M. K. SKAKIR

HUMAN IDENTIFICATION AND VERIFICATION BY HAND GEOMETRY INFORMATION

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY

MUSTAFA KANAAN SHAKIR

A MASTER OF SCIENCE THESIS
IN
COMPUTER ENGINEERING

JANUARY 2022

ATILIM UNIVERSITY 2022

HUMAN IDENTIFICATION AND VERIFICATION BY HAND GEOMETRY INFORMATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY

BY

MUSTAFA KANAAN SHAKIR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JANUARY 2022

Approval of the Graduate School of Natural and Applied Sciences, Atilim University.

_____

Prof. Dr. Ender KESKINKILIÇ
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science in Computer Engineering, Atilim University.

_____

Assoc. Prof. Dr. Gökhan ŞENGÜL
Head of Department

This is to certify that we have read the thesis **"HUMAN IDENTIFICATION AND VERIFICATION BY HAND GEOMETRY INFORMATION"** Submitted **By Mustafa Kanaan Shakir Mustafa** and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science

_____

Assoc. Prof. Dr. Gökhan ŞENGÜL
Supervisor

Assoc. Prof. Dr. Gökhan ŞENGÜL
Computer Engineering Department, Atilim University, _____

Asst. Prof. Dr. Güzin TİRKEŞ
Computer Engineering Department, Atilim University, _____

Assoc. Prof. Dr. Erol ÖZÇELİK
Department of Psychology, Çankaya University, _____

Date: *21-01-2022*

I hereby declare that all information in this document has been obtained and presented following academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

MUSTAFA KANAAN SHAKIR

Signature: _____

ABSTRACT

HUMAN IDENTIFICATION AND VERIFICATION BY HAND GEOMETRY INFORMATION

SHAKIR, MUSTAFA KANAAN

M.S. Computer Engineering

Supervisor: Assoc. Prof. Dr. Gökhan ŞENGÜL

January 2022, 71 pages

In this thesis, a hand geometry-based human identification system is proposed. The hand is a vital component of the human body. It consists of many unique features that can be used for human identification and verification systems. This study presents an approach for recognizing the human using the features extracted from hand images. The proposed method is implemented in three steps, namely preprocessing, feature extraction and classification phases. In the preprocessing step, the hand images are resized for the feature extraction model. In the feature extraction phase, the convolutional neural network (AlexNet model) is used in order to extract hand features. The extracted features are classified using the well*known Support Vector Machines (SVM) and k-nearest-neighbourhood classifiers. The proposed method is tested on the CASIA-MS-Palmprint dataset, using a different number of training and testing images. in used for the hand geometry-based recognition system. The average accuracy, sensitivity, and specificity were 94.36, 89.96 and 90.36. We conclude that the recognition accuracy rate is reasonable when the system is trained with an adequate number of images.

Keywords: Hand geometry, Deep learning, Deep neural network, Verification and identification, AlexNet.

ÖZ

EL GEOMETRİ BİLGİLERİ İLE İNSAN TANIMLAMA VE DOĞRULAMA

SHAKIR, MUSTAFA KANAAN

Yüksek Lisans, Bilgisayar Mühendisliği

Danışman: Assoc. Prof. Dr. Gökhan ŞENGÜL

Ocak 2022, 71 sayfa

Bu tezde, el geometrisi tabanlı bir insan tanımlama sistemi önerilmiştir. El, insan vücudunun hayati bir bileşenidir. İnsan tanımlama ve doğrulama sistemleri için kullanılabilecek birçok benzersiz özellikten oluşur. Bu çalışma, el görüntülerinden çıkarılan öznitelikleri kullanarak insanı tanımaya yönelik bir yaklaşım sunmaktadır. Önerilen yöntem, ön işleme, özellik çıkarma ve sınıflandırma aşamaları olmak üzere üç aşamada gerçekleştirilmiştir. Ön işleme adımında, özellik çıkarım modeli için el görüntüleri yeniden boyutlandırılır. Öznitelik çıkarma aşamasında, el özniteliklerini çıkarmak için evrişimli sinir ağı (AlexNet modeli) kullanılır. Çıkarılan özellikler, iyi bilinen Destek Vektör Makineleri (SVM) ve k-en yakın komşu sınıflandırıcıları kullanılarak sınıflandırılır. Önerilen yöntem, farklı sayıda eğitim ve test görüntüsü kullanılarak CASIA-MS-Palmprint veri kümesi üzerinde test edilmiştir. el geometrisi tabanlı tanıma sistemi için kullanılır. Ortalama doğruluk, duyarlılık ve özgüllük 94.36, 89.96 ve 90.36 idi. Sistem yeterli sayıda görüntü ile eğitildiğinde tanıma doğruluğu oranının makul olduğu sonucuna varıyoruz.

Anahtar Kelimeler: El geometrisi, Derin öğrenme, Derin sinir ağı, Doğrulama ve tanımlama, AlexNet.

ACKNOWLEDGMENTS

Throughout the process of writing this thesis, I was given a lot of help and encouragement.

I would first like to thank my advisor, Assoc. Prof. Dr Gökhan ŞENGÜL, whose expertise wasinvaluable in formulating the research questions and methodology. Your insightful feedback pushed me to sharpen my thinking and brought my work to a higher level.

Furthermore, I would like to thank the jury member for their wise counsel and sympathetic ear.

Finally, without the help of my family and my friends, I would not have been able tofinish my thesis.

To the wellspring of love, altruism and generosity, my great mother….

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF SYMBOLS/ABBREVIATIONS

PIN	Personal identification number
FAR	Rate of rejection
FAA	Rate of acceptance
ROC	Receive operating characteristic
CB	Cancelable biometrics
KL	Karhunen loeve
CNN	Convolutional neural networks
DET	Detection error tradeoff
GAR	Genuine accept rate
EER	Equal error rate
HMMs	Hidden markov models
GMM	Gaussian mixture model
ICA	Independent component analysis
PCA	Pincipal component analysis
LDA	Linear discriminant analysis
LPP	Locality preserving projections
VGG	Visual geometry group
SVM	Support vector machines
RBF	Radial based function
K-NN	K nearest neighbour algorithm
ECOC	Error-correcting output codes
ROC	Receiver Operating Characteristics
HMM	Hidden Markov Model
RNN	Recurrent Neural Network

CHAPTER 1

INTRODUCTION

1.1 Introduction

Recognition based on biometric data has been more popular in recent years, and various research projects are underway to develop safer and more precise recognition algorithms. Biometric systems are commonly utilized to secure personal identity and verification details. As the number of safety breaks and transaction cheating rises, identity and verification solutions become increasingly crucial. Biometrics is employed by the federal, state, and local governments and the military and private sectors. These technologies are also used in secure electronic banking, government IDs, retail transactions, health and social services. Network, data protection, remote access to resources, workstations, transactions, and web security are authentication-based applications. Electronic commerce in biometrics is necessary for the global economy's healthy growth. The ability to trust these computerized transactions is critical to the global economy's continued growth. Biometrics are combined with other technologies such as bank cards, code keys, and authentication. Biometrics stands as a technology that we use regularly. For authentication and identification of a person, biometric methods are more accurate and convenient.

For security, there are three primary forms of authentication. They are something you know (a PIN, password, or details of a person's information), something you have (a bank card, Magnetic card, or particular ID card), and something you are (biometric information). Biometric systems, for example, provide a safe and convenient authentication tool that cannot be stolen, borrowed, or forgotten, and faking biometric information is extremely difficult.

To identify personal identity, traditional access control methods use various ways of identification, such as an identity document, magnetic card, password, or personal identification number (PIN) code. Unauthorized use of these tools for profit can result

in significant financial and moral losses [1]. Moreover, with these tools, access to an area can be blocked, but it cannot be controlled by who accesses it. In addition, the use of traditional tools does not provide sufficient confidence in applications where personal identification is essential, such as national security, electronic commerce and access to computer networks. Typically, to control access to applications, passwords (knowledge-build protection) and ID smart cards (token-build protection) are commonly employed [1].

1.2 A Biometric System

Biometrics has attracted investigators' attention because of its rapidly rising applications in various regions. It has been dramatically as authentic and reliable as possible as protection is concerned. An individual investigation is an essential topic in the field of security. The biometric method is a pattern recognition system that uses particular physiological or behavioural traits to identify a person. Biometric systems are divided into two categories: unimodal and multimodal. Unimodal biometric systems (UBS) are a method for ensuring data verification by managing individuals' unique characteristic sequences. However, the performance of unimodal biometric systems is restricted because to their sensitivity to spoofing efforts, non-universality, significant intra-user variability, and noise in sensed data. Multimodal biometric systems overcome the constraints of unimodal biometric systems by allowing diverse biometric sources to compensate for one another's intrinsic limitations. A multimodal biometric system, on the other hand, utilizes two or more biometric traits for recognition. A unimodal biometric system, on the other hand, uses just one biometric trait for verification or identification. Multimodal biometric systems are more noise robust and less susceptible to spoofing. The main problem in the uni-modal systems is intra-class variations and noisy data [2], which can be resolved by expanding multimodal biometric systems. In the latent palm-print, hand geometry and its shape, finger structure, fingerprint, Finger-Knuckle-Print, palm-vein is integrated along various modalities to elaborate multimodal biometric systems [3], [4]. This was done for multiple applications of separate identification. The following are the two modalities in which biometric systems can be used.

- Verification (1-1): Identify the issue of verifying or rejecting his stated identity ("Am I who I say I am?").
- Identification(1-n): Ascertain the claimed identity ("Who am I?") by choosing based on the templates in the database.

A biometric-based identity system has two modes of operation: enrollment and authentication. Each one is described in full further down.

(1) **Enrollment mode:** In this mode, using a biometric reader, user biometric is taken. The data will be stored in a database and labelled with the person's distinctiveness (such as recognition name, number, etc.).

(2) **Authentication mode (verifying mode or identifying mode):** If the system is in identification mode, it confirms a user's identity by comparing the biometric data entered with the biometric template of that individual stored in the system's database, or it ensures a user's identity by matching it with one of the templates in the database (verification mode), as shown in the diagram below.

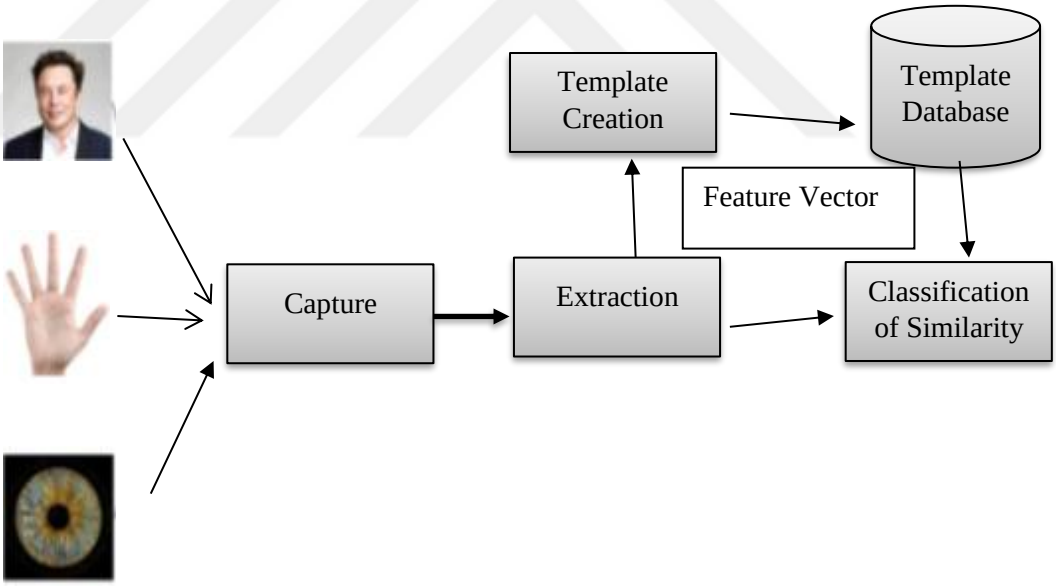


Figure 1.1: General biometric approach.

A biometric system is made up of four major components:

(1) scaner module: This is where the biometric data is received. A fingerprint sensor, for example, captures the fingerprint structure.
(2) Feature extraction module: It takes the information which is treated to get characteristics values. Eg. The direction and place of minutiae points in the fingerprint image.
(3) Match module: The data obtained by the feature extraction module will compare to the data of the registered models to determine how similar the two biometric datasets are by the match module.
(4) Ruling module: Based on the matching module outcome, the identity of the user is allowed or asserted specification is established or forbidden.

The following are some of the measures used to evaluate the accuracy of a biometric system:

- FAR: the percentage of identification cases of unauthorized persons are wrongly accepted in a certain number.
- FRR: the percentage of times authorised people are wrongly refused during identification.

FAR and FRR are two terms that are used interchangeably. These phrases are essential for anyone evaluating or comparing the biometric security systems performance. The values of FAR and FRR have a direct impact on biometric verification accuracy. The accuracy of biometric verification will be improved if the FAR and/or FRR values are reduced

FRR and FAR of a biometric system may be estimated by announcing them at various verges. Both the elements are taken concurrently in a ROC curve. At different thresholds, the ROC is used to plot FRR against FAR. The false acceptance and rejected rates are calculated by creating probable, fake and original matching outcomes. A threshold is determined for acceptance or rejection. When characteristic vectors equivalent to the two equal individuals are compared, then the original matching outcome is needed. Similarly, when two characteristic

vectors from distinct individuals are compared, then the fake matching outcome is needed. Usually, a threshold is used for system decisions. The acceptance threshold determines FRR and FAR, and their tradeoff curve which is (ROC) curve.

1.3 Hand Geometry

Security, user acceptability, cost, performance, and other considerations all differ across biometric techniques. One of the physiological features for recognition is hand geometry, which is based on the fact that every human hand is unique. In the case of multiple entry control applications, fingerprint and iris can not be preferable due to the individual's acceptance. These applications can be border command and dormitory meal plan entry, immigration, and particularly biometrics. In this case, the biometric measure which can only distinguish between individuals is sufficient. Recognition is not necessary. Hence, in this case, the use of a hand geometry data biometrics system was helpful.

Hand geometry recognition is considered low-medium secure, but it has several advantages over other biometrics, including ease of use (with almost no user rejection), low cost (since only an average resolution camera is required and no specialized sensors are required), and low computational cost (allowing for faster results). Many advantages are associated with hand geometry-based authentication. It is a rapid and relatively simple sensing technique. Also, it is not required high-intensity image optics. For fingerprint imaging systems, individuals worthy of frictional skin is required, while for the iris or retina-based identification systems, a particular illumination setup is required. These high-end designs are unnecessary for the hand-geometry system, and they can be integrated with other biometrics for specification. For instance, frequent confirmation can be done on the hand geometry and occasional proof on the fingerprint system. Also other advantages like:

- Ease of acquisition and strong verification performance
- Acceptable for applications requiring medium to low levels of safety.
- Integration-related relief.
- The readers are long-lasting and can handle a high number of users for several years without experiencing reader failure.

- Works in high-stress situations.
- Low template size reduces storage requirements.

1.4 Deep Neural Network

Deep learning is one of the most important aspects of data science, which also includes statistics and forecasting models. A deep neural network is a network that has more than two layers and has a certain level of complexity. Deep neural networks use advanced mathematical modeling to evaluate data in complex ways. Many experts define deep neural networks as networks with an input layer, an output layer, and at least one hidden layer in the middle. In a process known as "feature hierarchy," each layer conducts sure sorting and ordering. Dealing with unlabeled or unstructured input is one of the essential applications of potent neural networks. Deep neural networks are also referred to as "deep learning" networks, as deep learning is a type of machine learning in which technologies based on artificial intelligence attempt to classify and order data in ways that go beyond simple input/output protocols. In general, a neural network is a system that simulates human brain activity, precisely pattern recognition and the flow of data through several levels of simulated neural connections. It is very beneficial to data scientists tasked with gathering, investigating and interpreting high amounts of data. At its ease, deep learning can be believed to as a way to automate predictive analytics. In this section, promising deep learning architectures are reviewed in detail. They are commonly used in the computer vision community.

