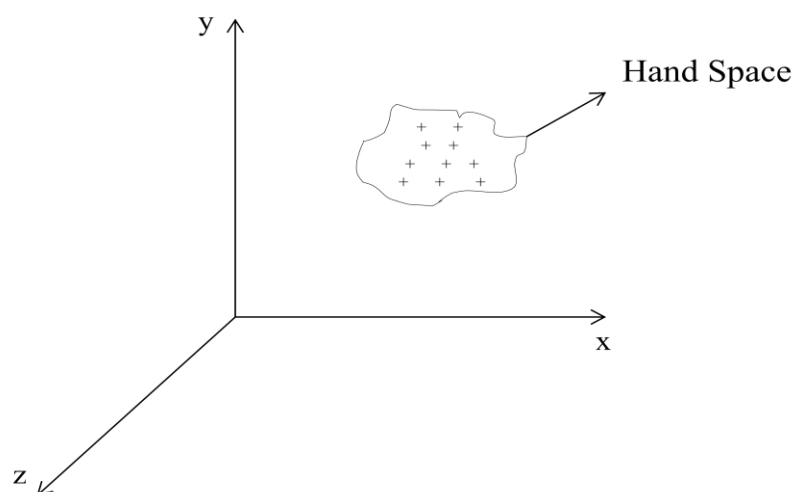


Figure 3.1: A collection of hand vectors in the image space [18]

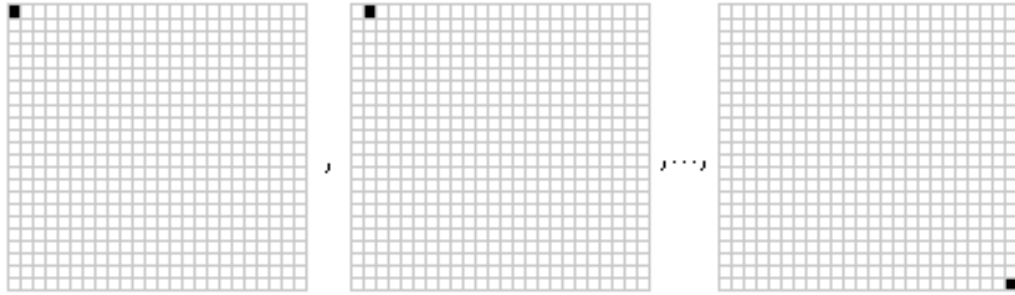


Figure 3.2: Basis of the Image Space [18]

3.2.1.1 Subtracting the mean

The aim of this part is to produce a data set whose mean is zero [13]. Hence the mean (eqn. 3.1) (summation of all images divided by the number of images) is subtracted from each of the image's dimensions, which are the average across each dimension (eqn. 3.2).

$$\psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i \quad (3.1)$$

$$\Phi_i = \Gamma_i - \psi \quad (3.2)$$

Where the selected training set of hand images are $\Gamma_1, \Gamma_2 \dots \Gamma_M$, ψ is a mean and $\Phi_1, \Phi_2 \dots \Phi_i$ are each hand differs from the average hand by the vector. Figure 3.3 shows the average hand.



Figure 3.3: The average hand

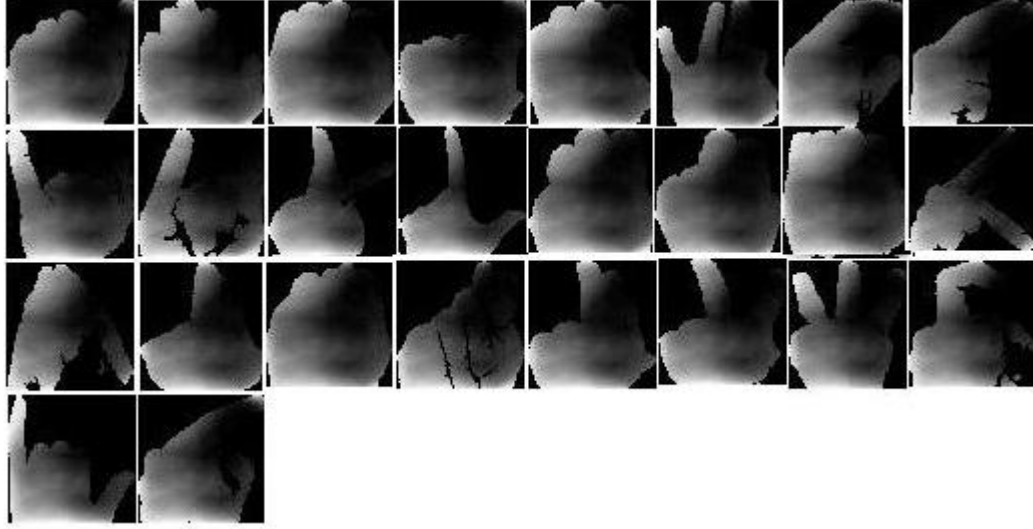


Figure 3.4: The difference of some selected hands and the average hand

3.2.1.2 Calculating the covariance matrix

Covariance matrix in a probability is a matrix whose elements in i, j position is the covariance between i^{th} and j^{th} elements of a random vector, which generalizes the concept of variance to multiple dimensions. The aim of the statistical analysis of covariance is to see, if there is any relationship between dimensions. Hence it helps to find out how much the dimensions vary from the mean with respect to each other.

The exact value of the covariance is not as important as its sign. If the value is positive then it indicates that both dimensions increase together. If the value is negative, then as one dimension increases, the other decreases. If the covariance is zero, it indicates that the two dimensions are independent of each other [13].

After calculating all the possible covariance values between all the different dimensions they are placed in a matrix which is in general presented by equation (3.3) [13]. In this case, it is better defined it as an equation (3.4).

$$C^{n \times n} = (c_{i,j}, c_{i,j} = cov(Dim_i, Dim_j)) \quad (3.3)$$

$$C = \frac{1}{M} \sum_{i=1}^M \Phi_i \Phi_i^T = AA^T \quad (3.4)$$

Where in equation (3.3), $C^{n \times n}$ is a covariance matrix with n rows and n columns, Dim_i is the i^{th} dimension and Dim_j is the j^{th} dimension. So if there is an n -dimensional data set then the matrix has n rows and columns which means it is square and each element of the matrix is the result of calculating the covariance between two separate dimensions [13] and in equation (3.4), $A = [\Phi_1, \Phi_2 \dots \Phi_M]$.

Each image is a matrix and its dimension is very large so calculating the covariance of the matrix is not possible. Hence it needs a new method to find covariance matrix which is:

$$C = A^* A^T \quad (3.5)$$

Where A is the difference between each hand and the average hand by the vector and C is a new covariance matrix.

3.2.1.3 Calculating the eigenvectors and eigenvalues of the covariance matrix

Since the covariance matrix is square, the eigenvectors and eigenvalues calculation are possible. In linear algebra finding the principal components of the distribution of images are equivalent to finding the eigenvectors of the covariance matrix of the hand images (data set). In this case, the eigenvectors is defined as a set of features, which they can differentiate the hand images. So the eigenvectors and eigenvalues of the covariance matrix are the principal component of the data set. In general the number of eigenhands would be equal to the number of hand images in the training set. So for large databases this could mean a lot of processing. Fortunately it is possible to represent hand images by using only the “best” (the ones with the largest eigenvalue) eigenhands. Using a finite number of best eigenhands will account for most of the variance in the set of hand images. In [19], Kirby and Sirovich showed that for a database of 115 images around 40 eigenfaces would be sufficient for a good description of the set of face images. In this work just it is

changed from eigenfaces to eigenhands. The process of generating the eigenhands and some selected eigenhands of 26 letter of ASL are illustrated in Figure 3.5 & 3.6.

