

ECG BIOMETRIC AUTHENTICATION: A COMPARATIVE ANALYSIS

*A Project report submitted in partial fulfilment of the requirements for the
award of the degree of*

**BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING**

Submitted by

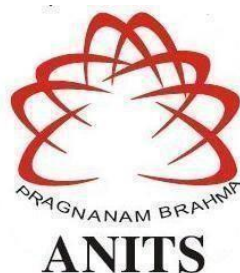
V.Charan Kumar (319126512185)

L.Vijay Kumar (319126512159)

B.Sowmya (319126512137)

S.Prasanth Kumar (319126512180)

**Under the guidance of
Prof. G. Manmadha Rao
M.E,Ph.D
Professor, Department of ECE
& Head - Training & Placements**



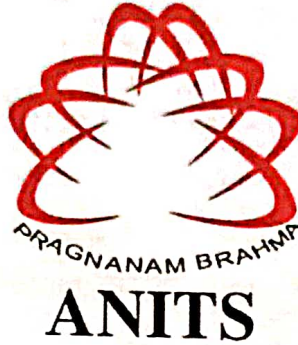
**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)**

*(Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC)
Sangivalasa, Bheemili Mandal, Visakhapatnam Dist. (A.P)
2022-2023*

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)

(Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA &
NAAC with 'A' Grade)

Sangivalasa, Bheemili Mandal, Visakhapatnam Dist.(A.P)



CERTIFICATE

This is to certify that the industrial training report entitled "ECG BIOMETRIC AUTHENTICATION: A COMPARATIVE ANALYSIS" submitted by V.Charan Kumar (Regd No:319126512185),L.Vijay Kumar (Regd No:319126512159),B.Sowmya(Regd No:319126512137),S.Prasanth Kumar(Regd No:319126512180) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Electronics & Communication Engineering is a record of bonafide work carried out under my supervision.

Project Guide

Prof.G.Manmadha Rao
Professor,Department of ECE
& Head – Training & Placements
ANITS

Professor
Department of E.C.E.
Anil Neerukonda
Institute of Technology & Sciences
Sangivalasa, Visakhapatnam-5311

Head of the Department

Prof.B.Jagadeesh
Professor & HOD
Department of ECE
ANITS

Head of the Department
Department of E C E
Anil Neerukonda Institute of Technology & Sciences
Sangivalasa - 531 162

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our project guide **Prof. G. Manmadha Rao** Professor, Department of Electronics and Communication Engineering, ANITS, for his guidance with unsurpassed knowledge and immense encouragement. We are grateful to **Prof. B. Jagadeesh**, Head of the Department, Electronics and Communication Engineering, for providing us with the required facilities for the completion of the project work.

We are very much thankful to the **Principal and Management, ANITS, Sangivalasa**, for their encouragement and cooperation to carry out this work.

We express our thanks to all **teaching faculty** of Department of ECE, whose suggestions during reviews helped us in accomplishment of our project. We would like to thank **all non teaching staff** of the Department of ECE, ANITS for providing great assistance in accomplishment of our project.

We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

PROJECT STUDENTS

V.Charan Kumar (319126512185)

L.Vijay Kumar (319126512159)

B.Sowmya (319126512137)

S.Prashanth (319126512180)

Abstract

The main objective of this study is to use ECG signals for user identity verification. An electrocardiogram is an electrical impulse signal that indicates different heart conditions, and ECG has exceptional discriminative qualities associated with human identification. Despite recent progress in ECG-based encryption, differential feature extraction and efficient pattern clustering still face several difficulties. In the current environment, attestation techniques have become a crucial and urgent responsibility to secure the confidentiality of equipment and sensitive information. ECG biometric authentication is the suggested authentication technique that can give users access. ECG biometric systems typically consist of basic processes: sensing, pre-processing (filtering, segmenting), feature extraction, and matching. The paragraph provides an overall description and flow of the enrollment and authentication stages of the ECG-based biometric authentication procedure. Our analysis shows that our approach to ECG biometric identification outperforms existing techniques that lack the ability to quickly extract features, filter, segment, and match.

Keywords:electrical impulse signal, encryption, filtering, segmentation, feature extraction, enrollment, authentication.

CONTENTS

ABSTRACT	iv
LIST OF FIGURES	vii
LIST OF SYMBOLS	ix
LIST OF ABBREVIATIONS	x
CHAPTER 1: INTRODUCTION.....	1
1.1 Introduction to ECG based Biometric Authentication:	1
1.2 What is Data acquisition?	1
1.3 Background on ECG	4
CHAPTER 2: BIOMETRIC LIVENESS DETECTION	5
2.1 What is Biometric Liveness Detection?.....	5
2.2 Types of Attacks:	5
2.3 Spoofing Attacks:.....	5
2.4 Face Spoofing:	6
2.5 Fingerprint Spoofing:	7
2.7 Counter measures:	9
CHAPTER 3: EXISTING SYSTEM.....	11
3.1 Feature Extraction	11
3.2 Classification Category	12
3.2.1 Support Vector Machine Algorithm(SVM):	12
3.3 Disadvantages:	19
CHAPTER 4: PROPOSED METHOD.....	20
4.1 Pre-Processing.....	20
4.1.1 Filtration:	20
4.2. Feature Extraction	22
4.2.1 Fiducial Points Based Feature Extraction	23

4.2.2 Non-Fiducial Features Extraction:	23
4.3.1 Euclidean Distance.....	24
4.3.2 Dynamic Time Warping :	24
CHAPTER 5: MATLAB	33
5.1 Introduction To Matlab	33
5.1.1 What is MATLAB?.....	33
5.2 MATLAB Working Environment:.....	35
CHAPTER 6: RESULTS AND DISCUSSIONS	47
CONCLUSION.....	51
References:.....	52
PAPER PUBLICATION DETAILS.....	54

LIST OF FIGURES

- Fig 1 ECG Waveform
- Fig 2.1 (a) In Brazil, doctors used fake silicone fingers to defraud a hospital's biometric punch-in clock.
(b) In Hong Kong, a young passenger boarded a plane while wearing an old-man mask and arrived in Canada to claim asylum.
(c) In New York, an African American man wore a silicone mask to disguise himself as white police officer to rob a check-cashing store.
- Fig 2.2 (a) a photograph
(b) a video
(c) a 3D mask
(d) a sketch
(e) a reverse-engineered face image (f)makeup
(g) excessive plastic surgery
- Fig 2.3 Spoofing attacks based on hardware and software methods
- Fig 3.1 Diagram where two different classes that are divided using hyperplane
- Fig 3.2 Linear SVM
- Fig 3.3 Classes are separated using multiple lines in Linear SVM
- Fig 3.4 Classes are separated using minimal hyper plane in Linear-SVM
- Fig 3.5 Non Linear SVM
- Fig 3.6 3D Sample Space
- Fig 3.7 Datasets are divided using SVM
- Fig 3.8 2D Sample Space
- Fig 4.1 Segment of ECG signal
- Fig 4.2 a single ECG pulse with identifying characteristics

Fig 4.3	KNN Classification of input value
Fig 4.4	KNN Algorithm
Fig 4.5	(a), (b) ,(c) Working of KNN algorithm
Fig 5.1	MATLAB function(left) and Simulink Model(right) with scatter plot for 16 QAM simulation.
Fig 5.2	Simulink model of an adaptive MIMO system with orthogonal space time block codes(OSTBC)
Fig 5.3	An ideal 16 QAM plot(left) impaired by a offset (middle) and a frequency offset (right).
Fig 6.1	Input ECG signal1
Fig 6.2	Input ECG signal2
Fig 6.3	Filtered signal
Fig 6.4	Segment of ECG signal
Fig 6.5	Feature extraction using symlet and daubechies wavelet
Fig 6.6	Verified Output
Fig 6.7	Accuracy comparison

LIST OF SYMBOLS

x, y	Independent variables
z	Dependent variable
l	length
z_t	Original ECG signal at time instant t
y_t	Filtered ECG signal at time instant t
$f(\cdot)$	State evolution function
$g(\cdot)$	Relation between state vector and the observations
w_t	Measured noise vector with covariance matrices $Q_t = E\{w_t w_t^T\}$
v_t	Measured noise vector with covariance matrices $R_t = E\{v_t v_t^T\}$
r_t	Innovation signal
$H(z)$	Transfer function
r_i, r_{i+1}	RR segmentation
D	Euclidean distance
T_i, q_i	Feature Vectors
x_q	Query feature set

LIST OF ABBREVIATIONS

LNA	Low-noise Amplifier
ECG	Electrocardiogram
FFR	False Reject Rate
EER	Equal Error Rate
FAR	False accept Rate
SA	Senatorial
AV	Atrioventricular
PAD	Presentation attack detection
AC	Auto-correlation
DCT	Discrete cosine transform
SVM	Support vector machines
DTW	Dynamic time warping
LSTM	Long short-term memory
RBNN	Radial Basis Function Neural Network
MLP	Multilayer Perceptron
EMG	Electromyogram
KNN	K-Nearest Neighbours Algorithm
OFDM	Orthogonal Frequency Division Multiplexing
MIMO	Multiple Input Multiple Output

CHAPTER-1

INTRODUCTION

1.1 Introduction to ECG based Biometric Authentication:

Devices for biomedicine are gaining popularity. This is a result of the integrated circuit (IC) fabrication industry's rapid progress. These gadgets are utilised as monitoring equipment as well as wearable or implantable devices. The bio signal is first preconditioned and transformed to digital in each of these applications. The digital data is subsequently processed for monitoring or diagnosing applications by a digital signal processor. Biomedical signal acquisition systems frequently contain a band pass filter, low-noise amplifier (LNA), an analogue sample-and-hold, and an analogue-to-digital converter.

The dynamic range requirements can be relaxed and the supply voltage can be kept to a minimum by removing the interferences at the system's input before significant gain is applied [2]. Due to their importance for implantable and multi-electrode systems, this can result in a decrease in both overall power consumption and area. In order to do this, digital blocks and mixed signal feedback are used. So, it would seem that using digital methodologies to create these systems could result in improved performance and better CMOS technology compatibility.

1.2 What is Data acquisition?

The process of sampling signals that characterize actual physical situations and turning the resulting samples into digital numeric values that a computer can work with is known as data acquisition [2]. Commonly referred to as DAS, data acquisition systems convert analogue waveforms into digital values for processing. An electrocardiogram (ECG) is a depictive recording of the heart's electrical activity over time. It is typically utilized for illness analysis by interpreting the biological signal since it is usually identified by that signal and has a non-invasive manner. Cardiovascular illnesses and their anomalies alter the ECG wave shape[2] Each part of the ECG waveform includes information that is important to the clinician in making a good diagnosis, as illustrated in Fig.1.

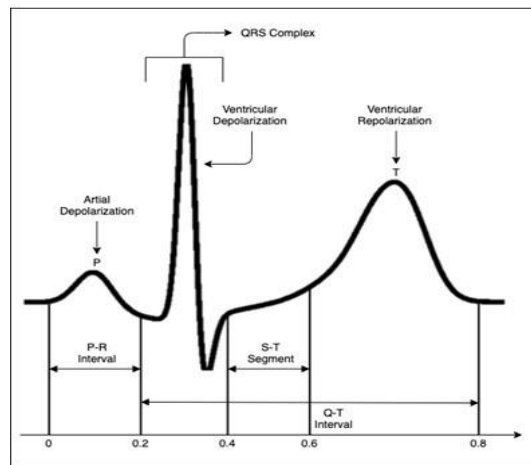
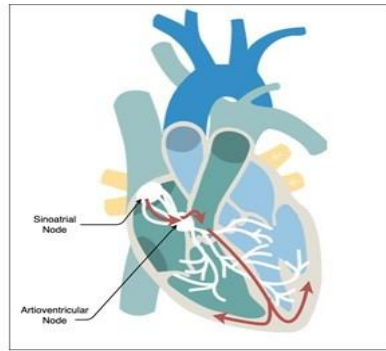


Fig. 1: ECG Waveform

The purpose of the ECG sensor is to measure the impact of human heart sounds. ECG signal, which is translated into electrical signal, should be based on the modulation circuit filter and amplification since the signal is typically erratic, accompanied by noise. For the evocation of information, a signal capture still requires hardware and software instrumentation, noise or other uniqueness filtering, and handing out [2]. In the great majority of systems, signal processing is used for ECG analysis and interpretation. It is employed to extract a number of defining factors. Biomedical signal processing is now focused on using signal analysis to quantitatively or objectively analyse physiological systems and events. The increase of measurement accuracy and repeatability is one of the many goals of ECG signal processing.

Despite the significant effort put into establishing the ECG as a biometric modality, it has not yet attained a level of technological maturity and acceptance that is acceptable [1]. The immaturity of ECG development is caused by a lack of dependable ECG data. Also, the exploration community is counting on a small ECG gallery that has good performance but

significant error rates. [1] The bulk of the current methodologies generally fail to give standard metrics for analysing the issues of ECG data assessment in order to balance a number of important evaluation criteria , similar as the false acceptance rate, false reject rate, and equal error rate. However, the major processes in creating the ideal algorithm for every application—filtering, segmentation, feature extraction, and matching—have not yet been thoroughly reviewed in most techniques.

The largest off-the-person ECG datasets will be used for the first time in this research, and we'll also learn about and analyze the efficiency of various approaches at various stages of ECG biometric systems. This paper offers a novel method of filtering to produce a bio template for biometric identification in addition to providing a full analysis of existing well-known techniques utilizing a number of ECG data sets, including our sizeable new data set. We present our findings to the ECG-based research community through extensive experiments and careful literature reviews. The following is a description of our primary contributions:

- We conduct a comprehensive analysis and debate based on the types of databases, the quantity of subjects, the subjects' states of health, metrics, and technique to summarise the current approaches from the literature on identity recognition systems based on the ECG.
- We look at several feature extraction, filtering, segmentation, and matching methods in order to provide a comprehensive analysis on ECG. Each of these performance techniques can be found on several ECG databases that include information on different medical issues. Our metrics also provide enough coverage for the important assessment metrics, including the false accept rate (FAR), false reject rate (FRR), equal error rate (EER), and identification rate.
- The extensive off-the-person ECG database is first made available. A total of 1,119 patients contributed 68,274 samples of ECG recordings to the new ECG database.
- Several tests were conducted utilising the ECGs of 1,119 different people to show the efficacy of the suggested plan. The result shows that, given the corresponding values of 1.2%, 1.48%, and 0% for EER, FAR, and FRR, we were successful in reaching 100% accuracy.
- The impact of segmentation is observed in ECG biometric authentication. We identified the ideal segment and the amount of samples each cycle to achieve high accuracy.

1.3 BACKGROUND ON ECG

A visual representation of the electrical activity of the heart using electrodes placed on the body's surface is called an electrocardiogram (ECG). The senatorial (SA) and atrioventricular (AV) nodes produce multiple action potentials, which are combined to form the ECG. The main distinguishing feature of a typical ECG is the PQRST complex, which is composed of the P, Q, R, S, and T waves. The depolarization wave travelling from the SA node through the atria node causes the P wave, which is the smallest and has the shortest duration (0.08 seconds). Ventricular depolarization leads to the much bigger QRS complex, which occurs before ventricular contraction. The T wave, which results from ventricular repolarization, typically lasts 0.16 seconds. Each person's ECG waveform has a unique pattern depending on a variety of factors, including size, shape, placement inside the chest, and conductive characteristics of the torso.