1.4.1 Convolutional Neural Networks (CNN)

1 CNN architectures are accessible and extensively used in deep learning. The mammalian optic cortex inspires the idea. And he was frequently used in computer vision tasks. Fukushima proposed this idea for the first time in his seminal “Neocognitron”[5]. It was based on the human visual system model suggested by Nobel laureates Hubel and Wiesel [6]. Later with a group of researcher-developed efficiently learned model weights for a CNN architecture based on the backpropagation[7].

2 CNN has three hidden layers viz. Convolutional layers with activation function (Relu), Nonlinear layers and Pooling layers. Lastly, fully connected layers with activation function Softmax, used for classification. A quick kernel is used in the convolutional layers to take out the images' characteristics. The nonlinear layers are used in the element-wise fashion on the traits to get the nonlinear task modelling by the network. The vicinity characteristic map is taken and interchanged in the pooling layers in its statistical data. In CNN, each node is locally joined. They accept input from the adjacent layer. It is also called an open-minded field. In the CNN, a quick kernel weighs the dividing mechanism. The kernel weights are separated across the whole image; CNNs have a notably less number of variables than a similar wholly joined neural network. Also, by assembling various convolution layers, the higher-level layers learn the characteristics from increasingly broad receptive fields. CNN's have been applied to various computer vision tasks such as semantic segmentation [8], medical image segmentation [9], object detection [10], super-resolution , [11], image enhancement [12], caption generation for image and videos [13], plus a lot more. AlexNet is one of the most well-known CNN architectures; ZF Net [14]; VGG Net [15]; ResNet [16]; GoogLenet [17]; MobileNet [18]; and Dense Net [19].

1.5 Motivation

Automatic human identification has become a significant concern in today's information and network-based society. Biometrics are strategies for automatically identifying a person based on physical or behavioural features. Biometric systems are already used in industries requiring user verification (e.g., access control or welfare disbursement programs). Fingerprints, face, voice, iris, and hand geometry are just a few of the identifying characteristics that have been utilized for personal identification. Fingerprint and iris patterns are widely acknowledged as uniquely identifying each member of a considerable population, making them appropriate for large-scale labels (identifying a subject's identity). However, due to privacy or resource constraints, we need to authenticate a person (confirm or deny the person's stated identity). In these instances, we can use qualities with less discriminating capacity, such as voice or hand form, due to hand ease of use, non-intrusiveness, and widespread acceptability. Deep

learning algorithms are used in a variety of identification and recognition systems. Then, the use of new deep learning-based techniques leads to accurate systems. Factors like distance, backdrop, illumination, and viewpoint hampered previous hand-biometric-based personal identification systems significantly. Therefore various research was carried out to overcome the limitations such as the environmental variety of workstations. [20], [21]. However, they focused only on the palm print region structure and not on the whole hand [22]. Subsequently, the computer vision field has been modernized with the deep learning renaissance [23], [24], mounting the latest high-tech image handling and examination. Deep learning was used in facial identification for biometrics [25], [26] . Many researchers reveal that machines are more powerful than humans at accepting individuals faces [27], [28]. Convolutional neural networks (CNN) were also used successfully in fingerprinting recognization [29], [30] and posture recognition [31]. In the last decade, precisely in 2015, the CNN was stronger for image operating work and can be merely deceived using an adversarial charge [32] . Additional palmprints and fingerprints can readily be added to an existing hand shape-based biometric system.

To overcome all factors that we mentioned biometric traits in fingerprint geometry, palm print geometry, previously studied handprint geometry, the new technique is proposed in this thesis. Hand build biometrics is considered in this thesis due to its advantages of minimum cost, low calculation complication, minimum template size. The study aims to use deep learning to develop a new technique for increasing the performance of hand geometry-based biometric devices..

1.6 Objective

The work focuses on utilising biometric-based authentication using hand geometry features in a biometric system, which uses a person's hand images to help verify that person. The following are the most important factors to consider:

- To assure the success of the design of our hand geometry-based biometric system.
- It is centred on developing a solution that improves the accuracy and speed of the person recognition procedure.
- Reduces the cost of the biometric system while also increasing user acceptability.

Each of the stated ways is explained in the study, and then each algorithm is implemented. This will demonstrate which algorithms are the most effective in solving recognizing challenges.

1.7 Thesis Outline

Chapter 2: shows a general review of the progress and state-of-the-art hand geometry biometrics.

Chapter 3: presents the methodology of hand recognition geometry using a deep learning network.

Chapter 4: This chapter provides the result of evaluating, comparing, and analysing the hand geometry system using a deep learning network.

Chapter 5: This chapter comes close by pointing forth a possible area for further research.

CHAPTER 2

LITERATURE REVIEW

In this chapter, a review of a biometric system and its execution was explained in detail. It also includes the performance of the most significant hand traits i.e. palm print, fingerprints and hand shape.

2.1 Introduction

This section reviews the research on hand geometry and other biometric features that researchers have done. A range of biometric methods is currently used to accomplish human identity or user verification. In recent years, the modern application area of Hand geometry recognition systems has received significant interest in pattern recognition research and study. A hand, geometry recognition system uses photographs of people's hands to identify them. It necessitates database pictures to use the tested photos and match them with training images. The test images must be evaluated to see if they reach some trained ideas. Feature extraction is preferred while performing a Hand geometry recognition system. The geometric component of human hand picture characteristics is extracted via feature extraction.

Consequently, it has become an essential aspect of pre-processing and geometric normalization. These features can then be used to find more photos with similar characteristics. A brief review of some prior research of various methodologies utilized for overcoming the challenges and problems in Hand geometry recognition systems, with varying accuracy rates, is presented in this chapter.

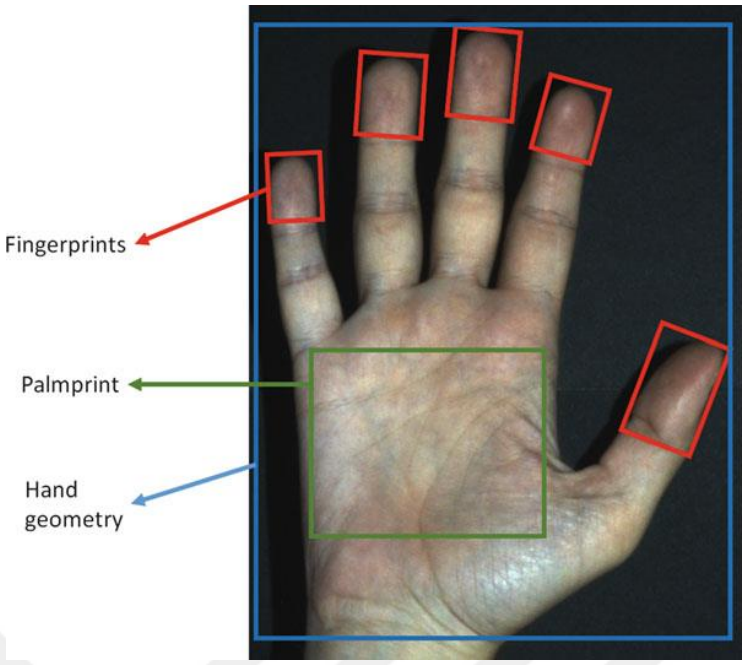


Figure 2.1 Human hand with its biometric traits [33]

2.2 Biometrics Are Now in Use In A Variety Of Settings

Nowadays, various biometrics are being used. These are as follows

1. Fingerprints
2. Face
3. Iris,
4. Palmprint
5. Hand geometry

Each one of these, we will study in detail. We focused more on hand geometry out of all the above in our study.

2.2.1 Fingerprint

The fingerprint contains valleys and ridge patterns that are used for the identification of individuals. This method has many features of an ideal recognition system. Those criteria are uniqueness, universality, accuracy, permanence, and low cost. It is a more reliable and popular biometric technology [34] . As per the archaeological records, Assyrian and Chinese ancient civilizations had this kind of technology in the 7000 to 6000 BC [35]. The minutiae characteristics were in the fingerprint matching by Henry

Foaled in 1880 and generated the scientific base for modern fingerprint identification [35].

The new era fingerprint identification systems can be divided into four groups viz. minutiae-build, connection-based ridge characteristic-based and incline based fingerprint identification [36], [37]. A minutiae points system is used in the considerable automatic fingerprint recognition [36]. Each finger minutiae pattern was relatively noisy, unique, and distorted during fingerprint acquisition, leading to spurious details and missing numbers [38]. A ridge feature-based technique can be applied to overcome this poor standard fingerprint image problem. It was the design of lines on a fingertip and used ridge features, such as ridges frequency and orientation, ridge texture and shape. However, the extended characteristic-created techniques bear their minimum distinction ability [35]. Two fingerprint pictures were superimposed for the correlation-based approaches. The correlation among them was calculated at the intensity level for different alignments. These methods are most sensitive to further finger pressure and alignment, non-linear deformation skin conditions, etc. [36], [39], [40]. The various strategies employed in the fingerprint detection technique are depicted in Table 2.1.

Table 2.1 Various Strategies Employed In The Fingerprint Detection Technique

Sr. No.	**Authors**	**Discovery**
1.	S. Chikkerur, S. Pankanti, A. Jea, N. Ratha, and R. Bolle [38]	Used confine texture characteristic of minutiae
2.	Z. Ouyang, J. Feng, F. Su, and A. Cai [41]	Utilized quality connection matching
3.	G. Aggarwal, N. K. Ratha, T.-Y. Jea, and R. M. Bolle [37]	The gradient-based approach captures textural data by dividing every minor neighbourhood into various native areas. Place gradient histograms are generated to characterize textural data around every little site.

4.	Z. A. Jhat, A. H. Mir, and S. Rubab [42]	a texture characteristic of The energy contained in a fingerprint might be utilized to verify it.

2.2.2 Face

Face recognition is a very user-friendly, non-intrusion and popular biometric technique [43]. Few reports are published regarding the existing human face recognition techniques [43], [44]. Till now, various algorithms were suggested for facial recognition. Appearance-based and geometric feature-based algorithms are the two primary groups of these algorithms. (Table 2.2).

Table 2.2 Classification of various face recognition algorithms

Appearance-based methods	**Geometry feature-based methods**
Eigenfaces [45]	Active Shape Mode [46], [47]
Fisherfaces [48]	Analysis of Local Characteristics [49]
Neural Networks [50]	Matching Elastic Bunch Graphs [51]
General Discriminant Analyses [52]	
Kernel Principal Component Analysis [53]	
Analysis of Independent Components [53]	
Support-Vector-Machine [54]	
Kernel-Fisher-Discriminant-Analysis (KFDA) [55]	

When the beginning has previously been watched under similar circumstances, recognising a face under specific lighting and attitude may be consistently achieved, which was an intrinsic limitation of appearance-based techniques. Furthermore, in appearance-based algorithms, the obtained characteristics are global aspects of the face photos, making facial occlusion challenging to manage. Changes in lighting and perspectives are resistant to geometric characteristics-based approaches, but they are

vulnerable to the feature extraction procedure. The geometric relationships between local facial features are examined using the geometry feature-based technique.

Methods based on geometrical features use to recognize faces from 2D or still images. This is very difficult due to illumination, change in pose and expression, etc. This will generate significant statistical differences, and the face identity will get hampered. To overcome this problem, a 3D face identification system is adopted. This can control the localization of features, illumination problems, pose and expression stability. The method was invented by Cartoux et al. [56]. The 3D face information gives better face formation, optimum illumination and strong face recognition. Achermann et al. [57] combine the Hidden Markov Models (HMMs) to 3D face confirmation. Mccool et al. [58] proposed the Gaussian Mixture Model (GMM) parts-based approach for 3D face confirmation. The 3D face identification has disadvantages, such as decreased ease-of-use for laser sensors, minimum sufficiently powerful algorithms, high cost, and minimum accuracy.

2.2.3 Iris

The iris is between the cornea and the lens, which is a thin circular diaphragm. Flom and Ara [59][59] proposed the notion of automatic iris recognition. The idea was implemented by Daugman [59], [60] as a computerised iris recognition system. Bowyer et al. [61] published advances in iris recognition technologies. Table 2.3 further details technological advancements.

Table 2.3 Milestone development in the iris recognition technologies

Sr. No.	**Authors**	**Discovery**
1.	S. R. Ganorkar and A. A. Ghatol [62]	Circular Hough transform-based automatic segmentation algorithm
2.	W. W. Boles [63]	Iris characteristics using a 1-D wavelet transform

3.	C. Sanchez-Avila et al. [64]	Progressed the iris presentation technique
4.	S. Lim, K. Lee, O. Byeon, and T. J. E. j. Kim [65]	The iris characteristic using 2-D Haar wavelet transform
5.	G. Park and S. J. S. Kim [66]	Progression filter banks were used to remove the normalized progression energy as a characteristic.
6.	A. Kumar, D. C. Wong, H. C. Shen, and A. K. Jain [67]	Use of connection filters
7.	L. Ma et al. [68]	Two iris identification algorithms were proposed, based on multi-channel Gabor filters and circular symmetric filters.
8.	L. Ma et al. [69]	An enhanced method characterises essential local dissimilarity with a specific collection of wavelets, recording a position sequence of local sharp dissimilarity points.
9.	Y. Chen, S. C. Dass, and A. K. Jain [70]	Daugman's 2-D Gabor filter with a quality rating was used to quantify enhancement.
10.	Y. Du et al. [71]	Patterns of 1-D local texture
11.	T. Sunet al. [72]	Iris blob matching based on the present moment

2.2.4 Palmprint

The palmprint is the print area between the fingers and wrist of the hand. The palmprint characteristics are minutia points, remarkable points, wrinkles, ridges, and main lines [73]. The palmprint confirmation systems have two types, *i.e.* minimum resolution and high resolution. The types are based on the image quality, *i.e.* low-quality or high-quality images. In generating minimum resolution palmprint, wrinkles, main lines, and palm texture are utilized. While in the high-resolution points, the palm's ridges and minutia are used. Palmprint verification techniques are broadly divided into four categories. Those are line-based [74], [75]; (2) texture-based [76]; (3) orientation based [76]; and aspect-based [77], [78]. Milestone development in palm recognition technologies is represented in Table 2.4.

Table 2.4 Milestone development in the palm recognition technologies

Sr. No.	**Authors**	**Discovery**
1.	C.-C. Han et al. [79]	To extract linelike characteristics from palmprint photos, use Sobel and morphological procedures.
2.	A. Kumar et al. etitive[76], [80]	Competitive Code was defined as the palm line orientation information for palmprint verification and their methodology.
3.	L. Zhang et al. [81]	Total wavelet extension and progression conditions modelling method to take out principal lines-like characteristics
4.	C.-L. Lin et al. [82]	Applied the nature of a hierarchy decay mechanism to take out principal palmprint characteristics, which involve progression and multi-resolution decay
5.	T. Wu et al. [83]	Palmprint orientation code is another way based on orientation (POC)

Wu et al. [84], [85]and Liu et al. [75]considered the palm line as a roof edge and studied it as a first-order derivative (zero-cross points of lines) and second derivative

(amplitude) [75], [84]. 2-D Gabor filter along with plam texture were the main approaches in some theories [76]. Subsequently, Kong et al. [76] developed the FusionCode consisting of feature layer fusion rules [76].

