

**FACIAL RECOGNITION USING SUPERVISED ALGORITHMS**

**(A CASE STURDY OF SUPPORT VECTOR MACHINES AND ARTIFICIAL NUERAL NETWORK)**

BY

**ADESAYO BUKUNMI ISRAEL**

**20D/7HEC/00628**

A Project Submitted to the Department of Electrical and Computer Engineering, Faculty of Engineering and Technology Kwara State University, Malete in Partial Fulfilment of the Requirements for the Award of Bachelor of Engineering (B.Eng.) Degree in Electrical and Electronics Engineering.

DECEMBER 2022

**DECLARATION**

I hereby declare that this project titled ‘**FACIAL RECOGNITION USING SUPERVISED ALGORITHMS’** is my own work and has not been submitted by any other person for any degree or qualification at any higher institution. I also declare that the information provided therein is mine and those not mine are properly acknowledged.

Adesayo Bukunmi Israel Signature and Date

# CERTIFICATION

This is to certify that this project titled “**FACIAL RECOGNITION USING SUPERVISED ALGORITHMS**” was carried out by **ADESAYO BUKUNMI ISRAEL**. The project has been read and approved as meeting the requirements for the award of Bachelor of Engineering (B.Eng.) Degree in Electrical and Electronics Engineering in the department of Electrical and Computer Engineering, Faculty of Engineering and Technology, the Kwara State University, Malete.

Dr. (MRS.) BALOGUN Date

Supervisor

Dr. Engr. Lambe Mutalab Adesina Date

Head of Department

External Examiner Date

# DEDICATION

I dedicate this research work to Almighty God who made this program a success for me and my lovely family for their support and words of encouragement rendered to me during my study. Also, to my colleagues and bosses who granted me the necessary permissions to run this program. May God bless and reward you all Amen.

# ACKNOWLEDGEMENT

I would like to acknowledge contributions from colleagues and individuals without whom this project will not be a success. Particular thanks are due to my amiable Supervisor (Dr. (Mrs.) Balogun), tireless coordinator (Dr. Engr. Olalekan Ogunbiyi), and evergreen H.O.D. (Dr. Engr. Lambe Mutalab Adesina) also to all my institution lecturers in the department of Electrical and Electronics Engineering. I will like to specially appreciate my family for their Love, kindness and support. I thank God for the lives of my friends and fellow colleagues throughout my Top-Up program. The Electrical and Computer Engineering Department of Kwara State University (KWASU) has provided assistance for this research project.

# TABLE OF CONTENTS

[CERTIFICATION iii](#_Toc127078445)

[DEDICATION iv](#_Toc127078446)

[ACKNOWLEDGEMENT v](#_Toc127078447)

[TABLE OF CONTENTS vi](#_Toc127078448)

[ABSTRACT viii](#_Toc127078449)

[CHAPTER ONE 1](#_Toc127078450)

[INTRODUCTION TO THE RESEARCH 1](#_Toc127078451)

[1.1 BRIEF OVERVIEW 1](#_Toc127078452)

[1.2 INTRODUCTION TO BIOMETRIC RECOGNISTION 1](#_Toc127078453)

[1.5 RESEARCH JUSTIFICATION 3](#_Toc127078454)

[1.6 SCOPE AND LIMITATION OF THE PROJECT 3](#_Toc127078455)

[1.7 REPORT ORGANIZATION 3](#_Toc127078456)

[CHAPTER TWO 5](#_Toc127078457)

[LITERATURE REVIEW 5](#_Toc127078458)

[2.1 INTRODUCTION 5](#_Toc127078459)

[2.2 LITERATURE REVIEW 5](#_Toc127078460)

[2.2.1 Machine Learning 13](#_Toc127078461)

[2.2.1.1 Machine Learning Algorithms 14](#_Toc127078462)

[2.2.2 A TOUR TOWARDS MACHINE LEARNING ALGORITHMS 15](#_Toc127078463)

[2.2.2.1 DECISION TREE ALGORITHMS 16](#_Toc127078464)

[2.2.2.2 SUPPORT VECTOR MACHINES (SVM) 17](#_Toc127078465)

[2.2.2.3 NEURAL NETWORK ALGORITHMS 19](#_Toc127078466)

[2.2.2.4 DECISION TREE ALGORITHM 21](#_Toc127078467)

[2.3 REVIEW OF RELATED WORKS 24](#_Toc127078468)

[CHAPTER THREE 27](#_Toc127078469)

[METHODOLOGY 27](#_Toc127078470)

[3.0 INTRODUCTION 27](#_Toc127078471)

[3.1 PRESSING RESEARCH QUESTIONS 27](#_Toc127078472)

[3.2 RESEARCH DESIGN 27](#_Toc127078473)

[3.3 RESEARCH STRATEGY 28](#_Toc127078474)

[3.3.1 Artificial Neural Network (ANN) 30](#_Toc127078475)

[3.3.2 Support Vector Machine (SVM) 35](#_Toc127078476)

[3.4 Performance Analysis of SVM and ANN 38](#_Toc127078477)

[3.4.0 Confusion matrix 38](#_Toc127078478)

[3.5 Summary 40](#_Toc127078479)

[CHAPTER FOUR: 41](#_Toc127078480)

[4.0 INTRODUCTION 41](#_Toc127078481)

[4.2 Data Preprocessing and Feature Extraction 42](#_Toc127078482)

[4.3 Comparative performance of Algorithms 44](#_Toc127078483)

[4.4 False Acceptance Rate 47](#_Toc127078484)

[4.5 False Rejection Rate 47](#_Toc127078485)

[4.6 Accuracy 47](#_Toc127078486)

[4.7 Precision 48](#_Toc127078487)

[4.8 Recall 48](#_Toc127078488)

[4.9 F1\_Score 48](#_Toc127078489)

[CHAPTER FIVE 52](#_Toc127078490)

[CONCLUSIONS AND RECOMMENDATIONS 52](#_Toc127078491)

[5.1 CONCLUSION 52](#_Toc127078492)

[5.2 RECOMMENDATION 53](#_Toc127078493)

[5.3 LIMITATIONS 53](#_Toc127078494)

[REFERENCES 54](#_Toc127078495)

# ABSTRACT

When compared to other existing technologies, it has been found that multiple antenna systems with a significant number of antenna components at base stations are energy efficient and capable of greatly increasing data rates without requiring any more bandwidth. Currently published research largely assume the effectiveness of tried-and-true tactics in a big MIMO scenario, but there are still unresolved issues with hardware complexity, computing complexity, and strength efficiency. This study aims to assess the advantages of massive MIMO deployment, particularly in terms of energy efficiency considering the rising cost of energy in the modern world. The ability to reduce power consumption in the 5G wireless community situation where very complex antenna strategies like large MIMO technology are used has become extremely important. As a result, this article has studied quite a few possibilities to which Energy Efficiency features provided by large MIMO (MM) structures can be of benefit.

# CHAPTER ONE

# INTRODUCTION TO THE RESEARCH

## 1.1 BRIEF OVERVIEW

Face recognition has been an active area of ​​research over the years. Unlike other biometric identification models/methods such as iris, finger print, face, DNA and so on, the facial recognition system does not rely heavily on the collaboration of the person. Among all these biometrics, face is more attractive as it provides information such as identity, expression, gender and age of a person. This inconsistency or lack of confidence in the system leads to the requirement for a face recognition system that is resilient to facial lighting conditions and different variations such as pose, facial expression, or occlusions such as glasses or beard. One of the most popular methods or security measures used nowadays is the increasingly powerful low-processing facial recognition systems, such as B. Telephones and Raspberry Pi's.

## 1.2 INTRODUCTION TO BIOMETRIC RECOGNISTION

Biometric recognition can be said to be a process of identification by defined structures example of which includes iris, finger print, face, DNA and so on,

Face recognition can be traced back to the sixties and seventies of the last century, traditional face recognition method depends solely on the structural features of the face and the color characteristics of the face, written algorithms identify facial features by extracting landmarks, or features, from an image of the subject's face, for example, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, cheekbones, and jaw, these features are then used to search for other images with matching features. These kinds of algorithms are complicated and require lots of computations which in turn leads to slow performance of the system. Sometimes these computations can also be inaccurate when the faces show clear emotional expressions, since the size and position of the landmarks can be altered significantly in such circumstance.

**1.3 PROBLEM STATEMENT**

An ideal face recognition system should be able to recognize new images of either a trained or an untrained face and should not pick up noise around the image introduced during the process of capturing an image. Most systems only work well with images captured under restricted or laboratory conditions where lighting, pose and camera parameters are tightly controlled. This requirement is far too stringent to be useful in many data mining situations when only a few sample images are available, such as detecting people from surveillance video of a planetary sensor network or searching historical film archives. The past decade has seen many improvements in research for a reliable face recognition system, and many techniques have been used in face recognition system performance modeling. Unfortunately, the existing facial recognition technology still needs a lot of improvement as data accuracy is still an issue in terms of the system's performance, not only that, other challenges such as sensitivity to changes in lighting, subject distance to camera and pose, and an increased computational load when searching or retrieving images still poses a major threat when it comes to compromising the security system in great distress. Hence, this work seeks to evaluate existing face recognition techniques and classifiers to see if they can be used in a system with limited computing power.

**1.4 AIM AND OBJECTIVES**

This aim of this work is to evaluate the performance of three supervised learning algorithms namely Decision tree, Support Vector Machine and Conventional Neural Network on face recognition. In doing so, it compares traditional methods with dive deep machine learning, the concept of supervised, unsupervised learning, reinforcement learning, and explains important concepts and theories needed to understand the problem area.

The specific objectives are:

1. Getting Multiple (Facial) Data (Data acquisition stage): to acquire face images, process the images acquired and extract salient features from the preprocessed image.
2. Testing case study with acquired data (testing stage): to use the decision tree, support vector machine and conventional neural network as classifiers.
3. Performance Evaluation Stage (Evaluation Stage): to evaluate the performance of the three classification algorithms.