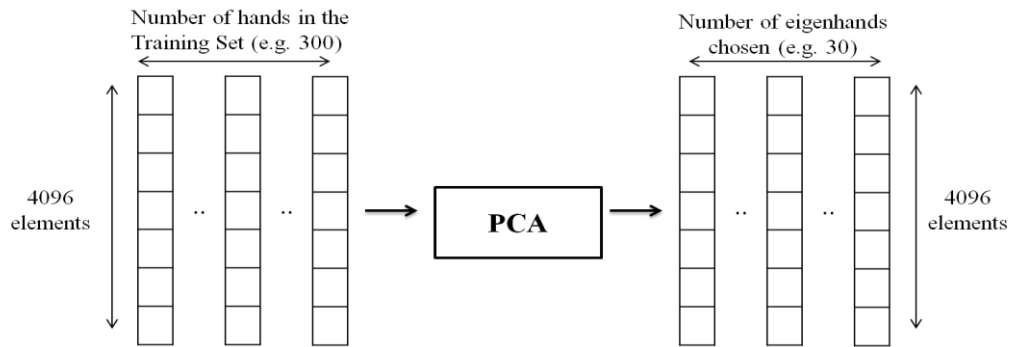


Figure 3.5: Eigenhands generation process [19]

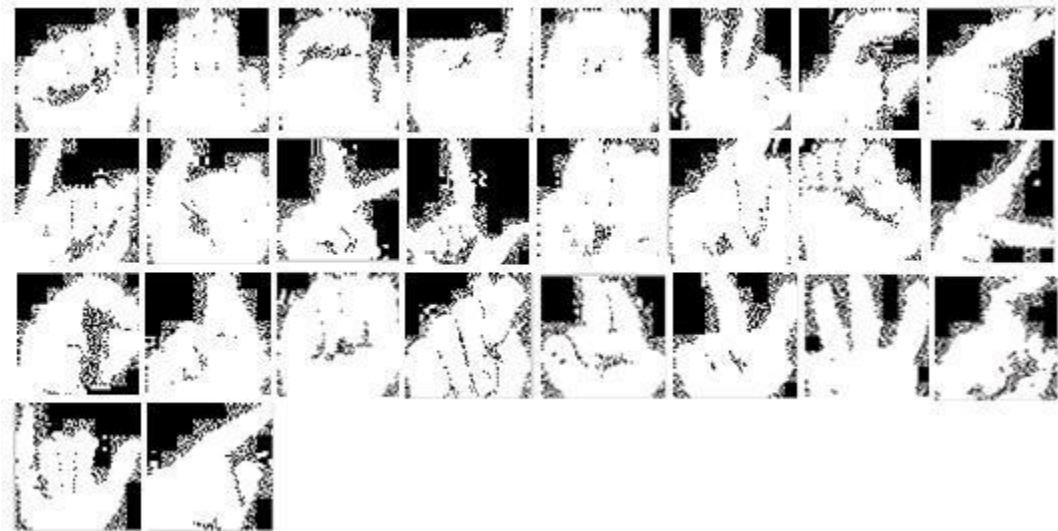


Figure 3.6: Visualization of eigenhands

By storing a small collection of weights in each hand and a small set of standard pictures (the eigenpictures), any collection of hand images can be reconstructed. It is possible to get the weights for each individual hand image by projecting it onto each eigenpicture. Hand images can then be reconstructed by a linear combination of the characteristic features for each hand (Figure 3.7).

As mentioned before in this method just the highest eigenvalue is going to be used and respective eigenvector has the best description of the data distribution. Hence the k^{th} vector u_k , is chosen such that λ_k is a maximum:

$$\lambda_k = \frac{1}{M} \sum_{n=1}^M (u_k^T \Phi_n)^2 \quad (3.6)$$

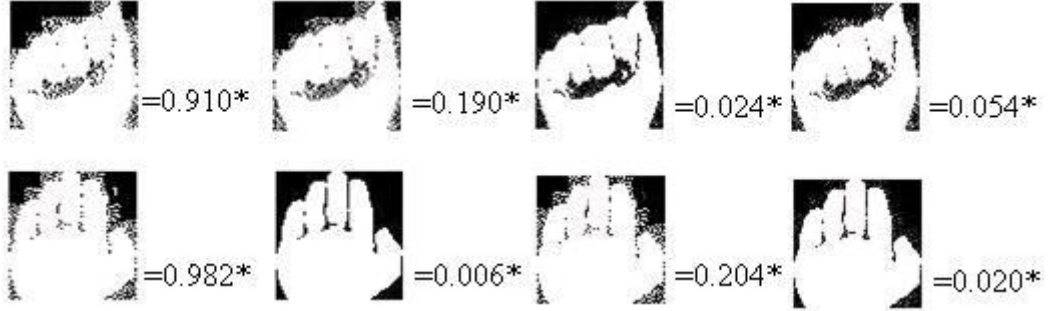


Figure 3.7: Reconstruction of hand in the hand space

The covariance matrix C has dimensions $N^2 \times N^2$ and determining N^2 eigenvectors and eigenvalues. In [17] has showed that if the dimension of space is greater than the number of data points $M \ll N^2$, instead of N^2 significant eigenvectors there will be only $M - 1$ meaningful eigenvectors. The remaining eigenvectors have eigenvalues close or equal to zero. Therefore it is an appropriate way to find a linear combination of the hand images ϕ_i and reducing analysis calculations. By this analysis the $(M \times M)$ matrix $L = A^T A$ is constructed where $L_{mn} = \phi_m^T \phi_n$, and finds the M eigenvectors, V_l of L . The M training set of hand images in the form of eigenhands is determined by these vectors linear combinations [19]:

$$u_l = \sum_{k=1}^M V_{lk} \phi_k \quad l = 1, \dots, M \quad (3.7)$$

3.3 Eigenhand classification of hand images

In [19], Kirby showed that about 40 eigenhands would be sufficient for a good description of the set of hand images. A new hand image (Γ) is transformed into its eigenhands components (projected into hand space) by the following operation:

$$w_k = u_k^T (\Gamma - \psi) \quad k = 1, 2, \dots, M' \quad (3.8)$$

Figure 3.8 shows some sample hand images and their projections onto the hand space where only the most significant M' eigenvectors are used.

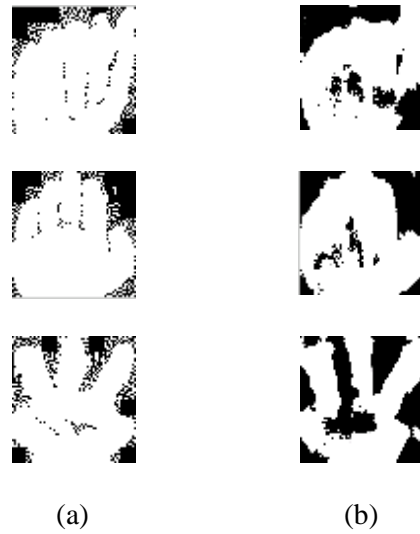


Figure 3.8: Three hand images and their projections onto the hand space
 (a) Sample images (b) projections to hand space.