Nowadays, the most promising method is the orientation code because it has the maximum discriminative power and is more robust for illumination change. Various approaches such as independent component analysis (ICA), PCA, linear discriminant analysis (LDA) and locality preserving projections (LPP) were used for the palmprint verification. [77], [78], [84], [86].

2.2.5 Hand Geometry

Hands are a vital component of the human body. It is essential for many operations related to the surrounding world and everyday work. Besides, it consists of many unique features that enable distinctive among separate personalities [33]. In the last few decades, the hand and its part have been used for biometric detection purposes, yielding ultra-modern Biometrics approaches; however, these are limited in their feasibility.

The width and lengths of fingers and the width of the palm make up the geometry of the human hand. It is a comparatively uncomplicated technique that utilizes minimum intention images. It gives good efficiency [87], [88]. Many articles were published based on the hand-structure confirmation systems [87], [89].

The hand structural characteristics are more or less unique and include many geometrical measures such as area, lengths, perimeters and widths of the fingers, palm and hand. Jain et al. [90] suggested geometrical characteristics of the hand that are not solely sufficient for discrimination. Thus, there is a need for alternative characteristic methods such as global hand formation, texture or appearance. Further, He described how he employed 16 preset axes with the help of five pegs. Sanchez-Reillo et al. [91] used the same geometric characteristic set. This includes four fingers (widths), three fingers (length) and a palm.

Along with these three parameters, inter finger baselines and fingertip regions were also included by Wong and Shi [89]. Bulatov et al. [92] designed a peg-free system

that consists of 30 measurements geometrical. In addition to this radii of engraving circles of the fingers was also included. Likewise, the contour-based approach was studied by Jain and Duta [93].

Accordingly, various techniques have been proposed to obtain and mathematically represent the hand feature [91]. Independent component analysis (ICA) and Hausdorff space of hand outline are used to get the most accurate results [94]. However, still, we required the most trusted, unique, detailed based, proper advanced identification system. In the present study, we aim to evaluate a novel technique for Taking use of the newest breakthroughs in deep learning, increasing the execution of biometric systems that are based on hands geometry.

2.3 History Of Hand Geometry Authentication

2.3.1 Hand Geometry systems Until 1900

In the Chauvet cave (France), 29000 BC, prehistoric paintings are discovered, consisting of hand stencils alongside animal paintings. These were considered as an artist unique signature. Here, the hand geometry was utilized for the identification progress [95]. Later on, in 1858, The author hypothesized that the handprint is necessary for project confirmation. The idea of this kind of practice was that handprint was considered as a more trusted thing than the only signature. Herschel used this system for a long time and eventually resolute that the single fingerprint was also sufficient for human identification [96].

2.3.2 From the 1960s until the 1980s, the origins of hand geometry

Robert Miller (from Jersey) worked on the apparel project for the US Army in the 1960s. He discovered that the measurements and structure of the hands might be utilized to identify individuals. As a result, the first automated equipment based on hand structure was created. The equipment and application were patented in 1971 by the Stanford Research Institute. When a user places its hand in this device, each finger length is measured by the four sprung rods. The corresponding hand figure in punched holes was generated as a user ID [97]. In the same year (1971), Richard Ernst has submitted a patent for a hand identification device and a unique design for an ID card.

However, the first marketable hand scanning device was based on the Robert Miller patent. The machine uses light sensors in the grooves for human finger length measurement. The magnetic stripe cards and verification card pattern was matched in this device after the reading hand by the light-sensing cell (1000 W). In this way, the user was verified. This system was effectively used in applications such as authentication of entry in the nuclear power plants (USA), nuclear weaponry security, employee attendance, data access to specific individuals, Wall Street investment companies, etc. The second generation of these sensors measures the length and breadth of all the fingers on one hand. In 1987, manufacture of this gadget came to a stop [97].

David Sidlauskas was the founder of the recognition system company in 1986. It has got lots of importance in daily life applications. Where a hand identification system was used for biometric authentication, this company is involved in the modern hand recognition scanners development and its sales.

He introduced the first-hand reading scanner in the mid-1980s, which was based on three measurements: an optical measuring plate, a top camera, and a side camera. The ID code was a numeric keypad that allowed the person to be identified. A handkey ID was the first commercially viable biometric scanner. This was a three-dimensional contraption [90], [95].

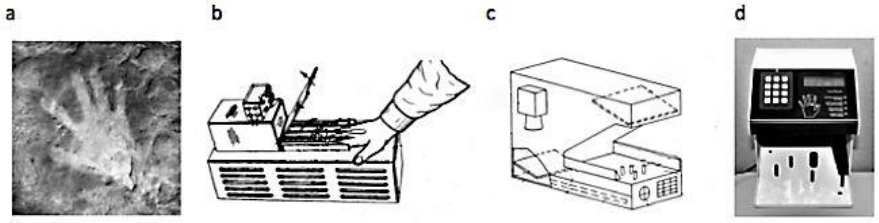


Figure 2.4 a) Chauvet cave has a prehistoric handprint. [98]; b) The mechanical hand reader is seen in this drawing.; c) Sidlauskas patent drawing, and d) Handkey ID3D scanner

2.3.3 Hand Geometry systems In The 1990s

Biomet Partners invented the geometry scanner using two fingers, namely the index and middle fingers, in 1992. It was known as Digi-2 in 1995. It was equipped with a CCD camera capable of capturing a full 3D image of the user. Hand geometry was first used in scientific institutes in 1998. "Vital Signs of Identity" and "Performance Evaluation of Biometric Identification Devices" are the only two research articles on this topic that have been published. A few patents were also filed [99]. After 1998, various new research articles were published every year and showed exponential growth of this field [100].

2.3.4 History Of The Use Of Hand Geometry Systems

- Identity was used at nuclear power facilities in the United States to track attendance and admission time for Wall Street personnel.
- US airports used the INSPASS named program, which used hand geometry recognition. Frequent travellers use it for faster clearance in airport checkups. The system and information scanned the travellers' hands matched the stored data. Based on the match, an ID was generated for the user.
- The Basel - Israel project for autonomous border control administration uses face recognition and hand geometry (Israel - Palestine). Fingerprint scans were no longer required for physical labourers [101].

- Hand geometry scanner was also used in voting in the Colombian legislature [96]

CHAPTER 3

METHODOLOGY

3.1 Dataset

The CASIA Multi-Spectral Palmprint Image Database V1.0 provided the data for this study As illustrated in figure 3.1, the datasets, including 7200 photos of human hands, were collected from 100 different persons using self-designed multiple spectrum imaging equipment.

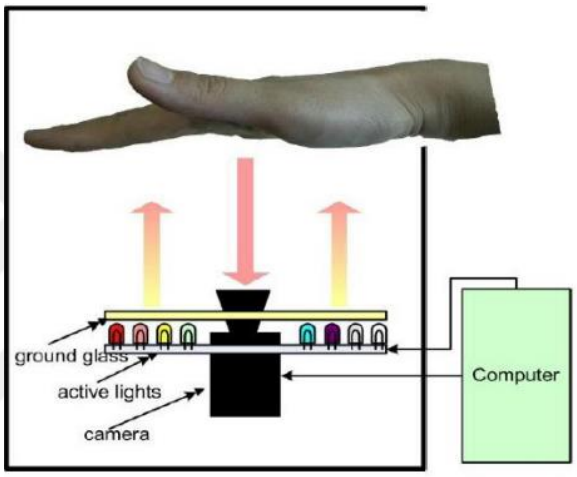


Figure 3.1 self-developed multi-spectral imaging device

They photograph palms in two sessions. Between the two sessions, there is a month's worth of time. There are three examples in each session. Six palm photos are recorded simultaneously, with six different electromagnetic spectrums in each sample. The illuminator's wavelengths correspond to the six spectrums: 460nm, 630nm, 700nm, 850nm, 940nm, and white light. They allow for a degree of hand posture variance between two samples. We want to broaden the range of intra-class models and emulate real-world applications. Figure 3.2 shows six typical palmprint pictures from the database.

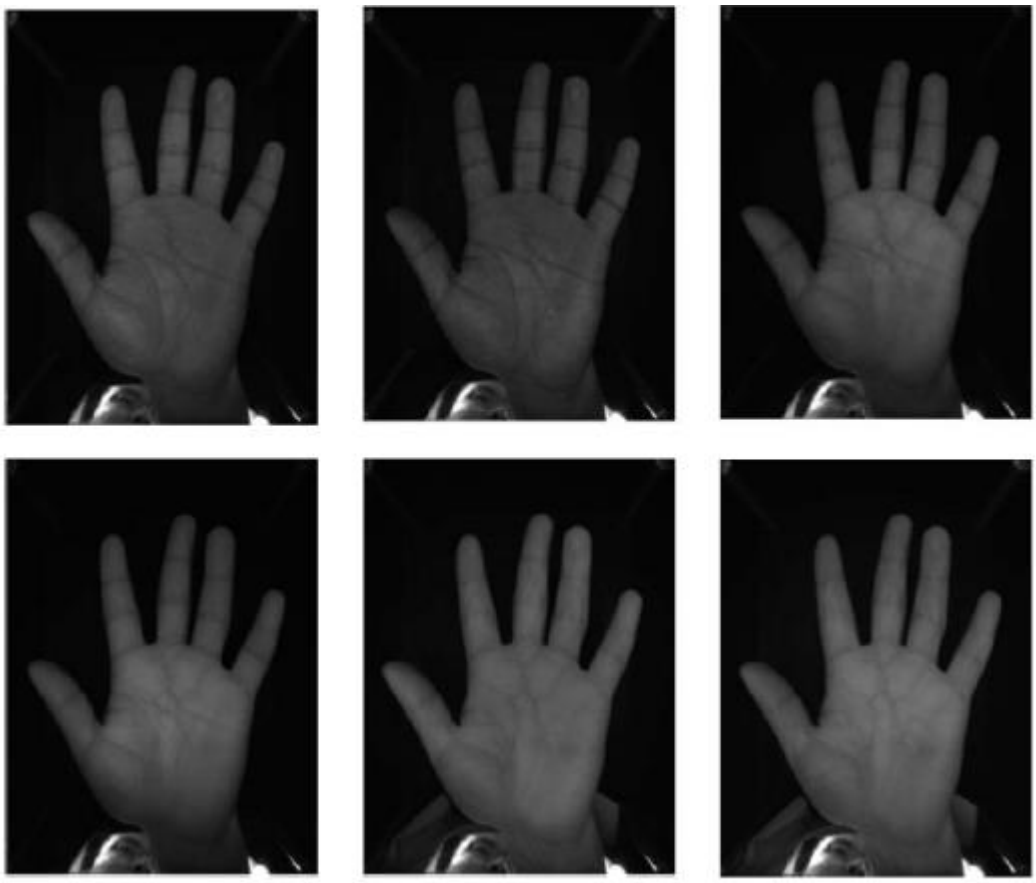


Figure 3.2: Six typical palmprint images in the database

The gadget offers uniformly distributed light and captures hand pictures using a CCD camera attached to the device's bottom.

3.2 Convolutional Neural Network (CNN)

SVM [30], HMM [102], CNN [103], RNN [104], and other machine-learning techniques have all been used to train classification models in recent yearsBecause of its capacity to extract the needed feature values from the input picture and teach the non-similarity between various samples using a more significant number of samples in its training, CNN is the most common in identification and has the best outcomes compared to others approaches. Therefore, in the previous, the rate of speed of hardware computing has limited its progress. The calculation rate of graphics processing units has increased in recent years due to developments in semiconductor fabrication. The bottleneck of hardware processing speed has been identified, allowing the CNN network to expand into a deep CNN network quickly.

AlexNet was the most well-known [24], an object recognition network that won the ImageNet [2017] championship in 2012. Moreover, we discovered if something or noise in the background was similar to the skin colour in the hand gesture identification steps (recognition plus identification), intervention could occur quickly, resulting in the observation of the incorrect region of interest, so background choice must be restricted to a small block to avoid trouble. The backdrop [105] was, for example, it is restricted to a small desktop area (employing a webcam facing down to the desktop). As a result, this research aims to overcome this constraint by focusing on detecting static hand movements. At the same time, the ROI of moving objects may be tracked during system construction, and recognition can be performed instantaneously. After accomplishing this goal, the identifying procedure was no longer limited to a small region. As a result, between the two processes of observation and identification, this study adds a tracking mechanism to eliminate issues caused by items in different backgrounds with the same skin colour and track hands that may be moving simultaneously.

Figure 3.5 depicts a “schematic model” of the suggested hand gesture identification concept, which integrates three main elements: “hand observation, hand tracking, and hand identification. For the hand observation part, we do skin segmentation on the input image to remove unnecessary background data”, then treat the noise to reduce minor damage in select photographs. Finally, they calculated the ROI of the hand position using the background subtraction approach [105], [106]. They employ the “kernelized correlation filters (KCF)” [107] method as the basis for calculation in the hand tracking part, which has been a widely used approach for image tracking in recent years. The main idea was to train a model by removing the ROI properties of the target position in the first frame. After that, when the next frame arrived, the trained model was used to compute a new anticipated position. They employed a” deep CNN network” to extract and identify the ROI's hand features in hand recognition. For comparison, this work used two deep CNN architectures based on AlexNet [24]and visual geometry group (VGG)Net [15]. When the model training is done, the test set's identification rate might reach more than 95%.

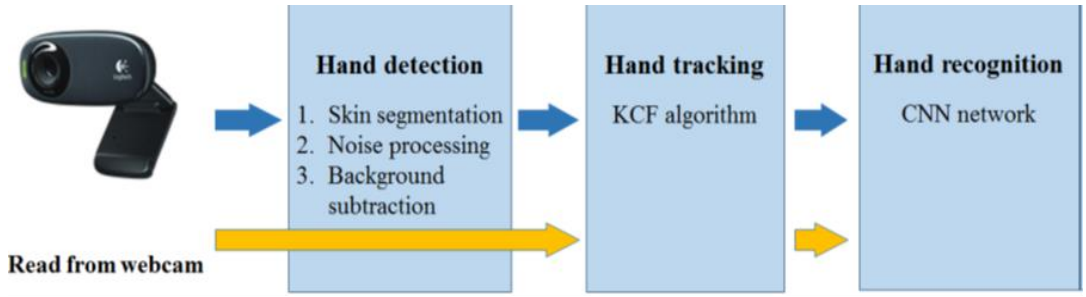


Figure 3.3 The diagram of the overall hand gesture recognition concept.

The camera initializes the tracking algorithm after recognizing the ROI of the first frame entering the lens, as illustrated by the blue arrow-guided route. The ROI block was shrunk to join the deep CNN network for identification. The tracking algorithm then continues to watch recent arriving frames (i.e., skip hand observation) and identify them, as shown by the yellow arrow-guided route.

Hand-based geometry is employed in this thesis since it is more favourable than another technique. Consequently, calculation times and function sizes were lowered, and performance was improved. The trained support vector machine uses the specified functions to decide whether the picture is in the original or cheat class. The findings of the comparative experimental study presented in this paper are often superior to competitor approaches, indicating that the recommended methodology is successful.

3.3 Alexnet Archıtecture

AlexNet is a CNN "convolutional neural network" Architecture is one of the types of neural networks with input and output layers. "The architecture used in the 2012 study was dubbed AlexNet after the original creator, Alex Krizhevsky". "AlexNet was the winning submission in the 2012 ILSVRC". It solves the photo classification problem by accepting an image from 1000 different classes (e.g. cats, dogs, etc.) and outputting a vector of 1000 numbers. It has seven rectified linear units (ReLu) layers, five convolutional layers, three pooling layers, three fully connected layers, and two normalizing layers in the hidden layers. "AlexNet" has a big influence on machine learning, especially when it comes to applying deep knowledge.. There are eight layers in the AlexNet convolutional neural network architecture, including five layers of convolution and three entirely linked (fully connected) layers. The convolution,

pooling and normalization layers are a composition of some of the convolution layers[24], as shown below.