## 1.5 RESEARCH JUSTIFICATION

This research work will provide an optimal solution for the face recognition model at the end of the project. In addition, it supports structural document representation and eliminates the tedium of flawed facial recognition systems. This research will also contribute to the existing literature in this area and serve as a guild or blueprint for undergraduate students.

## 1.6 SCOPE AND LIMITATION OF THE PROJECT

The work aim at comparing the performance of three common supervised learning algorithm namely: decision tree support vector machine and conventional neural network on face recognition. The three algorithms were applied at the classification stage of biometric based facial recognition system.

## 1.7 REPORT ORGANIZATION

This report has been carefully organized into five chapters for easy comprehension and understanding by the readers:

Chapter one presents the general overview into the topic, statement of the problem, aim and objective and so on. Chapter two contains the literature review carried out to gain better knowledge about the topic. Chapter three contains detail of methodology employed towards achieving the aim of embarking on this work. Chapter four presents the result and discussion of results. Chapter five contains the conclusion an recommendation for future improvement on this work

# CHAPTER TWO

# LITERATURE REVIEW

## 2.1 INTRODUCTION

This chapter provides a comprehensive face recognition classification enriched with different variables, which facilitate introducing organized categories of solutions for face recognition. In the 1970s, Goldstein (1971) used 21 specific subjective markers such as hair color and lip thickness to automate the recognition and over the years technology reinvents itself and this phenomenon is a healthy sign for the advancement to technology (Tajinder Kumar et al 2017). This has always been the case in technology development, thus accepting programs or software that could still be improved is no game in computational science or the engineering world and the same is the case in biometric technology where the technical processing of earlier experiments has become more fascinating and effective with the passage of time. Although humans can generally perform facial recognition with greater accuracy than any computer, the human memory is less adept at memorizing a large dataset of faces, which makes automatic facial recognition algorithms vital (V. Kazemi et al June 2014) but a face recognition system where there will be input of an arbitrary image will search in database to output people’s identification in the input image, bridging the gap of large dataset and thereby reducing the lag that may occur as a result of human inefficiency.

## 2.2 LITERATURE REVIEW

**Theoretical Background of Biometrics**

Biometrics are measurements and calculations of the human body that are mostly based on human traits. Biometrics (from the Greek bios, which means "life," and metron, which means "measure") refers to two distinct domains of study and application. The collection, synthesis, analysis, and management of quantitative data on biological communities such as forests is the first, and it is utilized in biological studies, including forestry. Biometrics, in the context of biological sciences, has been researched and utilized for numerous generations and can be thought of as "biological statistics" (Secure Technology Alliance – The Digital Security Industry’s Premier Association, n.d.). The act of establishing or certifying something (or someone) as authentic is known as biometric authentication, and it is employed in computer science as a type of identification and access control.

**Historical Background of Biometrics**

The explorer from Europe Joao de Barros was the first person known to record an example of fingerprinting which is a form of biometrics. 14th century. Chinese merchants used ink to take children’s fingerprints for identification purposes (Bhattacharyya et al., 2009) .

Alphonse Bertillon studied body mechanics and measurements in 1890, Alphonse Bertillon studied body mechanics and measurements to help in identifying criminals. The police used his method known as the Bertillonage method, until it falsely identified some subjects. This Bertillon method was soon abandoned in favor of fingerprinting, which was brought back into use by Richard Edward Henry of Scotland Yard.

Karl Pearson, an applied mathematician, studied biometric research at University College London in the early twentieth century. Through his research into statistical history and correlation, which he applied to animal evolution, he made significant contributions to the discipline of biometrics. The method of moments, the Pearson system of curves, correlation, and the chi-squared test were among his historical contributions.

Signature biometric authentication processes were created in the 1960s and 1970s, but the biometric sector remained stagnant until the military and security agencies investigated and developed biometric technology that was not limited to fingerprints.

Biometric authentication is a rapidly emerging and contentious sector in which civil liberties organizations are concerned about privacy and identification issues. Biometric legislation and regulations are currently being developed, and biometric industry standards are being tested. Face recognition biometrics has not yet reached the level of widespread adoption that fingerprinting has, but with constant technological advancements and the threat of terrorism, researchers and biometric developers will continue to push this security technology forward into the twenty-first century. Biometric traits can be classified into two categories in a modern approach:

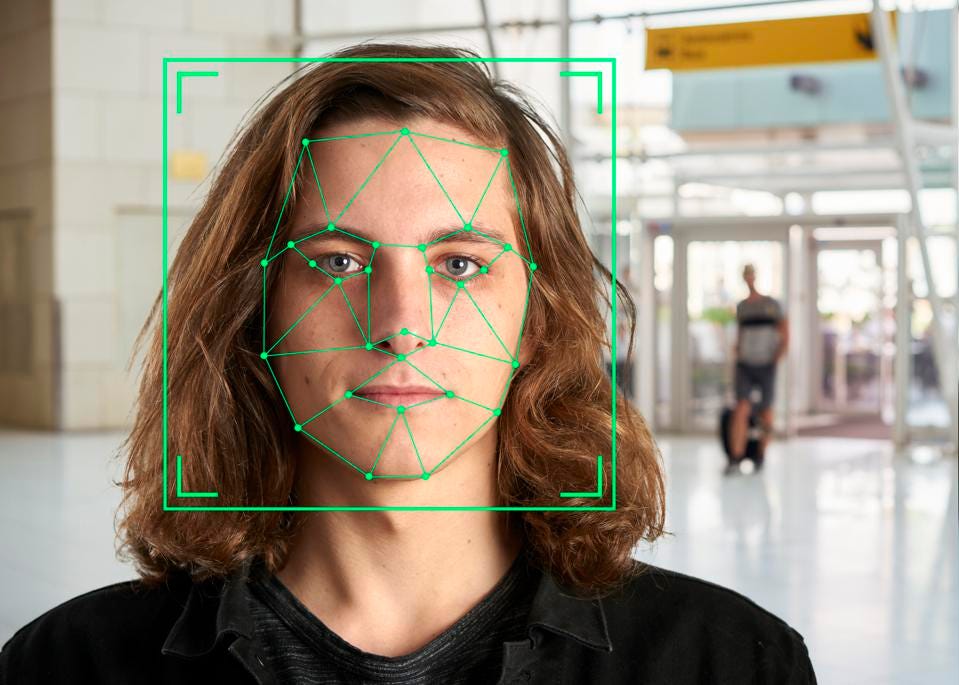
* Physiological is linked to the shape of the body, thus it differs from one individual to the next. Biometrics include fingerprints, face recognition, hand geometry, and iris recognition, e.t.c.
* Behavioral refers to a person's actions. Signatures, keystroke dynamics, and face are some examples in this instance. Because face differs from person to person, it is sometimes considered a physiological biometric. This project aims at the behavioral approach in biometrics to determine the more superior algorithm.

In a brain-machine interface, a new trend has emerged that combines human perception with a computer database. Cognitive biometrics is the name given to this method. Cognitive biometrics is based on distinct brain responses to stimuli that might be used to initiate a database search on a computer.

**Techniques and Technologies for Biometric Authentication**

After reviewing the historical background and terminology of Biometrics, the techniques and technology for biometric authentication have been developed for the physiological and behavioral biometric characteristics. The commonly used techniques for Biometric Authentication are briefly discussed below.

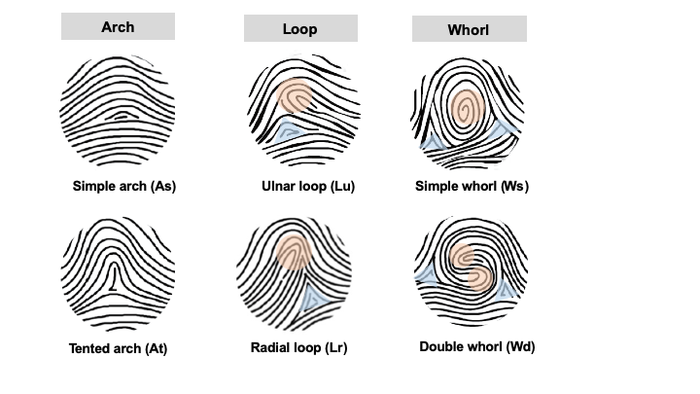
**Facial Recognition Technology**: A computer application for automatically identifying or validating a person from a digital image or a video frame from a video source is known as a facial recognition approach. It's the most natural way of identifying yourself biometrically (Bhattacharyya et al., 2009). Facial recognition technology has lately split into two categories: Facial metric and Eigen faces. Facial metric technology is based on the creation of specific facial features (the system often looks for the placement of eyes, nose, and mouth, as well as the distances between these features), as shown in Figures 2-1 below.



**Figure 2.1 Facial Recognition**

The facial area is resized to a predetermined size (e.g. 150-100 points). This normalized face image is called the canonical image. The facial metrics are then calculated and saved in a face template. A common template of this type is between 4 - 6 KB in size. There are systems with sizes as little as 100 bytes. The Eigen Face approach shown in figure 2-2 uses a predetermined collection of 100 to 150 eigen faces to categorize faces based on their degree of similarity. The generated eigen faces will look as light and dark regions grouped in a specified pattern. This pattern demonstrates how different aspects of a face are highlighted. It must be assessed and graded. A pattern will be used to assess symmetry, facial hair, the location of the hairline, as well as the size of the nose and mouth. Other eigen faces have patterns and are far less simple to identify, and the image of the eigen face may look hardly anything like a face.  Each of the 150 eigen faces is awarded a degree of fit, and only the 40 template eigen faces with the highest degree of fit are required to rebuild the face with 99 percent accuracy. Face Recognition software is used throughout the process (Bhattacharyya et al., 2009).

**Fingerprint Technology:** The automatic technique of determining an individual's identity based on a comparison of stored fingerprint information with input fingerprint information is known as fingerprint recognition. It is one of the most well-known biometrics for computer system authentication. The impressions/patterns present on a human finger are known as fingerprints. These impressions become more noticeable as people become older, but the structures remain the same (Ramaswamy et al., 2010). Fingerprints do not alter unless there is a physical disruption, such as an accident or working in a field where caustic or hot materials are used, which can destroy fingerprints (Pakutharivu & Srinath, 2015).