ECG recording setups can be either "on-the-person" or "off-the-person". Body sensors that require conductive paste or gel to be applied to the subject's surface are used in on-the-person recording to acquire precise measurements. This technique is mostly employed in medical apparatus to get ECG signals from the chest. This method is quite invasive for individuals while being a highly efficient diagnostic tool, which restricts the potential industrial uses of ECG-based biometrics. The majority of biometric research has demonstrated the applicability of ECG signals obtained via a "on-the-person" method. On the other hand, off-the-person recording does not require the subject to be carefully prepared with objects or surfaces because it uses technology to measure ECG signals. This technique can utilize devices with dry electrodes, such as fingertips or wearable devices. The fact that this method doesn't need the user's involvement for sensor placement is one of its main benefits.

CHAPTER-2

Biometric Liveness Detection

2.1 What is Biometric Liveness Detection?

The term "biometrics" refers to the identification of individuals based on their physical or behavioral characteristics, derived from the Greek words "bios" (life) and "metrikos" (measurement). For identifying purposes, biometrics largely rely on "who you are" rather than "what you have" (such as an ID i. card) or "what you know" (such i. as a password). Biometric systems are commonly used in government-mandated processes such as border crossings (as shown in Figure 1) and in business operations such as banks. Nowadays, many mobile i. devices use i. biometrics as a means of authentication rather than passwords, with fingerprint recognition being utilized by the iPhone 5s and facial recognition being used by Android phones. Physiological or behavioural biometric signals are captured by a biometric system, which works as a pattern recognition system,extractsi. a set of distinctive features from this signal, comparesti these features with the i.feature set (template) stored i. in a database, and ultimately determines the identification of the individual providing the biometric signal.

2.2 Types of Attacks:

In the context of biometric systems, attacks can be categorized into two types: direct attacks or presentation attacks, and indirect attacks. Presentation threats happen when an attacker purposefully presents an artefacti. or human trait to the biometric capture subsystem. in an effort to interfere with the operation of the system. These attacks take place at the external sensor level and do not utilize digital security measures such as hashing, encryption, or digital signatures. In contrast, indirect attacks occur when attackers, such as hackers or cybercriminals, circumvent the feature extractor to carry out attacks within the system itself.

2.3 Spoofing Attacks:

Presentation attacks are becoming more likely as a result of the use of biometric technologies. These attacks involve fraudsters impersonating real users by copying their biometrics, enabling them to gain unauthorized access and benefits. Spoofing attacks can be

carried out by a wide range of potential attackers without requiring advanced technical expertise. The US National Institute of Standards and Technology's National Vulnerability Database now includes a section on the spoofability of biometrics. Direct assaults can be carried out on various biometric types, including the face, iris, and fingerprints. Spoofing attacks are a significant problem for businesses that market identity management systems based on biometrics. Examples of such attacks include doctors at a hospital in Brazil using fake silicone fingers to claim overtime pay[1]. Despite improvements in mobile device biometrics, their vulnerability to spoofing attacks has been largely overlooked. In 2013, the German hacker collective Chaos Computer Club demonstrated how a fake fingerprint could trick the iPhone 5s's fingerprint reader.

2.4 Face Spoofing:

Face spoofing continues to be a severe danger to face recognition systems despite improvements in its detection. These systems might be fooled by a picture, a video, a 3D face model (mask), a drawing, a reverse-engineered face image made from the real user's face, or by using cosmetics or plastic surgery to make an imposter appear to be the real user. The two broad categories of face liveness detection techniques are motion analysis and texture analysis [1]. When an attacker provides the system with 2D counterfeits, such as photos or films, motion analysis recognises the spontaneous movement hints that are created [1]. For instance, one study team proposed eye blink-based liveness detection for photo-based spoofing, based on the fact that people blink every 2 to 4 seconds.

Face spoofing continues to be a serious danger to facial recognition systems despite improvements in face spoofing detection. These systems can be deceived by various counterfeits, such as photos, videos, 3D face models (masks), drawings, or even through the use of cosmetics or plastic surgery to resemble the real user. There are two main categories of face liveness detection techniques: motion analysis and texture analysis. Motion analysis identifies spontaneous movement cues generated when an attacker provides 2D counterfeits like photos or videos [1]. For example, some researchers proposed liveness detection based on eye blinks, which occur every 2 to 4 seconds. Texture analysis looks at skin characteristics like texture and reflectivity to differentiate between genuine faces and prints. Microtexture analysis has been used to identify printed picture spoofs. Some techniques use multispectral imaging, which evaluates the reflectance of object surfaces, as a basis for further protection against spoofing attacks. Other researchers have employed Lambertian and

variational Retinex reflectance models or scene context matching and optical flow[2]. Fourier spectra analysis techniques have also been used but are less effective with higher-quality data.

2.5 Fingerprint Spoofing:

Fingerprint spoofing is an old trick used to deceive fingerprint recognition systems. A 2D fake fingerprint, a 3D synthetic fingerprint, a reverse-engineered [1] picture of a user's actual fingerprint, a cadaver or severed finger of the legitimate person can all be used to fool the system. Even the use of easily accessible materials like gelatin can create fake fingerprints.

Consensual and non-consensual approaches are used to create forged fingerprints. In consensual approaches, the user allows the creation of fake fingerprints using their actual fingers. In contrast, the non-consensual method uses the user's latent fingermarks to produce the forged prints without their involvement. The detection of liveness in fingerprints can be based on several factors, such as pore size, skin deformation, sweat, and image quality.

Sweat pattern analysis can be used to detect fake fingerprints by tracking the sweating process of a finger. Skin elasticity is used in methods based on skin deformation, which requires specialised expertise. Texture attributes or picture quality are used in image quality-based approaches. Local phase quantization texture properties can be used for determining liveness. Pores are also used as an indicator of a fingerprint's life, and high-pass filters and correlation techniques are frequently employed in these methods.

2.6 Iris Spoofing :

Iris recognition is considered the most precise method of individual identification. Nonetheless, iris photos, movies, printed contact lenses, glass or plastic prosthetic eyes, reverse-engineered iris pictures based on the genuine user's iris, and real eyes removed from the authentic user's body can fool iris identification systems [1]. Frequency spectrum analysis, reflectance analysis, dynamics analysis, and texture analysis are some of the techniques used to determine if an iris is alive [1]. Although reflectance analysis employs various light wavelengths to illuminate the eye and measure how the sclera and iris respond differently, frequency spectrum analysis believes that spoofing efforts will have frequency irregularities.

Dynamics analysis collects many photos while adjusting the light intensity to measure changes in pupil dilation. Texture analysis categorizes texture characteristics to

identify counterfeit. Researchers have also developed techniques for detecting false iris using Purkinje pictures based on a human eye model, although this method often fails for contact lenses. For instance[1], one team examined the statistical texture of four different variables to identify fake contact lenses. Another team utilized motion features with picture quality characteristics brought on by either the iris's motion or the motion of the sensor to detect counterfeit

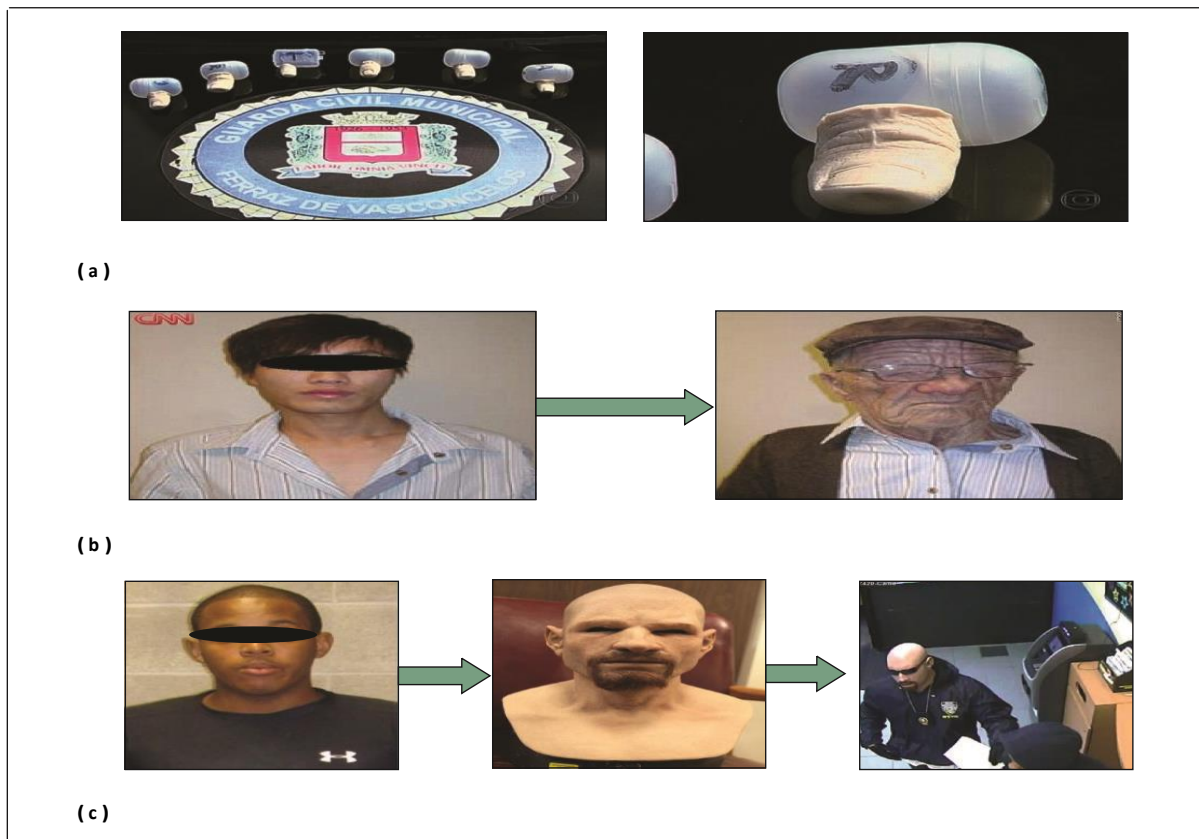


Figure 2.1:Examples of spoofing attacks. (a) In Brazil, medical professionals tricked a hospital's biometric punch-in clock by using phoney silicone fingers. (b) A young traveller from Hong Kong flew to Canada while hiding his identity by donning an old-man disguise. (Reproduced with permission from CBS News.) (c) An African American guy robbed a check-cashing business in New York while pretending to be a white police officer by donning a silicone mask. Used with permission from Composite Effects/CFX as a source.

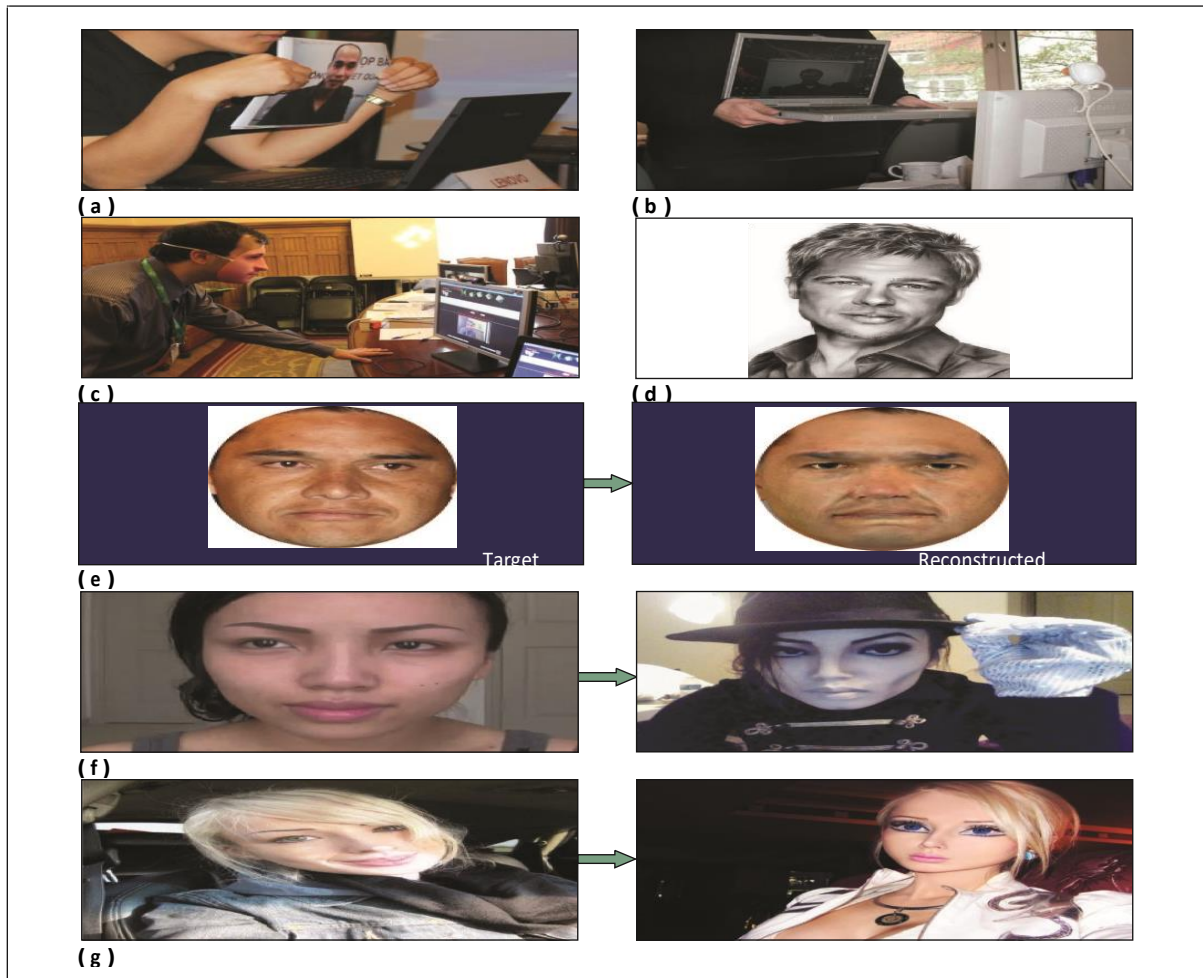


Figure 2.2 Face spoofing can be achieved using (a) a photograph, (b) a video, (c) a 3D mask (Source: Sébastien Marcel; used with permission), (d) a sketch, (e) a reverse-engineered face image, (f) makeup, and (g) excessive plastic surgery (here, to look like a doll).

2.7 Counter measures:

In order for biometric applications to be effective, it is crucial to quickly address the threat presented by presentation attack. One approach to defend against such attacks is to incorporate a presentation attack detection (PAD) mechanism into the system. Liveness detection is a PAD technique that assumes a hacker attempting to use fake biometric information to impersonate a genuine user is attempting to breach the system. This article uses the terms PAD and liveness detection interchangeably, and PAD operates in four different modes as illustrated in Figure 2.3.

- using available sensors to detect patterns characteristic of attacks in the signal,
- employing specialised sensors to find authenticity proof (which isn't always feasible),

- utilising user interaction-based challenge-response mechanisms, and
- using recognition methods that are inherently resistant to presentation threats.

While many people believe that multimodal biometric systems are less vulnerable to fraudulent activities, recent research has indicated that attackers can deceive such systems by falsifying even one biometric feature. Liveness detection can be implemented either through software or hardware, with the latter being more expensive and less invasive. The most commonly used biometric features such as face, fingerprint, and iris, along with their corresponding software-based liveness detection methods [1], are discussed below.