The input image is represented by the contribution of each eigenhand which treats as a basis by the weights form of a vector $\Omega^T = (w_1, w_2, \dots, w_{M'})$ for hand image. The vector will be used in recognition to find which predefined hand classes are the best matches of the hand. The simplest method for determining which hand class provides the best description of an input class Ω_k is through the use of Euclidean distance and trying to find which hand class minimizes it.

$$\varepsilon_k = \|\Omega - \Omega_k\| \quad (3.9)$$

Here Ω_k is the vector describing the k^{th} hand class.

A hand is belonged to class k when the minimum ε_k is less than some predefined threshold θ_δ , which defines the maximum allowable distance from hand space. Otherwise a new class will be created to classify the unknown hand.

Projecting the images onto a low dimensional space is useful specially for the images are not similar to hand shape since the projection acts as the vector of weights which all images will project onto the pattern vector. The image and the hand space

distance can be computed by the squared distance between hand differ $\varphi = \Gamma - \psi$ and its projection $\varphi_f = \sum_{i=1}^{M'} w_i u_i$, onto the hand space:

$$\varepsilon^2 = \|\varphi - \varphi_f\| \quad (3.10)$$

Hands in the training set, should has a small distance with the hand space, which shows that images are hand. For an input hand image and its projection there are four possibilities as shown in Figure 3.9.

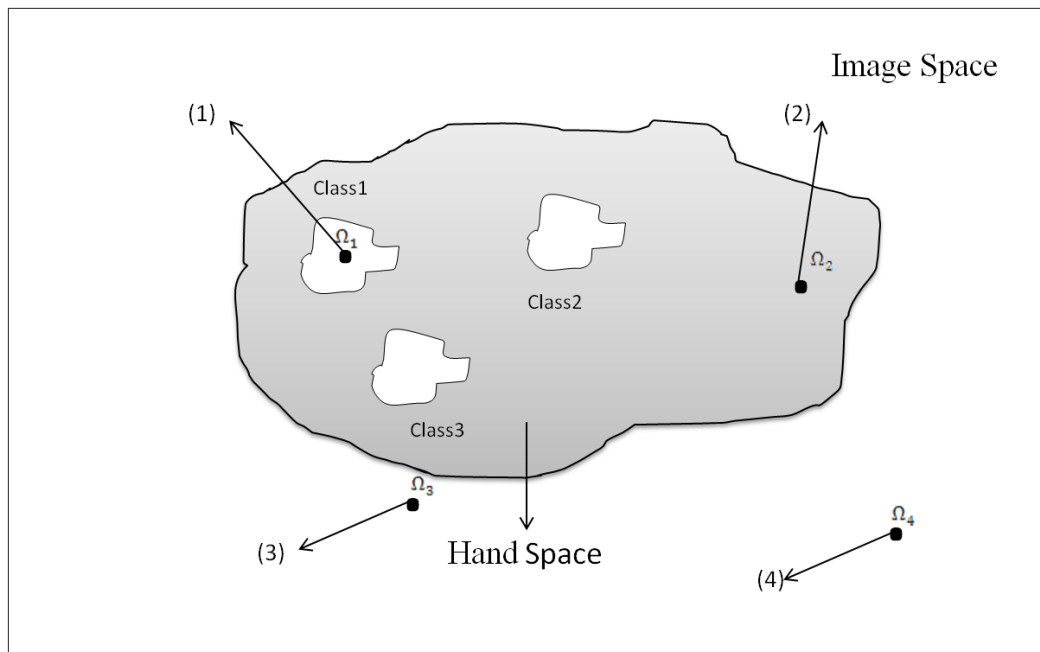


Figure 3.9: Possible projection into hand space [17].

The four possible scenarios are:

- 1) Belongs to a hand space and a predefined hand class.
- 2) Belongs to a hand space but not belongs to a predefined hand class.
- 3) Out of hand space but has a short distance to a predefined hand class.
- 4) Out of hand space but neither does not belong to a predefined hand class nor has a short distance to it.

Chapter 4

DCT COEFFICIENTS BASED CODE ASSIGNMENT

4.1 Introduction

DCT has been introduced first time by Ahmed, Natarajan and Rao [20]. Later Wang in [21] categorized the DCT into eight different transformation where each transform is identified as even or odd type, which the evens are DCT-I, DCT-II, DCT-III and DCT-IV that are used in image processing and signal. Out of four different even DCT that Wang defined, DCT-II was the one suggested by Ahmet (Eq. 4.1).

$$[C_N^{II}] = \sqrt{\frac{2}{N}} \left[\varepsilon_k \cos \frac{\pi(2n+1)k}{2N} \right] \quad n, k = 0, 1, \dots, N-1 \quad (4.1)$$
$$\varepsilon_P = \begin{cases} \frac{1}{\sqrt{2}}, & P = 0 \text{ or } P = N \\ 1, & \text{otherwise} \end{cases}$$

DCT is one of the best methods in image processing and recognition for reducing image information redundancy. Since it just keeps transform coefficients which are essential to conserve the most important features which are particularly transform coding systems for data compression/decompression. In cosine transform, sinusoidal basis function is used. The only difference is that the functions use only cosine functions without sine functions hence, it is not complex. As this type of frequency transform is real, orthogonal, and separable, the algorithms for its computation are computationally efficient.

4.2 The DCT Process

As DCT is a linear transform it can represent a finite sequence of data points in terms of a sum of cosine functions which are oscillating at different frequency [5]. The direct DCT transform apply to the original signal for converting it to the frequency domain. After transforming, the importance of the frequencies that are present in it will be reflected by DCT coefficients. The signal's lowest frequency is related to the very first coefficient and it carries the most representative information from the original signal. The signal's higher frequencies are related to the last coefficient and they represent more detailed of signal which probably have been caused by noise [6]. The coefficients which are between the first and the last coefficients carry various information of the original signal. After extracting DCT coefficients for the training and testing sets, minimum distance will be found by using minimum Euclidean norm.

In image recognition and processing, DCT is used to reduce image information redundancy, because only a few transform coefficients are necessary to preserve the most important features [7].

In this method, to classify a hand, its DCT coefficients are extracted. Next, the distances between the DCT coefficients of test image and the DCT coefficients of all hand images are computed. The minimum distance is probably related to the same sign.

4.2.1 One-dimensional DCT

One dimensional transforms (1-D) can disintegrate every multidimensional transform in the proper directions. A spatial domain waveform will be transformed into its basic frequency components by the one-dimensional DCT-II (1-D DCT) which a set of coefficients will correspond to representing the transformation. The

two-dimensional DCT (2-D DCT) is computed by using 1-D DCT. Figure 4.1 illustrate the 1-D process.

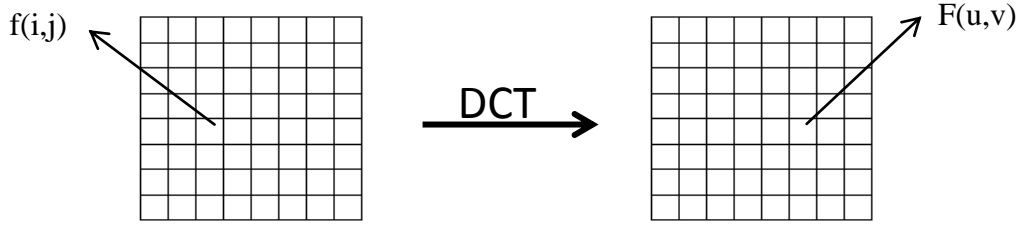


Figure 4.1: 1-D DCT process [7].