**Figure 2.2 Finger Prints**

This is quite helpful because to obtain fingerprints of missing persons, or of unidentified criminals gathered from crime scenes can be used in tracking and finding the persons themselves. Fingerprint recognition technologies are popular because they have a variety of advantages. The fact that it is widely accepted in the legal profession is one of its most significant advantages. It is the most cost-effective, fast, and reliable method of identifying a person. Due to the rarity of two people having identical finger prints, fingerprint identification is universally acknowledged as a highly accurate means of authentication (Swann, 2008).

**Speech Recognition Technology**: Speech or face can be defined as physiological trait due to pitch of a human face, but since face recognition is concerned with the uniqueness of how a person makes the speech for identification purposes, it is considered a behavioral trait. (Bhattacharyya et al., 2009). Speech can be utilized in biometric systems to identify a person based on the speech that was captured during the enrollment phase (Habeeb, 2019). Person or face recognition systems and approaches have two key uses. Verification or authentication occurs when a person claims to be of a particular identity and the face is utilized to verify that assertion. Identification, on the other hand, is the effort of discovering the identity of an unknown person. In certain ways, person verification is a 1:1 match, in which one person's face is matched to a single template, whereas person identification is a 1:N match, in which the face is matched to several templates. Face recognition is concerned with the vocal features that make speech, rather than with the sound or pronunciation of speech. The size of the vocal tract, mouth, nasal cavities, and other speech processing mechanisms in the human body influence face characteristics. The acoustic aspects of speech that have been shown to differ across persons are used in person recognition. Acoustic patterns reflect both anatomy (such as throat and mouth size and shape) and acquired behavioral habits. (for example, pitch of face, speaking manner) (Bhattacharyya et al., 2009).

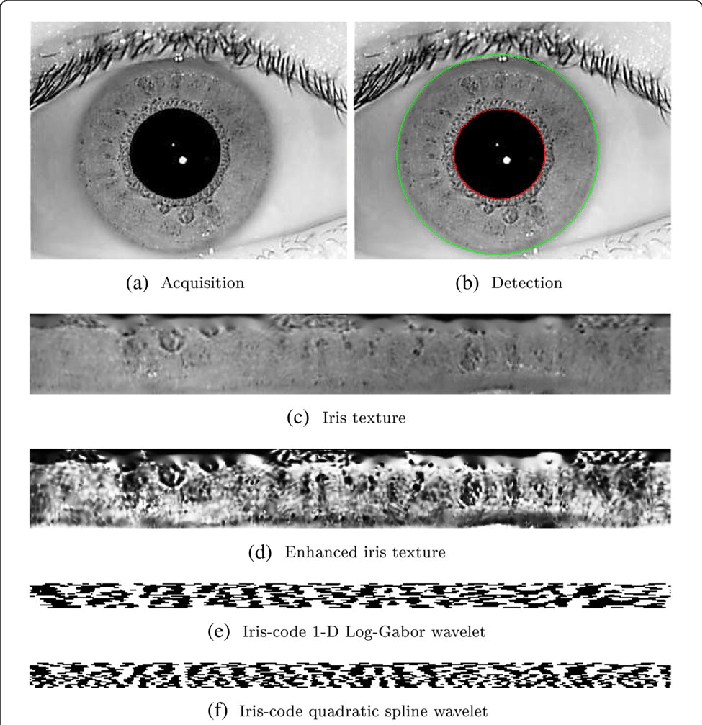
**Signature Verification Technology:** Signature recognition is based on the dynamics of signing rather than a straightforward comparison of the signature thereafter. The direction, pressure, acceleration, and length of the strokes are used to calculate the dynamics, as well as the number of strokes and their duration. The most apparent and significant benefit is that a fraudster will not be able to learn how to make a signature by merely glancing at one that has already been made. The trademark dynamics are captured using a variety of technologies. These are either regular tablets or gadgets with a specific function. Tablets are capable of capturing 2D coordinates as well as pressure (Bhattacharyya et al., 2008).



**Figure 2.3 Electronic Signature on a device**

Movements in all three dimensions may be captured using special pens. Tablet computers have two significant drawbacks. To begin with, the resulting digitalized signature does not resemble a typical user signature. Second, the user does not see what he or she has already written while signing. To view the signature, he or she must glance at the computer monitor (Furui, 1997). For many (inexperienced) users, this is a significant disadvantage. Some unique pens function like regular pens; they have an ink cartridge and may be used to write on paper.

**Iris Recognition Technology:** Iris recognition is a biometric identification method that relies on pattern recognition algorithms based on high-resolution photographs of a person's irises. An infrared imaging process is used to capture the iris, which distinguishes it from the pupil and sclera of the eye. An examination of the detail inside the trabecula meshwork of the iris is then used to create a template (Wildes, 1997). Iris identification technology employs camera technology to capture photos of the iris' detail-rich, complicated structures, with modest infrared light decreasing specular reflection from the convex cornea. These photos are transformed into digital templates in order to create mathematical representations of the iris that allow for unequivocal affirmative identification of a person. A unique gray scale camera captures the iris pattern at a distance of 10- 40 cm from the camera. After obtaining a grayscale picture of the eye, the program attempts to detect the iris within the image.



**Figure 2.4 The Iris Scan**

If an iris is discovered, the program generates a net of curves to cover it. The program generates the iris code based on the darkness of the spots along the lines. Two factors must be considered in this case. First, because the total darkness of the image is modified by lighting conditions, the darkness threshold used to determine whether a specific location is dark or bright cannot be static; instead, it must be dynamically calculated based on the overall picture darkness. Second, when the pupil size varies, so does the size of the iris. An appropriate transformation must be performed before computing the iris code.

**Retina Geometry Technology:** This technology is based on the physiological blood vessel pattern in the retina of the eye (Hildebrand & Fielder, 2011), since the blood vessels at the rear of the eye have a distinct pattern that differs from eye to eye and person to person. Because the retina is not directly visible, it requires illumination from a coherent infrared light source. Blood veins in the retina absorb infrared radiation quicker than the surrounding tissue. After that, the picture of the retina blood vessel pattern is examined.

Retina scans need the removal of the person's spectacles, placing their eye close to the scanner, staring at a precise spot, remaining steady, and focusing on a specific position for around 10 to 15 seconds until the scan is finished. A retinal scan includes projecting a low-intensity coherent light source into the retina, which illuminates the blood vessels, which are then photographed and studied. The blood vessel patterns are read using a coupler. It is presently impossible to manufacture a human retina, hence a retina scan cannot be falsified. Furthermore, a deceased person's retina decays far too quickly to be utilized to fool a retinal scan.

The evolution of facial recognition systems is mainly caused by scientific advances in machine learning with powerful data analysis algorithms. Consequently, facial recognition is becoming a reliable technology for identity verification. However, Machine learning is the concept that machines can perform specific tasks without instructions but rather based upon identifying patterns and receiving rewards (Russel & Norvig 2010). Training time for the machine learning algorithms is when the model is built out of the data. When a model has been created, it can be used to make decisions on any future data. There are three main types of machine learning algorithms, they include supervised, unsupervised and reinforcement learning (Timmy Schenkel et al 2019).

## 2.2.1 MACHINE LEARNING

Machine learning can appear in many guises. We now discuss a number of applications, the types of data they deal with, and finally, we formalize the problems in a somewhat more stylized fashion. The latter is key if we want to avoid reinventing the wheel for every new application. Instead, much of the art of machine learning is to reduce a range of fairly disparate problems to a set of fairly narrow prototypes. Much of the science of machine learning is then to solve those problems and provide good guarantees for the solutions (Alex Smola etval 2008).

Machine learning consists of designing efficient and accurate prediction algorithms. Theoretical learning guarantees for an algorithm depend on the complexity of the concept classes considered and the size of the training sample. Since the success of a learning algorithm depends on the data used, machine learning is inherently related to data analysis and statistics (Mehryar Mohri et al 2012). More generally, learning techniques are data-driven methods combining fundamental concepts in computer science with ideas from statistics, probability and optimization.

It is worth at this point, to pause and consider the problem that a machine learning algorithm is designed to solve, lets consider the problem of predicting annual salary from age where x = age and y = salary. This is a supervised learning problem where for example we have a label yn (the salary) associated with each example xn label (the age). A dataset is written as a set of pairs {(x1, y1), . . . ,(xn, yn), . . . ,(xN, yN)}. The table of examples {x1, . . . , xN} is often concatenated, and written as X ∈ RN×D.

This will lead us into the types of machine learning algorithms, which had been spoken earlier in the earlier part of this chapter and will be further discussed in the chapter three (3) of this project.

## 2.2.1.1 MACHINE LEARNING ALGORITHMS

There are four main classification of machine learning algorithms, they include supervised, semi-supervised Learning, unsupervised and reinforcement learning

**Supervised learning** is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labelled training data consisting of a set of training examples. In supervised learning, all data must be labelled with what the data contains (Russel & Norvig 2010). An example would be trying to determine if a picture contains either a cat or a dog. If the set of output is infinite, it becomes a regression problem (Timmy Schenkel et al 2019).

**Semi-Supervised Learning:** when dealing with semi-supervised Learning Algorithms, Input data is a mixture of labelled and unlabelled examples. There is a desired prediction problem but the model must learn the structures to organize the data as well as make predictions. Example problems are classification and regression. Example algorithms are extensions to other flexible methods that make assumptions about how to model the unlabelled data.

**Unsupervised Learning** there are no labels which the algorithm can train on, so the algorithm must try to find patterns in the data without any help (Russel & Norvig 2010). The most common task of unsupervised learning is clustering, which is to detect useful clusters in some input data. An example of clustering would be that an algorithm may identify a set of articles being in the same category, even though it was not explicitly told to make splits in the data on categories.

**Reinforcement Learning** (Russel & Norvig 2010). This type will either reward or punish the algorithm depending on the choices it makes. The goal is to get as many rewards as possible. An example of reinforcement learning is the program AlphaGo developed by Google Deep mind which learned to play the game Go. This program was used to beat the Go grandmaster Lee Sedol in 2016 (Wang et al. 2016).