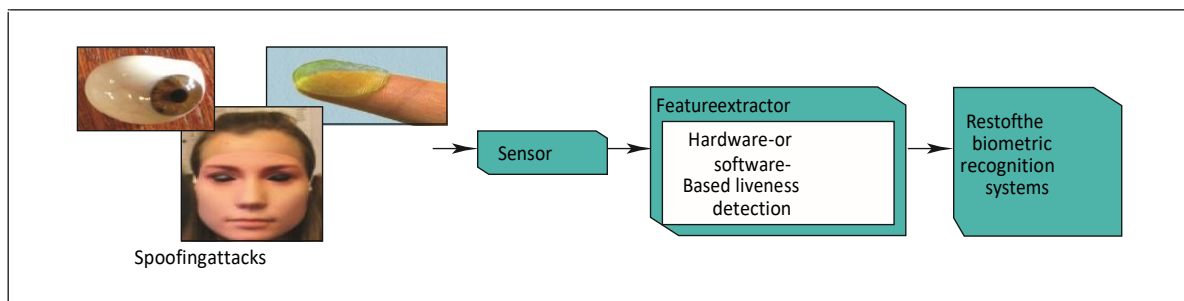


Figure 2.3. Liveness detection techniques based on hardware or software may be able to identify spoofing attacks. Software-based approaches make use of signal-processing techniques, making them affordable and non-intrusive.

CHAPTER-3

Existing system

This section covers recent developments in ECG biometrics, focusing on key aspects which are how many records, types of ECG-data sources, filtering techniques, segmentation methods, and feature extraction approaches. The section also highlights specific evaluation metrics, including identification-rate, false rejection rate, false acceptance rate, and EER, which are summarized in tables to aid in method selection for various applications. The past decade has seen the emergence of several manually built feature extraction methods, including fiducial point feature extraction wavelet transform, AC, and DCT, for ECG based biometric authentication and verification. Additionally, researchers have explored various public records, including on- and off the person datasets with normal and unhealthy ECG data. Finally, the section presents a summary of the methodologies used for ECG analysis, based on feature extraction and classification schemes.

3.1 FEATURE EXTRACTION

1) MANUALLY BUILT FEATURE EXTRACTION METHODS:

Manually built feature extraction for ECG biometrics can be classified into two categories: fiducial and non-fiducial. Fiducial based features are depended on specific local characteristics of ECG beats, such as onset, offset and peak are taken from individual beats or segments. Examples of fiducial-features include the P Q R S T peaks, the interval between the Q to T peaks as well as the Q-T interval. Different subsets of these features have been used in various studies. In contrast, non-fiducial feature extraction uses time and frequency analysis to extract statistical data from the complete ECG waveform without the need of fiducial points. This approach is used to capture interpreting data gleaned from the ECG waveform comprehensively. Examples of non-fiducial features include autocorrelation, discrete cosine transform, and wavelet transform.

2) ALGORITHMS BASED ON NON-MANUALLY BUILT FEATURE EXTRACTION METHODS:

Most manually built feature extraction methods involve a preprocessing stage that prepares ECG data, like extracting fiducial or nonfiducial features through statistical

analysis. However, deep learning has emerged as a promising alternative for feature extraction, as it can achieve better performance and robustness without relying on handcrafted features. Handcrafted approaches require separate steps for feature transformation and noise removal, which can result in suboptimal performance. In contrast, deep learning methods can bypass these restrictions and achieve better performance by learning features directly from the raw ECG data. Therefore, researchers have begun exploring the use of deep learning for ECG biometrics to improve performance and overcome the limitations of handcrafted feature extraction.

3.2 CLASSIFICATION CATEGORY

ECG biometric systems can be categorized into two types: identification and verification. As both types have distinct purposes, it is important to differentiate between them. Therefore, we will discuss the ECG biometric systems in these two separate categories to provide a clear understanding of their respective characteristics and applications.

1) ECG VERIFICATION

The majority of verification algorithms assess the degree of similarity or dissimilarity between a query feature vector and a template using matching scores. The matching scores are then compared to a preset threshold during authentication. The stated identification is accepted if the score exceeds the cutoff. Various classifiers are used in this manner, such as svm's, Euclidean distance, hamming distance, and dynamic time warping, among others, which have been extensively studied in the literature.

3.2.1 Support Vector Machine Algorithm(SVM):

The SVM-algorithm is a widely used Supervised Learning technique that is used for both Decision making and Regression tasks. In the field of Machine-learning, it can be used to solve regression difficulties, but its main application is to solve classification problems.

In an n-dimensional space, the SVM algorithm seeks to define an ideal boundary or decision line, which can effectively separate different classes. The goal is to appropriately categorise future new data points. A hyperplane is a boundary or line that serves as a decision.

To create the hyperplane, the Support Vector Machine (SVM)-algorithm chooses the most important vectors or points. Support vectors, which are these important occurrences, are

what give the method its name. In the example that follows, a decision boundary such as hyperplane is used to divide two distinct classes:

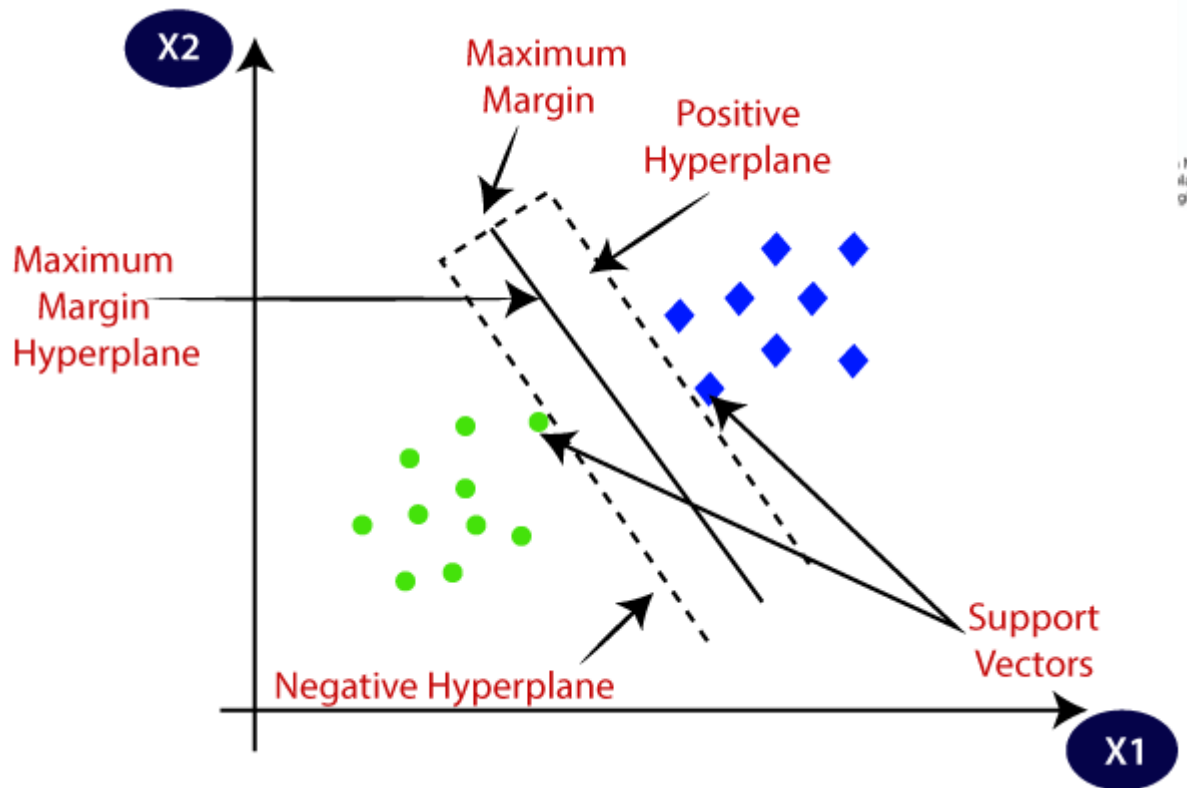


Figure3.1:Diagram where two different classes that are divided using hyperplane

SVM are of two types:

- **Linear-SVM:** Linear-SVM is a type of SVM used to classify linearly -separable data. This means that a dataset/record is considered to be linearly separable data if it can be classified into two groups using a single straight line, and the classifier used for this task is known as a Linear-SVM.
- **Nonlinear-SVM:** Nonlinear-SVM is a form of SVM employed to classify non-linearly separable data. In other words, if Non-linear data is defined as a dataset that cannot be classified using a straight line; the classifier used for this function is known as a Non-linear SVM classifier.

Hyperplane and Support Vectors in the SVM algorithm:

Hyperplane: Within an n-dimensional space, there may exist numerous lines or decision boundaries that can segregate the classes. However, it is imperative to identify the optimal decision boundary that assists in accurately classifying the data points. The SVM hyperplane is the most suitable boundary for this purpose.

The dimensions of the SVM hyperplane are determined by the features of the dataset. For instance, if there are only two features in the dataset (as depicted in the image), then the hyperplane will be a straight line. Conversely, if the dataset has three features, then the hyperplane will be a two-dimensional plane.

When creating a hyperplane using SVM, the primary objective is to maximize the margin, which represents the maximum distance between the data points.

Support Vectors:

Support Vectors are the data points or vectors that are in close proximity to the SVM hyperplane and significantly influence its position. These vectors are referred to as Support Vectors as they provide support to the hyperplane.

How does SVM works?

In Linear-SVM's:

To comprehend the working of the SVM-algorithm, let us consider an example. Consider a dataset with two tags (green and blue), two unique features (x_1 and x_2), and two tags. Our aim is to build a classifier capable of accurately classifying pairs of coordinates (x_1 , x_2) into either green or blue category. The below image provides a visual representation of this example:

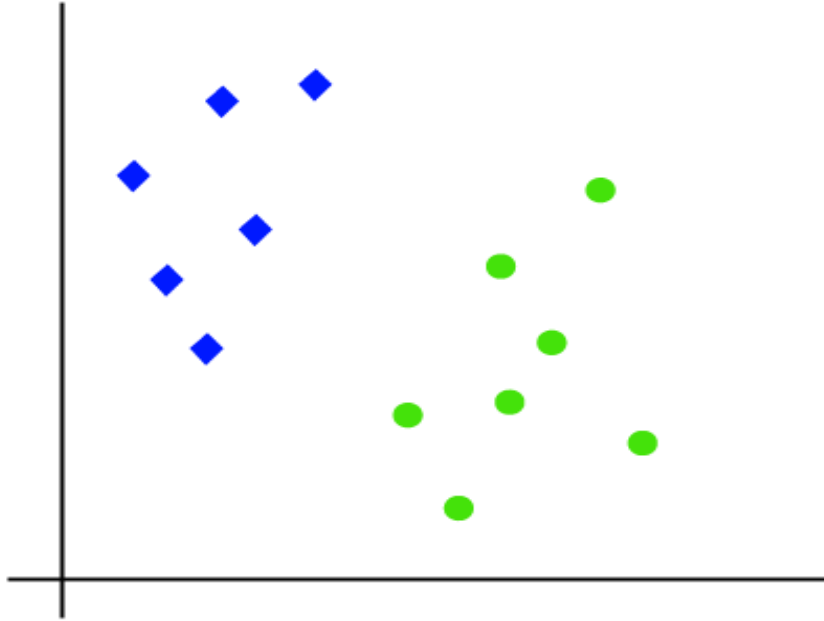


Figure3.2 :Linear SVM

Since the example is a 2-dimensional space, it is feasible to separate the two classes using a straight line. However, there may exist multiple lines capable of segregating these classes. The following image depicts an illustration of this scenario.

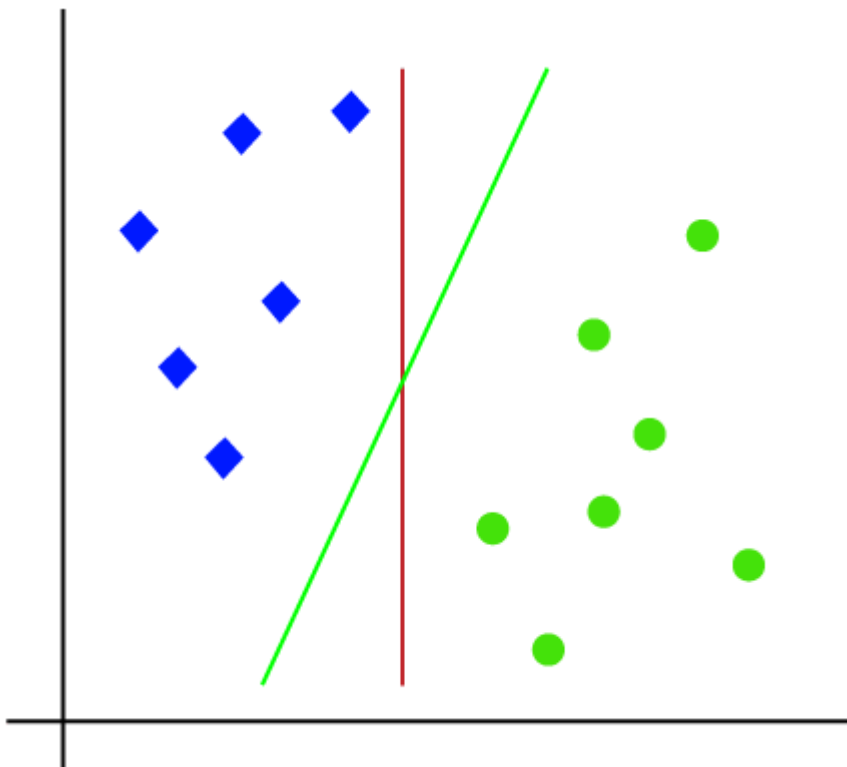


Figure 3.3: Classes are separated using multiple lines in Linear SVM

The hyperplane, also known as the best line or decision border, is found with the use of the SVM algorithm. It completes this work by identifying the support vectors, which are the spots where the lines from both classes are closest to one another. These vectors' separation from the hyperplane is denoted by the margin. The SVM algorithm's main goal is to maximise this margin. This leads to the term "optimal hyperplane" being used to describe the hyperplane with the highest margin.

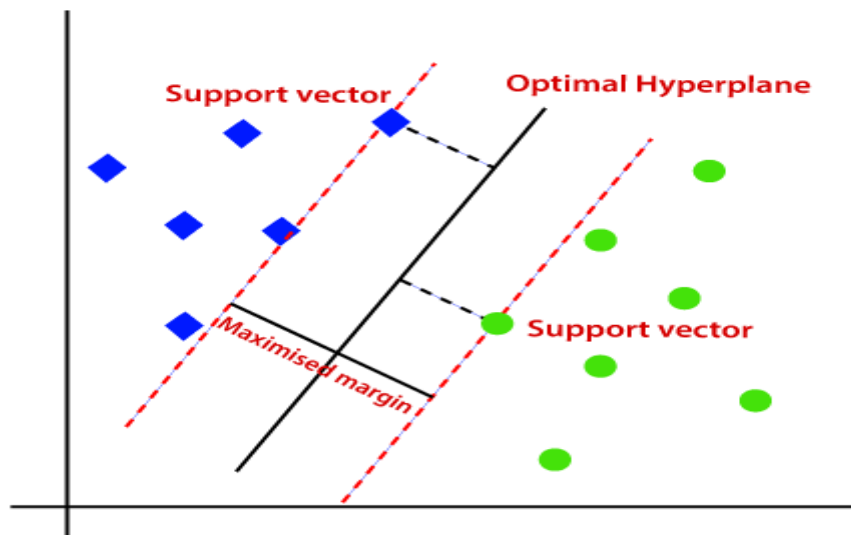


Figure 3.4: Classes are separated using minimal hyper plane in Linear-SVM

NonLinear SVM:

If the data is arranged in a linear fashion, we can effortlessly segregate it by using a straight line. However, it is impossible to draw a one straight line to separate non-linear data. The following image illustrates this scenario.