1-D DCT is calculated by:

$$F(u) = \alpha(u) \sum_{x=0}^{N-1} f(x) \cdot \cos \left[\frac{\pi(2x+1)u}{2N} \right] \quad (4.2)$$

$$\alpha(0) = \sqrt{\frac{1}{N}} \quad \text{and} \quad \alpha(m) = \sqrt{\frac{2}{N}}$$

The function for the IDCT is given by:

$$f(u) = \sum_{x=0}^{N-1} \alpha(u) F(x) \cdot \cos \left[\frac{\pi(2x+1)u}{2N} \right] \quad (4.3)$$

$$\alpha(0) = \sqrt{\frac{1}{N}} \quad \text{and} \quad \alpha(m) = \sqrt{\frac{2}{N}}$$

4.2.2 Two-dimensional DCT

In the literature various transformation exist that take an input image and transform it into a linear combination of weighted basis functions just like does.

A 2-D DCT can be computed by applying a 1-D DCT to rows and columns of an input image in an iterative manner. The DCT is of an $(N \times M)$ image $f(x,y)$ is defined by:

$$F(u, v) = \frac{2}{\sqrt{MN}} \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) \cos \left[\frac{(2x+1)u\pi}{2N} \right] \cos \left[\frac{(2y+1)v\pi}{2M} \right] \quad (4.4)$$

$$\alpha(n) = \begin{cases} 1, & n = 0 \\ \sqrt{2}, & n > 0 \end{cases}$$

Where n denotes u or v .

The 2-D IDCT is defined by:

$$f(x, y) = \frac{2}{\sqrt{MN}} \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} \alpha(u)\alpha(v)C(u, v) \cos \left[\frac{(2x+1)u\pi}{2N} \right] \cos \left[\frac{(2y+1)v\pi}{2M} \right] \quad (4.5)$$

The two replaceable steps in computing the 2-D DCT are:

- Step1: 1-D DCT is applied vertically.
- Step 2: 1-D DCT is applied horizontally to the result of Step 1.

Given in Figure 4.2 is an illustration of 2-D DCT process.

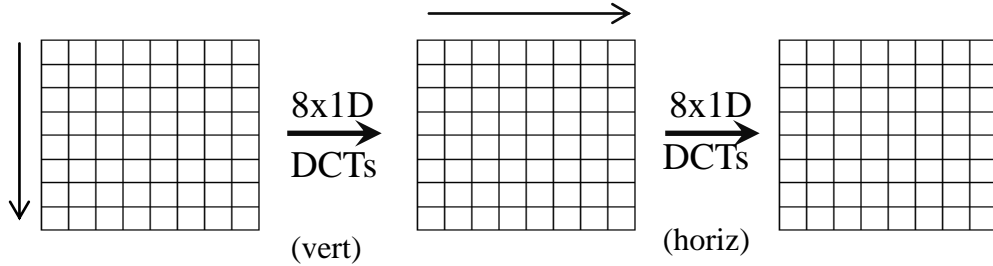


Figure 4.2: 2-D DCT process [7].

An 8×8 sub image or blocks has an optimal of trade-off between compression efficiency and computational difficulty, hence it is applied to the 2-D DCT [22].

While using DCT the input image is transformed into the frequency components. In most images, commonly the signal energy lies at low frequencies which are represented by high amplitude of DCT coefficients and they are located in upper-left corner of the DCT. On the other hand, higher frequencies are presented by the lower-right corner of the DCT with smaller amplitude [23].

4.2.3 Reorganization of DCT Coefficients

For DCT coding, an input image of size 64×64 is first divided into 8×8 blocks, then each sub block is transformed into the DCT domain [24]. The most DCT coefficient magnitudes are very small. For some images, useful and necessary information lay on in the coefficients which are placed in upper-left corner of DCT with a large magnitude. A threshold has set to remove unnecessary and small coefficients which are less than threshold amount [5] [7].

4.2.4 Recognition

After extracting all DCT coefficients of the training and the testing sets, the distance between test image and all training dataset is measured by Euclidean distance. The shortest distance will probably be the match for recognition.

Chapter 5

SINGULAR VALUE DECOMPOSITION COUPLED WITH CIRCULARITY MEASURE

5.1 Introduction

Singular Value Decomposition (SVD) is the factorization of a real or complex matrix which has $m \times n$ size. SVD plays a key computational role throughout linear algebra [25]. If a matrix $A \in \mathbb{R}^{m \times n}$, then there exist two orthogonal matrices which are $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ such that

$$A = U\Sigma V^T = \sum_{i=1}^r \sigma_i u_i v_i^T \quad (5.1)$$

where $\Sigma = \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix}$, $S = \text{diag}(\sigma_1, \dots, \sigma_r) \in \mathbb{R}^{r \times r}$, and $\sigma_1 \geq \dots \geq \sigma_r > 0$. The set of $\{\sigma_1, \dots, \sigma_r\}$ is called the set of singular values (SVs) of matrix A which are unique.

5.2 Significance of SVD

One of the early works in the field of image recognition making use of singular value decomposition was reported by Hong [26]. He conveyed that SVD had good stability and invariance on algebra and geometry and was less sensitive to noise [27]. Also the fact the image matrix could be re-factored and that only the diagonal elements of the S in the Σ matrix is used greatly reduces the set of features one needs to store [28].

5.3 Circularity measure

Circularity: It is defined by ($perimeter^2/area$) but practically it is calculated by (μ_R/σ_R).

$$\mu_R = \frac{1}{N} \sum_n \|(x_n, y_n) - (\bar{x}, \bar{y})\| \quad (5.2)$$

$$\sigma_R = \left(\frac{1}{N} \sum_n [\|(x_n, y_n) - (\bar{x}, \bar{y})\| - \mu_R]^2 \right)^{1/2} \quad (5.3)$$

N is the number of pixels, n iterates over all pixels, (\bar{x}, \bar{y}) is the center of area, (x_n, y_n) is the n^{th} pixel coordinate, and $\| \cdot \|$ denotes the Euclidean distance [13].

5.4 Euclidean distance for selecting the best match

Once the singular values are extracted and the circularity measure is computed for images in the training and testing sets the best match for each one of the test samples is found by computing the Euclidean distance between the acquired features of the test sample and every other image in the entire set and taking the minimum distance among all.

Chapter 6

CUSTOM DATABASE

6.1 Custom Database

6.1.1 Camera and Programming Platform

In this work the experimental platform consist of a Cannon Power shot A1100 IS 12.1 Mega pixels camera, and a personal computer with an Intel(R) Pentium(R) Dual CPU, operating frequency of 2.00 GHz and 1.00 GB of RAM. For the camera orientation there are two options that are commonly considered: the top down view [29] and a frontal view [30] [31].

6.1.2 Lighting

If lighting is carefully controlled, the differentiating skin pixels from background pixels will be easier extracted. In the general case, it is not reasonable to apply the requirement of special lighting equipment. Hence, the system would be tested under normal room conditions with fluorescent lighting.

6.1.3 Background

In order to simplify the hand skin detection process it is required that the background color differs as much as possible from the skin. For this work a blackboard was found to be suitable.