## 2.2.2 A TOUR TOWARDS MACHINE LEARNING ALGORITHMS

We have already considered the classification of Machine Learning but it goes further than that, Machine learning algorithms can be further broken down and we have already seen that, the performance of the system is solely or singularly dependent on the type of approach applied and to get optimum response from the system, it needs to be trained although performance on the training set is not a good indicator of precise predictive performance but if the model is iterated many times using a limited size data set, then some over fitting may occur, we should consider a range of different types of model in order to find the best one.

Below is the list of commonly used Machine Learning (ML) Algorithms:

* Linear regression
* Logistic regression
* Decision tree
* SVM algorithm
* Naive Bayes algorithm
* SVM algorithm
* K-means
* Random forest algorithm
* Dimensionality reduction algorithms
* Gradient boosting algorithm and AdaBoosting algorithm

Most of these algorithms will not be discussed in this project but we will have a drive into Decision Tree, Support Vector Machines (SVM) and Neural network (NN)

## 2.2.2.1 DECISION TREE ALGORITHMS

Decision trees are powerful and attractive approach for classification and prediction. The attractiveness of decision trees is due to the fact that, in contrast to neural networks. Decision trees represent set of rules that helps in decision making. These Rules are within the ambit of human intellect and they may be applied accordingly or even directly used in any computer programming languages like SQL etc (tajinder kumar Saini 2018).

Decision tree methods construct a model of decisions made based on actual values of attributes in the data. Decisions fork in tree structures until a prediction decision is made for a given record. Decision trees are trained on data for classification and regression problems. Decision trees are often fast and accurate and a big favorite in machine learning.

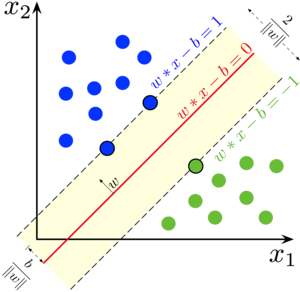
The most popular decision tree algorithms are:

* Classification and Regression Tree (CART)
* Iterative Dichotomiser 3 (ID3)
* C4.5 and C5.0 (different versions of a powerful approach)
* Chi-squared Automatic Interaction Detection (CHAID)
* Decision Stump
* M5
* Conditional Decision Trees

## 2.2.2.2 SUPPORT VECTOR MACHINES (SVM)

These were popular some years ago for solving problems in classification, regression, and novelty detection, an important property of support vector machines is that the determination of the model parameters corresponds to a convex optimization problem, and so any local solution is also a global optimum (M. Jordan 29 (2006).

The classification algorithm support vector machine (SVM) is one of the most influential algorithms in supervised learning (Zhang et al. 2016). The algorithm has been applied to numerous different areas in recognition, like identifying brain disorder such as Alzheimer's disease (Zhang, Wang & Dong 2014), spatiotemporal activity (Behmann, Hendriksen, Müller, Büscher & Plümer 2016) and multiple sclerosis (Zhang et al. 2016).



**Figure 2.5 Support Vector Algorithm**

There are multiple different SVM versions that can be used for facial recognition classification (Hsu, Chang & Lin 2008). Three examples of these SVM versions are the radial based function (RBF), polynomial and the linear version. These SVM versions have an impact on the accuracy and training time. The linear SVM shows to have the optimal accuracy and training time in relation to the other SVM versions when used in facial recognition.

The SVM is a classification method that can provide better accuracy for some other methods and in many fields (Z. Chent et al. 2007). In short, an SVM is an algorithm that works as follows. It uses nonlinear mapping to transform the original training data into a higher dimension. Within this new dimension, it searches for the linear optimal separating hyperplane (i.e., a "decision boundary" separating the tuples of one class from another). With suitable non-linear mapping to a sufficiently high dimension, data from two classes can always be separated by a hyperplane. The SVM finds this hyperplane using support vectors ("essential" training tuples) and edges (defined by the support vectors). SVM creates a line or hyperplane between two datasets for classification. SVM performs best among the current classification techniques because of its ability to capture non-linearities. The learning process involves optimizing a cost function that is provably convex. This contrast to neural network approaches where the presence of false local minima in the error function can complicate the learning process. In contrast to neural network approaches, the architecture is determined by the algorithm and not found through experimentation. Different kernel functions can result in different performance. Two types of kernel functions for SVM are presented; the linear and the polynomial. Both are used in these experiments. TS The TS (J. Gaast et al. 2014) works with neighborhood structures and uses a short-term memory structure called Tabu, which is essentially a list of forbidden moves or solutions. For the TS problem, we consider the well-known one-flip neighborhood. The measure of similarity is now based on the Euclidean norm such that points that are closest together in Euclidean space are grouped under one and only one cluster. The TS can find the border instances that belong to neighboring classes and have the minimum distance. The closest pair can be defined as min(xi - xj), where xi ȯ&i and xj ȯ &j. These pairs (pij) are separated from all other instances (resti and restj) of the two neighboring clusters. Randomly selected instances (rani) of resti (from Ci) are also included in SVM training. The prohibited moves (instances of Ci – pij – rani) are contained in a taboo list for Ci.

The final list of Ci contains the closest pairs (pij) of two neighboring clusters and the randomly chosen instances (rani). For example, if the total number of instances of Ci is 20, the closed pairs are 5 (pij), and the randomly chosen instances of Ci are 3 (rani), then the forbidden moves are: 12 (20-5-3). The total instances for the training SVM will be 8 instead of 20. The TS is used to support searching to find boundaries for the SVM. Instances closer to decision boundaries are most important to SVM. DT\_SVM For each cluster Ci, a subset of instances Si is selected: the boundary points between two neighboring clusters (Ci and Cj) and other randomly selected instances from the Ci. Instead of the entire set of instances of a cluster Ci, this subset is used for SVM training.

The proposed method (DT\_SVM) has the following steps:

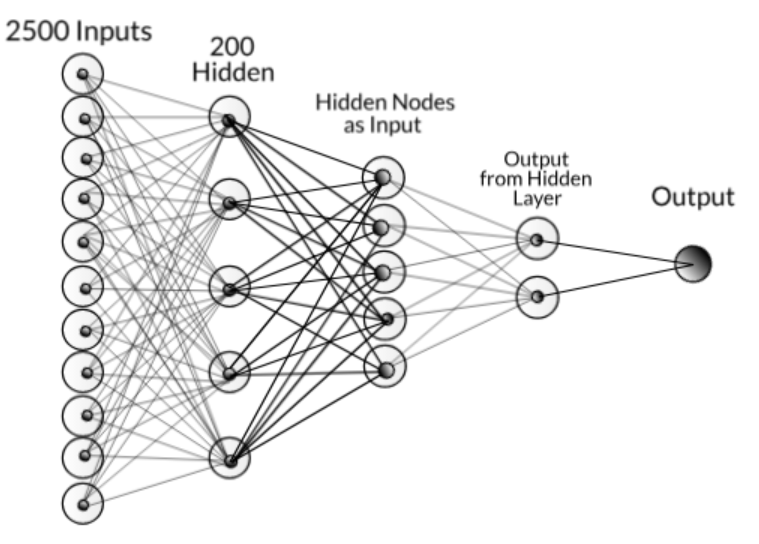
1. Use DTPL for the training set and receive the leaves which contain instances in clusters.
2. Gather all the leaves of DTPL and use them as clusters.
3. From a cluster prepare a subset of instances containing the closest pairs (CP) of instances that belong to adjacent clusters -using TS- and a randomly selected instance from the rest of instances of the cluster.
4. SVM training: The minority class instances (examples) from each leaf can be removed in order to have pure clusters.

## 2.2.2.3 NEURAL NETWORK ALGORITHMS

Neural Networks are models that are inspired by the structure and/or function of biological neural networks. A neural network is a collection of units connected in a specific pattern to enable communication between the units. The test dataset and the training dataset should be disjoint (so that test data is not used during training). For the neural network in the input layer, the neurons correspond to dataset prediction attribute values, and the output layer represents the predicted classes (S, Haykinl 2005), (Fausett L. 1994). The specification of a typical neural network model requires choosing the type of inputs, the number of hidden units, the number of hidden layers, and the connection structure between the input and output layers. The main characteristic of neural networks (NN) is their ability to generalize information and their tolerance to noise. NNs are often viewed as black boxes because their predictions cannot be clearly explained. The performance of NNs is sensitive to the specific architecture used to lay out the compute units. NN, which are nonlinear statistical data modeling tools, can be used to model complex relationships between inputs and outputs or to find patterns in data. Therefore, for a number of items in different classes, the neural network can learn to classify items. It takes a while to learn, but then it can instantly classify new inputs.

They are a class of pattern matching that are commonly used for regression and classification problems but are really an enormous subfield comprised of hundreds of algorithms and variations for all manner of problem types. The most popular artificial neural network algorithms are:

* Perceptron
* Multilayer Perceptron (MLP)
* Back-Propagation
* Stochastic Gradient Descent
* Hopfield Network
* Radial Basis Function Network (RBFN)



**Figure 2.6 Neural Network Algorithm**

## DECISION TREE ALGORITHM

DTs are one of the most popular data mining techniques (John Tsiligaridis 2018). S. Acharia et al. (1995) used a tree structure to represent a division of space. A DT algorithm based on high frequency classes is built from data using probabilities (DTPL). A theorem for finding a complete DT and identifying don't care attributes is developed. To avoid repetition or repetition, the criterion of eliminating a branch is also applied. For classification purposes to reduce the training time of SVM using the whole data, the DT\_SVM algorithm is developed based on DTPL and TS. TS is a heuristic method for solving optimization problems (F. Glover 2012). TS works with a number of limitations. The taboo list contains the forbidden moves. A specific set of trains and eligibility criteria are used (J.Gaast et al. 2014). An iterative learning algorithm based on a sample selection called "SVC" is developed in (Z. Chent et al. 2007). In (H. Yu et al. 2003) a clustering-based (CB\_SVM) for hierarchical clustering of large datasets is developed. In many real-world classification applications, interest has focused on the use of NNs (S, Haykinl 2005), (Fausett L. 1994). The clusters created from the DTs can be linearly separable. In this case, the majority decision is applied. To do this, a node N is converted into a leaf and labeled with the most common class in D.