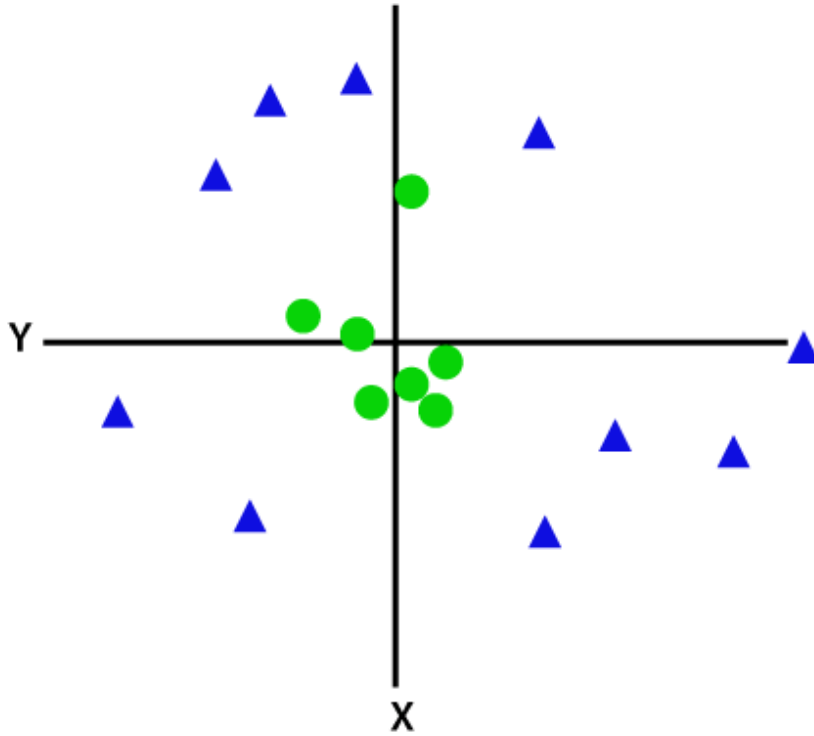


Figure 3.5: Non-Linear SVM

We must add a new dimension in order to separate non-linear data points. We employed two dimensions— x and y —in the case of linear type of data. We introduce a third dimension while working with non-linear data, z , which can be calculated by adding the squares of x and y , i.e., $z = x^2 + y^2$. The addition of this third dimension transforms the sample space into a three-dimensional space, as shown in the following image:

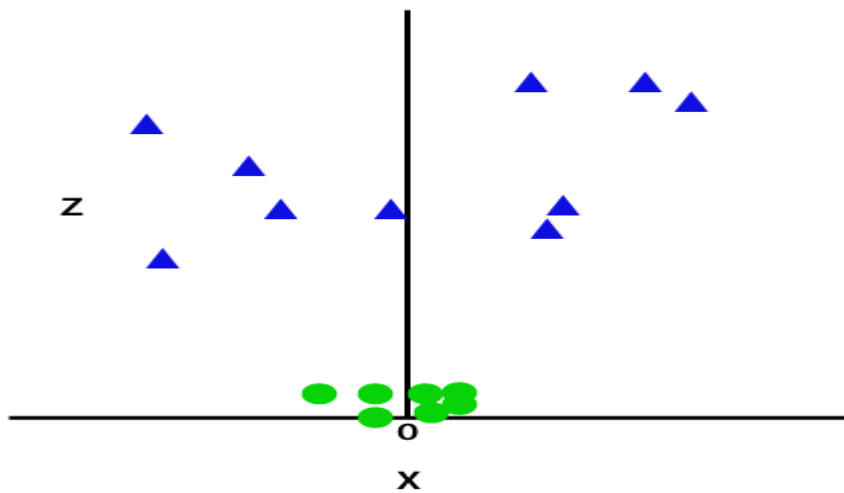


Figure 3.6: 3D Sample space

The data records will now be split into the following classes using SVM. Think on the photo below:

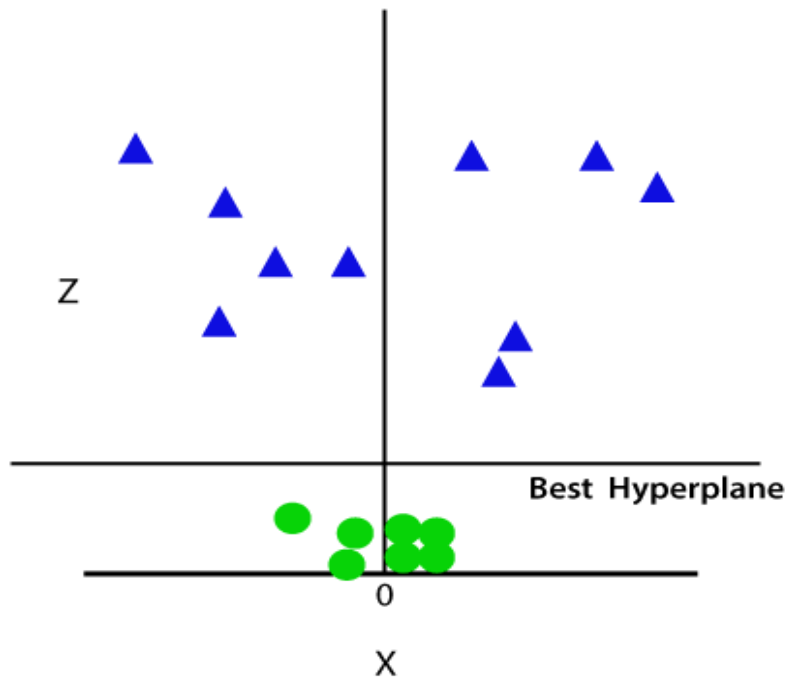


Figure 3.7:Datasets are divided using SVM

The separating hyperplane appears to be parallel to the x-axis in the three-dimensional representation of the current dataset. Yet, if we fix the z-coordinate to 1, we can project this dataset onto a two-dimensional space.

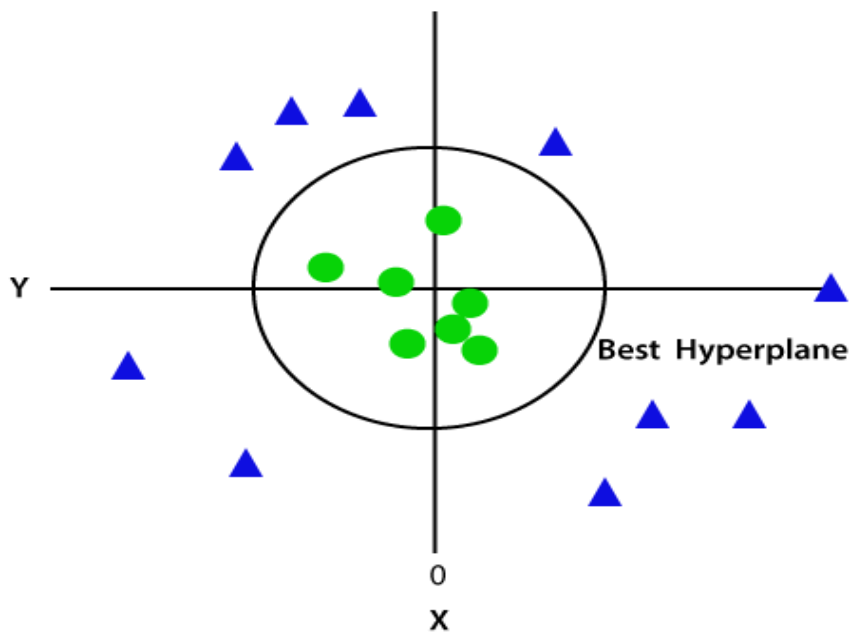


Figure 3.8:2D Sample space

So, in the case of non-linear data, we obtain a circumference of radius 1.

2) ECG IDENTIFICATION

In the existing literature on ECG biometric systems, deep learning methods have been extensively explored. These methods typically involve using a fully connected layer and soft max function as a classifier, where the template that yields the highest matching score is identified as the query signal. Neural networks are particularly useful for non-linear classification problems, and a variety of classifiers have been used for ECG identification, including the Dynamic Time Warping (DTW), Multilayer Perceptron (MLP), the Radial Basis Function Neural Network (RBFNN) and the Long Short-Term Memory (LSTM). While these classifiers employ different node activation functions for optimization, they often use a similar approach for the loss function. Few studies have investigated decision-based discriminant and component analysis techniques for ECG biometric systems.

3.3 DISADVANTAGES:

The literature on ECG biometric systems indicates that there is a lack of standard techniques for demonstrating and evaluating the effectiveness of various approaches and algorithms. Factors such as pre processing (filtering, segmentation), feature-extraction, and matching techniques, as well as the type of records used, can all impact the performance of these systems. Selecting an appropriate kernel function is not a straightforward task, and training large datasets can be time-consuming.

CHAPTER-4

Proposed Method

Generally, Sensing, Pre-processing (filtering, segmentation), features extraction and verification are the main processes involved in systems using ECG based biometric authentication. Two steps are involved in ECG-based biometric authentication: enrollment and authentication. During enrollment, the user's ECG data is recorded to create a template, while during authentication, the system compares raw data from the user to the previously saved template in order to ascertain access rights. In the following sections, we will detail the many processes involved in the authentication process and describe our strategy for implementing the ECG-based biometric authentication algorithm into practise.

4.1 PRE-PROCESSING

The primary objective of data pre-processing is to extract the relevant biometric property from ECG signal while minimizing the effect of ambient noise in ECG-based biometric authentication. Typically, baseline drift, power-line disruption, muscle artifacts and other low- and high-frequency noise components contaminate ECG readings. ECG waves are slower than the electromyogram (EMG) changes that can be produced by muscular action. High-pass filters are used to remove low-frequency components including muscle actions, respiratory fluctuation and baseline fluctuations, whereas lowpass filters are used to remove high-frequency noise. Filtering ECG data during pre-processing reduces noise that could interfere with the signal of biometric property. In order to accomplish this, our study used two filters: Kalman and IIR, which we will go into more depth about below.

4.1.1 FILTERATION:

1) KALMAN FILTER

A discrete-time series signal with non-stationary behaviour is the electrocardiogram (ECG) signal. To model this signal, it is possible to employ an observation vector (filtered ECG) at time t and an unobserved state vector (original ECG signal) at time t . The equations below can be used to represent the model:

$$h_{t+1} = f(h_t, k_t, t)$$

$$x = g(h_t, m_t, t) \tag{1}$$

In this case, the state evolution function is denoted by $f()$ and the link between the state vector and observations is denoted by $g()$. Covariance matrices are used to measure the noise vectors k_t and m_t . $S_t = E\{k_t k_t^T\}$ and $T_t = E\{m_t m_t^T\}$. An approximate linear solution of Equation (1) near a reference point $(\hat{x}_t, \hat{y}_t, \hat{z}_t)$ [17],[18] is required to use the Kalman filter (KF) formalism needs to be derived. This linear model is obtained by linearizing Equation (1) using Taylor series expansion

$$\begin{aligned} x_{t+1} &\approx f(\hat{x}_t, \hat{y}_t, t) + B_t(x_t - \hat{x}_t) + H_t(y_t - \hat{y}_t) \\ y_t &\approx g(\hat{x}_t, \hat{z}_t, t) + J_t(x_t - \hat{x}_t) + K_t(z_t - \hat{z}_t) \end{aligned} \quad (2)$$

where

$$B_t = \left. \frac{\partial f(x,y,t)}{\partial z} \right|_{x=\hat{x}}$$

$$H_t = \left. \frac{\partial f(\text{OECG},y,t)}{\partial y} \right|_{y=\hat{y}_t}$$

$$J_t = \left. \frac{\partial g(z,v,t)}{\partial x} \right|_{x=\hat{x}}$$

$$K_t = \left. \frac{\partial g(z,v,t)}{\partial z} \right|_{z=\hat{z}}$$

2) IIR BUTTERWORTH FILTER:

The characteristic known as infinite impulse response is applicable to number of linear time-invariant systems. IIR filters are systems with this characteristic. As a recursive filter, the current output is calculated using the inputs and outputs from both the current and prior iterations. An IIR filter's transfer function is defined in a specific way.

$$H(z) = \frac{Y(z)}{X(z)} = \frac{p_0 + p_1 z^{-1} + \dots + p_k z^{-K}}{1 + q_1 z^{-1} + \dots + p_L z^{-L}}$$

We used an IIR butterworth filter with a sample frequency of 250 Hz here.

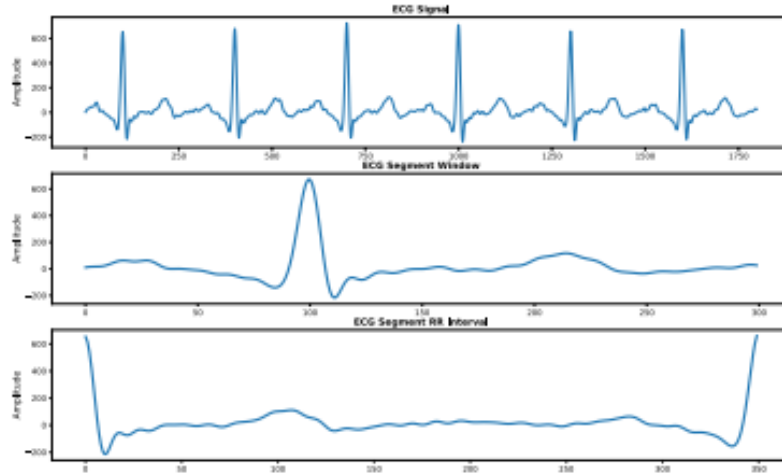


FIGURE 4.1 Segment of ECG signal

4.1.2. SEGMENTATION

The P,Q,R,S,T waves are among the five primary peaks that make up the electrocardiogram (ECG) signal. Analyzing recurrent heartbeats that are associated is ineffective because every heartbeat contains the same information. A common signal processing method that aids in limiting the data size for extracting features is ECG segmentation. To make template matching easier and drastically reduce template size, the segmentation procedure entails locating the ECG signal's recurrent patterns, such as P,Q,R,S,T waves. In ECG biometric, segmentation is performed by identifying a reference point, typically the R-peak and fixed distances initial and after it. This approach, known as Fixed Length Segmentation, involves selecting a partial ECG data ($R - t, R + t_0$) that embraces the majority of the PQRST fragment, with t and t_0 being predefined fixed times. Alternately, it is possible to choose the entire ECG waveform using the RR segmentation method, which entails picking the ECG R peaks at cycles t and $t+1$. ECG signals are frequently divided into distinct heartbeats with framed windows. Fig.4.1 depicts our approach for employing framed windows to segment ECG signals.

4.2. FEATURE EXTRACTION

At the feature extraction stage of ECG-based biometric systems, the segmented ECG signal is transformed into a representation that enhances discriminative and intra-class variations while minimising the impact of intrasubject variability. Feature extraction techniques in ECG biometrics can be classified as either fiducial (or) nonfiducial points method.

4.2.1 FIDUCIAL POINTS BASED FEATURE EXTRACTION

In this technique ECG-based biometric systems focuses on Local heartbeat characteristics include the spatial or intensity variations between adjacent fiducial sites. To extract these features, accurate detection of the main ECG characteristic points, including P,Q,R,S,T waves, is necessary. The timing and amplitude characteristics of each waveform are specific to each user. Yet, due to their sensitivity to noise, precisely recognising each waveform is a difficult operation. In addition, the absence of fiducial points in anomalous signals may lead to sizable mistakes, making it inappropriate as a general characteristic. Therefore, nonfiducial points method have become the preferred approach for ECG biometric systems.