6.1.4 Signers

In this work an ASL fingerspelling recognition system for the 26 letters of English alphabet has been developed. The custom database used in the thesis work has been created by using four different signers. Each person has been asked to sign a total of

6 times for each letter in the alphabet. This will provide 156 images per person and the total number of images for all people will be 624. The original RGB images acquired are all of 640×480 pixels resolution.

To create the four data sets for simulations, the original custom hand dataset containing 624 images has been shuffled and three new data sets with randomly picked images have been generated. Figures 6.1 - 6.4 that follow depict the RGB images for each letter and for each signer in the original data set.

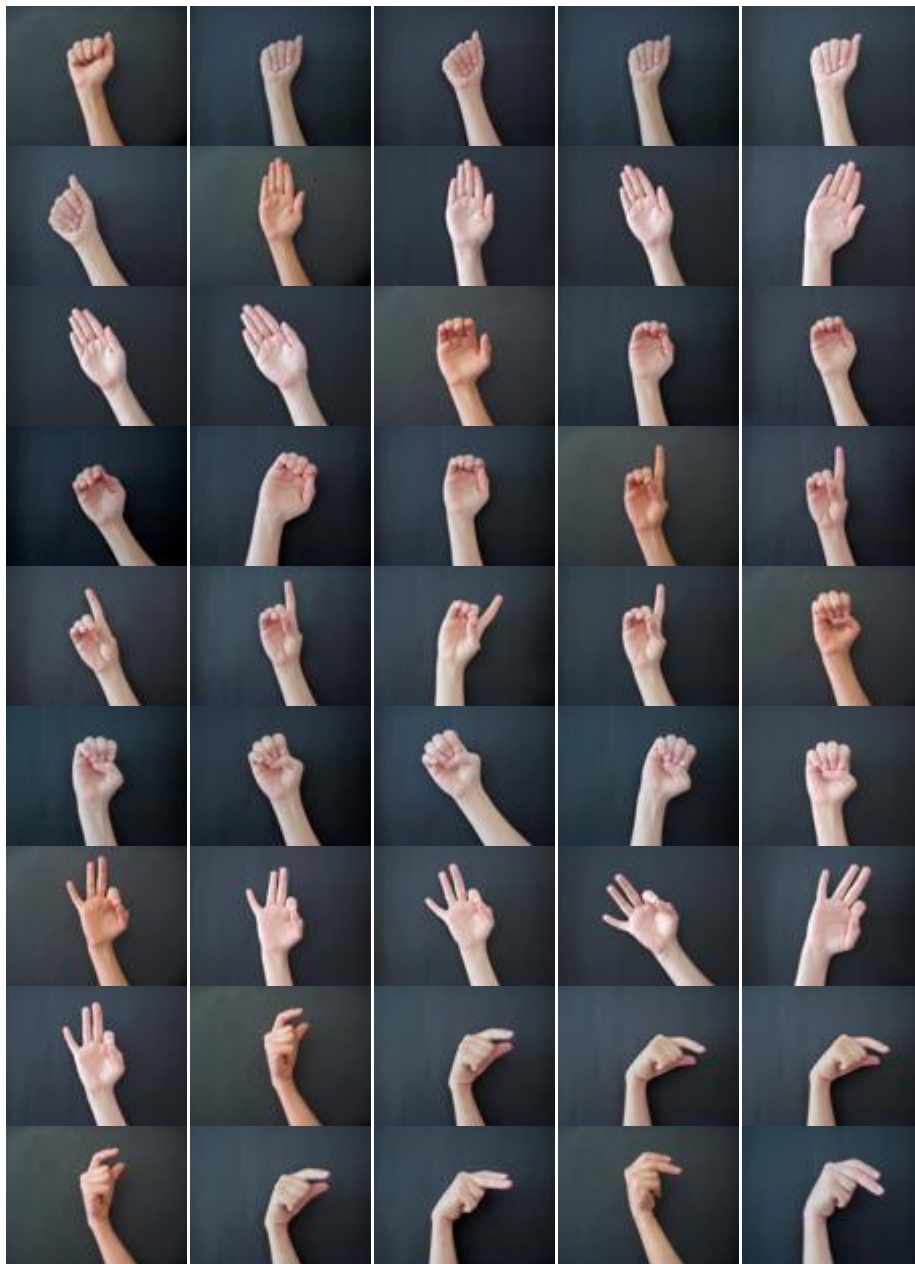
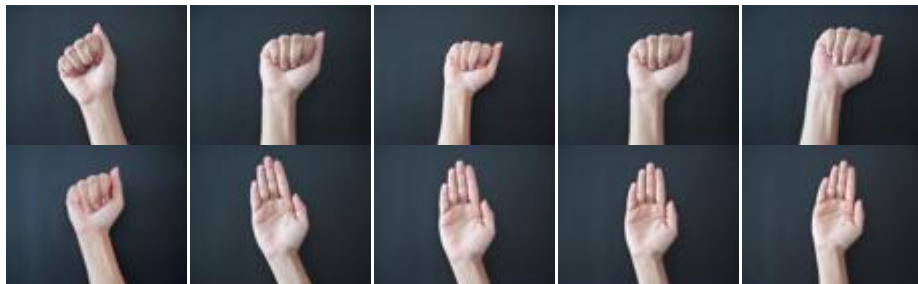






Figure 6.1: Alphabet for signer-1 with 6 copies of each letter.







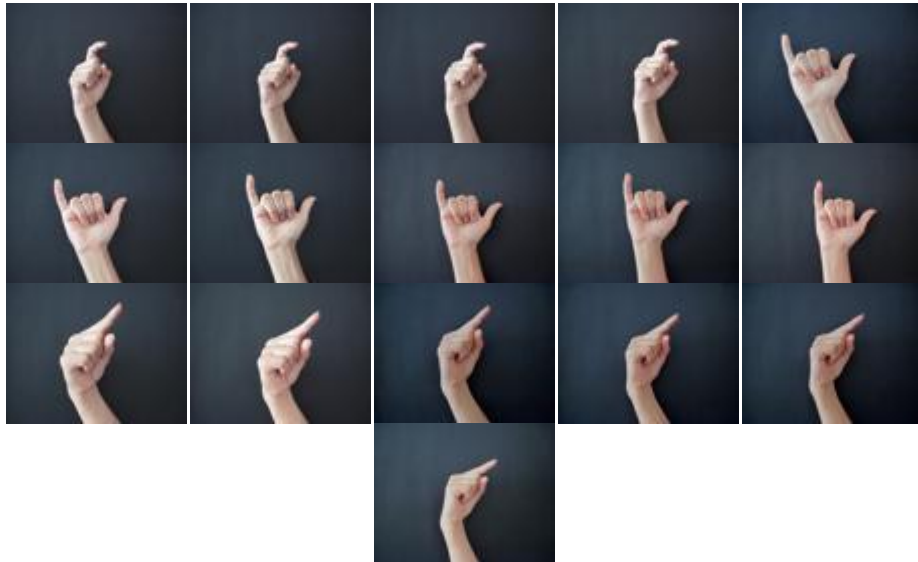


Figure 6.2: Alphabet for signer-2 with 6 copies of each letter.







Figure 6.3: Alphabet for signer-3 with 6 copies of each letter.





Figure 6.4: Alphabet for signer-4 with 6 copies of each letter.