DTPL DTs divide the input space into hyper rectangles according to the target. The DT classifier is a method that can be used as a pre-processing step for SVM. The DTPL can be created in the following phases:

Phase 1: Discover the root (from all attributes)

where A: the attributes of the tuples and C the classes (attribute test). MP = max (P(EA)) //max attribute test criterion

Phase 2: Split the data into smaller subsets so the split is as pure as possible using the same formula. The measure of node contamination is the MP. Continue to the end of the attributes. The large frequency classes (DTPL) are also extracted. The CEB criterion eliminates redundant branches. Most decision tree inducers must rebuild the tree from scratch to reflect newly available data. For an attribute (attr1) with value v1, if there are tuples of attr2 that all have values ​​related to v1 (of attr1), then attr2 is named: whatever attribute. The branch elimination criterion (CEB) is used to avoid repetition and replication and is given by:

If the PCEB= 0, between two attributes (A1, A2) then A2 is don’t care attribute. The CEB criterion is valid when PCEB z 0

Theorem: the CEB criterion can determine the existence of a small DT with the best accuracy (100%, or complete) avoiding repetitions and replications.

Proof: Because, when the CEB criterion is valid, discourages the repetition.

The DTPL is used for the generation of the clusters from the leaves.

**Programming in OpenCV:** OpenCV acronym for Open Source Computer Vision Library is a library containing functions for computer vision. It is developed by Intel and now handled and supported by Willow Garage. The library is functional cross platform and runs on Windows, Android, FreeBSD, Maemo, iOS, OpenBSD, Linux and Mac OS. The current release of the library is obtained from the Sourceforge and they also provide the binaries for the user, so that they can develop according to their requirements. OpenCV makes use of CMake to compile source files to start using the library.

The main focus of this library is on the real-time image processing functionality and implementing the machine learning algorithms. By using this we can improve the cost of computation and take an initiative to advance the CPU – intensive applications. The areas of application where openCV can be useful are facial recognition system, mobile robotics, gesture recognition, segmentation, object identification, motion tracking and many more.

OpenCv also includes a statistical machine learning library that supports the above areas of application. The name of the functions that supports this library are decision tree learning, expectation maximization, gradient boosting trees, , Naïve Bayes classifier, k- nearest neighbor, artificial neural network, support vector machine(SVM) and many more.

The goals of the developing libraries of opencv are manifold. I have utilized these goals to the benefit of my project:

• Advance vision research by providing open source as well as optimized code for basic vision infrastructure.

• Distribute vision knowledge by providing a common infrastructure for all developers so that they can share their research and knowledge.

• Making portable, performance-optimized code available for free, so that others can contribute computer vision-based application.

The library was previously written in C language and because of this C interface the language is portable to different platforms. To increase the adoption of the openCV language, the wrapper class for different languages like C#, Python, Ruby and Java have been written.

## 2.3 REVIEW OF RELATED WORKS

A comparison between seven different facial recognition techniques was conducted on two different datasets by Parkhi et al. (2015). Their conclusion was that Face Net with alignment had the best accuracy and Fisher face had the worst accuracy. Their study only focused on the metric accuracy.

Tajinder Kumar et al 2017 in their research paper attempted to show the technological phenomenon where decision tree-based face recognition approach was proposed using SVM and SURF. In this approach, Pre-processing comes first that includes both the input image and all the images stored in the database. Secondly, the image processing operations is used to extract face feature. At last decision tree with SVM and SURF base technique is applied for training and testing purpose, they concluded by reaching a verdict that the proposed approach produced better result in respect of Error Rate, Matching Time and average accuracy graph.

In Evaluation of Face Recognition Algorithms Under Noise by Ansam Almatarneh 2019, The performance of each algorithm was measured by the accuracy of the algorithm, in this thesis, the influence of noise on traditional face recognition algorithms including PCA, 2DPCA, LDA, ICA, and DCT and SVM modem algorithms (convolution neural network (CNN) and Alex Net were introduced and Deep learning using CNN and Alex Net were the best accuracy results compared to traditional algorithms. While A. A. Abayomi-Alli et al in their International Journal of Computer Applications (0975 – 8887) Volume 105 – No. 18, November 2014 titled decision tree approach to facial image retrieval from databases,the decision tree was constructed using the coordinates and features of the images in the database; the tree was constructed based on the information gain generated. Implementation was done using C# programming language. Results from the performance experiments showed that the proposed model was able to perform facial image retrieval from large databases with an accuracy of 0.99 when used to retrieve 897 images of 31 subjects in the testing dataset. The writers agreed that the recognition performance of the system was satisfactory, it can further be improved with some modifications.

However, Harouna Naroua and his colledges in the year 2022 decided that they would go in search of the best performing method by undergoing a comparative study of classification algorithms. Using specifieded methodology, many systems were implemented with different supervised machine learning techniques such as: SVM, LR, ANN, RF, SVM, NB. For the evaluation of each technique on the ORL, Yale and Grimace databases, three performance measures were used simultaneously by calculating their average value: recognition rate, f-measure of the confusion matrix, and area under the ROC curve. The results obtained were good. And, the SVM, LR, ANN, RF, SVM, NB classifiers produced respectively the following average performances: 99.05%, 98.91%, 98.89%, 97.14%, 96.82% and 95.89% with SVM taking lead in the course of ranking.

Thai Hoang Le wasn’t satisfied so in his Research Article titled **Applying Artificial Neural Networks for Face Recognition,** he introduced Face recognition “as a visual pattern recognition problem”, he went on to conclude that for all steps of the recognition of human faces in 2-dimentional digital images, The experiments were done on a difficult face detection database which were widely studied (MIT + CMU database). The result showed that ABANN (AdaBoost and Artificial Neural Network) did not only get approximate detection rate and processing time AdaBoost detector, but also minimized false detections.

Meanwhile, Ansam Almatarneh submitted a dissertation to the School of Graduate Studies, Faculty of Engineering & Applied Science, Memorial University of Newfoundland, in October 2019 titled EVALUATION OF FACE RECOGNITION ALGORITHMS UNDER NOISE, he claimed Noise was one of the major deterrents affecting the efficiency of existing facial recognition systems. He went further to implement three cases of experiments which included no noise, added Gaussian noise, and added salt-and pepper noise. The performance of each algorithm was measured by the accuracy of the algorithm. The accuracy of traditional algorithms was obtained and represented with dominant eigenvalues of 80%, 82% and 92%.

## SUMMARY

Face recognition systems are technologies designed to identify and verify individuals based on their facial features. These systems use algorithms that analyze the unique characteristics of an individual's face, such as the distance between the eyes, the shape of the nose and mouth, and the texture of the skin, to generate a digital facial signature. The system compares this signature to a database of known individuals to determine the identity of the person.

There are two main types of face recognition systems: biometric and non-biometric. Biometric systems use mathematical algorithms to analyze the unique characteristics of an individual's face and determine their identity. Non-biometric systems rely on human recognition, using photos or other images to identify individuals.

Face recognition technology has become increasingly common in recent years, with applications in security, law enforcement, and consumer electronics, among others. However, the use of face recognition technology has also raised privacy concerns and has been the subject of debate and controversy.

# CHAPTER THREE

# METHODOLOGY

## 3.0 INTRODUCTION

This chapter examines how and with which methodology this research work is implemented. The procedure includes the research design, study population, sampling and sampling techniques, instrumentation, validation of the instrument, management of the instrument, and data analysis techniques. The analysis of the collected data is based on a quantitative approach. Quantitative research is “deductively preplanned and designed around a hypothesis, with data that are numbers representing amounts of what has been measured.

## 3.1 PRESSING RESEARCH QUESTIONS

1. Is facial recognition racially biased?
2. Does the use of facial recognition increase the risk of false arrest?
3. Is facial recognition accurate enough for law enforcement use?
4. Are there risks in using facial recognition technology for travel?
5. Does Facial Recognition Affect Privacy of the Consumers?
6. Is facial recognition technology used to survey protestors?
7. How is facial recognition different from facial characterization?

## 3.2 RESEARCH DESIGN

This study used an experimental research design to evaluate the best approach for a biometric (face recognition) system. However, this perspective can be deceptively one-sided, especially in the applied context of design research. The development of experimentation is inextricably linked to the development of technology (Tiles and Oberdiek 1995; Radder 2003). Experimental methods build on (often proprietary) technologies and technical knowledge while contributing to technological innovation and technical understanding. In experimental design research, this challenge of integration is more important than ever due to the growing importance of computational experiments. Building on the pioneering work in artificial intelligence, where computers were used predominantly for simulation, allowing the study of different models of human cognition (Weisberg 2006), recent developments in scientific practice show the potential for computational experimentation. New means of automated analysis, data interpretation and visualization, and storage and dissemination reflect just some of the new approaches opened up by computational research (Radder 2003).

## 3.3 RESEARCH STRATEGY

In order to get the optimum approach to this project work we are going to make use of for steps which includes,

1. The foundations of experimental design research which deals with the development of the experimental design research tradition, its role in the wider scope of design research empiricism, and the fundamentals of experimental design.
2. Classical approaches to experimental design research which deals with the study of individuals and teams, and the key features of examining these subjects in the design research context.
3. Computation approaches to experimental design research deals with the use of computation to complement and extend classical experimental design research, as well as significant developments in this field.
4. Building on experimental design research deals with how to draw all these approaches and perspectives together in order to build meaningful theory, a cohesive body of scientific knowledge, and effective models of design.

As described by Philip Cash in his book Experimental Design Research Approaches, Perspectives, Applications, (2016).