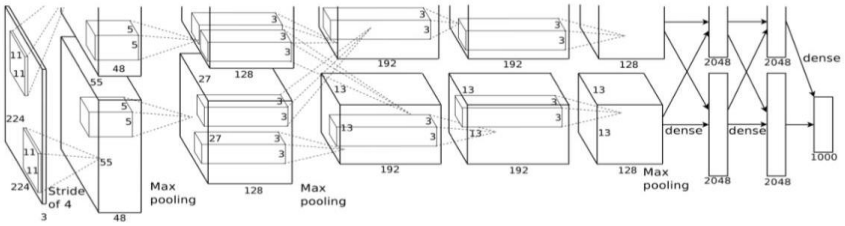


Figure 3.4 A typical architecture of AlexNet

- AlexNet architecture includes convolutional layers (5), normalization layers (2), fully connected layers (2), max-pooling layers (3), and softmax layers (1).
- The convolutional layer includes a nonlinear activation function ReLU and convolutional filters.
- The "pooling" layers are utilized to execute max pooling.
- The input size is stable owing to the existence of wholly linked layers.
- The input size: Maximum positions are 224x224x3. If some padding is present, then it works out to be 227x227x3.
- AlexNet had 60 million parameters.

3.3.1 ReLU Nonlinearity

Using ReLU non-linearity, AlexNet presents us that deep CNN's can be trained much more rapid with the help of soak activation functions such as Tanh or Sigmoid. Figure 3.6 illustrates that with ReLUs(curve), AlexNet can extend a 25% training error

rate. This is six times more rapid than an identical network that uses tanh(dotted curve). This was tested on the CIFAR-10 dataset[24].

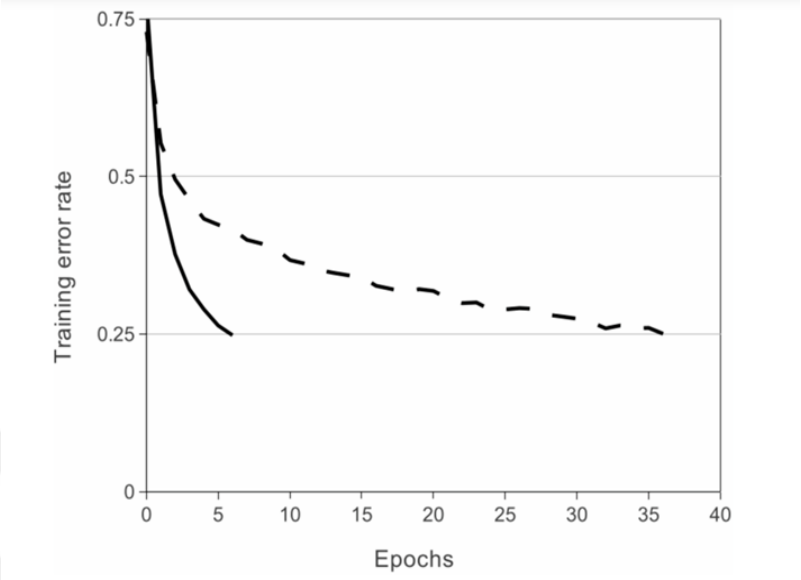


Figure 3.5 ReLU Nonlinearity[24]

3.3.2 Dropout

A neuron is dropped from the Neural Network with a probability of 0.5 during dropout. A neuron that falls vertically does not contribute to forward or backward propagation. As shown in the diagram below, each input is routed through a different neural network design. As a result, the acquired weight parameters are stable and don't quickly get overfitted.

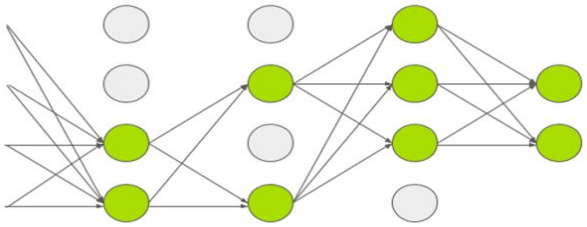


Figure 3.6: Neural network architecture[24]

3.4 Support Vector Machine (SVM) Algorithm

One simple, highly effective, and different classification method used in classification problems is support vector machines (SVM). SVM, one of the statistical learning algorithms defined by Vapnik-Chervonenkis, has given successful results in many real problems. Support vector machines (SVM) is a controlled grouping algorithm built on a statistical learning study. Support vector machines are known as training algorithms based on the probability distribution of statistical techniques. There is not enough information and distribution about the distribution laws that form the basis of statistical methods in many practical situations.

The working principle of support vector machines (SVM) aims to maximize the perpendicular distances of these examples to the separator plane (which will separate the two classes) by finding the closest examples of the courses while classifying the data, in other words, to the hyperplane. Thus, the misclassification error of the data in both training and testing sets is minimized.

The separator plane can have many alternatives without changing its success on the dataset. Thanks to SVM, the separator plane is at the same and maximum distance from both classes. The basic idea is to define the best separator plane for linearly different data structures. Sometimes the data structure can not be analyzed linearly. This problem was overcome by moving the data to an altered dimension with the transformation technique.

In the simplest pattern recognition tasks, vector machines use a linear separating high plane to create a classifier with a leading edge. The learning problem is taken as a

nonlinear optimization problem to implement this. When the given classes space is not linearly separable at the original input, SVM first linearly transforms a higher-dimensional feature space into the actual input space. This transformation can be achieved using various non-linear mappings: Polynomial, sigmoidal as in multilayer perceptron, radially symmetrical tasks can be RBF (Radial Based Function) mappings to have principal tasks where it is Gaussian. After the nonlinear transformation step is done, it is the task of the SVM to find the optimal linear separation. It solves the optimization problem of the same type as the separator plane calculation in the original input space for linearly separable classes. The high plane in the feature space gives the optimal result when there is a maximum edge classifier.

3.4.1 Support Vector Machines for Linearly Separable Data

In classification with support vector machines, it is aimed to separate the samples belonging to two classes, which are usually shown with class labels as (-1,+1), with the help of a resolution function acquired from the training data. Using the decision function in question, a hyper-plane is found that can split the training data most appropriately.

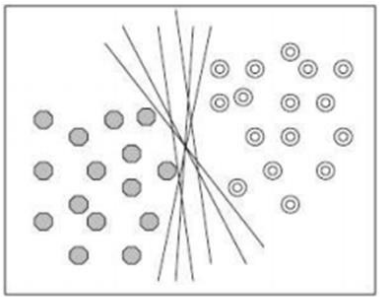
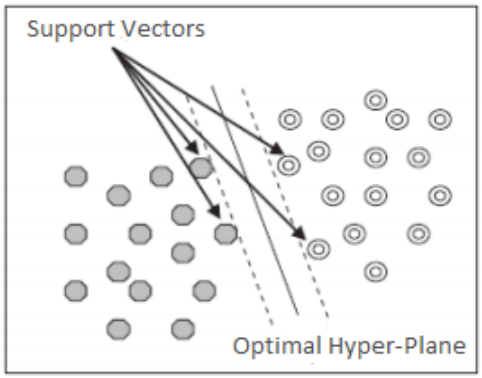


Figure 3.7 (a) hyperplanes for a two-class problem

However, the SVM goal is to invite the hyperplane, which increases the distance between the points closest to it. Figure 3.7 b showed that the hyperplane maximizes the boundary and makes the optimal separation is called the optimal hyperplane. The issues that limit the boundary width are called support vectors.

In Figure 3.7 b. two classes are depicted on a two-dimensional plane. This plane and its dimensions might be considered characteristics. In other words, for each input that entered the system, feature extraction was performed, and a separate point representing each input was obtained in this two-dimensional plane. Classifying these points entails categorizing the information based on the extracted characteristics.



(b)

Figure 3.7 (b) Support vectors and optimal hyperplane.

For training of SVM in a linearly separable two-class classification problem, assuming that the training data consisting of k samples $\{xi, Yi\}, i = 1, \dots, k$ & $Yi \in \{-1, +1\}$ the inequalities of the optimum hyperplane will be as follows:

$$w\, x_i + b \geq 1 \; each \; y = 1 \tag{3.1}$$

$$w\, x_i + b \leq 1 \; each \; y = -1 \tag{3.2}$$

In this model, x∈R^N represents an N-dimensional space, y ∈ {-1, +1} the class labels, w the weight vector (normal to the hyperplane) and b the trend value. To determine the optimum hyper-plane, two hyper-planes forming its borders and its parallel plane should be selected (Figure 3.7). The points forming these hyperplanes are called support vectors, and these planes are expressed as the given equation:

w·x_i+b=±1

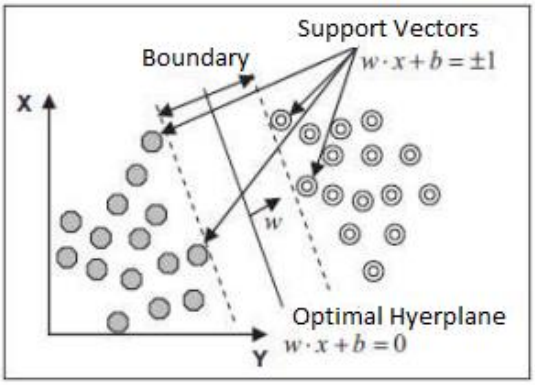


Figure 3.8 Determination of the hyperplane for linearly separable datasets.

As might be expected, the hyperplane between the two classes can't be unidirectional. Although there are two different hyper-plane possibilities, the one with the largest tolerance (offset) is taken in the SVM method.

To maximize the limit of the optimum hyperplane, ‖W‖ expression should be minimized. In this case, determining the optimal hyperplane requires solving the following constrained optimization problem.

$$\min \left[\frac{1}{2}\|\mathcal{W}\|^2\right]$$

(3.2)

It is expressed as

$$y_i(w \cdot x_i + b) - 1 \geq 0 \;\&y_i \in \{1, -1\} \tag{3.3}$$

Lagrangian equations can be used for resolving optimization problems. After this process;

$$L\,(w, ba) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{k} a_i\,(y_i(w \cdot x_i + b) - 1) \tag{3.4}$$

equality is achieved

The traditional solution of the Lagrangian equation for some datasets may be too easy. However, a different path must be followed to produce a computer-aided solution to this equation.

$$\frac{\partial L(w,b,a)}{\partial w} = 0 \tag{3.5}$$

$$\frac{\partial L(w,b,a)}{\partial b} = 0 \tag{3.6}$$

$$\frac{\partial L(w,b,a)}{\partial a} = 0 \tag{3.7}$$

According to the traditional application of the Lagrangian equation, it is necessary to evaluate the above three equations in a standard calculation. Instead, the dependency on three variables in the solution method can be reduced to one by discovering a typical relationship theoretically.

The Lagrangian equation should depend on a single variable by applying the classical solution. Thus:

$$\frac{\partial L(w,b,a)}{\partial w} = 0 \Rightarrow \text{w} = \sum_i a_i y_i x_i \quad (3.8)$$

$$\frac{\partial L(w,b,a)}{\partial b} = 0 \Rightarrow \sum_i a_i y_i = 0 \quad (3.9)$$

These expressions are substituted in the Lagrangian equation. Thus, the Lagrangian equation is made dependent on a single variable.

$$L(w,b,a) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{k} a_i(y_i(w \cdot x + b) - 1) \quad (3.10)$$

$$L(a) = \sum_i a_i - \frac{1}{2}\sum_i \sum_j a_i a_j y_i y_j x_i x_j \quad (3.11)$$

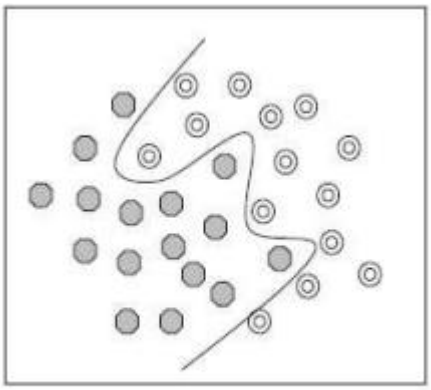
As an outcome, the decision function for a linearly separable two-class problem can be written as.

$$f(x) = sign\,(\sum_{i=1}^{k} \lambda_i y_i (x \cdot x_i) + b) \quad (3.12)$$

The λ_i variables must be $\lambda_i \geq 0$ to represent the Lagrangian coefficients.

3.4.2 Support Vector Machines For Linear Non-Separable Information

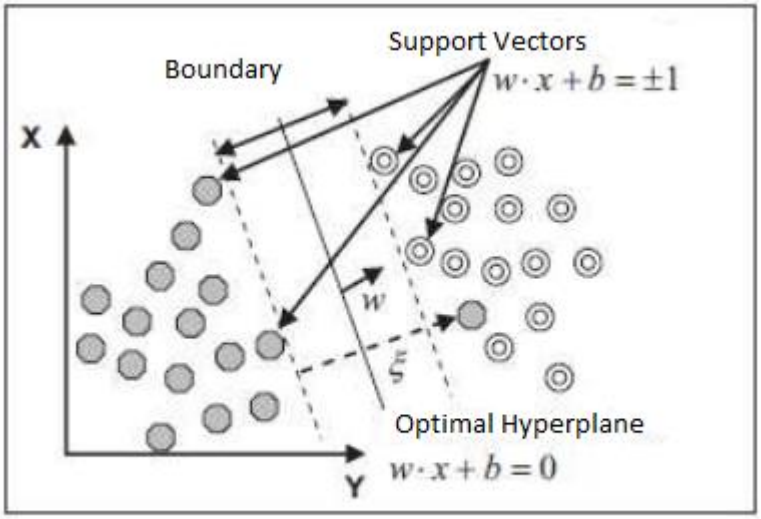
Linear data separation is not possible in many problems (Figure 3.9 a). In this case, a non-linear line is needed. Rather than fitting the data with nonlinear curves, a more consistent separation is achieved by moving the SVM to another space via a kernel function. The kernel function can transfer the data to higher dimensions to be classified. Upgrading the kernel function is a powerful approach. It allows SVM models even in cases separated by very complex boundaries.



(a)

Figure 3.9 a non-linearly separable data set.

In this situation, the remaining part of the data shows the problem. The optimal hyperplane is answered by defining a positive artificial variable (đi) (Figure. 3.9 b).



(b)

Figure 3.9 b The hyperplane for nonlinearly separable datasets.

The equilibrium between maximizing the limit and minimizing misclassification errors can be checked ($0 < C < \infty$). This will have positive values and is represented by C. The optimization problem for data that cannot be linearly differentiated using the editing parameter and the dummy variable:

$$min = \left[\frac{\|w\|^2}{2} + C \cdot \sum_{i=1}^{r} \xi_i\right] \quad (3.13)$$

The limitations related to this are;

$$y_i(w \cdot \varphi(x_i) + b) - 1 \geq 1 - \xi_i \quad (3.14)$$

$$\xi_i \geq 0 \; and \; i = 1, \ldots, N$$

For the solution of the optimization problem expressed in Equations (3.13) and (3.14), as can be seen in Figure 3.7, the data that cannot be separated linearly in the input space is displayed in a high-dimensional area defined as the feature space. Thus, the linear separation of the data can be made, and the hyper-plane between the classes can be determined.