Throughout the simulations the following steps were followed: first the RGB images were processed by a YC_bC_r skin detector and some morphological operations were applied to avoid small components or to fill holes that may have occurred due to variations in illumination. Later the segmented hand images were oriented so that the hands are upright and thirdly the images were cropped from the wrist area. Afterwards feature extraction using one of the four methods described in the earlier chapters was carried out. Finally the rate of recognition for a given set of test images was computed under various training to testing ratios.

Chapter 7

SIMULATION RESULTS

7.1 Simulation approach

In this work, the RGB images acquired each has 640 by 480 pixels resolution. The original custom hand dataset containing 624 images is shuffled to create three new copies of the original data so that the ordering of the images is different in each new set. The four algorithms discussed in chapters 2-5 are applied to each one of these four set of data and overall recognition performance under preselected training/testing scenarios were obtained. The training/testing ratios used are 2:4, 3:3, and 4:2. All simulations have been carried out using the MATLAB programming platform. The following sections will present the simulation results corresponding to each method.

7.1.1 Prominent Features based Recognition Results

To obtain the recognition rates for the signed characters under this method three different training vs. testing scenarios are applied. The first scenario assumes 2:4 training/testing ratios. This implies that for training a total 104 (26×4) images and for testing a total of 52 (26×2) images are used. In the second scenario which adopts 3:3 ratio there are 78 (26×3) training images and 78 (26×3) test images and similarly in the third scenario using 4:2 ratio there are 52 (26×2) training 104 (26×4) testing images. The prominent features based system performance for each scenario has been summarized in Tables 1, 2 and 3.

Table 7.1: Prominent features based recognition results using 4:2 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	104	52	104	52	104	52	104	52
Signer2	104	52	104	52	104	52	104	52
Signer3	104	52	104	52	104	52	104	52
Signer4	104	52	104	52	104	52	104	52
Total	416	208	416	208	416	208	416	208
Recognition Rate (%)	80.77		84.61		80.76		69.23	
Overall Rate (%)	78.84							

Table 7.2: Prominent features based recognition results using 3:3 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	78	78	78	78	78	78	78	78
Signer2	78	78	78	78	78	78	78	78
Signer3	78	78	78	78	78	78	78	78
Signer4	78	78	78	78	78	78	78	78
Total	312	312	312	312	312	312	312	312
Recognition Rate (%)	73.07		73.07		69.23		69.23	
Overall Rate (%)	71.15							

Table 7.3: Prominent features based recognition results using 2:4 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	52	104	52	104	52	104	52	104
Signer2	52	104	52	104	52	104	52	104
Signer3	52	104	52	104	52	104	52	104
Signer4	52	104	52	104	52	104	52	104
Total	208	416	208	416	208	416	208	416
Recognition Rate (%)	53.84		53.84		57.69		53.84	
Overall Rate (%)	54.80							

We note that when the system is trained well enough then the performance attained would be better. However it is also clear from Tables 2 and 3 that the performance is not merely based on the partitioning of the data but the segmentation and orientation also play an important role.

7.1.2 PCA Results

Performance analysis in this section has been carried out using the three different training vs. testing scenarios explained earlier in section 7. 2. 1. Our aim while using PCA was to recognize any image in the test set by using as few eigenvectors as possible. Hence, the simulations were repeated for 1, 5, 10, 15, 20, 25, 30, 40 eigenvectors. Tables 4, 5 and 6 provide recognition rates for images which are resized to 128×128 and Tables 7, 8 and 9 for images which are resized to 256×256 resolution.

Table 7.4: PCA performance using 4:2 training/testing ratios while using 128×128 resized images.

Recognition Rate (%)	Error Rate (%)	Image Size	Number of Eigenvectors
16.34	83.65	128x128	1
55.76	44.23	128x128	5
61.29	38.07	128x128	10
62.50	37.50	128x128	15
63.70	36.29	128x128	20
65.14	34.85	128x128	25
66.10	33.89	128x128	30
66.58	33.41	128x128	40

Table 7.5: PCA performance using 3:3 training/testing ratios while using 128×128 resized images.

Recognition Rate (%)	Error Rate (%)	Image Size	Number of Eigenvectors
14.42	85.57	128x128	1
54.80	45.19	128x128	5
62.98	37.01	128x128	10
65.38	34.61	128x128	15
64.42	35.57	128x128	20
64.90	35.09	128x128	25
65.38	34.61	128x128	30
64.90	35.09	128x128	40

Table 7.6: PCA performance using 2:4 training/testing ratios while using 128×128 resized images.

Recognition Rate (%)	Error Rate (%)	Image Size	Number of Eigenvectors
12.50	87.50	128x128	1
49.67	50.32	128x128	5
56.09	43.91	128x128	10
59.93	40.06	128x128	15
60.57	39.42	128x128	20
60.89	39.10	128x128	25
63.46	36.53	128x128	30
64.10	35.89	128x128	40

Table 7.7: PCA performance using 4:2 training/testing ratios while using 256×256 resized images.

Recognition Rate (%)	Error Rate (%)	Image Size	Number of Eigenvectors
16.58	83.41	256x256	1
58.17	41.82	256x256	5
65.86	34.13	256x256	10
67.78	32.21	256x256	15
69.23	30.76	256x256	20
69.23	30.76	256x256	25
71.39	28.60	256x256	30
72.11	27.88	256x256	40

Table 7.8: PCA performance using 3:3 training/testing ratios while using 256×256 resized images.

Recognition Rate (%)	Error Rate (%)	Image Size	Number of Eigenvectors
15.38	84.61	256x256	1
54.8	45.19	256x256	5
62.98	37.01	256x256	10
64.9	35.09	256x256	15
64.42	35.57	256x256	20
65.38	34.61	256x256	25
64.9	35.09	256x256	30
65.86	34.13	256x256	40

Table 7.9: PCA performance using 2:4 training/testing ratios while using 256×256 resized images.

Recognition Rate (%)	Error Rate (%)	Image Size	Number of Eigenvectors
12.17	87.82	256x256	1
49.67	50.32	256x256	5
56.73	43.26	256x256	10
59.93	40.06	256x256	15
60.57	39.42	256x256	20
60.89	39.10	256x256	25
63.46	36.53	256x256	30
64.10	35.89	256x256	40

A quick look at the results obtained, points out that as the number of eigenvectors used is increased the system performance also improves. However performance would stable out after some time regardless of the number of eigenvectors. In comparison to performance of other methods the PCA results are the poorest. The highest overall recognition rate is 72.11% using 40 eigenvectors. We believe this degradation in performance is due to the fuzzy mean image that results since by nature of hand patterns there is not much correlation between the 26 signs for the alphabet.

7.1.3 Performance Analysis for DCT Coefficients based Code Assignment

Performance analysis in this section has been carried out using the three different training vs. testing scenarios explained earlier in section 7. 2. 1. Tables 10, 11 and 12 provided below summarize the results obtained under each scenario. After applying the DCT to each 8×8 sub-block only the coefficients above a pre-selected threshold has been taken since only a few transform coefficients are essential to conserve the most important features. In this work the threshold value assumed was 0.5.