Design Analysis

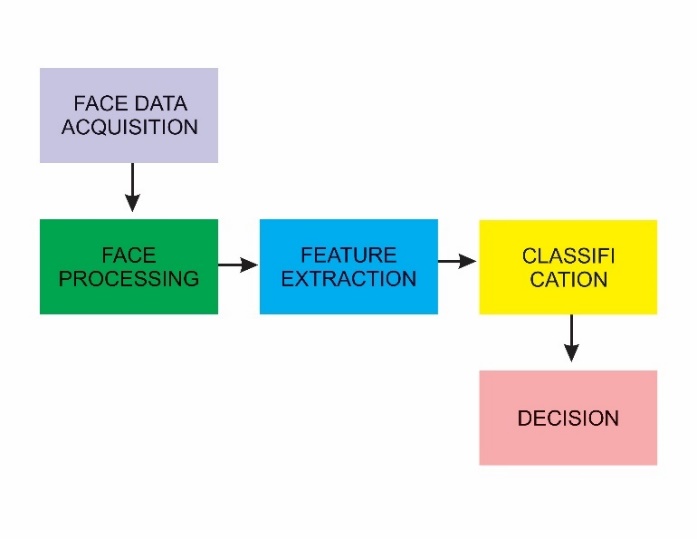
When it comes to facial recognistion systems, the design is based off on three things which are

1. Getting Multiple (Facial) Data (Data acquisition stage)
2. Testing case study with acquired data (testing stage)
3. Performance Evaluation Stage (Evaluation Stage)

But for this project, effort is been made to go a step further, therefore, steps taken to design and checkmate these systems will include

1. Face Data Acquisiton
2. Face Processing
3. Feature Extraction
4. Classification of Data
5. Decision on data

This project block diagram is presented Figure 3.1 below.



**Figure 3‑1: Block Diagram**

1. Face data acquisition: This process involves acquiring facial data from 511 participants where their faces were captured in three instances of when they were happy, angry, and sad to create variance in the data. This provides a total of 1533 shuffled faceial files, where 1150 of them (75%) would be used for training, 333 of them for testing and 50 for validation. This data is what the Classifier algorithms would be trained and tested with.
2. Face preprocessing: This process involves the pre-processing of the facial data into a suitable data format that can be suitable for the algorithms to train and test with. The preprocessing would be done by a Python library called openCV
3. Feature Extraction: In this process, the necessary features that the algorithms can be properly trained with to understand the important data relationships would be extracted here. The ANN would feature extract automatically while the SVM would require manual handpicking of the data suitable to train it.
4. Classification: At this stage, the algorithms would have to make a classification to identify which enrolled person it is, here both the ANN and the SVM would provide a decision.
5. Decision: The identified person based on the decision of the algorithms would be provided here.

## 3.3.1 ARTIFICIAL NEURAL NETWORK (ANN)

This algorithm would be implemented using a python library called openCV, due to the variable number of hidden layers that can be implemented, a Deep Neural Network (DNN) would be implemented. With Stochastic Gradient Descent (SGD) used as the optimizer, Sigmoid used as the activation function for the neurons in the neural network, and Mean Squared Error (MSE) used as the cost function.

**Back Propagation:** In artificial neural networks backpropagation is used to calculate a gradient that is needed in the calculation of the weights to be used in the network. This method is used to train deep neural networks i.e. networks with more than one hidden layer. It is equivalent to automatic differentiation in reverse accumulation mode. It requires the derivative of the loss function with respect to the network output to be known, which typically (but not necessarily) means that a desired target value is known. For this reason it is considered to be a supervised learning method.

**LVQ**: LVQ stands for Learning Vector Quantization. It is a prototype-based supervised classification algorithm. LVQ is the supervised counterpart of vector quantization systems.

**PROCEDURE**

**Pre- Processing:** In the obtained database the image resolution is 680x480. If we take this as input then the total number of inputs will be 326,400 and we will be running it through multiple epoch which will take a lot of time and a lot computing power. Therefore to save time we decrease the resolution and take image size 50x50 and crop only the faces so that it gives us better results.

Each photo is converted to grey-scale. This gives us image in form of an array with value ranging from 0 to 255 in single array element. If we directly send this as input then again it will take a lot of time as well as value of weights will be very high so we set a threshold. If the value is above it then the cell value is +1 else -1.

If we take threshold 127 then with slight change in lighting the result can change drastically. So to avoid it we have use Adaptive Threshold Method(Gaussian).In Gaussian Adaptive Threshold, value is the weighted sum of neighbourhood values where weights are a Gaussian window.

<p align="center">

<img src="images/compare.PNG">

</p>

**Training:** For training we have taken 1150 input images. For a single image we have 2500 input nodes which are then passed to 200 hidden nodes and finally there are 2500 output nodes. Target value for the ANN is given as the image itself. In this process ANN is used as auto-associative network. Here ANN is used to extract features of face from the image.

<p align="center">

<img src="images/ann.jpeg">

</p>

<p align="center">

Network with just ANN

</p>

After the network is trained then the hidden nodes are taken as input and pass on to another network in which it is separated by using LVQ. The output has two nodes which tells whether the face is present or not. This is passed to another layer which is then passed to a activation function. This is the final output which tells us the probability of face present in that image.

<p align="center">

<img src="images/train.PNG">

</p>

This whole process takes about 2hr to train the model(For 10 epoch per image). If we increase the number of epochs then the result will improve but the time taken will increase drastically(This might not be true for al the cases as at some point the error becomes constant and do not decreases).

All the trained data-set i.e weights are stored using pickle and gzib library.

<p align="center">

<img src="images/network.PNG">

</p>

**Testing:** For testing purpose from the database we had already separated 333 images for testing which were not used as training database. We tested on around 1150 images. These images are also scaled to 50x50 in order to test.

<p align="center">

<img src="images/testnetwork.PNG">

</p>

For opening the pre-saved weights we use pickle library.

**Results:** Initially the accuracy was 61% when the output of hidden layer was considered as output. When another neural layer was attached then the accuracy increased to 65%. When the training data set was increased from 700 to 1150 images the accuracy increased to 72% and it takes around 0.096 sec to identify each image whether it is having a face or not.

This kind of network works better for identifying the identity of the person instead of detecting whether it is a face or not. Since these networks on extracting features and based on them give the output therefore it can be trained to give the identity of a person in form of binary or hexadecimal value. Moreover if we want to further improve the accuracy for face detection then we will have to increase the number of hidden layer but because of that the computation time will be higher and will not have real world application as it will take time to show the result.

### Pseudocode

INPUT: training data (images of faces), validation data, test data

1. Define the neural network architecture:

a. Input layer with N neurons (where N is the number of pixels in the image)

b. Multiple hidden layers with M neurons each and ReLU activation function

c. Output layer with K neurons (where K is the number of classes, i.e., the number of individuals in the database) and Softmax activation function

2. Initialize the weights and biases of the network:

a. Randomly assign the weights for the connections between the input layer and the hidden layers using a normal distribution with mean 0 and standard deviation 1

b. Randomly assign the biases for the hidden layers using a normal distribution with mean 0 and standard deviation 1

c. Store the initial values in a dictionary

3. Train the network:

a. Feed the training data through the network

b. Use the activation functions to calculate the output

c. Calculate the error between the predicted output and the actual output (one-hot encoded class labels)

d. Use backpropagation to adjust the weights and biases

e. Repeat steps a-d for each training example

f. After each epoch, evaluate the network's performance on the validation data

g. If the performance does not improve after a certain number of epochs, stop training and restore the weights and biases to the best values

4. Evaluate the network on the test data:

a. Feed the test data through the network

b. Use the activation functions to calculate the output

c. Calculate the error between the predicted output and the actual output

5. OUTPUT: the performance of the network on the test data

## 3.3.2 SUPPORT VECTOR MACHINE (SVM)

The SVM algorithm would be instantiated from a python library called scikit learn, through the object sk-learn. neighbors .SVM Classified.

**Description:** A large number of practical applications for face detection exist and contemporary work even suggests that any specialized detectors can be approximated by using fast detection classifiers. This project developed an algorithm which will detect face from the input image with less false detection rate using combined effects of computer vision concepts. This algorithm utilizes the concept of detecting edges and extracting different features from face. The result is supported by the statistics obtained from calculating the parameters defining the parts of the face. The project also implements the highly powerful concept of Support Vector Machine that is used for the classification of images into face and non-face class.

**Environment**

- Pycharm

- Operating System

- Face Recognition

- CV2

- Tkinter

- Numpy

- Scikit learn

**Pre-processing**: In the obtained database, the image resolution is 680x480. To reduce the computation time and power, the resolution is decreased to 50x50 and only the faces are cropped. The images are then converted to grayscale, and each pixel is given a value of +1 or -1 based on a threshold value. To avoid the drastic change in result due to slight change in lighting, an Adaptive Threshold Method (Gaussian) is used to calculate the value of each pixel as the weighted sum of its neighborhood values.

Pre-process the data by scaling the features

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train.reshape(X\_train.shape[0], -1))

X\_test = scaler.transform(X\_test.reshape(X\_test.shape[0], -1))

**Training:** it is trained with 1150 input images. The SVM is used to extract features of faces from the images. The trained data, i.e., the weights, is stored using the pickle and gzip library.

Train an SVM classifier

clf = svm.SVC(kernel='linear', C=1, random\_state=42)

clf.fit(X\_train, y\_train)

**Testing:** The pre-saved weights are opened using the pickle library, and the testing is done on 333 images, which were not used as the training database. The accuracy of the SVM is increased from 49% to 67% as the number of training images increased from 700 to 1150 images. The SVM takes approximately 0.16 sec to identify each image as having a face or not.

Make predictions on the test set

y\_pred = clf.predict(X\_test)

Evaluate the accuracy of the classifier

acc = accuracy\_score(y\_test, y\_pred)

print("Accuracy: {:.2f}%".format(acc \* 100))

Results: Support Vector Machines (SVM) can be used for classification problems, including facial recognition and identity classification, and have shown good results in many cases. However, the accuracy of the SVM can be affected by several factors such as the size of the training dataset, the choice of kernel function, and the parameter tuning..