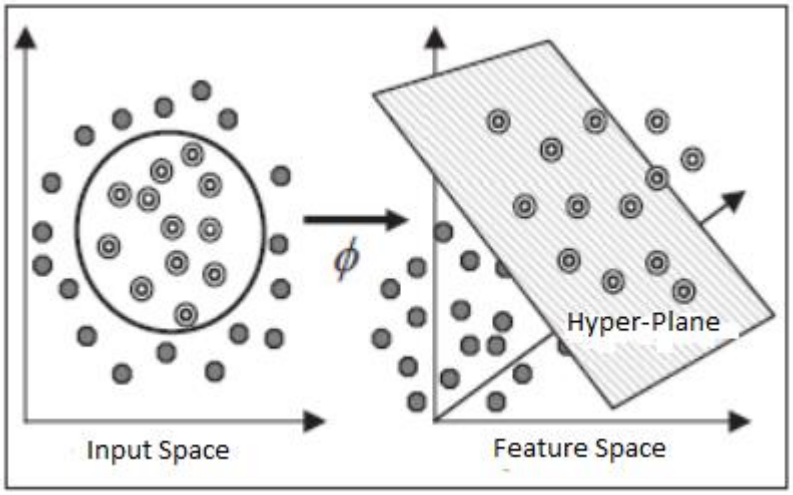


Figure 3.10 Converting the data to a higher dimension with the kernel function.

3.4.3 Kernel Functions

Support vector machines can make non-linear transformations with the help of a kernel function, which is mathematically expressed as $K\,(x_i, x_i) = \varphi(x) \cdot (x_j)$ and in this way, it allows the data to be separated linearly in high dimensions. As a consolation,

the following is the decision rule for solving a nonlinearly separable two-class issue using the kernel function:

$$f(x) = sign\left(\sum_i a_i y_i \varphi(x)\varphi(x_i) + b\right) \quad (3.15)$$

It is necessary to find the kernel function for a grouping operation to be executed with support vector machines and the minimum parts of this function. The radial basis function, polynomial, Pearson VII (PUK) function and normalized polynomial kernels most frequently used as kernel functions in the literature are presented in Table 3.1 with their formulas and parameters. As can be seen from the table, the user must determine specific parameters for each kernel function. While the PUK kernel has only two variables to compute, it needs to identify a variable in the model structure that will serve as the foundation for grouping for subsequent functions.

Table 3.1 Essential kernel functions and parameters used in support vector machines.

kernel function	mathematical expression	parameter
polynomial kernel	$k(x,y) = ((x \cdot y) + 1)^d$	polynomial degree (d)
normalized polynomial kernel	$k(x,y) = \frac{((x \cdot y) + 1)^d}{\sqrt{((x \cdot x) + 1)^d((y \cdot y) + 1)^d}}$	polynomial degree(d)
radial basis function kernel	$k(x,y) = e^{-y\|(x-x_i)\|^2}$	kernel size (γ)

Pearson VII (PUK) kernel	$\frac{1}{\left[\left(1+\frac{\sqrt[2]{\|x-y\|^2\sqrt[2]{2\left(1/\omega-1\right)}}}{}\right)\right]^{2-\omega}}$	Pearson width parameters (σ, ω)

When comparing kernel functions, it can be said that polynomial and radial-based kernels are more straightforward to comprehend. The procedure becomes more difficult as the degree of the polynomial increases, even though it appears to be mathematically simple. After a certain point, this significantly increases processing time and degrades classification accuracy. Changes in the radial basis function parameter, represented as kernel size (v), on the other hand, have been found to have a more minor impact on classification performance. With two variables (o, 0) known as Pearson width, the PUK kernel has a larger composite mathematical structure than other kernel functions. These two factors impact classification accuracy, and it's impossible to predict which set of parameters will produce the most outstanding results. As a result, determine the most appropriate parameter pair in the use of the PUK kernel.

Aside from kernel function-specific parameters, the user must specify the editing parameter C for all support vector machines. The ideal hyperplane cannot be computed accurately if too small or too large values are used for this parameter, resulting in a significant loss in classification accuracy.

As can be seen, choosing optimal values for the parameters is a factor that directly impacts the SVM classifier's performance. Although most people utilize a trial-and-error technique, cross-validation provides more accurate outcomes.

Ideally, an SVM analysis should produce two utterly different classification results and the multidimensional plane separating the feature vectors. However, the perfect classification may not be possible. There may be times when the model does not generalize well to the data and produces too many feature vectors. Therefore, the cross-validation precision is the percentage of data that has been correctly classified.

Nonconformity may be avoided via the cross-validation technique. The cross-validation method is used to assess the performance of the classification model that has been constructed. The data set is separated into two pieces for this purpose. The first component serves as training data for the model, which is the foundation for classification, while the second serves as test data for determining the model's performance. The number of successfully categorized samples represents the classifier's performance after applying the model generated using the training set to the test data set. As a result, the kernel parameters that yield the highest classification performance were established using the cross-validation approach. The model that would serve as the basis for classification was produced.

3.4.4 Multi-Class Support Vector Machines

The main difference of the support vector machine method from other classification methods is that it can only distinguish two classes. Various algorithms have been developed to use the SVM method in a system with more than two classes. The most

commonly used of these algorithms are one against all and one against one. In this study, one versus all algorithm is used to implement the application with SVM.

All against one; In this method, while one of the classes is accepted as (+) during the training phase, the information of all other classes is obtained as (-), and M SVMs are created for the M class. The sample to be recognized belongs to that class which SVM classifies as (+). An example application of the one against all methods is shown in Figure 3.11.

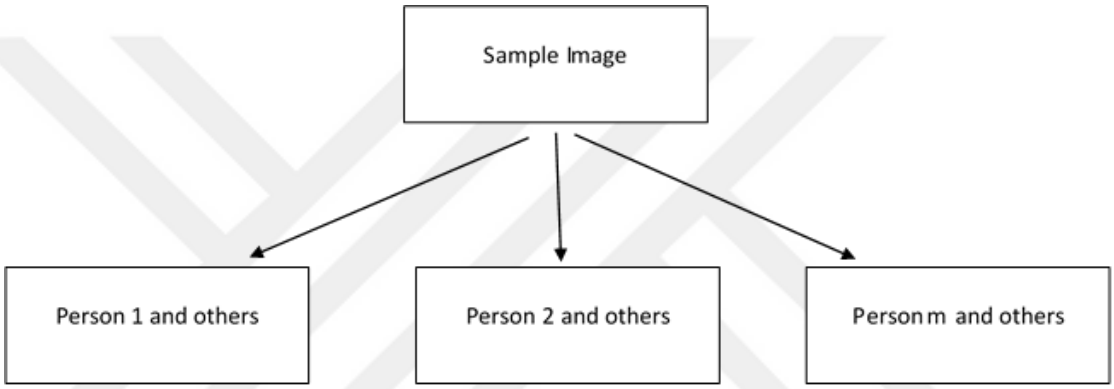


Figure 3.11 Example of one versus all for SVM.

The support vector is very useful in learning, building on simple ideas, and showing high performance in practical applications. The number of instances to use in SVMs is not essential. SVM also classifies data not seen during training without any problems. This demonstrates the SVM's ability to generalize. Its generalization feature makes SVM an excellent alternative to other techniques (ANN, decision tree, etc.) and provides an advantage.

3.5 K Nearest Neighbour (KNN) Algorithm

The (K-NN) is a trained sample-build grouping algorithm. Grouping of a vector in KNN is done using vectors of well-known class. The sample to be tested is handled

one by one with every sample in the training set. To calculate the type of sample to be tested, the K samples nearest to that sample in the training set are chosen. It is said that the sample to be checked belongs to this class, which class has the more samples in the group comprised of selected samples. The Euclidean distance finds the space between samples. All space values evaluated using Euclidean space are sorted. The minor K is calculated based on the number of K out of the ordinal values. The K nearest neighbor samples to the sample to be tested are calculated. Class labels of K nearest neighbors are utilized to group the sample to be tested.

All space values determined using the Euclidean space are sorted. The minor K is calculated depending on the multiple K among the rank values. The K nearest neighbouring samples to the sample to be tested are calculated. Class labels of K nearest neighbors are used to group the sample to be tested.

Two factors can occur to select the K value. In the first case, the K value is chosen as an odd number to stop the number of "+1" and "0" samples from existing the same. Even if the K value is selected as an even number, the samples from each class for K samples are summed up together, and their averages are found. There is no limitation on the various classes for this algorithm. Grouping can be done by calculating the desired number of classes. The flow chart followed in the KNN algorithm is given in Figure 3.12.

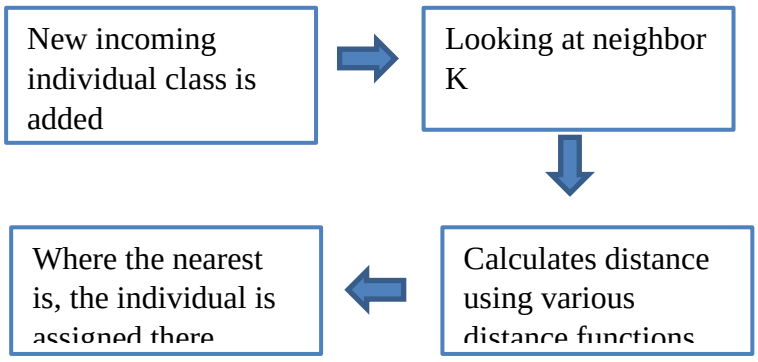


Figure 3.12 KNN algorithm

Distance functions utilized in KNN technique; Manhattan Distance Function, Minkowski Distance Function, Euclidean Distance Function.

The steps of the KNN algorithm.

Each information in the test set X = {x1, x2, x3, x4, …, x n}; information in the learning set D = {d1, d2, d3, d4, …, dm} proximity is determined.

$$sim(x_i, d_1) = \frac{x_i * d_1}{\| x_i \| \| d_1 \|} \quad (3.16)$$

$(i = \{1,2,3, \ldots . n\}, \mathrm{l} = \{1,2,3 \ldots, \mathrm{m}\})$

1. The closeness of every information to the data in the learning set is noted and the average is determined by taking the first “k”.

$$\text{sim\_avg}(x_j) = \frac{max[\sum_{1=1}^{k} \text{sim}(x_i, d_1)]}{K} \quad (3.17)$$

2. Those whose mean values are more than the specified threshold value are grouped as normal, and those below the threshold value are grouped as abnormal.

Training – Test Phases of K-NN Algorithm

For the information to be categorized, the K-NN method does not require any training stages. The information vector coordinates must be preserved with the class label throughout the training phase. The class label is usually used as the information vector's identification. During testing, this is utilized to categorize information vectors.

a) The goal is to compute the class label for the most recent point given the information points for testing. The procedure is used for the k = 1 rule, which is the closest neighbour rule, and then expanded for the k = k rule, which is the k-nearest neighbour rule.

b) 1-Nearest Neighbor Rule (k=1)

This approach is a simple scenario for grouping. Consider 'x' to be the point to be marked.

- The nearest point to 'x' is found in the training information set. Let this closest point be 'y'
- The closest neighbor rule now requests that label 'y' be assigned to 'x'. When the number of data points isn't too great, this method is utilized.

If the number of information points is very large, the x and y labels likely are the same.

c) 3-Nearest Neighbour Rule (K=3)

To consider the subject, K=3 was taken and the details are listed below.

- In the first step, the training information is assigned to the classes.

- By calculating the coordinate of the latest element to be grouped, the span between all elements are computed.
- The computed values are sorted and the class is calculated according to the multiple of the votes of the first 3 neighbours in the circle shown with dashed lines in Figure 3.13.

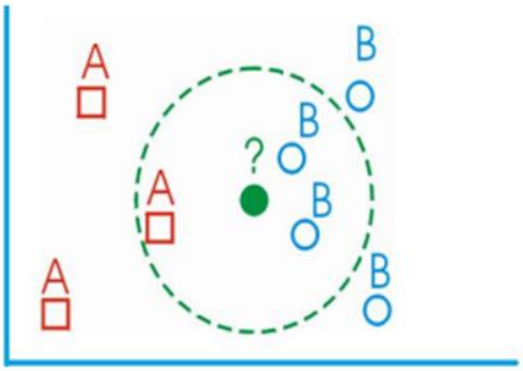


Figure 3.13 Initial information

d) Generalization of the K-NN Algorithm.

- An initial value, 'K', is arranged.
- The training data set is complete by recording the coordinates and class labels of the information points. Figure 3.14 depicts this procedure graphically.

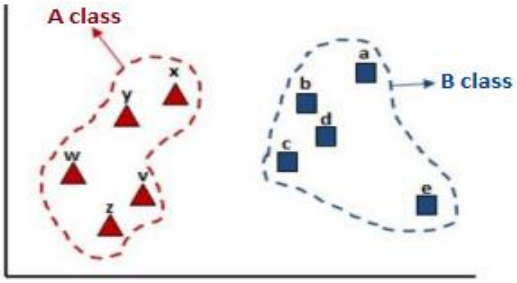


Figure 3.14 Distance calculation

- The information point is loaded from the test information set.
- Multiple votes is made between the "K" nearest neighbours of the test information from the training data built on a distance metric. This process is graphically shown in Figure 3.15.

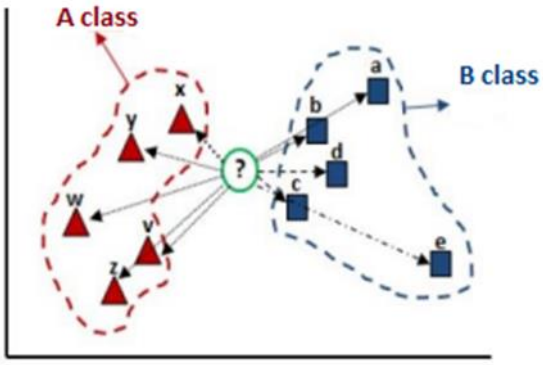


Figure 3.15 Calculating neighbours and voting for labels

3.6 Biometric Systems Evaluation

The terminology related to the system and scores used in the biometric system evaluation was studied in the biometric system calculation. The exact match was searched between the database and hash stored on the card in the card reader based systems. However, it was unusual to get an exact match in biometrics due to various factors like the surroundings, sensor sound, and human behaviour. As a result, the two feature vectors derived from two independent acquisitions of the same biometric

characteristic are both problematic and exceptional. So, the following terms should be understood deeply

- changeability in biometric feature traits derived from the same person (intra-class liability to vary).
- Variability between biometric feature traits generated from the same characteristic (class liability to change).
- Inter-class variability: changeability between biometric feature vectors created from two dissimilar characteristic.
- Biometric entropy: It is an information measurement in a specific biometric trait. More information leads to the greater discriminative power

Generally, it is preferred to have a small, sizeable inter-class variability and intra-class variability. It facilitated effectively distinguishing between dissimilar individuals instead of searching the similarities in the same individuals. The similarity score represented two biometric feature vectors [108].

- Genuine score: A score calculated by comparing two feature vectors from the same person.

- Impostor score: A score calculated by comparing two feature vectors created by two dissimilar people.