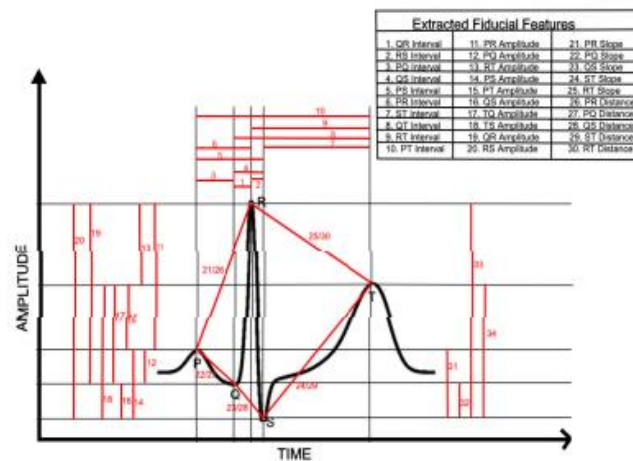


Figure 4.2 a single ECG pulse with identifying characteristics

4.2.2 NON-FIDUCIAL FEATURES EXTRACTION:

In this technique it examines the entirety of an ECG to find significant features. To accomplish this, time or frequency analysis is often used to determine statistical features. Symmlet and Daubechies wavelet families, which were chosen for their resemblance to the ECG signal in our investigation, were among the wavelet families used. To assess the performance of these wavelet mother transformations, we conducted multiple tests at different levels of decomposition, but we only present results for four decompositions.

During the matching stage of ECG analysis, two important functions are performed: identification and verification. This was accomplished using a KNN classifier. The major goal of this stage is to generate match result by comparing the input ECG-feature sets of a query to the saved templates. These scores serve as a quantitative measure of similarity between the

sets of feature of the query and the templates. The degree of correlation between the two is larger when the match score is higher. Our study examined two distinct methods for performing this matching process. The Classification KNN model is a classification system that makes use of nearest-neighbors techniques, allowing for customization of both the number of closest neighbours and the distance measure are applied. As the model stores training data, it can generate resubstitution predictions. Additionally, this model can categorize new observations by employing the “predict” method.

4.3.1 EUCLIDEAN DISTANCE

In the authentication section of our study, Euclidean distance is used as a matching method to evaluate the veracity of identity claims. In order to determine whether (X, P, q) belongs to a real user or an imposter user, the feature vectors of a claimed identity X and P, q is a feature vector of a query are compared. To calculate Euclidean distance Y_d between the a pair of feature vectors T_i and q_i , we use a specific formula.

$$Y_d(\{B_X^T\}, \{P^Q\}) = \sqrt{\sum_{a=1}^n (B_X^T[a] - P^Q[a])^2}$$

Where , B_X^T is a saved template with respect to X . For the purpose of verification, we compare the feature vectors B_X^T and P^Q in order to determine their level of similarity. The stated identity is taken into consideration and recognised as a legitimate user if the calculated distance Y_d or score exceeds a prespecified threshold. Conversely, if the score falls below the threshold, the declared identity is disregarded and labelled as fake.

4.3.2 DYNAMIC TIME WARPING :

It is most efficient method for sequence comparison that are of differing lengths, as they do not necessarily need to be the same [18],[19]. The DTW algorithm basically minimizes the sum of deviations among each pair of matched points to determine the minimized alignment between two sequences of variable lengths. This makes it a valuable tool for comparing templates and queries that are not aligned, such as those in fingerprint analysis or ECG with differing lengths. Our study utilized RR segmentation, which is based

on heart rate variability, to analyze different segments of RR that are unique from one another. Since DTW aligns the template and query ECG feature sets before quantifying their similarity, it was crucial in our case study

K-Nearest Neighbor(KNN) based Algorithm:

- KNN is a straightforward algorithm in the domain of Machine-Learning that makes use of the idea of Supervised Learning.
- KNN algorithm, assuming that the new case/data and the previous cases are comparable, places the new case in the category that is most comparable to the available categories.
- KNN algorithm categorises new data based on similarity and preserves all of the existing data. This indicates that new data can be reliably and quickly categorised using the K-NN approach.
- Although the KNN technique can be used for both decision making and regression problems, classification tasks are where it is most typically applied.
- KNN does not make any assumptions about the underlying data because it is a non-parametric approach.
- Because it saves the training dataset rather than instantly learning from it, the technique is often referred to as a lazy learner algorithm. Instead, it performs an action while classifying data by using the dataset.
- During the training phase, the KNN algorithm merely saves the dataset and then classifies new data into a category that is quite similar to the new data.

For example: Consider the case when we have a photo of a species that resembles both cats and dogs but we are unsure of which it is. We can use the KNN method for this identification though because it is based on a similarity metric. In order to categorise the new data set as cat- or dog-related, our KNN model will search for similarities between the properties of the new data set and those in the photographs of cats and dogs.



Figure 4.3 Classification of input value

Why is a KNN algorithm necessary?

A Kth Nearest Neighbor (KNN) algorithm can be used to identify which category a new data point, x_1 , belongs to out of Category A and Category B. KNN helps us classify a dataset into different categories or classes. The majority class of the K data points that are K closest to the new data point x_1 in the dataset are found by this algorithm. The KNN algorithm's operation is shown in the following diagram.

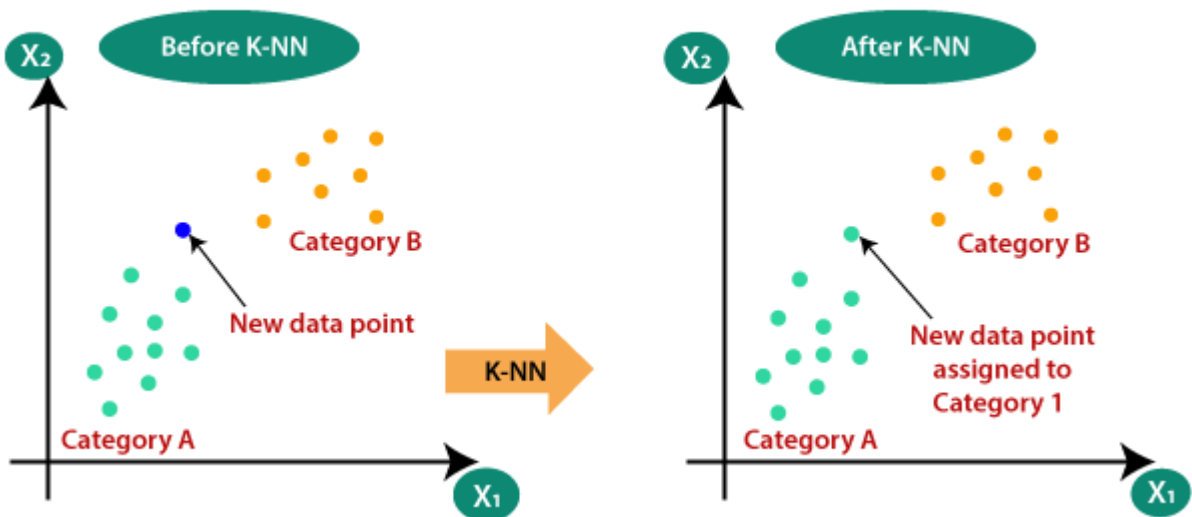


Figure 4.4:KNN Algorithm

How does this algorithm work?

The KNN operates according to the following algorithm:

Step-1: Let us consider the Kth neighbor's number.

Step-2: Find the distance in Euclidean space between those neighbours.

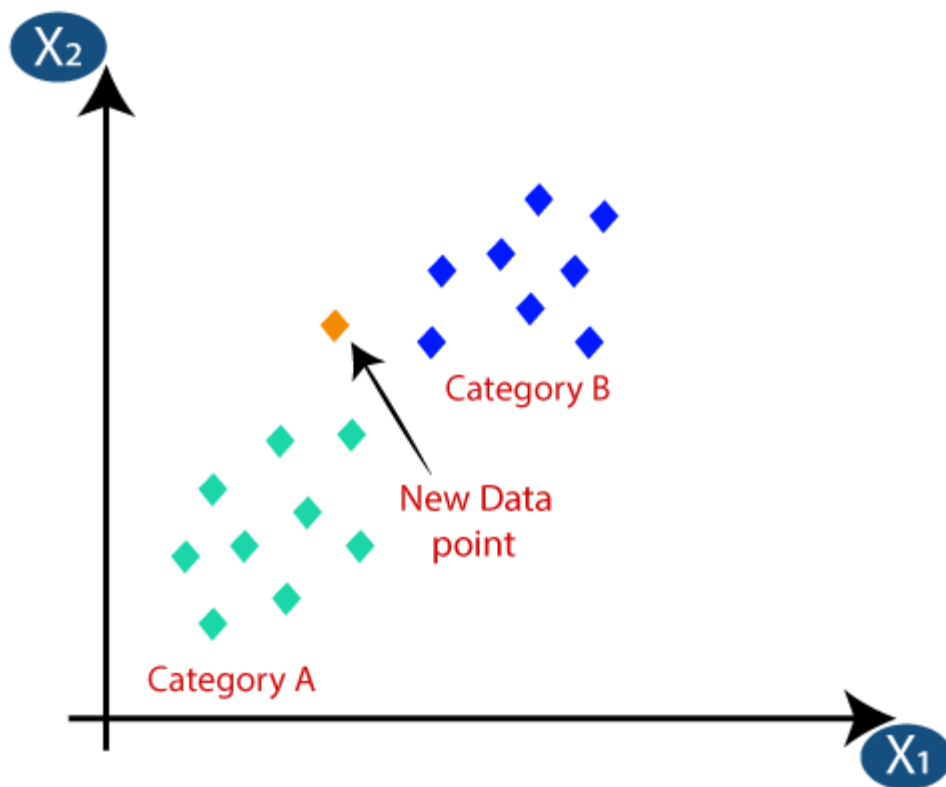
Step-3: Pick K neighbours who are the closest in terms of Euclidean distance.

Step-4: Within these neighbours, total the amount of data points in individual category.

Step-5: Place the extra data points in the category with the highest neighbour count.

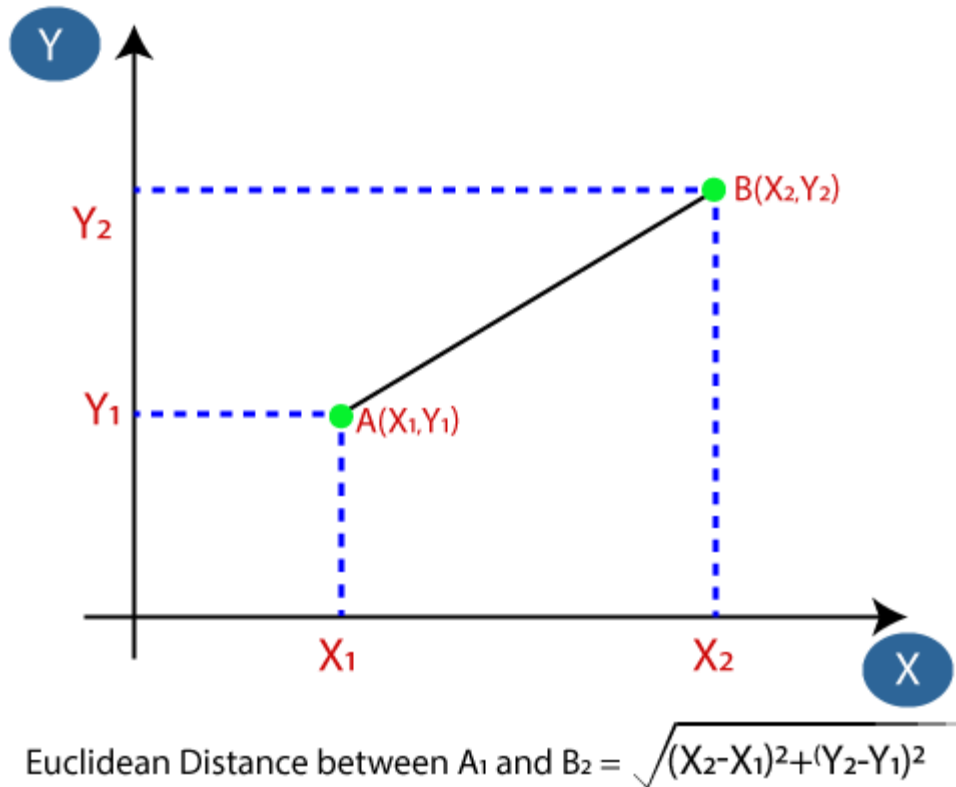
Step-6: Our model has been finished.

Say that we need to classify a new data-point in order to use it. Think on the photo below:



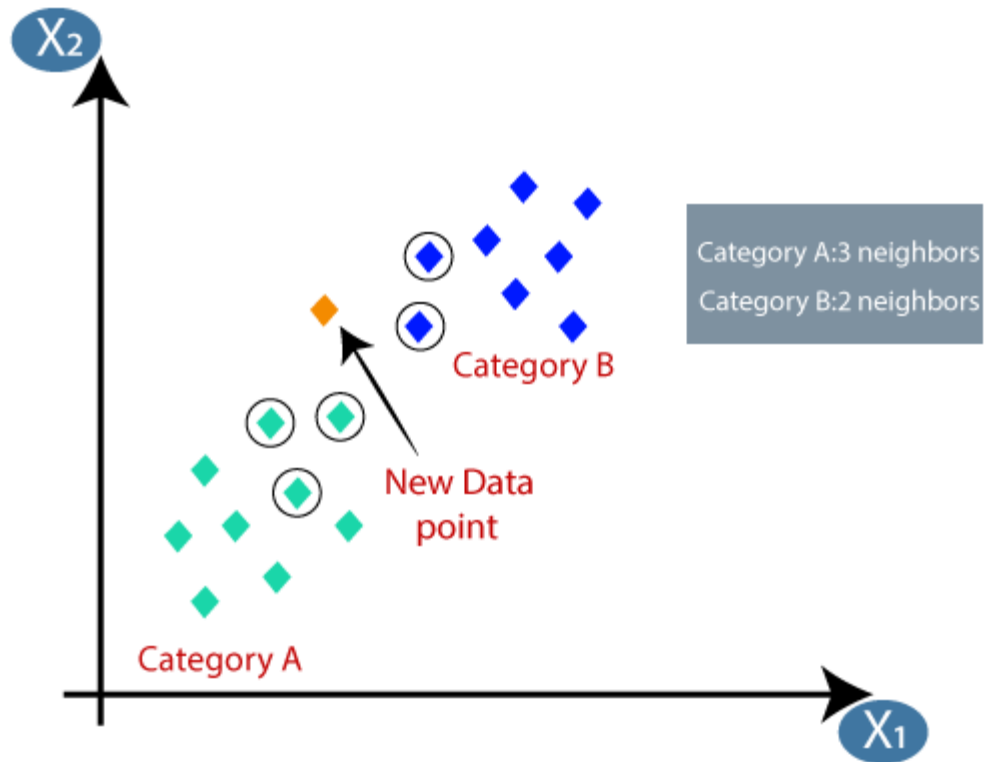
- Initially, we'll decide on the No. of neighbours, thus we'll pick k as 5.

- After that, the Euclidean distance between the data points can be calculated. The spacing between two points is known as the Euclidean distance, which we have already covered in geometry. It can be calculated as:



(b)

- To identify, the Euclidean distance was calculated for the neighbours that were nearest to one another. In group A, there were three nearest neighbours, and in category B, there were two. Think on the following photo:



(C)

FIGURE 4.5(a),(b),(c) Working of KNN algorithm

As we can see, this new data point must fall under category A as the three nearest neighbours are all in group A.

How do we choose K in the Algorithm?