Table 7.10: DCT coefficients based code assignment results using 4:2 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	104	52	104	52	104	52	104	52
Signer2	104	52	104	52	104	52	104	52
Signer3	104	52	104	52	104	52	104	52
Signer4	104	52	104	52	104	52	104	52
Total	416	208	416	208	416	208	416	208
Recognition Rate (%)	100		96.15		96.15		96.15	
Overall Rate (%)	97.11							

Table 7.11: DCT coefficients based code assignment results using 3:3 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	78	78	78	78	78	78	78	78
Signer2	78	78	78	78	78	78	78	78
Signer3	78	78	78	78	78	78	78	78
Signer4	78	78	78	78	78	78	78	78
Total	312	312	312	312	312	312	312	312
Recognition Rate (%)	76.92		73.07		73.07		76.92	
Overall Rate (%)	75							

Table 7.12: DCT coefficients based code assignment results using 2:4 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	52	104	52	104	52	104	52	104
Signer2	52	104	52	104	52	104	52	104
Signer3	52	104	52	104	52	104	52	104
Signer4	52	104	52	104	52	104	52	104
Total	208	416	208	416	208	416	208	416
Recognition Rate (%)	65.38		53.84		53.84		26.92	
Overall Rate (%)	50							

Clearly the overall recognition rates for DCT based code assignment supersede that of PCA and prominent feature based approaches. However it is clear that the overall performance for DCT based code assignment method will get better as the system is better trained.

7.1.4 Recognition rates for SVD Coupled with circularity

While performing the simulations with SVD + circularity based method, first we have applied the SVD to the global images and in the second round the SVD was applied to 8×8 sub-blocks of each image. For each sub-block the entire set of singular values were obtained and they were concatenated to create a longer vector. The performance of the SVD coupled with circularity method was obtained for each training/testing scenario using either the entire image (global) or sub-blocks of each image (local) and the performance of the SVD without circularity method was

obtained for 4:2 training/testing scenario and the results attained are summarized in Tables 13, 14, 15, 16, 17, 18, 19 and 20.

Table 7.13: Recognition rates for SVD Coupled with Circularity method applied to global images under 4:2 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	104	52	104	52	104	52	104	52
Signer2	104	52	104	52	104	52	104	52
Signer3	104	52	104	52	104	52	104	52
Signer4	104	52	104	52	104	52	104	52
Total	416	208	416	208	416	208	416	208
Recognition Rate (%)	92.3		96.15		88.46		92.3	
Overall Rate (%)	92.3							

Table 7.14: Recognition rates for SVD method applied to global images under 4:2 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	104	52	104	52	104	52	104	52
Signer2	104	52	104	52	104	52	104	52
Signer3	104	52	104	52	104	52	104	52
Signer4	104	52	104	52	104	52	104	52
Total	416	208	416	208	416	208	416	208
Recognition Rate (%)	89.6		92.3		88.46		92.3	
Overall Rate (%)	90.49							

Table 7.15: Recognition rates for SVD Coupled with Circularity method applied to global images under 3:3 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	78	78	78	78	78	78	78	78
Signer2	78	78	78	78	78	78	78	78
Signer3	78	78	78	78	78	78	78	78
Signer4	78	78	78	78	78	78	78	78
Total	312	312	312	312	312	312	312	312
Recognition Rate (%)	61.56		69.23		53.84		53.84	
Overall Rate (%)	59.61							

Table 7.16: Recognition rates for SVD Coupled with Circularity method applied to global images under 2:4 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	52	104	52	104	52	104	52	104
Signer2	52	104	52	104	52	104	52	104
Signer3	52	104	52	104	52	104	52	104
Signer4	52	104	52	104	52	104	52	104
Total	208	416	208	416	208	416	208	416
Recognition Rate (%)	69.23		65.38		46.15		53.84	
Overall Rate (%)	52.84							

Table 7.17: Recognition rates for SVD Coupled with Circularity method applied to 8×8 sub-blocks under 4:2 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	104	52	104	52	104	52	104	52
Signer2	104	52	104	52	104	52	104	52
Signer3	104	52	104	52	104	52	104	52
Signer4	104	52	104	52	104	52	104	52
Total	416	208	416	208	416	208	416	208
Recognition Rate (%)	100		100		100		100	
Overall Rate (%)	100							

Table 7.18: Recognition rates for SVD method applied to 8×8 sub-blocks under 4:2 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	104	52	104	52	104	52	104	52
Signer2	104	52	104	52	104	52	104	52
Signer3	104	52	104	52	104	52	104	52
Signer4	104	52	104	52	104	52	104	52
Total	416	208	416	208	416	208	416	208
Recognition Rate (%)	100		100		100		100	
Overall Rate (%)	100							

Table 7.19: Recognition rates for SVD Coupled with Circularity method applied to 8×8 sub-blocks under 3:3 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	78	78	78	78	78	78	78	78
Signer2	78	78	78	78	78	78	78	78
Signer3	78	78	78	78	78	78	78	78
Signer4	78	78	78	78	78	78	78	78
Total	312	312	312	312	312	312	312	312
Recognition Rate (%)	69.23		76.92		73.07		69.23	
Overall Rate (%)	72.11							

Table 7.20: Recognition rates for SVD Coupled with Circularity method applied to 8×8 sub-blocks under 2:4 training/testing ratio.

	Set1		Set2		Set3		Set4	
	Train	Test	Train	Test	Train	Test	Train	Test
Signer1	52	104	52	104	52	104	52	104
Signer2	52	104	52	104	52	104	52	104
Signer3	52	104	52	104	52	104	52	104
Signer4	52	104	52	104	52	104	52	104
Total	208	416	208	416	208	416	208	416
Recognition Rate (%)	65.38		50		65.38		53.84	
Overall Rate (%)	58.65							

As mentioned in the Chapter 5, in the SVD, σ is an $(M \times N)$ matrix which is zero everywhere except in the main diagonal. This mainly results in enormous dimensionality reduction of an input pattern from matrix $(M \times N)$ to a vector of N components [25]. A quick look at the results indicate that the SVD coupled with circularity and without circularity applied to sub-blocks give the best overall recognition rate among all methods for 4:2 training/testing. In fact the recognition rates for block based SVD (in both case) for each on the four sets of data is 100%. For the case when SVD coupled with circularity and without circularity are applied to full images the overall recognition rate was 92.3% and 90.04. The results in the tables point out that when SVD is applied to 8×8 sub-blocks whatever the training to testing ratio the overall recognition rates would be higher than that of SVD applied globally.

Chapter 8

CONCLUSION AND FUTURE WORK

The focus of this thesis was American Sign Language (ASL) based one-hand fingerspelling recognition under four main methods: namely the prominent features based recognition, principal component analysis (PCA), discrete cosine transform (DCT) coefficients based code assignment and singular value decomposition (SVD) coupled with circularity.

To carry out the simulations an ASL fingerspelling recognition system was developed for the 26 letters of English alphabet. The custom database was generated by using four signers where each person had been asked to sign a total of six times for each letter. Hence 156 images were acquired for the twenty six letters of the alphabet per person and the total number of images for all signers was 624. The resolution for the individual images was 640 by 480. During simulations the custom hand-dataset and three randomly shuffled versions of this original set were obtained and simulations under various partitioning scenarios were carried out to obtain the system's overall recognition rates.

To each one of the four data sets, 416:208, 312:312, and 208:416 partitioning (training vs. testing) was applied. After applying each method to all datasets the overall recognition rates were as follows: the prominent features based recognition had 78.84%, PCA had 72.11%, DCT coefficients based code assignment had 97.11%, SVD coupled with circularity and without circularity applied to the entire image (globally) had 92.3% and 90.04% and SVD coupled with circularity and

without circularity applied to (8×8) sub-blocks of the image (locally) had 100% and 100% correct detection rate. The results indicate that when DCT and/or SVD were applied locally (to sub-blocks instead of the global image) the fingerspelling system provides high recognition performances in both cases. In fact SVD using all the singular values had a better performance than the DCT using only the most important coefficients. In the condition that complexity was not concerned SVD would give the highest overall recognition rate whereas if reduction in complexity was a must DCT was the best contender. The third best results were obtained while using the prominent features based approach. In contrast, PCA had the poorest result which was about 75.96%. Since hand images are not correlated the mean hand would be quite dispersed and this fuzzy mean image would cause a drop in the PCA's overall recognition performance.