3.6.1 Error Visualization And Metrics

In the biometric system, the acceptable (FAR) and the rejectable (FRR) rate can be commanded through the final threshold T. FAR and FRR are opposite, i.e. increased in one parameter will decrease other parameters. In the Detection Error Tradeoff (DET) curve, different values of T can be obtained by changing FAR and FRR values [109]. The dependence of FRR on FAR with different T values can be generated on a basic deviation scale (Figure 3.16). The terminologies used in the visualization metrics are defined below

- False Accept Rate (FAR): Impostor scores fraction that exceeds threshold T.
- False Reject Rate (FRR): Genuine scores fraction falls below threshold T.
- Genuine Accept Rate (GAR): Genuine scored fraction that exceeds threshold T

$$GAR = 1 - FRR$$

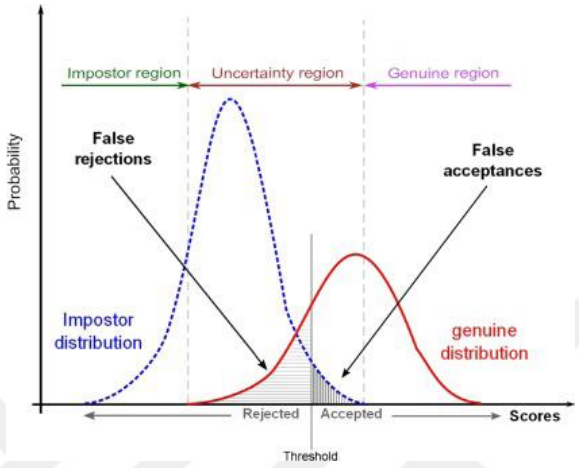


Figure 3.16 An example of two biometric feature vector score distributions. [110]

ROC curve can be utilised as an alternative method [111]. In this, dependency of GAR on FAR is plotted on linear, semi-logarithmic or logarithmic scale Figure 3.17. A single value can measure biometric system performance at the Equal Error Rate (EER). EER is the point on DET where the values for FAR=FRR. Similarly, a point on the ROC curve, where the importance of GAR=FAR.

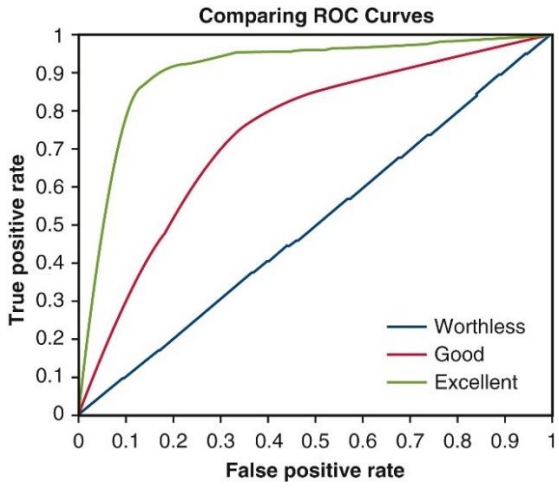


Figure 3.17 An example of the Receiver Operating Characteristic (ROC) curve [33]

CHAPTER 4

EXPERIMENTAL RESULTS

4.1 Hand Images preparation

For implementing the proposed study in the Matlab R2019a package and using the database of 7200 images (CASIA-MS-Palmprint V1 database). Division of the images was like that 72 images for each person, and I labelled them into 100 classes for implementation purposes. Datastore was created for all hand images to efficiently manage a image collection with a convolutional neural network.

The hand recognition process for our study is described in the flowchart in Figure 4.1.

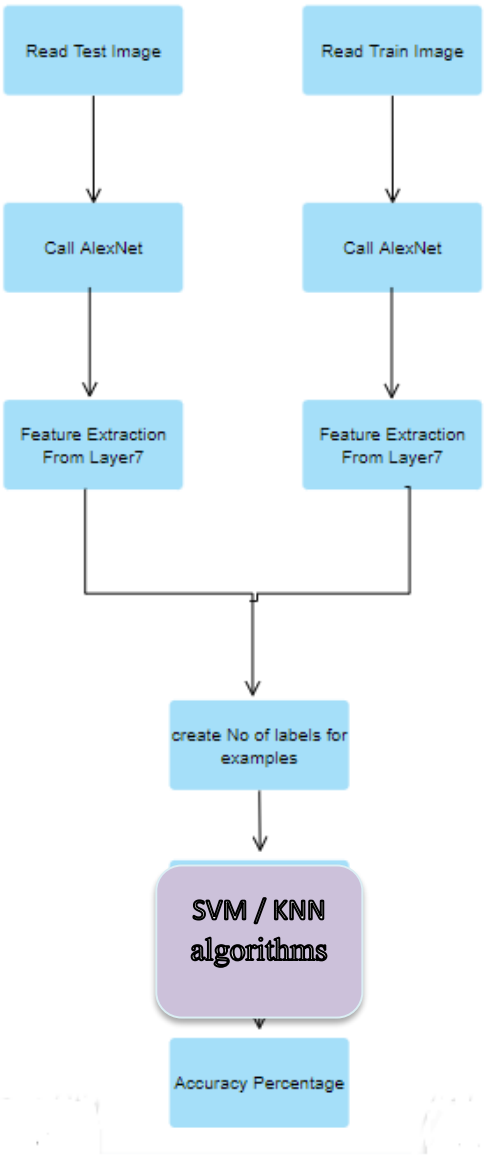


Figure 4.1 The proposal method

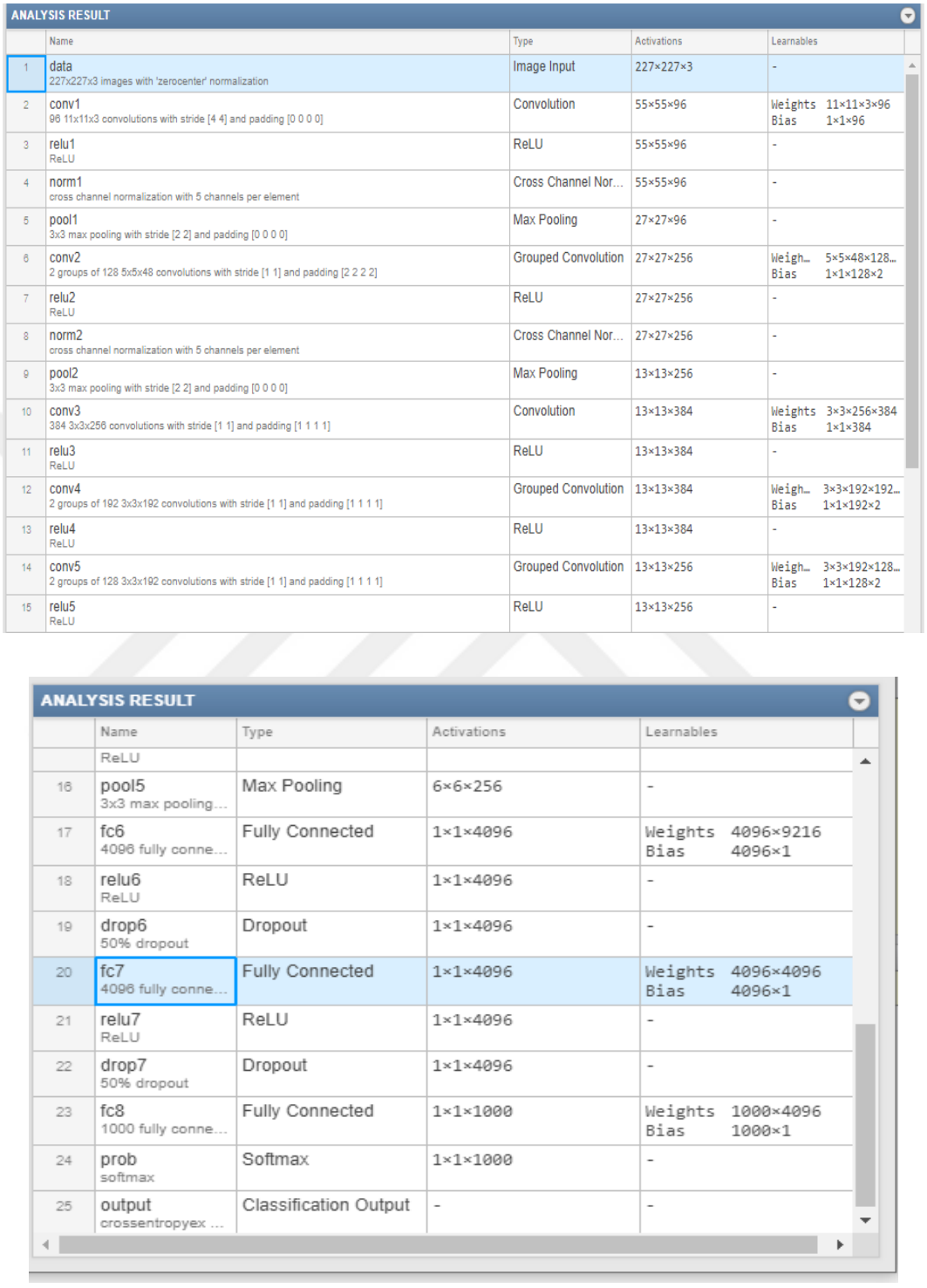
4.2 Resize Input Images

In this thesis the size of each input image is resized to 227*227*3; the AlexNet can only process the input with 227*227*3.

4.3 Implementation

This section focuses on the computations involved in the experiment of a computer system to complete the implementation and then test it using the algorithms and classifiers listed below. In this section, the AlexNet Convolution neural network

software is shown in Figure 4.2. was applied. This study aims to apply transfer learning techniques to extract high-level features.



ANALYSIS RESULT

	Name	Type	Activations	Learnables
1	data 227x227x3 images with 'zerocenter' normalization	Image Input	227×227×3	-
2	conv1 96 11x11x3 convolutions with stride [4 4] and padding [0 0 0 0]	Convolution	55×55×96	Weights 11×11×3×96 Bias 1×1×96
3	relu1 ReLU	ReLU	55×55×96	-
4	norm1 cross channel normalization with 5 channels per element	Cross Channel Nor...	55×55×96	-
5	pool1 3x3 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	27×27×96	-
6	conv2 2 groups of 128 5x5x48 convolutions with stride [1 1] and padding [2 2 2 2]	Grouped Convolution	27×27×256	Weigh… 5×5×48×128… Bias 1×1×128×2
7	relu2 ReLU	ReLU	27×27×256	-
8	norm2 cross channel normalization with 5 channels per element	Cross Channel Nor...	27×27×256	-
9	pool2 3x3 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	13×13×256	-
10	conv3 384 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	13×13×384	Weights 3×3×256×384 Bias 1×1×384
11	relu3 ReLU	ReLU	13×13×384	-
12	conv4 2 groups of 192 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]	Grouped Convolution	13×13×384	Weigh… 3×3×192×192… Bias 1×1×192×2
13	relu4 ReLU	ReLU	13×13×384	-
14	conv5 2 groups of 128 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]	Grouped Convolution	13×13×256	Weigh… 3×3×192×128… Bias 1×1×128×2
15	relu5 ReLU	ReLU	13×13×256	-

ANALYSIS RESULT

	Name	Type	Activations	Learnables
	ReLU			
16	pool5 3x3 max pooling...	Max Pooling	6×6×256	-
17	fc6 4096 fully conne...	Fully Connected	1×1×4096	Weights 4096×9216 Bias 4096×1
18	relu6 ReLU	ReLU	1×1×4096	-
19	drop6 50% dropout	Dropout	1×1×4096	-
20	fc7 4096 fully conne...	Fully Connected	1×1×4096	Weights 4096×4096 Bias 4096×1
21	relu7 ReLU	ReLU	1×1×4096	-
22	drop7 50% dropout	Dropout	1×1×4096	-
23	fc8 1000 fully conne...	Fully Connected	1×1×1000	Weights 1000×4096 Bias 1000×1
24	prob softmax	Softmax	1×1×1000	-
25	output crossentropyex ...	Classification Output	-	-

Figure 4.2 AlexNet.architecture

4.4 Feature Extraction

The training and testing features were extracted from layer 7, a fully connected layer containing a high level and linear features. Using AlexNet, which has been trained with over one million photos, we loaded the pre-trained CNN. As previously stated, our method fine-tuned AlexNet; for example, we deleted the last layer of AlexNet and used the data from the final completely linked layer. We used CNN to compute the training features and test images based on the data from the last fully connected layer. Then, the class labels were extracted from the training and test picture sets.

4.5 Support Vector Machine (SVM) And K-Nearest Neighbour (KNN)

CNN uses the classifier to classify the collected features in the last phase. To assess the accuracy of this thesis, we used a support vector machine (SVM) and k closest neighbours (KNN).

The SVM was trained using the retrieved training features and then tested using the testing features.

If a collection of labelled data is provided in training set for the method, a support vector machine is a machine-learning model that can generalize between two classes. The SVM's main task is to look for a hyperplane to differentiate between the two classes..

The KNN is trained by the extracted training features and tested on testing features.

KNN works by calculating the distances between a query and all of the examples in the data, selecting the K samples closest to the question, and then voting for the most frequent label (in the case of classification) or averaging the titles (in the case of regression). The maximum error percentage is less than 0.01, which is considered acceptable.

- Based on the predictor data X and response Y, a k-nearest neighbour classification model was created.
- INPUT ARGUMENTS:-
  - NumNeighbors (K) = 1 by default.

- X =Training data

Format: each row in X represents one training point, while each column represents one feature variable.

- Y = Class labels

K(K – 1)/2 binary (SVM) models with one-versus-one coding are used to fit multiclass models for support vector machines, where K is the number of unique class labels (levels). Utilizing the defaulting choices, provide multiclass models train a multiclass ECOC model. Fit multiclass models empty the SVM characteristics (Alpha, SupportVectorLabels, and SupportVectors for all linear SVM binary learners; Fit multiclass models list Beta rather than Alpha) in the model shown by default and for efficiency.

An error-correcting output codes (ECOC) model converts a three- or more-class classification problem to a series of binary classification problems.

Binary classification is dividing a set of components into two categories using a classification rule.

The one-versus-one method Converts a multiclass classification problem into a series of binary classification issues. One trains (K - 1) / 2 binary classifiers for a K-way multiclass problem using the one-vs-one (OvO) reduction; each is given samples of a couple of classes from the initial training group and is expected to learn to distinguish between them. One kind is positive, another is negative, and the remaining types are ignored for each binary learner. All combinations of class pair assignments are exhausted in this design.

The characteristics of the SVM template object are blank. The software sets the applicable attributes to their default settings for training the ECOC classifier.

Features Test: Points of observation (vectors) that we wish to identify and so test the correctness of each model.

4.6 Result

The outcomes of using the proposed method are shown in this section. The training data are divided each time for each rating from 10% to 90%. The procedure is that when we used a training data rate of 10%, the testing rate was 90%; after that, when the training data was 20%, we used 80 % of the data for testing purposes. After this step, divided data to 30% for training and 70% for testing and recording the accuracy in each classification algorithm. And so on, in each different training data rate case, calculate the time, which process takes training and testing data. The last scenario is the best one. So, in SVM, Accuracy is 99.14%, and KNN Accuracy is 99.86%

Table 4.1: The Proposal Result

Sr. No.	**Training data**	**Testing data**	**SVM accuracy**	**Time in seconds**	**KNN Accuracy**	**Time in seconds**
1.	10%	90%	57.60%	511.4549	56.94%	22.0418
2.	20%	80%	79.36%	475.07	75.52%	23.4617
3.	30%	70%	87.36%	455.374	87.60%	28.4821
4.	40%	60%	93.47%	434.8061	91.91%	35.5436
5.	50%	50%	96.11%	427.4489	95.56%	38.1322
6.	60%	40%	97.45%	423.0125	97.21%	34.2737
7.	70%	30%	98.77%	408.5823	98.00%	31.6215
8.	80%	20%	98.93%	397.8564	98.79%	24.2993

9.	90%	10%	99.14%	402.0703	99.86%	14.2781

4.6.1 Results Analysis

Although the average accuracy of SVM is slightly higher than the average accuracy of the KNN model since the training and the testing times is essential factors in designing biometric verification systems. Given the average time taken by the KNN model is 13 times less than the time taken by the SVM model. This resulted from the low computational complexity of KNN $O(n)$[112] compared to the SVM high computational complexity $O(n^3)$[113]. Therefore, the KNN is considered the most appropriate approach to use.