The following considerations should be made when determining K's value:

- As there is no one right way to select the ideal number for "K," we must test out many values to determine which one performs the best. K is most accurately expressed in the number 5.
- An extremely low K-value, such as K=0 or K=1, may be noisy which can cause adversarial effects in the model.
- K-variable must have huge values, however there might be some problems.

Advantages:

- Putting it into practise is simple.
- It can handle erratic training data.
- If there is a lot of trained data, it might work better.

KNN execution in MATLAB:

Description:

Specifying a category The number of nearest neighbours and the distance measure can both be customised with a particular kind of nearest-neighbor classification model called KNN. The training data are stored by this classifier, enabling the creation of resubstitution predictions. Using the predict method and the stored model to classify fresh observations is another choice.

We can create a Classification of KNN-model using “**fitcknn**”.

Properties :

Tie-breaking algorithm

smallest(default) | closest | random

Distance metric

cityblock | cosine | correlation | chebychev | euclidean

Distance weighting function

'equal' | 'in'squaredinverse' | 'inverse'

Parameter for distance metric

matrix of positive definite covariance | positive scalar

Tie inclusion flag

'false' (by default) | 'true'

NSMethod : using the nearest neighbour method

kdtree

NumNeighbors : count of closest neighbours

positive value

Some Other Properties are:

CategoricalPredictors

[]:-vector of positive integer values

RowsUsed :-Rows used in fitting

[] :- logical vector

ClassNames : classes in training a data X or Y

logical vector, numeric vector, categorical array, character array, and cell array of character vectors

Cost : Misclassification costs

square matrix values

Expanded predictor names

character vector cell array

ModelParameters:-Parameters used in training ClassificationKNN

object

Mu : Predictor means

vector of numericals

ResponseName :-Response variable name

character vector

Number of observations

positive integer

Prior probabilities for each class

numeric vector

Predictor variable names

character vector cell array

Sigma : standard deviations for predictors

numeric vector

W : Observation weights

a collection of positive values

X : unpredictable predictor data

numeric matrix

ScoreTransform :- Score transformation

ismax | doublelogit | invlogit | logit | none (default)

4.4ADVANTAGES&Its APPLICATIONS:

Advantages:

Only one segment of ECG for each user is saved in the template in this work, as opposed to the bulk of the implied ECG based biometric systems that saved many segments in the layouts for comparison during the authentication phase.

Applications:

In systems for biomedical acquisition.

CHAPTER-5

MATLAB

5.1 INTRODUCTION TO MATLAB

5.1.1 What Is MATLAB?

MATLAB is a specialised programming language. It combines calculation, representation, and programming in an easy-to-use format in which problems and treatments are presented in flowery numerical documentation. Commonplace applications include

- Math and calculation
- Algorithm advancement
- Data obtaining
- Modeling, re-enactment, and prototyping
- Data examination, investigation, and representation
- Scientific and designing illustrations
- Application advancement, including graphical UI building

MATLAB is an intuitive framework whose important statistics aspect is an show off that does not require dimensioning. This allows you to tackle several specialized processing issues, particularly those with framework and vector info, in a small quantity of the time it'd take to compose a program in a scalar non intuitive dialect, as an instance, C or FORTRAN.

The name MATLAB remains for the grid studies facility. MATLAB was initially intended to provide simple access to framework programming created by the LINPACK and EISPACK initiatives. Today, MATLAB motors combine the LAPACK and BLAS libraries, introducing the cutting edge in network calculation programming.

MATLAB has advanced over a time of years with contribution from several customers. In university situations, it's far the usual academic apparatus for early on and

propelled guides in mathematics, designing, and science. In enterprise, MATLAB is the tool of choice for excessive-profitability studies, advancement, and exam.

Tool booths are a collection of more utility-specific configurations highlighted by MATLAB. Device kits, which are important to the majority of MATLAB users, allow you to understand and implement specific innovation. Tool compartments are thorough collections of MATLAB capabilities (M-records) that reach out to the MATLAB condition to handle exact problem training. Flag handling, manipulation frameworks, neural structures, fluffy reason, wavelets, amusement, and several others are among the territories where tool stash is accessible.

The MATLAB System:

The MATLAB system consists of five main parts.

Development Environment:

This is a collection of tools and centres that assist you in using MATLAB features and files. Many of the devices include graphical user interfaces. It features the MATLAB desktop and Command Window, a command history, an editor and debugger, and browsers for the workspace, files, and the hunt direction.

The MATLAB Mathematical Function:

This is a great collection of computational algorithms ranging from standard capabilities like sum, sine, cosine, and complex arithmetic, to extra sophisticated features like matrix inverse, matrix eigen values, Bessel functions, and speedy Fourier transforms.

The MATLAB Language:

This is a high-level matrix/array language with features such as control flow statements, functions, data structures, input/output, and object-oriented programming. It enables both "programming in the small" and "programming in the large" to quickly develop quick and dirty throw-away programs, as well as "programming in the large" to create complete massive and complicated application programmes.

Graphics:

MATLAB offers numerous options for graphing vectors and matrices, as well as annotating and publishing those graphs. It includes high-level features for 2D and 3D record visualization, photo processing, animation, and presentation graphics. It also includes low-level capabilities that allow you to completely alter the appearance of graphics as well as create complete graphical user interfaces for your MATLAB programmes.

The MATLAB Application Program Interface (API):

This is a package that enables you to create C and Fortran applications that interface with MATLAB. It includes tools for calling MATLAB workouts (dynamic linking), using MATLAB as a computational engine, and studying and producing MAT-documents.

5.2 MATLAB WORKING ENVIRONMENT:

MATLAB DESKTOP:

MATLAB Desktop is the primary MATLAB application window. The desktop is separated into five sub-windows: the call window, workspace programme, existing catalogue window, order records window, and at least one figure window, which can be shown simply while the user suggests anything reasonable.

The outcomes of these charges are shown in the order window, where the client organises the MATLAB orders and expressions at the beginning (>>). The workspace, as defined by MATLAB, is the conglomeration of elements that the client develops throughout a work session. These components are displayed in the workspace application along with some related information. The Array Editor, which can be used to acquire data and salary instances as well as alter the variable's unique characteristics, appears when you double-tap a variable in the workspace application.

The content of the current registry, which is viewable within the current index window, is displayed in the current Directory tab over the workspace tab. 1 For instance, the technique may be as follows when used with the Windows operating system: C:MATLAB Work that proves that the primary catalogue "MATLAB," which is offered in pressure C, is a

subfolder of the register "paintings." A list of recently used techniques appears when you tap on the bolt inside the current index window. By touching on the catch to one side of the display, the customer can switch out the current catalogue.

M-data and other MATLAB-related documents are stored in catalogues within the PC file system, and MATLAB uses an inquiry method to find them. Every file that is constantly moving in MATLAB must be stored in either the ebb and flow registry or a method-specific index. Naturally, the study approach incorporates the data provided by the MATLAB and math works equipment kits. To determine if indices are at the inquiry stage, use the least stressful method possible. The easiest method to choose which catalogues will be the search way, or to add or amend an inquiry way, is to utilise the set way exchange container after choosing set way from the computer's File menu. To prevent having to regularly change the current index, it is a wonderful idea to include any frequently used catalogues to the search technique.

The commands that the client entered in the charge window, including those from earlier and current MATLAB sessions, are recorded in the Command History Window. Previously entered MATLAB orders from the charge history pane can be picked and re-executed by right-tapping on a summon or arrangement of orders. While performing the orders, this action generates a menu from which to choose different alternatives. This enables you to consider several options while still responding to the summons. When experimenting with various ordering during a work session, this function is helpful.

Using the MATLAB Editor to create M-Files:

The MATLAB manager functions as both a graphical MATLAB debugger and a word editor with special M-statistics capabilities. The laptop's sub-window or a separate window can display the proofreader. M-facts are meant to be a part of the expansion. Pixelup.M is pronounced "M." The MATLAB editorial management window has a number of drop-down choices for operations including document saving, document reading, and document debugging. This content device is recommended as the tool of choice for creating and altering M-capabilities since it runs a few simple checks and furthermore uses colour to distinguish between different code modifications. The M-report filename is opened by the sort control at the beginning to launch the proofreader. M is waiting for change in a supervisor

window. As previously stated, the record must be in the momentum catalog or in an index within the pursuit method.

Getting Help:

The MATLAB help application, which may be started as a separate window by pressing the question mark icon on the computer toolbar or by typing help program on the keyboard within the order window, is the most important way to receive online assistance. The assistance Browser is a web application built into MATLAB that provides Hypertext Markup Language (HTML) data. The Help Browser comprises two sheets: the assistance pilot sheet for finding data and the display sheet for seeing statistics. To conduct a pursuit, clear as crystal tabs and a pilot sheet are used. We cannot exactly adjust item motion within the sum of difference map, which makes the situation more challenging in the event of motion pictures produced with a transferring camera setup because the background may change as a result of the changing shot.

As a consequence, the objective is achieved in this instance by reusing the old seam and applying it to the modern body. We find the modern seam in the current frame by referencing the seam of the preceding body. For pixels near to the location of the previous seam, we determine how much the selected pixel may change from the pixel of the prior seam. The degree of temporal coherence is determined using the difference between the two pixels. If the distinction value of the first seam pixel above the threshold, we can search for the following seam pixel on three potential pixels (in yellow, blue, and brown) in the subsequent row until we discover five consecutive pixels that are also above the threshold. We recalculate the energy for a new seam when we are unable to find the matching seam. As a seam must be computed for the modern frame, we presume that a seam $l-1$ has already been calculated inside the preceding body. We want to create a new seam with the same index that is near to the old seam in order to maintain the temporal coherence. We utilise the difference between the previous seam and each and every pixel at the present body as the measurement.

As a result, before calculating a seam L , we upload the temporal coherence price $Tc(i, j)$ to the strength map. Tc is 0, however body pixels carry the same fee as prior seam pixels. Using our temporal coherence pricing, we will find the seam with the least electricity and the closest proximity to the preceding seam in the previous frame. As a result, we will reduce the jittery artifacts inside the films.

COMMUNICATION:

Communications System Toolbox™ provides algorithms and equipment for the design, simulation, and analysis of communication networks. These functionalities are provided by MATLAB® features, MATLAB System gadgets™, and Simulink® blocks. The machine toolbox includes algorithms for source coding, channel coding, interleaving, modulation, equalization, synchronization, and channel modeling. Tools for analyzing bit errors, making eye and constellation diagrams, and displaying channel parameters are provided. Adaptive methods for versioning dynamic communications designs that use OFDM, OFDMA, and MIMO approaches are also included in the machine toolbox. Algorithms can do fixed-point fact arithmetic as well as generate C or HDL code.

Key Features

- Algorithms for building the physical layer of communications systems, which includes supply coding, channel coding, interleaving, modulation, channel designs, MIMO, equalization, and synchronization.
- GPU-enabled System objects with Turbo, LDPC, and Viterbi decoders for computationally expensive algorithms
- Eye diagrams, constellations, and channel scattering capabilities are included in interactive visualization equipment.
- A graphical tool for comparing a machine's simulated bit error rate with analytical results.
- Channel models, consisting of AWGN, Multipath Rayleigh Fading, Rician Fading, MIMO Multipath Fading, and LTE MIMO Multipath Fading
- Nonlinearity, section noise, thermal noise, and section and frequency offsets are all examples of basic RF flaws.
- Algorithms available as MATLAB features, MATLAB System objects, and Simulink blocks
- Support for fixed-point modeling and C and HDL code technology

System Design, Characterization, and Visualization:

The design and modeling of a communications device necessitates studying its response to the noise and interference inherent in real-world contexts, reading its behavior

using graphical and quantitative methods, and determining whether the overall performance satisfies acceptable requirements. Communications System Toolbox meets a wide range of needs for communications machine layout and simulation. Many of the device toolbox's functions, System objects™, and blocks perform computations relevant to a specific piece of communications equipment, such as a demodulator or equalizer. Other talents are designed to aid with visualization or evaluation.

System Characterization

The system toolbox offers several standard methods for quantitatively characterizing system performance:

- Bit error rate (BER) computations
- Adjacent channel power ratio (ACPR) measurements
- Error vector magnitude (EVM) measurements
- Modulation error ratio (MER) measurements

Because BER computations are essential for characterizing any communications system, the system toolbox contains the following tools and capabilities for developing BER test scenarios and speeding up BER simulations:

BER tool— A graphical user interface for analyzing communication system BER performance. A simulation-based, semi-analytic, or theoretical technique can be used to analyze performance.

Error Rate Test Console — A MATLAB object that simulates communication systems to assess error rate performance. It enables the generation of parametric performance charts and surfaces as well as user-specified test points.

When running on a multi-core computing platform, performance can be accelerated.

Multi core and GPU acceleration — Parallel Computing Toolbox™ provides the potential to accelerate simulation performance by utilizing multi-core and GPU technology on your computer.

Distributed computing and cloud computing support — You can employ the computing capability of your server farms and the Amazon EC2 Web service with the Parallel computing Toolbox and MATLAB Distributed Computing Server™ features.

The representation of performance. The system toolbox contains the following tools for visualizing system performance:

Channel visualization tool — To depict the characteristics of a fading channel

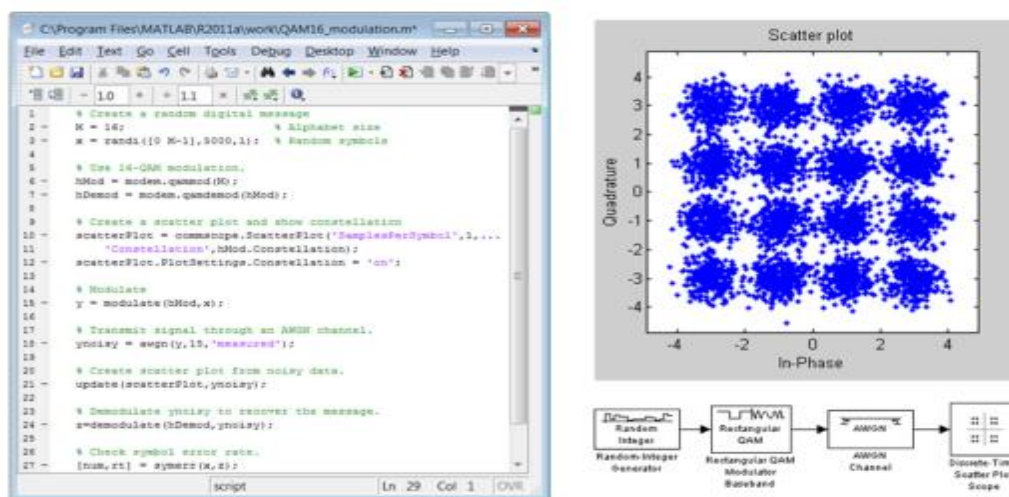
Eye diagrams and signal constellation scatter plots — to gain a qualitative, visual grasp of system behavior and make first design decisions

Signal trajectory plots — for a continuous picture of the signal's trajectory between decision points

BER plots — for visualizing quantitative BER performance of a design candidate, parameterized by metrics such as SNR and fixed-point word size

Analog and Digital Modulation

Analog and digital modulation techniques encode the information flow into a signal suitable for transmission. Certain modulation and demodulation capabilities are included in the Communications System Toolbox. These capabilities are provided as MATLAB features and gadgets, and the toolbox includes the MATLAB system modulation kinds listed below:



MATLAB function (left) and Simulink model (right) with scatter plot for 16 QAM simulation.

Figure 5.1 :Matkab function(left) and Simulink model(right) with scatter plot for 16 QAm simulation

Source and Channel Coding

Communications System Toolbox has source and channel coding capabilities that enable you to easily develop and analyze communications structures, allowing you to find what-if scenarios and avoid having to create coding skills from start.