8.1 Future Work

Some of the ideas for future work include the following:

We would like to investigate how to split a sequence of characters into individual words that can be pronounced by a text-to-speech processor. In this regard one can employ the mutual information (MI) which majors the coherence of adjacent characters and compares them to a previously selected threshold [32].

- Since the success rate for correct recognition is heavily based on segmenting the skin parts belonging to signed characters and this is a hard task under changing illumination conditions, it is worth trying to do the segmentation using near infrared lighting instead of visible or fluorescent like illuminators. This idea was suggested by A. Moralles in [33].

REFERENCES

- [1] S. Hollar, K. S. J. Pister, "Wireless Static Hand Gesture Recognition with Accelerometers-The Acceleration Sensing Glove", in *Technical Report University of California*, Berkeley, June 2000.
- [2] M. Lamar, M. Bhuiyant, "Hand Alphabet Recognition Using Morphological PCA and Neural Networks", in *Proceedings of International Joint Conference on Neural Networks*, Washington, USA, pp. 2839-2844, 1999.
- [3] R. Lindeman, N. Kyriakopoulos, J. Rebollar, "A Multi-class Pattern Recognition System for Practical Fingerspelling Translation", in *International Conference on Multimodal Interfaces*, Pittsburgh, USA, pp. 200-204, 2004.
- [4] S. Foo, J. Isaacs, "Hand Pose Estimation for American Sign Language Recognition", in *Thirty-Sixth Southeastern Symposium on IEEE System Theory*, pp. 132-136, 2004.
- [5] J. K. Poel, L.V. Batista, D. Omaia, "2D-DCT Distance Based Face Recognition Using a Reduced Number of Coefficients", *Computer Graphics and Image Processing, 2009 XXII Brazilian Symposium on IEEE*, pp. 291-298, October 2009.
- [6] R. Woods, R. C. Gonzalez, *Digital Image Processing Using Matlab*, 3rd ed.: Prentice Hall, 2007.

- [7] C. Yu, L. Jinghong, X. Xinhe, Z. Lihong, "Face Recognition Based on Discrete Cosine and Support Vector Machine", *Neural Networks and Brain. International Conference on IEEE*, pp. 1248-1252, 2005.
- [8] L. Cao, "Singular Value Decomposition Applied to Digital Image Processing", in *Division of Computing Studies, Arizona State University, Polytechnic Campus Mesa, Arizona*, 2008.
- [9] S. Albayrak, A. Ekinici, B. Bükün, O. Altun, "Turkish Fingerspelling Recognition System Using Axis of Least Inertia Based Fast Alignment", *Lecture Notes in Computer Science*, vol. 4304, pp. 473-481, 2006.
- [10] N. K. Ngan, D. Chai, "Locating Facial Region of a Head-and-Shoulders Color Image," in *Third IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, Japan, pp. 124-129, April 1998.
- [11] F. A. Merchant, K. R. Castleman, Q. Wu, *Microscope Image Processing*, in *Academic Press*, April 2008.
- [12] K. Pearson, "On Lines and Planes of Closest Fit to Systems of Points in Space", in *Philosophical Magazine*, pp. 559-572, 1901.
- [13] I. S. Lindsay, "A tutorial on Principal Component Analysis", February 2002.
- [14] P. E. Spector, *Summated Rating Scale Construction: An Introduction*, in *Newbury Park, California Sage*, 1992.

- [15] L. J. Williams, H. Abdi, "Principal Component Analysis", in *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, pp. 433-459, 2010.
- [16] P. J. A. Shaw, "Multivariate Statistics for the Environmental Sciences", in *Hodder-Arnold, International Standard Book Number-10: 0340807636*, 2003.
- [17] A. Pentland, M. Turk, "Eigenfaces for Recognition", *Journal of Cognitive Neuroscience*, vol. 3, pp. 71-86, March 1991.
- [18] J. Shlens, "A Tutorial on Principal Component Analysis", in *Center for Neural Science and Systems Neurobiology Laboratory, Salk Institute for Biological Studies La Jolla*, version 3.01, vol. 3, New York, April 2009.
- [19] L. Sriovich, M. Kirbly, "Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces", *IEEE Transaction PAMI*, vol. 12, pp. 103-108, 1990.
- [20] T. Natarajan, K. Rao, N. Ahmed, "Discrete Cosine Transform", *IEEE Transaction on Computers*, vol. 1, pp. 90-93, 1974.
- [21] Z. Wang, "Fast Algorithms for the Discrete W Transform and for the Discrete Fourier Transform", *IEEE Transaction Acoustics, Speech and Signal Processing*, vol. 32, pp. 803-816, 1984.
- [22] L. A. Rowe, S. R. Smoot, "Study of DCT Coefficient Distributions", in *Proceedings of the SPIE Symposium on Electronic Imaging*, p. 2657, 1996.

- [23] C. T. Chen, "Transform Coding of Digital Images using Variable Block Size DCT with Adaptive Thresholding and Quantization", in *SPIE*, pp. 43-54, 1990.
- [24] W. Gao, Y. K. Chan, D. zhao, "Morphological Representation of DCT Coefficients for Image Compression", *IEEE Transactions On Circuits And Systems For Video Technology*, vol. 12, September 2002.
- [25] S. Subbaraman, B. G. Patil, "SVD-EBP Algorithm for Iris Patten Recognition", *Computer Vision and Pattern Recognition*, vol. 2, no. 12, December 2011.
- [26] Z. Q. Hong, "Algebraic Feature Extraction of Image Recognition", in *Pattern Recognition*, pp. 211-219, 1991.
- [27] G. Ji, H. Zheng, S. Shian, "Improved Approach for Cell Image Recognition based on Fractal Coding and Fractal Singular Value Neighbor Distance", *Knowledge Acquisition and Modeling Workshop, 2008. KAM Workshop. IEEE International Symposium on*, pp. 432-435, 2008.
- [28] L. Cao, "Singular Value Decomposition Applied To Digital Image Processing", 2005.
- [29] A. Pentland, T. Starner, "Real-time American Sign Language Recognition from Video using Hidden Markov Models", in *Systems and Applications*, pp. 5-7, 1995.
- [30] M. Assan, K. Grobel, "Video based Sign Language Recognition using Hidden Markov Models", in *Proceedings of Gesture Workshop*, pp. 97-109, 1997.

- [31] G. Lee, R. Owens, E. J. Holden, "Automatic recognition of colloquial Australian Sign Language", in *Proceedings of IEEE Workshop on Motion and Video Computing* , vol.2, pp. 183-188, 2005.
- [32] B. Yin, Y. Sun, Y. Cheng, "Research on Word Segmentation for Chinese Sign Language", vol. 20, pp. 407-413, 2006.
- [33] M. A. Ferrer, C. M. Travieso, J. B. Alonso, A. Morales, "Comparing Infrared and Visible Illumination for Contactless hand based Biometric Scheme", pp. 191-197, 2008.