4.6.2 Discussion

Our system's limitations are long training time and flexibility, i.e. the system is implemented for 100 classes (persons). Suppose there is a new user to add to the system. In that case, we have to retrain the system from the beginning, for instance, if we developed our approach to accommodate 10000 employees. We have 72 images for each employee, and if we add a new one, we have to collect 72 images for that employee. This will lead to an increase in the total database size and computational cost. To overcome the addressed limitation, we can improve the system by using a one-shot learning technique[114] , allowing for smaller training samples for each class. Therefore, it will result in less computational time and more flexibility when introducing new employees' data to the system.

Table 4.2 Comparison with Related Works. Ref: Authors.

Ref	**Techniques Applied for Recognition**	**Size of The Database**	**Accuracy**
Villegas et al. [115]	Nearest Neighbour	120	70.20%
F. M. Al-Fiky et al. [116]	BPNN with MSE value	100	93%
R. Gross et al. [117]	AAMs	54	90.70%
R. Sanchez-Reillo et al. [91]	SVM (Radial basic function)	220	90%
S. Prabu et al. [118]	Hybrid Adaptive Fusion (HAF)	**Same Database**	98.5%
S. A. Angadi et al. [119]	SVM	570	94.74%
Own approach based on the last scenario	AlexNet + KNN	7200	99.86%
	AlexNet + SVM		99.14%

Own approach presented the best performance in accuracy. In [115], the author applied Nearest Neighbour (a supervised learning algorithm) to identify the system and showed low results compared with other studies. Finally, the system was trained and tested to 120 images. The experimental result as a total recognition rate of 70.2 % was found.

In[116], BPNN with MSE value presented 93%, and it's suitable than other methods but also lower than our method’s result. The system has 66 features for the right hand and is applied to 100 test images using BPNN with a threshold of 0.8. finally, the system was compared with other methods, and it was the best.

In[117], an AAM fitting algorithm was applied to 54 hand images from 18 volunteers, and the system achieved identification accuracies above 90.70%.

In [91], the system was applied to 10 different persons; the performance achieved by SVM was 90% lower than our method.

In [118] also SVM applied. The author created a multimodal biometric system using three types of hand images databases using hand traits and palmprint traits. The system achieved 94.74 % accuracy, and it's the best compared to other methods. In , the researchers applied a new algorithm method called Hybrid Adaptive Fusion(HAF), and the technique tested to the same database we used; they reached 98% accuracy.

Table 4.3 Comparison With Other Biometric Systems.

Ref	**Biometric Characteristic for Recognition**	**Size of the Database**	**Accuracy**
X. Liu et al. [120]	Iris	4249	97.80%
D. M. Monro et al. [121]	Iris	5111	100%
H. Cho et al. [122]	Face	16128	97.30%
M. Sharif et al. [123]	Face	400	99.50%
N. A. Mngenge et al. [124]	Fingerprint	3904	89.60%
A. K. Jain et al. [125]	Fingerprint	29257	74%
Our method	Hand geometry	7200	KNN = 99.86%
			SVM = 99.14%

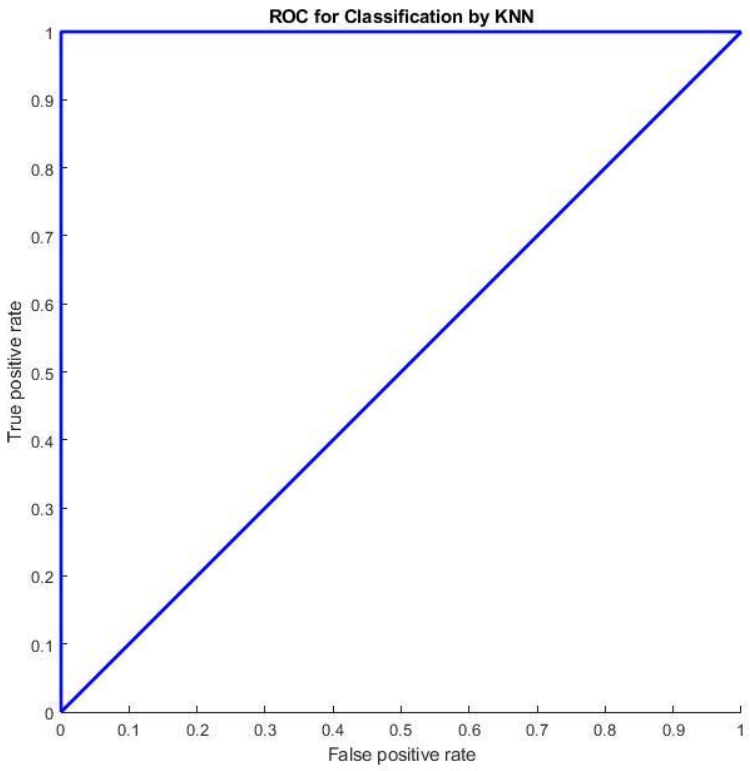


Figure 4.5 ROC Carve for KNN

The high-performance graph of the proposed approach is shown in figure 4.5. The ROC is a curve used to evaluate how well a system works. The actual positive rate is usually represented on the Y-axis in ROC curves, while the false positive rate is generally described on the X-axis. The "ideal" point, with a false positive rate of zero and an actual positive rate of one, appears in the picture's upper left corner. The testing image labels were utilized as the Y-axis on the ROC curve for KNN implementation, and the KNN model projected output was used as the X-axis. The figure has two lines, the first of which is the performance line, which is roughly equal to 1 because the system obtained a 99.86 per cent accuracy rate with KNN. The second one, A straight line from the origin (0.0, 0.0) to the top right corner, is always drawn by a classifier with a random performance level (1.0, 1.0). A simple estimation of the performance level is indicated by two areas separated by this ROC curve. ROC curves in the top left corner (0, 1) indicate good performance, whereas ROC curves in the bottom right corner (1,0) suggest poor performance.

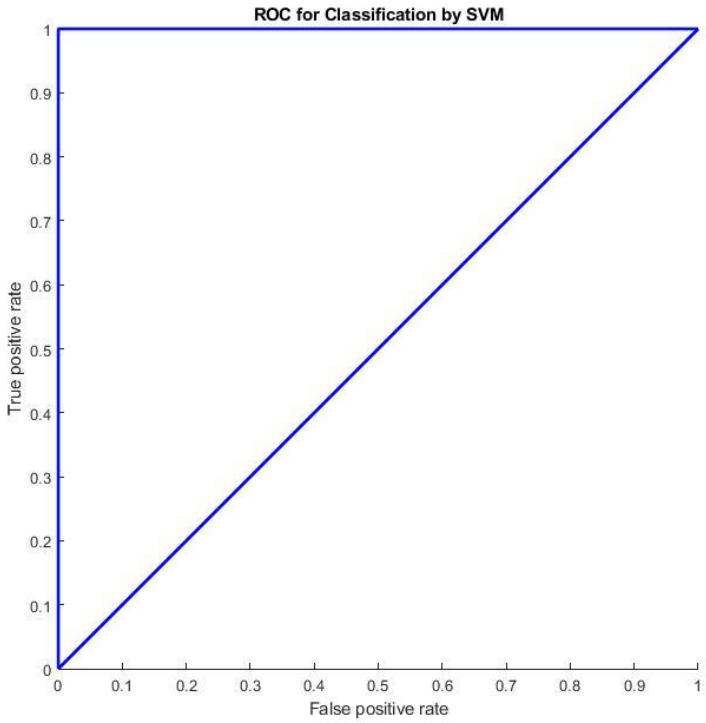


Figure 4.6 ROC Carve for SVM

The figures above also show the proposed method high-performance graph. The actual positive rate is usually represented on the Y-axis in ROC curves, while the false positive rate is generally described on the X-axis. The "ideal" point, with a false positive rate of zero and an actual positive rate of one, appears in the picture's upper left corner. The testing image labels were utilized as the Y-axis on the ROC curve for SVM implementation, and the SVM model projected output was used as the X-axis. The figure has two lines; the first is the performance line, which is roughly equal to 1 because the system obtained a 99.14 per cent accuracy rate with SVM. The second one, A straight line from the origin (0, 0) to the top right corner, is always drawn by a classifier with a random performance level (1, 1). A simple estimation of the performance level is indicated by two areas separated by this ROC curve. ROC curves in the top left corner (0.0, 1.0) indicate good performance, whereas ROC curves in the bottom right corner (1.0, 0.0) suggest poor performance.

CHAPTER 5

CONCLUSION

5.1 Conclusion

The suggested system incorporates various classification approaches for human hand geometry verification algorithms in this thesis. “The CASIA Multi-Spectral Palm-print Image Database V1” database is used for training and testing. Matlab R2019a is utilized to carry out the planned research. We used (SVM) and (K-NN) methods to create a classifier. We use Alexnet (the first CNN to win Image Net), the ImageNet Large Scale Visual Recognition Challenge 2012, as a model for feature extraction. This research shows which technique has greater accuracy processing speed and which classifier works better than the other. In this thesis, we have recorded the accuracy of the result. As we can see, the results are different in each test. When we used fewer data in training, the accuracy for SVM implementation was 57.60% and for KNN is 56.94%. But when we used more data for training, the result is good to consider. For instance, the last scenario’s implementation results in the SVM classifier is 99.14%, and the KNN classifier is 99.86%. This study shows that KNN is more resilient and faster at classifying and recognizing objects based on hand geometry features from photos with multiple classifications used in the implementation. Our system’s limitations are long training time and flexibility. We can improve the system by using a one-shot learning technique to overcome the addressed limitation. The suggested approach can be used in a variety of settings, including car parking, a cash vault, a computer station, and anti-passback system, a point of sale, time and attendance.

5.2 Future Work

In the future, I will use different techniques for classifiers to get more information about approaches. We can use Euclidean Distance Classifier, another model of CNN (GoogleNet, ResNet), decision trees, linear or logistic regression as a classifier in the future. As feature extraction methods, we can use local bınary pattern (LBP), a popular way for feature extraction or using a GRNN network without feature extraction. We can get high accuracy and compare with another approach using another database.

REFERENCES

[1] J. Jang-Jaccard and S. Nepal, "A survey of emerging threats in cybersecurity," *Journal of Computer and System Sciences*, vol. 80, pp. 973–993, 2014.

[2] A. bin Mansoor, H. Masood, M. Mumtaz, and S. A. Khan, "A feature level multimodal approach for palmprint identification using directional subband energies," *Journal of Network and Computer Applications*, vol. 34, pp. 159–171, 2011.

[3] S. Malassiotis, N. Aifanti, and M. G. Strintzis, "Personal authentication using 3-D finger geometry," *IEEE Transactions on Information Forensics and Security*, vol. 1, pp. 12–21, 2006.

[4] G. Gao, J. Yang, J. Qian, and L. Zhang, "Integration of multiple orientation and texture information for finger-knuckle-print verification," *Neurocomputing*, vol. 135, pp. 180–191, 2014.

[5] K. Fukushima and S. Miyake, *"Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition,"* New York: Springer, 1982, pp. 267–285.

[6] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture in two nonstriate visual areas (18 and 19) of the cat," *Journal of neurophysiology*, vol. 28, pp. 229–289, 1965.

[7] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, 1998.

[8] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *Proceedings of the IEEE Conf on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[9] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conf on Medical image computing and computer-assisted intervention*, 2015, pp. 234–241.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-CNN: Towards real-time object detection with region proposal networks," *Journal Advances in neural information processing systems*, vol. 28, pp. 778–789, 2015.

[11] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conf. on computer vision*, 2014, pp. 184–199.

[12] K. G. Lore, A. Akintayo, and S. Sarkar, "LLNet: A deep autoencoder approach to natural low-light image enhancement," *Pattern Recognition*, vol. 61, pp. 650–662, 2017.

[13] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," *Proceedings of the IEEE Conf. on computer vision and pattern recognition*, 2016, pp. 4651–4659.

[14] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conf. on computer vision*, 2014, pp. 818–833.

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR Conf.*, 2015, pp. 71–82.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Conf. on computer vision and pattern recognition*, 2016, pp. 770–778.

[17] C. Szegedy *et al.*, "Going deeper with convolutions," *Proceedings of the IEEE Conf. on computer vision and pattern recognition*, 2015, pp. 1–9.

[18] A. G. Howard *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint Journal*, vol.01, pp.1-7,2017

[19] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proceedings of the IEEE Conf. on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[20] C. Wang, H. Liu, and X. Liu, “Contact-free and pose-invariant hand-biometric-based personal identification system using RGB and depth data,” *Journal of Zhejiang University SCIENCE C*, vol. 15, pp. 525–536, 2014.

[21] J. Svoboda, M. M. Bronstein, and M. Drahansky, “Contactless biometric hand geometry recognition using a low-cost 3D camera,” in *2015 International Conf on Biometrics (ICB)*, 2015, pp. 452–457.

[22] A. Genovese, V. Piuri, and F. Scotti, “Touchless and Less-Constrained Biometric Systems,” in *Touchless Palmprint Recognition Systems*, New York: Springer,2014, pp. 33–48.

[23] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, “Stacked convolutional auto-encoders for hierarchical feature extraction,” in *International Conf. on artificial neural networks*, 2011, pp. 52–59.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Journal of Advances in neural information processing systems*, vol. 25, pp. 84–90, 2012.

[25] Z. Sun, Y. Wang, T. Tan, and J. Cu, “Robust direction estimation of a gradient vector field for iris recognition,” *Proceedings of the 17th International Conf. on Pattern Recognition*, 2004, pp. 783–786.

[26] G. Hu *et al.*, “When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition,” *Proceedings of the IEEE International Conf. on computer vision workshops*, 2015, pp. 142–150.

[27] K. Fukushima, “A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” *Biol. Cybern.*, vol. 36, pp. 193–202, 1980.

[28] Y. LeCun, “The MNIST database of handwritten digits,” in *IEEE Signal Processing Magazine*, vol. 29, pp. 141-142,2012.

[29] R. Wang, C. Han, Y. Wu, and T. Guo, “Fingerprint Classification based on depth neural network,” *arXiv preprint Journal*, vol.02, pp. 251-266, 2014.

[30] D. L. Nguyen, K. Cao, and A. K. Jain, “Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge,” *Proceedings International Conf. on Biometrics,* 2018, pp. 9–16.

[31] C. Wan, L. Wang, and V. v. Phoha, “A survey on gait recognition,” *ACM Computing Surveys* Journal, vol. 51, pp 1–35, 2018

[32] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and Harnessing Adversarial Examples,” *3rd International Conf. on Learning Representations,* 2015,p. 6265.

[33] J. Svoboda, A. Anoosheh, C. Osendorfer, and J. Masci, “Two-stage peer-regularized feature recombination for arbitrary image style transfer,” *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2020, pp. 13816–13825.

[34] A. K. Jain, K. Nandakumar, X. Lu, and U. Park, “Integrating faces, fingerprints, and soft biometric traits for user recognition,” *International Workshop on Biometric Authentication Conf*. 2004, pp. 259–269.

[35] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of fingerprint recognition*. New York: Springer Science & Business Media, 2009, pp. 50-66.

[36] A. Jain, A. Ross, and S. Prabhakar, “Fingerprint matching using minutiae and texture features,” *Proceedings 2001 International Conf. on Image Processing (Cat. No. 01CH37205)*, 2001, pp. 282–285.

[37] G. Aggarwal, N. K. Ratha, T.-Y. Jea, and R. M. Bolle, “Gradient-based textural characterization of fingerprints,” *IEEE Second International Conf. on Biometrics*, 2008, pp. 1–5.

[38] S. Chikkerur, S. Pankanti, A. Jea, N. Ratha, and R. Bolle, “Fingerprint representation using localized texture features,” *International Conf. on Pattern Recognition (ICPR’06),* 2006, pp. 521–524.

[39] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, pp. 4–37, 2000.

[40] A. A. A. Youssif, M. U. Chowdhury, S. Ray, and H. Y. Nafaa, "Fingerprint recognition system using hybrid matching techniques," *IEEE/ACIS International Conf. on Computer and Information Science*, 2007, pp. 234–240.