Source Coding

Source coding, also known as quantization or signal formatting, is a method of processing data that reduces duplication or prepares it for later processing. The system toolbox includes a variety of algorithms for imposing source coding and interpreting, including:

- Quantizing
- Companding (μ -law and A-law)
- Differential pulse code modulation (DPCM)
- Huffman coding
- Arithmetic coding

Channel Coding

- orthogonal area-time block code (OSTBC) (encoder and decoder for MIMO channels)
- Turbo encoder and decoder examples

The gadget toolbox includes application functions for creating your own channel coding. Generator polynomials, coefficients, and syndrome deciphering tables can be generated, as well as product parity-check and generator matrices.

The system toolbox also includes block and convolutional interleaving, as well as delinters leaving functions, to mitigate data errors caused by burst errors in a conversation machine:

Block, including General block interleaver, algebraic interleaver, helical scan interleaver, matrix interleaver, and random interleaver.

Convolutional, including General multiplexed interleaver, convolutional interleaver, and helical interleaver

Channel Modeling and RF Impairments

Channel Modeling

The Communications System Toolbox provides tools and methods for simulating distortions in communications channels such as noise, fading, interference, and other possibly common occurrences. The computer's toolbox supports the following channel types:

- Additive white Gaussian noise (AWGN)

- Multiple-enter multiple-output (MIMO) fading
- Single-enter single-output (SISO), Rayleigh, and Rician fading
- Binary symmetric

A MATLABchannel object offers a condensed, adaptable channel model implementation that enables you to specify variables like:

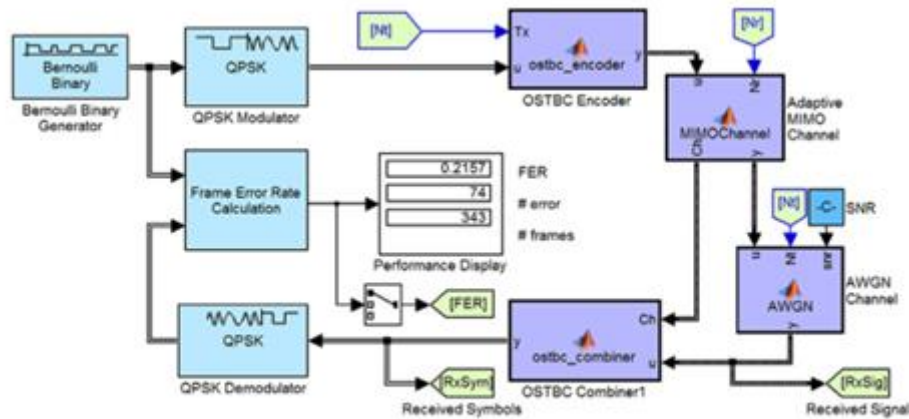
- Path delays
- Average path gains
- Maximum Doppler shifts
- K-Factor for Rician fading channels
- Doppler spectrum parameters

For MIMO systems, the MATLAB MIMO channel object expands these parameters to also include:

- Number of transmit antennas (up to 8)
- Number of receive antennas (up to 8)
- Transmit correlation matrix
- Receive correlation matrix

The system toolbox includes block and convolutional coding and decoding algorithms for error identification and correction in order to mitigate the effects of noise and channel pollution. For simple error detection without inherent correction, a cyclic redundancy check capability is also available. Channel coding elements of the system toolbox include:

- BCH encoder and decoder
- Reed-Solomon encoder and decoder
- LDPC encoder and decoder
- Convolutional encoder and Viterbi decoder



Simulink model of an adaptive MIMO system with orthogonal space-time block codes (OSTBC).

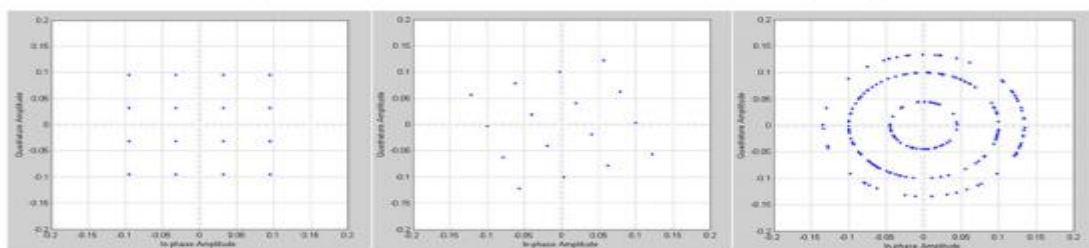
Figure 5.2 : An adaptive MIMO system with orthogonal space-time block codes is modeled in Simulink. (OSTBC)

RF Impairments

You can include the following impairments into your communications system to imitate the consequences of a less-than-ideal RF front end in order to analyze and characterize performance with real-world impacts:

- Memory less nonlinearity
- Phase and frequency offset
- Phase noise
- Thermal noise

You can include more complex RF impairments and RF circuit models in your design using SimRF™.



An ideal 16 QAM scatter plot (left) impaired by a phase offset (middle) and a frequency offset (right).

Figure 5.3: An ideal 16 QAM plot(left) impaired by a offset (middle) and a frequency offset (right)

Equalization and Synchronization

Using the Communications System Toolbox, you may learn about synchronisation and equalisation strategies. These strategies are usually adaptive and difficult to construct and symbolise. The machine toolbox contains tools and algorithms that allow you to quickly select the optimum strategy for your communications machine. Equalization The device toolbox includes adaptive algorithms such as the following to compare different approaches of equalization:

- LMS
- Normalized LMS
- Variable step LMS
- Signed LMS
- MLSE (Viterbi)
- RLS
- CMA

These adaptive equalizers are available in both nonlinear decision feedback equalizer (DFE) and linear (symbol or fractionally spaced) variants.

Synchronization

The device toolbox includes algorithms for synchronizing each service segment and timing phase. The machine toolbox provides a MATLAB Timing Phase Synchronizer object for timing section synchronization, with the following implementation techniques:

- Early-late gate timing method
- Gardner's method
- Fourth-order nonlinearity method

Stream Processing in MATLAB and Simulink

The majority of verbal exchange systems combine temporal processing with concurrent multi-frequency and multi-channel processing to accommodate streaming and frame-based input. Complex communication schemes using MIMO and OFDM may be

observed to leverage this form of streaming multidimensional processing. By helping with move processing and frame-based simulation in MATLAB and Simulink, Communications System Toolbox makes it easier to simulate complex communications networks.

System items™ in MATLAB, which employ MATLAB objects to represent time-based and data-driven algorithms, sources, and sinks, enable flow processing. In addition to information about information indexing, buffering, and other flow processing, system objects automatically store the state of a set of rules. System items™ in MATLAB, which employ MATLAB objects to represent time-based and data-driven algorithms, sources, and sinks, enable flow processing. In addition to information about information indexing, buffering, and other flow processing, system objects automatically store the state of a set of rules. It is possible to combine system devices with current MATLAB operations and functions. A Simulink block similar to most System objects with the same properties. Simulink handles circulation processing implicitly by controlling information flow across the blocks that make up a Simulink model. Hierarchical diagrams are used in Simulink, an interactive graphical interface for modelling and simulating dynamic systems, to represent a machine version. For modelling device hierarchy, resources, sinks, and algorithms, it comes with a library of frequently used, pre-configured blocks.

Implementing a Communications System

Modeling Using Fixed Points Many communication systems make use of hardware that necessitates a fixed-point representation of your design.

Fixed-point modeling is supported by Communications System Toolbox in all essential blocks and System objects™ with tools that help you configure fixed-point attributes. Fixed-point support in the system toolbox includes:

- Word sizes from 1 to 128 bits
- Arbitrary binary-point placement
- Overflow handling methods (wrap or saturation)
- Rounding methods includes ceiling, convergent, floor, nearest, round, simplest, and zero.

Simulink's Fixed-Point Tool Fixed Point™ enables the conversion of floating-point data formats to fixed-point data types. The tool tracks overflows, maxima and minima when configuring fixed-point properties.

Code Generation

You may instantly create C code from your developed set of rules or communications device for implementation, quick prototyping, and verification. Most of the system tools, features, and building blocks in the Communications System Toolbox may produce ANSI/ISO C code using MATLAB Coder™, Simulink Coder™, or Embedded Coder™. Certain Simulink blocks and System gadgets can produce HDL code. To take use of the present highbrow resources, you may choose optimisations for particular processor architectures and combine legacy C code with the produced code. For both fixed-point and floating-point data formats, C code may be generated.

DSP prototyping DSPs are used in communication system development for hardware finalization, rapid prototyping, and verification. You can validate the generated source code and compiled code by executing the implementation code for your method on a target processor using Embedded Coder's processor-in-the-loop (PIL) simulation capability.

FPGA Prototyping

High-speed signal processing techniques are implemented in communication systems using FPGAs. RTL code for any current HDL code, whether manually written or mechanically manufactured, may be tested in real hardware using HDL's FPGA-in-the-loop (FIL) capability Verifier™.

CHAPTER-6

RESULTS AND DISCUSSIONS

Since verification is our first priority, we have evaluated complex metrics utilising a variety of databases, including accuracy, false accept rate (FAR), equal error rate (EER), and equal false reject rate (FRR). FRR is the proportion of real users who were denied access to the ECG recognition system, whereas The FAR is the percentage of imposters who were successful in gaining access to the system. Finding the ideal and desired EER necessitates balancing the two error rates, FAR and FRR. The findings are put into practise using the strategy described in Section III. Figures 6 show the experimental results from the MIT-BIH database using our methods. In this study, we created our template from a single ECG pulse (segment), but our feature extraction, segmentation, and filtering methods did not work very well. Due to the requirement to remove some people, we had to exclude several topics from the associated work. We put several feature extraction, segmentation, filtering, and matching methods to the test. Using fixed window segmentation, IIR filter, Kalman filter, non-fiducial feature extraction, Euclidean distance, and dynamic temporal warping, we looked into a number of ECG records.

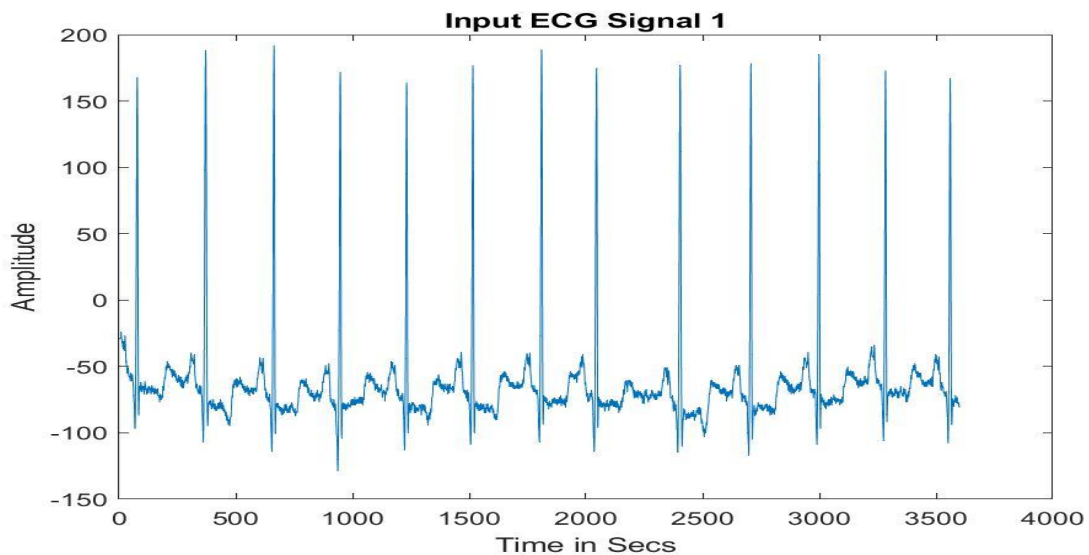


Figure 6.1 Input signal-1

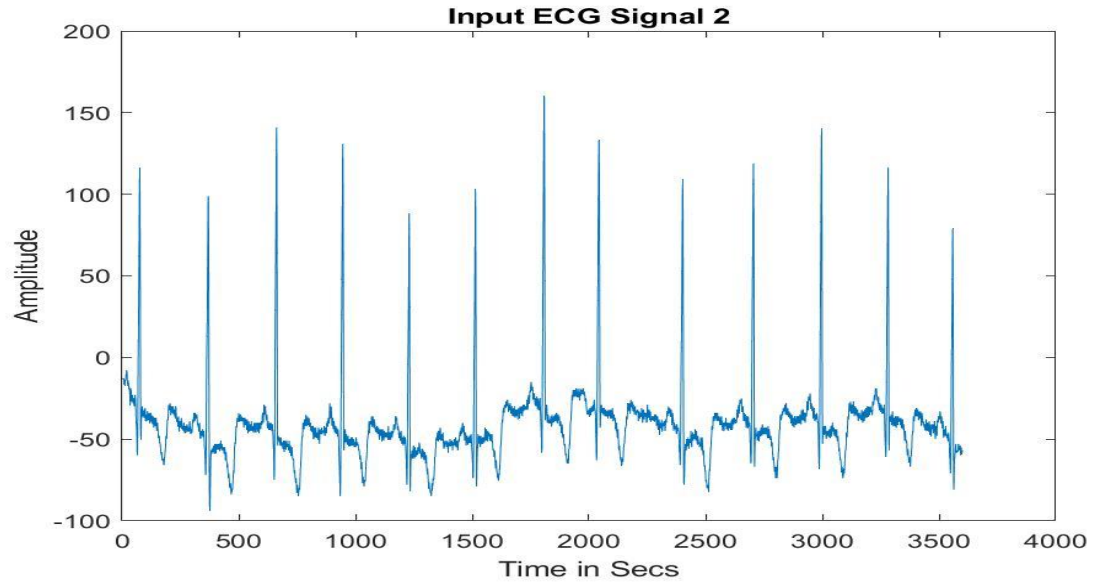


Figure 6.2 Input signal-2

The two images above show an individual's ECG signal that was collected from the MIT-BIH database. Now we are using our suggested technique on this. (filtering, segmentation, feature extraction and Matching).