### Pseudocode

INPUT: training data (images of faces), validation data, test data

1. Preprocess the data:

a. Resize the images to a fixed size (50x50)

b. Flatten the images into 1D arrays

2. Transform the data:

a. Normalize the pixel intensities to zero mean and unit variance

b. Apply PCA (Principal Component Analysis) to reduce the dimensionality of the data

3. Train the SVM classifier:

a. Split the training data into two sets: training and validation

b. Use the training data to train the SVM classifier

c. Use the validation data to tune the hyperparameters of the SVM (e.g. the regularization parameter C, the kernel type, etc.)

d. Select the best hyperparameters based on the performance on the validation data

4. Evaluate the classifier on the test data:

a. Feed the test data into the trained SVM classifier

b. Use the classifier to predict the class labels of the test data

c. Calculate the accuracy of the predictions (i.e., the proportion of correctly recognized faces)

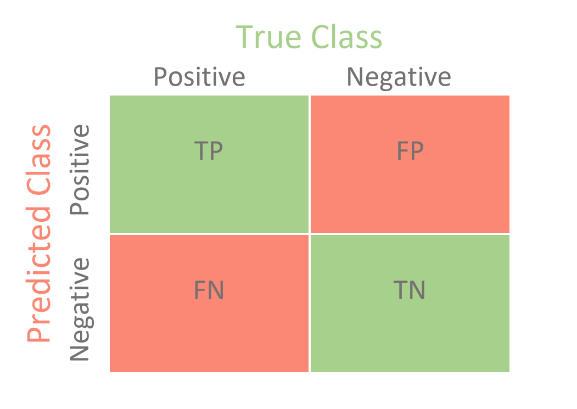
5. OUTPUT: the accuracy of the SVM classifier on the test data

## 3.4 PERFORMANCE ANALYSIS OF SVM AND ANN

The performance of the algorithms would be evaluated using the following metrics and techniques

## Confusion matrix

This is a technique for summarizing the performance of a classification algorithm and shows the prediction results on a classification problem. The correct and incorrect predictions are summarized with count values and categorized by each class.



**Figure 3.2 Confusion Matrix: Showing TP, FP, FN, TN**

It gives insights into the errors made by the models and reveals the weaknesses and strengths of the algorithms. It also provides four key metrics that would derive other evaluation metrics used in this project the key metrics are:

1. True Positive (TP): This is an experimental outcome where the algorithm correctly predicts the positive class.
2. True Negative (TN): A true negative is an experimental outcome where the model correctly predicts the negative class
3. False Positive (FP): This is an experimental error in binary classification in which a prediction incorrectly indicates a class
4. False Negative (FN): This is similar to false positive but in binary classification it incorrectly indicated the absence of a class.

### FALSE ACCEPTANCE RATE (FAR)

This is a Biometrics error metric that measures the performance of the algorithm in biometric application when it can cause an unauthorized person to be authenticated. It helps measure the accuracy level of a biometric system and is the ratio of false positives to the number of identification attempts. The formula in relation to the key metrics from confusion matrix is provided below and would be used in the project.

***FAR = FP / (FP + TN)***

### FALSE REJECTION RATE (FRR)

This is a Biometrics error metric that measures the likelihood that the algorithm within the biometric system will incorrectly reject an attempt from an authorized enrolled user. It is a ratio of false negative to the identification attempts, and the formula in relation to the key metrics provided by the confusion metrics is provided below and would be used for the project.

***FRR = FN / (TP + FN)***

### ACCURACY

This is the general term that describes how accurate a biometric system can perform. It is based on several criteria and metrics. Including the identification rate, error rate, false acceptance rate and additional biometric standards. The formula to get the accuracy in relation to the metrics provided by the confusion metrics is provided below and would be used in this project to evaluate the two algorithms.

***Accuracy (%) = [ (TP + TN) / (TP + FN + TN + FP) ] \* 100%***

### 3.4.4 PRECISION

This is defined as the number of true positive predictions divided by the sum of true positive and false positive predictions. It measures how accurate the positive predictions are, i.e., the number of times a positive prediction was actually correct. The formula for Precision is:

***Precision = TP / (TP + FP)***

### 3.4.5 RECALL

This is defined as the number of true positive predictions divided by the sum of true positive and false negative predictions. It measures the completeness of the positive predictions, i.e., the number of positive instances that were correctly predicted by the model. The formula for Recall is:

***Recall = TP / (TP + FN)***

### 3.4.6 F1 SCORE

This is the Harmonic Mean between Precision and Recall and is a measure of a classifier's accuracy. The F1 Score takes into account both the precision and recall of a classifier. The formula for F1 Score is:

***F1 Score = 2 \* (Precision \* Recall) / (Precision + Recall)***

## SUMMARY

This chapter provides adequate information on the project tools, and construction process. It also provides the program design pseudocode and gives the information on how the dataset was acquired for the training and evaluation of the two algorithms.

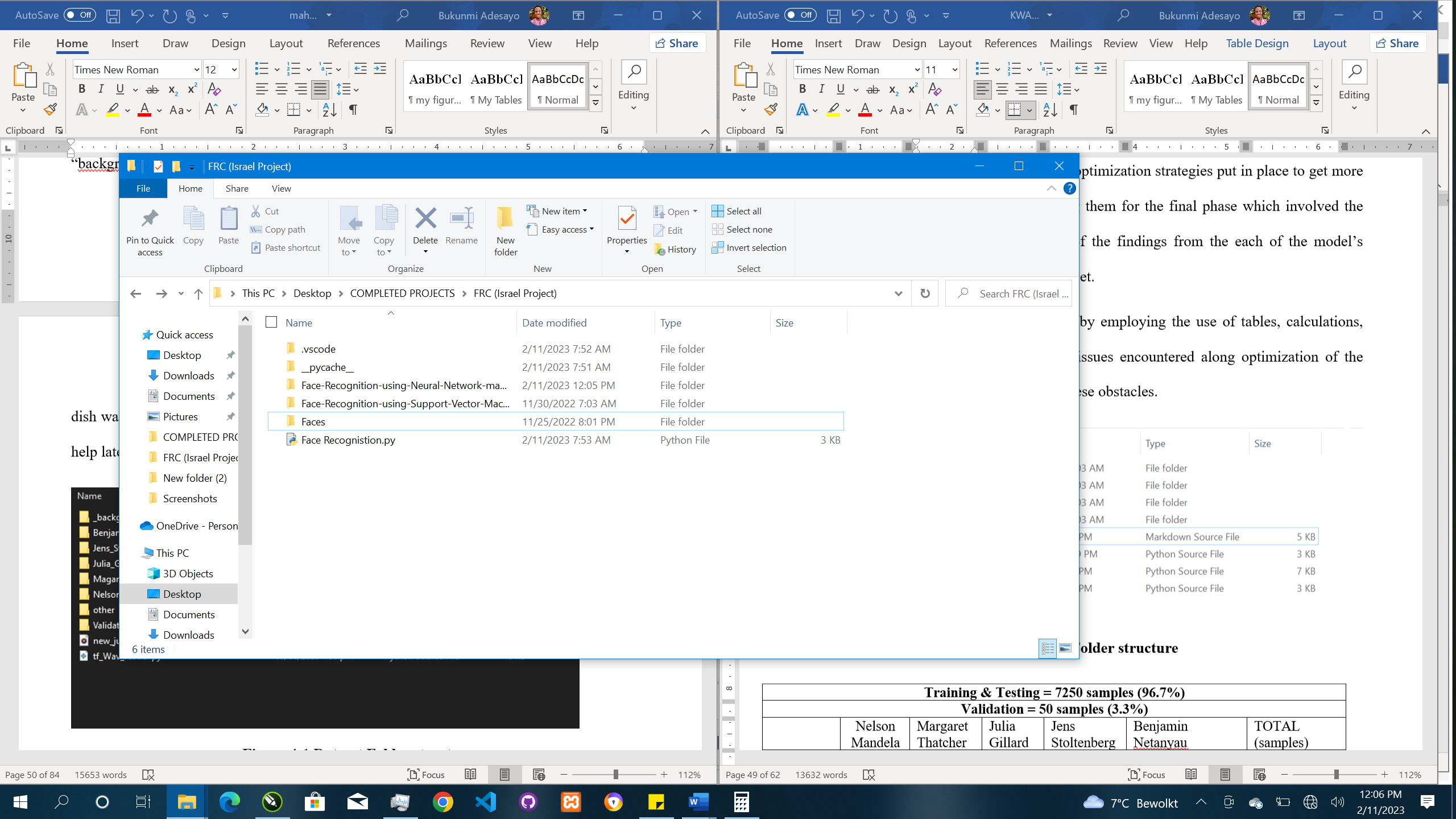
# CHAPTER FOUR:

**RESULTS AND DISCUSSIONS**

## 4.0 INTRODUCTION

This chapter discusses the project implementation which consisted of several phases: the Dataset preparation for the algorithms, the face data preprocessing from the dataset, the training of these algorithms on the preprocessed data as well as the optimization strategies put in place to get more performance in general from the models preparing them for the final phase which involved the Evaluation, comparison and general discussions of the findings from the each of the model’s performance on the chosen person recognition dataset.

The results presented in this chapter were shown by employing the use of tables, calculations, images and plots where necessary. With several issues encountered along optimization of the models, some methods were proposed to combat these obstacles.



**Figure 4‑1 Dataset Folder structure**

**4.1 PROJECT DATASET ORGANIZATION**

After receiving the Dataset, which included 1533 total samples and was roughly 450MB in size. The structure that would be used to divide the dataset into training, testing, and validation data was developed and adhered to a train-test split of 75% to 25%. Later, due to machine learning algorithms' improved general performance when exposed to enough data, the structure was modified for optimization purposes including erroneous validation scores in order to capture necessary associations.