[41] Z. Ouyang, J. Feng, F. Su, and A. Cai, "Fingerprint matching with rotation-descriptor texture features," *International Conf. on Pattern Recognition*, 2006, pp. 417–420.

[42] Z. A. Jhat, A. H. Mir, and S. Rubab, "Fingerprint Texture Feature for Discrimination and Personal Verification," *International Conf. on Emerging Security Information, Systems and Technologies*, 2009, pp. 230–236.

[43] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proceedings of the IEEE*, vol. 83, pp. 705–741, 1995.

[44] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2D and 3D face recognition: A survey," *Pattern recognition letters*, vol. 28, pp. 1885–1906, 2007.

[45] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, pp. 71–86, 1991.

[46] A. L. Yuille, "Deformable templates for face recognition," *Journal of Cognitive Neuroscience*, vol. 3, pp. 59–70, 1991.

[47] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Computer vision and image understanding*, vol. 61, pp. 38–59, 1995.

[48] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisher face: Recognition using class specific linear projection," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, pp. 711–720, 1997.

[49] P. S. Penev and J. J. Atick, “Local feature analysis: A general statistical theory for object representation,” *Network: computation in neural systems*, vol. 7, pp. 477–500, 1996.

[50] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, “Face recognition: A convolutional neural-network approach,” *IEEE transactions on neural networks*, vol. 8, pp. 98–113, 1997.

[51] L. Wiskott, N. Krüger, N. Kuriger, and C. von der Malsburg, “Face recognition by elastic bunch graph matching,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, pp. 775–779, 1997.

[52] G. Baudat and F. Anouar, “Generalized discriminant analysis using a kernel approach,” *Neural computation*, vol. 12, pp. 2385–2404, 2000.

[53] K. I. Kim, K. Jung, and H. J. Kim, “Face recognition using kernel principal component analysis,” *IEEE signal processing letters*, vol. 9, pp. 40–42, 2002.

[54] K. Jonsson, J. Kittler, Y. P. Li, and J. Matas, “Support vector machines for face authentication,” *Image and Vision Computing*, vol. 20, pp. 369–375, 2002.

[55] C. Liu, “Gabor-based kernel PCA with fractional power polynomial models for face recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, pp. 572–581, 2004.

[56] J.-Y. Cortex, J.-T. LaPresté, and M. Richetin, “Face authentification or recognition by profile extraction from range images,” in *Proceedings. Workshop on Interpretation of 3D Scenes Conf.*, 1989, pp. 194–195.

[57] B. Achermann, X. Jiang, and H. Bunke, “Face recognition using range images,” in *Proceedings. International Conf. on Virtual Systems and Multimedia VSMM*, 1997, pp. 129–136.

[58] C. McCool, V. Chandran, S. Sridharan, and C. Fookes, “3D face verification using a free-parts approach,” *Pattern Recognition Letters*, vol. 29, pp. 1190–1196, 2008.

[59] J. G. Daugman, “High confidence visual recognition of persons by a test of statistical independence,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 15, pp. 1148–1161, 1993.

[60] J. Daugman, “How iris recognition works,” in *The essential guide to image processing*, Elsevier, 2nd ed., vol. 2, pp. 715–739, 2009.

[61] P. L. Bowyer *et al.*, “A user’s manual for the Short Child Occupational Profile (SCOPE)(v.2.2),” *Chicago, IL: Model of Human Occupation Clearinghouse*. 2005.

[62] S. R. Ganorkar and A. A. Ghatol, “Iris recognition: An emerging biometric technology,” *Proceedings of the 6th WSEAS International Conf. on Signal Processing, Robotics and Automation*, 2007, pp. 91–96.

[63] W. W. Boles and B. Boashash, “A human identification technique using images of the iris and wavelet transform,” *IEEE transactions on signal processing*, vol. 46, pp. 1185–1188, 1998.

[64] C. Sanchez-Avila and R. Sanchez-Reillo, “Two different approaches for iris recognition using Gabor filters and multiscale zero-crossing representation,” *Pattern Recognition*, vol. 38, pp. 231–240, 2005.

[65] S. Lim, K. Lee, O. Byeon, and T. Kim, “Efficient iris recognition through the improvement of feature vector and classifier,” *ETRI Journal*, vol. 23, pp. 61–70, 2001.

[66] G. Park and S. Kim, “Hand biometric recognition based on fused hand geometry and vascular patterns,” *Sensors*, vol. 13, pp. 2895–2910, 2013.

[67] A. Kumar, D. Wong, H. C. Shen, and A. K. Jain, “Personal verification using palmprint and hand geometry biometric,” *International Conf. on audio-and video-based biometric person authentication*, 2003, pp. 668–678.

[68] L. Ma, T. Tan, Y. Wang, and D. Zhang, “Personal identification based on iris texture analysis,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, pp. 1519–1533, 2003.

[69] L. Ma, T. Tan, Y. Wang, and D. Zhang, "Efficient iris recognition by characterizing key local variations," *IEEE Transactions on Image processing*, vol. 13, pp. 739–750, 2004.

[70] Y. Chen, S. C. Dass, and A. K. Jain, "Localized iris image quality using 2-D wavelets," in *International Conf. on biometrics*, 2006, pp. 373–381.

[71] Y. Du, R. Ives, D. M. Etter, and T. Welch, "Use of one-dimensional iris signatures to rank iris pattern similarities," *Optical Engineering*, vol. 45, p. 037201, 2006.

[72] T. Sun, A. Serrano, D. Gutierrez, and B. Masia, "Attribute-preserving gamut mapping of measured BRDFs," *Computer graphics forum Conf.*, 2017, pp. 47–54.

[73] W. Shu and D. Zhang, "Palmprint verification: an implementation of biometric technology," in *Proceedings. Fourteenth International Conf. on Pattern Recognition*, 1998, pp. 219–221.

[74] L. Liu and D. Zhang, "Palm-line detection," in *IEEE International Conf. on Image Processing*, 2005, pp. 269–272.

[75] L. Liu, D. Zhang, and J. You, "Detecting wide lines using isotropic nonlinear filtering," *IEEE Transactions on image processing*, vol. 16, pp. 1584–1595, 2007.

[76] A. Kong, D. Zhang, and M. Kamel, "Palmprint identification using feature-level fusion," *Pattern Recognition*, vol. 39, pp. 478–487, 2006.

[77] Q. Liu, R. Huang, H. Lu, and S. Ma, "Face recognition using kernel-based fisher discriminant analysis," *Proceedings of Fifth IEEE International Conf. on Automatic Face Gesture Recognition*, 2002, pp. 197–201.

[78] D. Hu, G. Feng, and Z. Zhou, "Two-dimensional locality preserving projections (2DLPP) with its application to palmprint recognition," *Pattern Recognition*, vol. 40, pp. 339–342, 2007.

[79] C.-C. Han, H.-L. Cheng, C.-L. Lin, and K.-C. Fan, “Personal authentication using palm-print features,” *Pattern Recognition*, vol. 36, pp. 371–381, 2003.

[80] A.-K. Kong and D. Zhang, “Competitive coding scheme for palmprint verification,” *Proceedings of the 17th International Conf. on Pattern Recognition*, 2004, pp. 520–523.

[81] L. Zhang and D. Zhang, “Characterization of palmprints by wavelet signatures via directional context modeling,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, pp. 1335–1347, 2004.

[82] C.-L. Lin, T. C. Chuang, and K.-C. Fan, “Palmprint verification using hierarchical decomposition,” *Pattern Recognition*, vol. 38, pp. 2639–2652, 2005.

[83] T. Wu, Y. Yang, and Z. Wu, “Improving speaker recognition by training on emotion-added models,” in *International Conf. on Affective Computing and Intelligent Interaction*, 2005, pp. 382–389.

[84] X. Wu, D. Zhang, and K. Wang, “Fisherpalms based palmprint recognition,” *Pattern recognition letters*, vol. 24, pp. 2829–2838, 2003.

[85] B. Wu, H. Ai, and C. Huang, “Real-time gender classification,” in *Third International Symposium on Multispectral Image Processing and Pattern Recognition Conf.*, 2003, pp. 498–503.

[86] C. Oden, A. Ercil, and B. Buke, “Combining implicit polynomials and geometric features for hand recognition,” *Pattern Recognition Letters*, vol.13, pp. 2145–2152, 2003.

[87] M. Golfarelli, D. Maio, and D. Malton, “On the error-reject trade-off in biometric verification systems,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 786–796, 1997.

[88] J.-W. JUNG, Z. BIEN, and T. SATO, “Biometrics-Personal Identification in Networked Society Biometrics-Personal Identification in Networked Society, 1998,” *IEICE transactions on fundamentals of electronics, communications, and computer sciences*, vol. 87, pp. 1393–1400, 2004.

[89] P. K. Rudravaram and V. Govindaraju, “Peg-Free Hand Geometry Verification System,” International Journal of Engineering Science and Technology vol. 4, pp. 2942-2949, 2012.

[90] A. Ross, A. Jain, and S. Pankati, “A prototype hand geometry-based verification system,” *Proceedings of 2nd Conf. on audio and video based biometric person authentication*, 1999, pp. 166–171.

[91] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos, “Biometric identification through hand geometry measurements,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, pp. 1168–1171, 2000.

[92] Y. Bulatov, S. Jambawalikar, P. Kumar, and S. Sethia, “Hand recognition using geometric classifiers,” *International Conf. on Biometric Authentication*, 2004, pp. 753–759.

[93] A. K. Jain and N. Duta, “Deformable matching of hand shapes for user verification,” *Proceedings in International Conf. on Image Processing*, 1999, pp. 857–861.

[94] E. Yörük, H. Dutağaci, and B. Sankur, “Hand biometrics,” *Image and vision computing*, vol. 24, pp. 483–497, 2006.

[95] S. Z. Li and A. Jain, *Encyclopedia of biometrics*. New York: Springer, 2015., pp. 672–679.

[96] A. K. Jain, K. Nandakumar, and A. Nagar, “Biometric template security,” *EURASIP Journal on advances in signal processing*, vol. 2008, pp. 1–17, 2008.

[97] B. Miller, “Vital signs of identity [biometrics],” *IEEE Spectrum*, vol. 31, pp. 22–30, 1994.

[98] K. Přihodová and M. Hub, “Biometric Privacy through Hand Geometry-A Survey,” *International Conf. on Information and Digital Technologies (IDT)*, 2019, pp. 395–401.

[99] J. P. Holmes, R. L. Maxwell, and L. J. Wright, "A performance evaluation of biometric identification devices," *Institute of nuclear materials management Conf.*,1990, pp.1-14.

[100] H. Dutagaci, B. Sankur, and E. Yörük, "Comparative analysis of global hand appearance-based person recognition," *Journal of electronic imaging*, vol. 17, p. 011018, 2008.

[101] N. Kingsbury, *"Technology Assessment: Using Biometrics for Border Security."* United States: Diane Publishing Co: 2003, pp. 84-104.

[102] T. Starner and A. Pentland, *"Real-time American sign language recognition from video using hidden Markov models,"* New York: Springer, 1997, pp. 227–243.

[103] B. Zhang, C. Quan, and F. Ren, "Study on CNN in the recognition of emotion in audio and images," *IEEE/ACIS 15th International Conf. on Computer and Information Science (ICIS)*, 2016, pp. 1–5.

[104] T. Koizumi, M. Mori, S. Taniguchi, and M. Maruya, "Recurrent neural networks for phoneme recognition," *Proceeding of Fourth International Conf on Spoken Language Processing. ICSLP'96*, 1996, pp. 326–329.

[105] R. Han *et al.*, "An assessment of multimodel simulations for the variability of western North Pacific tropical cyclones and its association with ENSO," *Journal of Climate*, vol. 29, pp. 6401–6423, 2016.

[106] P. Kaew, *"An improved adaptive background mixture model for real-time tracking with shadow detection,"* New York: Springer, 2001, pp. 135-144.

[107] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, pp. 583–596, 2014.

[108] A. K. Jain, P. Flynn, and A. A. Ross, *Handbook of biometrics*. New York: Springer, 2007, pp. 91-107.

[109] A. Martin, G. Doddington, T. Kamm, M. Orlowski, and M. Przybocki, "The DET curve in assessment of detection task performance," *Proceeding* EUROSPEECH *Conf*,1997, pp.1-4.

[110] K. Fakhar, M. el Aroussi, M. N. Saidi, and D. Aboutajdine, "Fuzzy pattern recognition-based approach to biometric score fusion problem," *Fuzzy Sets and Systems*, vol. 305, pp. 149–159, 2016.

[111] J. P. Egan, *"Signal detection theory and ROC analysis "* New York: Academic Press, 1975, pp.80 -120.

[112] Y. Wu, K. Ianakiev, and V. Govindaraju, "Improved k-nearest neighbor classification," *Pattern Recognition*, vol. 35, pp. 2311–2318, 2002.

[113] A. Bordes, S. Ertekin, J. Weston, L. Botton, and N. Cristianini, "Fast kernel classifiers with online and active learning.," *Journal of Machine Learning Research*, vol. 6, pp. 1579–1619, 2005.

[114] I. Sucholutsky and M. Schonlau, "Less than one-shot learning: Learning N classes from M< N samples*," ICLR Conf.*, 2020, pp. 60–80.

[115] O. O. V. Villegas, H. de J. O. Domínguez, V. G. C. Sánchez, L. O. Maynez, and H. M. Orozco, "Biometric human identification of hand geometry features using discrete wavelet transform*," astes Journal* , vol. 5, pp. 689-698, 2020.

[116] M. Firas and S. Zainab, "A new features extracted for recognizing a hand geometry using BPNN," *Int. Journal Sci. Eng. Res*, vol. 5, pp. 232–237, 2014.

[117] R. Gross, Y. Li, L. Sweeney, X. Jiang, W. Xu, and D. Yurovsky, "Robust hand geometry measurements for person identification using active appearance models," in *2007 First IEEE international Conf.*

on biometrics: Theory, applications, and systems, 2007, pp. 1–6.

[118] S. Prabu, M. Lakshmanan, and V. N. Mohammed, “A multimodal authentication for biometric recognition system using intelligent hybrid fusion techniques,” *Journal of medical systems*, vol. 43, pp. 1–9, 2019.

[119] S. A. Angadi and S. M. Hatture, “Biometric person identification system: a multimodal approach employing spectral graph characteristics of hand geometry and palmprint,” *International Journal of Intelligent Systems and Applications*, vol. 8, p. 48, 2016.

[120] X. Liu, K. W. Bowyer, and P. J. Flynn, “Experiments with an improved iris segmentation algorithm,” in *Fourth IEEE Workshop on Automatic Identification Advanced Technologies (AutoID’05)*, 2005, pp. 118–123.

[121] D. M. Monro, S. Rakshit, and D. Zhang, “DCT-based iris recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, pp. 586–595, 2007.

[122] H. Cho, R. Roberts, B. Jung, O. Choi, and S. Moon, “An efficient hybrid face recognition algorithm using PCA and GABOR wavelets,” *International Journal of Advanced Robotic Systems*, vol. 11, p. 59, 2014.

[123] M. Sharif, J. H. Shah, S. Mohsin, and M. Raza, “Subholistic hidden Markov model for face recognition,” *Research Journal of Recent Sciences*, vol. 2277, p. 2502, 2013.

[124] N. A. Mngenge, L. Mthembu, F. v Nelwamondo, and C. H. Ngejane, “A fingerprint indexing approach using multiple similarity measures and spectral clustering,” *12th Conf. on Computer and Robot Vision*, 2015, pp. 208–213.

[125] A. K. Jain and J. Feng, “Latent fingerprint matching,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 33, pp. 88–100, 2010.