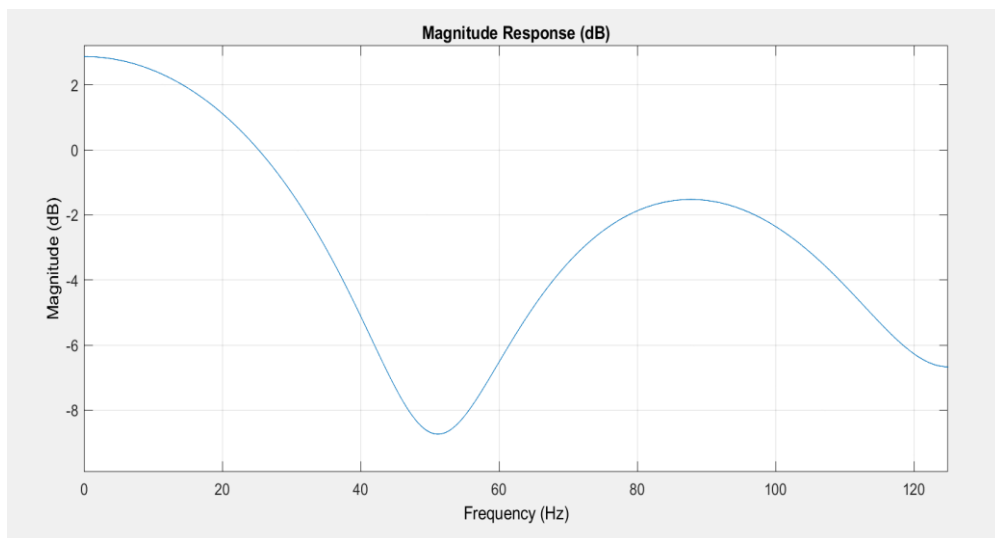


Figure 6.3 Filtered Signal

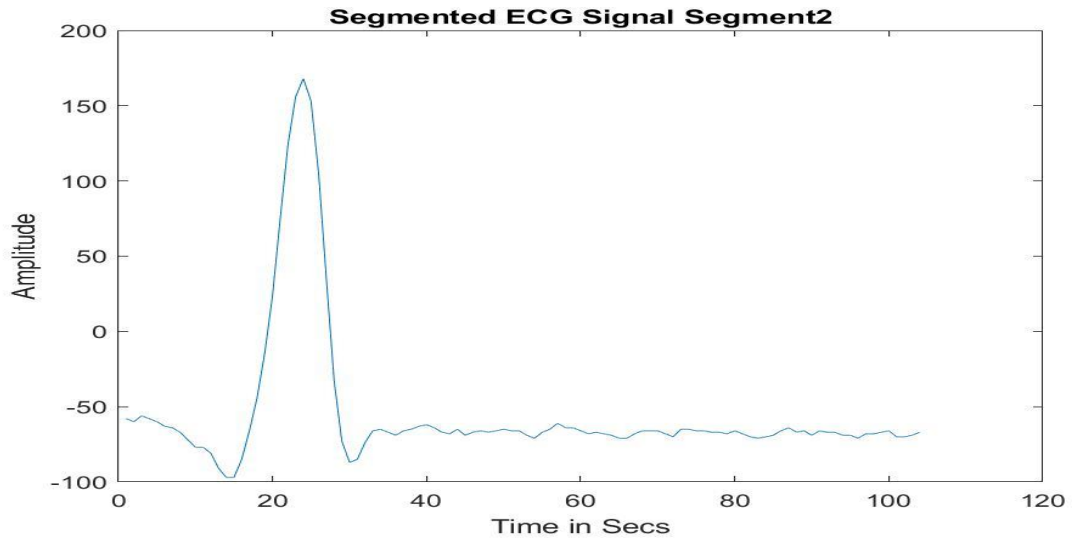


Figure 6.4 Segment of ECG signal

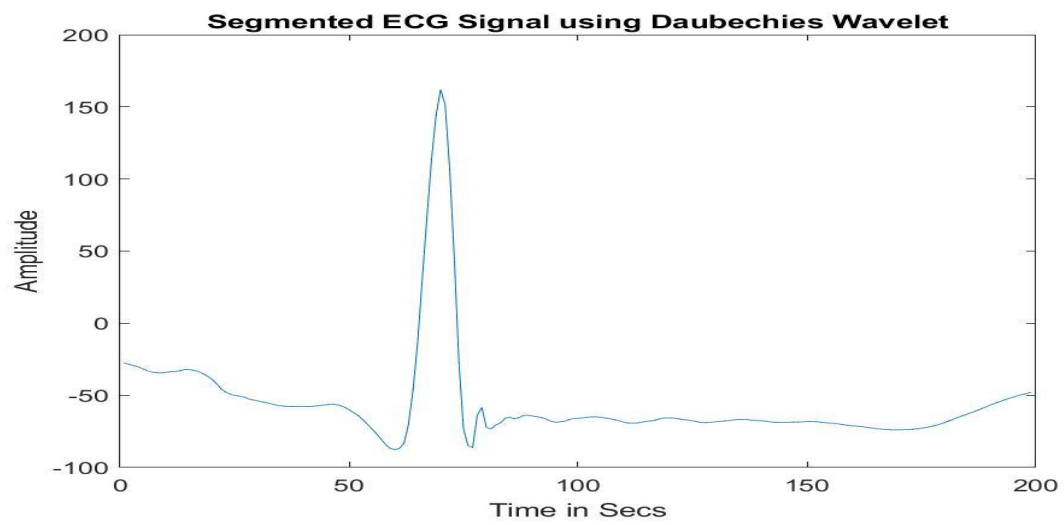
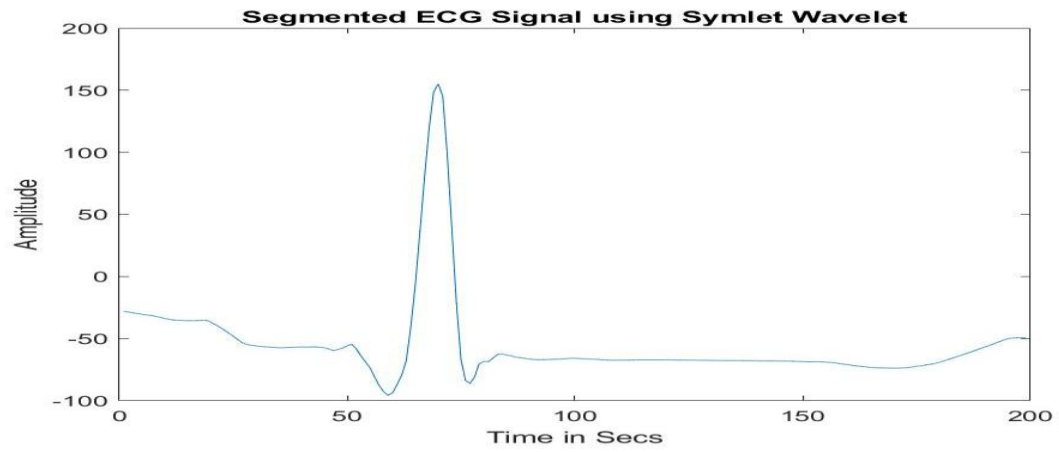


Figure 6.5 Feature extraction using symlet and daubechies wavelet

```
Command Window

out =

    1×1 cell array

    {'Yes Verified'}
```

Figure 6.6 VerifiedOutput

We achieved 100% accuracy with FAR=8, FRR=2 and EER=6 when we performed our proposed method on some of records taken from MIT-BIH database. This study demonstrates that our new proposed approach outperforms previous works in terms of FAR, EER, FRR, and detection rate while highlighting the drawbacks of previous works. As can be observed from figure-6.7 shows average accuracy comparison plot using various segmentation of on-the-person MIT-BH data base.

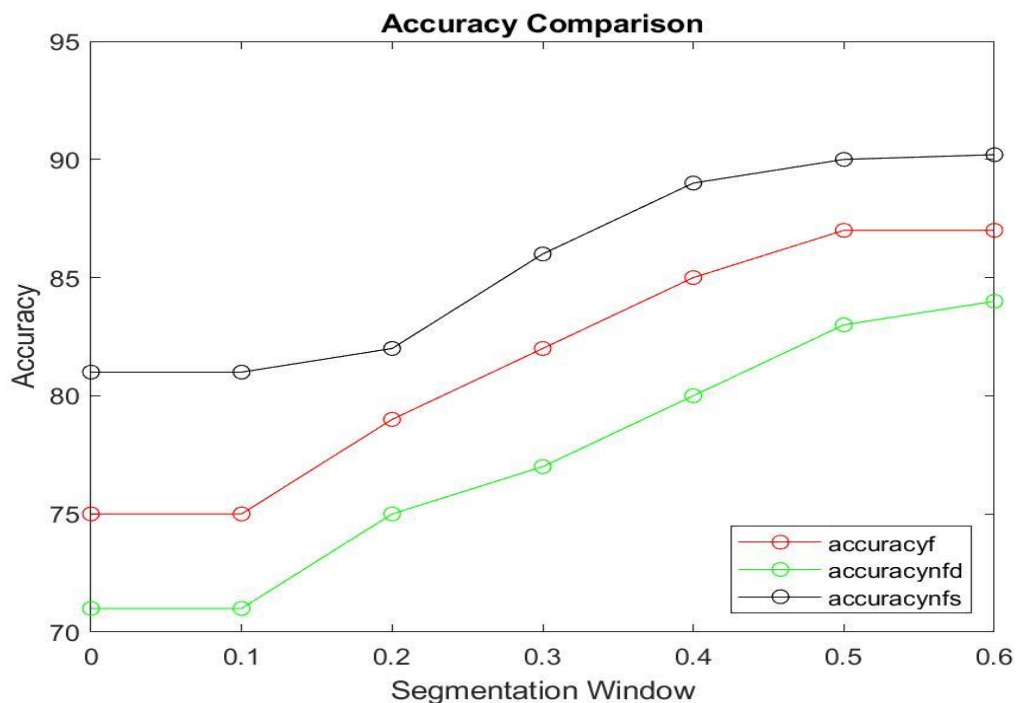


Figure 6.7 Accuracy comparison

CONCLUSION

Despite significant attempts to construct ECG-based biometric systems, some significant challenges have been disregarded when creating new algorithms. At first, the ECG data are imperfect and include noise and unfavourable information. As a result, the efficacy of ECG data has not been adequately examined, nor have the filtering, segmentation, and feature extraction techniques utilised in prior algorithms. This article examines in detail the effects of filtering type, segmentation, feature extraction, and person health state on ECG biometric systems utilising assessment metrics including FAR, EER, FRR, and accuracy. The proposed methodology not only highlights the shortcomings of past research but also improves on it in terms of FRR, EER, and FAR. The findings of this study will have a substantial impact on future ECG research subjects.

References:

- [1]. Zahid Akhtar, Christian Micheloni, G.L. Foresti, "Biometric Liveness Detection: Challenges and Research Opportunities", *IEEE Security and Privacy Magazine* 13(5):63-72, pp.1540-7993, Sep 2015
- [2]. D. Sasikala, S. Muthukumar and R. Sivaranjani, "Power efficient digital circuits for ECG front end data acquisition mobile system" *Inderscience online Ltd*, Vol. 9, No. 3-4, pp 261-272, Sep 2018.
- [3]. N. Erdogmus and S. Marcel, "Spoofing face recognition with 3D masks," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 7, pp. 1084–1097, Jul. 2014.
- [4]. A. Hadid, N. Evans, S. Marcel, and J. Fierrez, "Biometrics systems under spoofing attack: An evaluation methodology and lessons learned," *IEEE Signal Process. Mag.*, vol. 32, no. 5, pp. 20–30, Sep. 2015.
- [5]. N. Kose and J.-L. Dugelay, "On the vulnerability of face recognition systems to spoofing mask attacks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 2357–2361.
- [6]. J. Galbally, R. Cappelli, A. Lumini, G. Gonzalez-de-Rivera, D. Maltoni, J. Fierrez, J. Ortega-Garcia, and D. Maio, "An evaluation of direct attacks using fake fingers generated from ISO templates," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 725–732, Jun. 2010.
- [7]. Z. Akhtar, C. Micheloni, and G. L. Foresti, "Biometric liveness detection: Challenges and research opportunities," *IEEE Secur. Privacy*, vol. 13, no. 5, pp. 63–72, Sep. 2015.
- [8]. M. Komeili, N. Armanfard, and D. Hatzinakos, "Liveness detection and automatic template updating using fusion of ECG and fingerprint," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 7, pp. 1810–1822, Jul. 2018.
- [9]. P. P. K. Chan, W. Liu, D. Chen, D. S. Yeung, F. Zhang, X. Wang, and C.-C. Hsu, "Face liveness detection using a flash against 2D spoofing attack," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 2, pp. 521–534, Feb. 2018.
- [10]. N. Karimian, Z. Guo, M. Tehranipoor, and D. Forte, "Highly reliable key generation from electrocardiogram (ECG)," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 6, pp. 1400–1411, Jun. 2017.
- [11]. Sairul I Safie, John J Soraghan, and Lykourgos Petropoulakis. "ECG biometric authentication using Pulse Active Width (PAW)". In: *2011 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications (BIOMS)*. IEEE. 2011, pp. 1– 6.
- [12]. Filipe Canento et al. "Review and comparison of real time electrocardiogram segmentation algorithms for biometric applications". In: *Proc. 6th Int. Conf. Health Inform.(HEALTHINF)*. 2013.

- [13]. Qingxue Zhang, Dian Zhou, and Xuan Zeng. "A novel framework for motion-tolerant instantaneous heart rate estimation by phase-domain multiview dynamic time warping". In: IEEE Transactions on Biomedical Engineering 64.11 (2016), pp. 2562–2574.
- [14]. JM Irvine et al. "Heart rate variability: a new biometric for human identification". In: Proceedings of the International Conference on Artificial Intelligence (ICAI'01). 2001, pp. 1106–1111.
- [15]. Mohit Ingale1 , Renato Cordeiro , Siddartha Thentu, Younghee Park, Nima Karimian,"Ecg Biometric Authentication: A Comparative Analysis" IEEE Access vol.8, 1no. ,pp. 117853- 117866,23 June 2020.
- [16]. Shraddha R. Bajare and Dr.Vaishali V. Ingale," Ecg Based Biometric For Human Identification Using Convolutional Neural Network" IEEE – 45670. 10th ICCCNT 2019 July 6-8, 2019.
- [17]. Binish Fatimah , Priyanka G , Rehana Sultana and Rekha N" Analysis of ECG for biometric identification" IEEE – 49239 11th ICCCNT 2020 July 1-3, 2020.
- [18]. S. Haykin, Kalman Filtering and Neural Networks, vol. 47. Hoboken, NJ, USA: Wiley, 2004.
- [19]. R. Sameni, M. B. Shamsollahi, C. Jutten, and G. D. Clifford, "A nonlinear Bayesian filtering framework for ECG denoising," IEEE Trans. Biomed. Eng., vol. 54, no. 12, pp. 2172–2185, Dec. 2007.
- [20]. D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in Proc. KDD Workshop, Seattle, WA, USA, 1994, vol. 10, no. 16, pp. 359–370.
- [21]. C. A. Ratanamahatana and E. Keogh, "Making time-series classification more accurate using learned constraints," in Proc. SIAM Int. Conf. Data Mining. Philadelphia, PA, USA: SIAM,2004,pp.11–22.

PAPER PUBLICATION DETAILS

Selected Journal : International Journal of Image, Graphics and Signal Processing(IJIGSP)

Paper number : PAPER008651

File Number : PAPER008651 -1 [Manuscript Download](#)

Paper Title : ECG BIOMETRIC AUTHENTICATION: A COMPARATIVE ANALYSIS

Paper Authors : Prof.G.Manamadha Rao,Vaddadi Charan Kumar, Lingala Vijay Kumar, Bontu Sowmya, Saragula Prasanth Kumar

Research Area Of The Paper : Signal, Image and Video Processing

Paper Keywords : electrical impulse signal, encryption,filtering, segmentation, feature extraction, enrollment, authentication

Paper abstract : The main objective of this study is to use ECG signals for user identity verification. An electrocardiogram is an electrical impulse signal that indicates different heart conditions, and ECG has exceptional discriminative qualities associated with human identification. Despite recent progress in ECG-based encryption, differential feature extraction and efficient pattern clustering still face several difficulties. In the current environment, attestation techniques have become a crucial and urgent responsibility to secure the confidentiality of equipment and sensitive information. ECG biometric authentication is the suggested authentication technique that can give users access. ECG biometric systems typically consist of basic processes: sensing, pre-processing (filtering, segmenting), feature extraction, and matching. The paragraph provides an overall description and flow of the enrollment and authentication stages of the ECG-based biometric authentication procedure. Our analysis shows that our approach to ECG biometric identification outperforms existing techniques that lack the ability to quickly extract features, filter, segment, and match.

Last upload Time : 2023-03-28 14:25:03

Paper Submission Time : 2023-03-28 14:25:03

Status : Under review