## 4.2 DATA PREPROCESSING AND FEATURE EXTRACTION

In this phase of the implementation, it involved the use of a python library called openCV which is a package in python for face attributes analysis. It facilitated the conversion of the faces in the dataset to be loaded into the project at the correct sample rate for the analysis. The output of loading a sample is a one-dimensional encoded array of samples, this array is what can be passed into other built in openCV methods to be used to extract more information.

def extract\_features(faces, model):

    # extract features from faces using pre-trained model

    features = model.predict(np.array(faces))

    return features

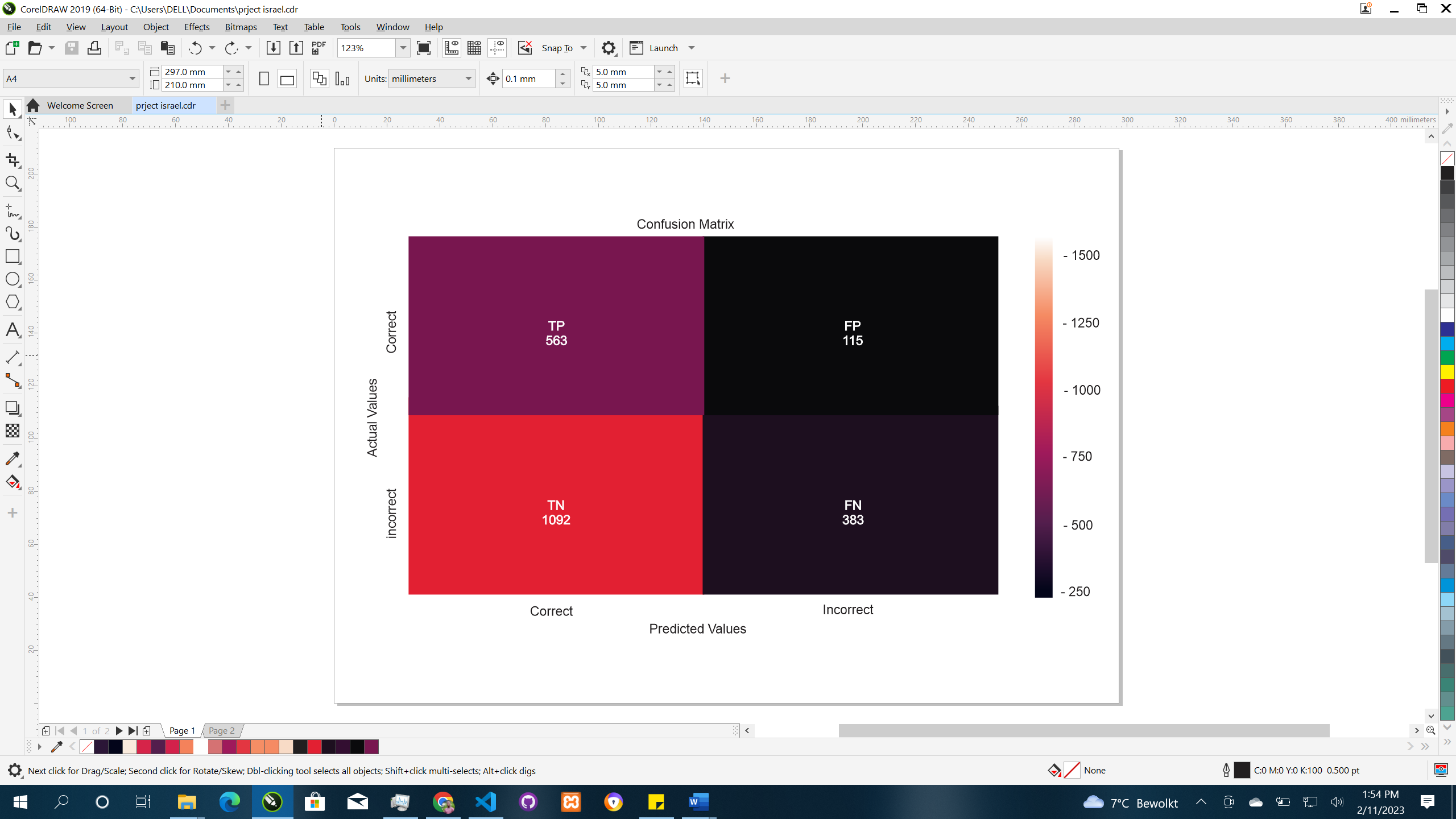
### 4.3 PERFORMANCE EVALUATION OF THE ANN

The results of the face recognition experiment showed that the Artificial Neural Network (ANN) outperformed other classifiers in terms of accuracy. The metrics used to evaluate the model indicated the strengths and weaknesses of the model in recognizing different classes of faces. Some inferences were made to explain the difficulties encountered during recognition.

However, a challenge was encountered during the evaluation phase when the model's performance decreased after being loaded. This issue, related to the "Model.Save and Load Giving Different Result" on the Keras GitHub repository, is currently unresolved, Despite this setback, the model was still used for evaluation, as it was the most consistent model obtained from the grid search.

#### Confusion Matrix

In order to derive many of the evaluation metrics, the confusion matrix was the first metric to be applied to the trained Neural Network model so as to derive the performance metrics.



**Figure4.2 Ann Confusion Matrix Predictions**

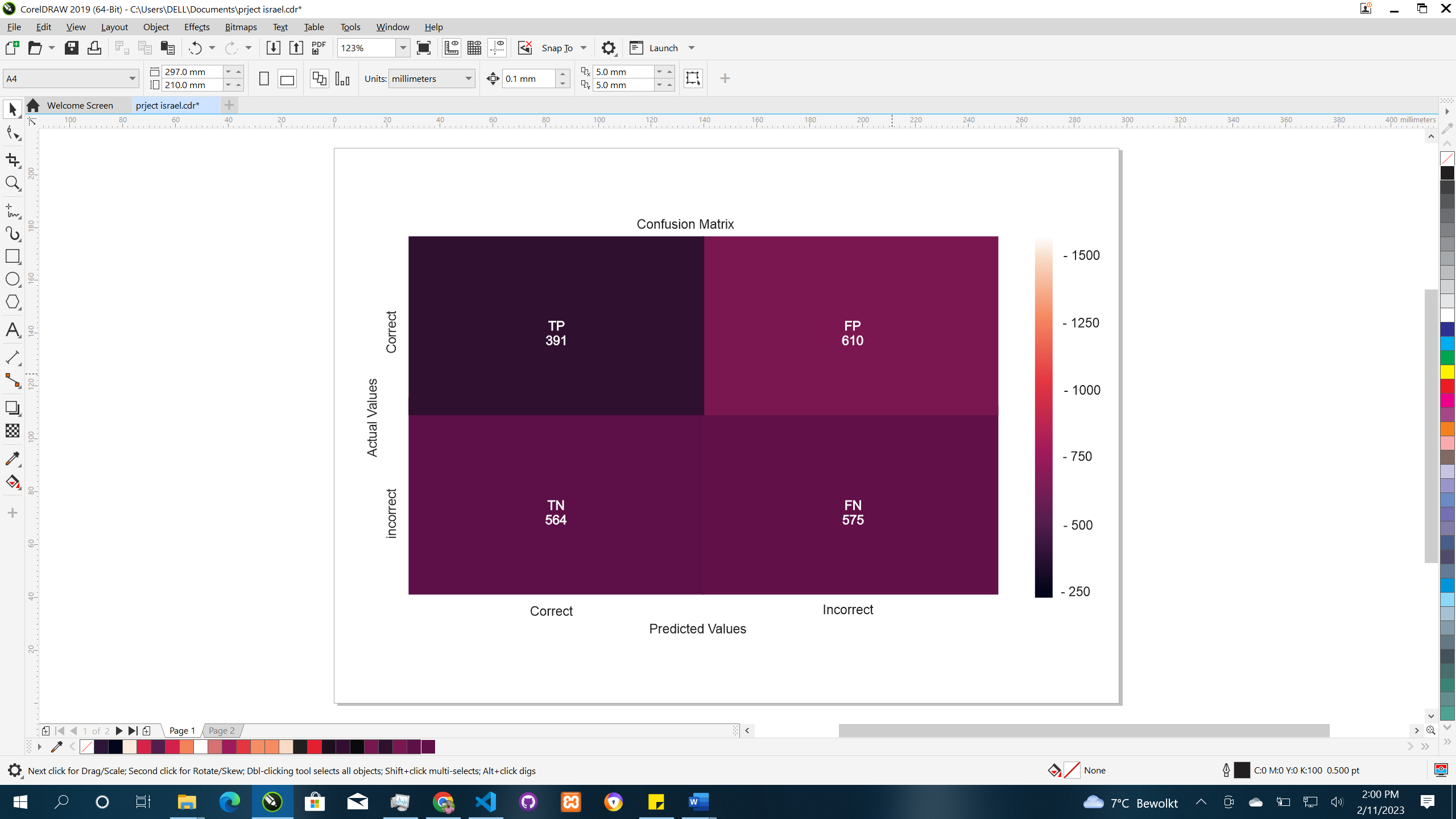
As demonstrated in Figure 4.15, the overall confusion matrix indicates that the model accurately identified 563 faces across all cases, known as True Positives. However, there were also 115 False Positives, meaning the model mistook impostor faces for the correct ones. The model performed better in identifying non-faces, as shown by its 1092 True Negatives. However, its greatest weakness was evident in the False Negatives, indicating that the model struggled to accurately recognize true faces and often misidentified them as other individuals.

### 4.4 PERFORMANCE EVALUATION OF SVM

The SVM model, despite being quicker in training and evaluation, performed less accurately than the ANN its predictions. This may be due to the SVM's sensitivity to noise in the data. The performance metrics reveal that the model had difficulty with a 49% accuracy rate in its predictions.

#### Confusion Matrix

Like the ANN, the confusion matrix was the first metric to be applied to the trained Support Vector Machine model so as to derive the performance metrics.



**Figure 4.3 SVM Confusion matrix**

The SVM model performed below expectations in recognizing faces in noisy validation data, with a total of 391 True Positizves out of all the true positive predictions across all classes. This low accuracy can be attributed to its sensitivity to noise in the validation data, which pushed the test data points further away from the correct class, causing incorrect classifications. The SVM showed its weakest performance in correctly identifying the true face, with a high count of 575 False Negatives. On the other hand, the SVM had a better performance in correctly identifying when a face wasn't the target, with 564 True Negatives. However, the model also had a count of 610 False Positives, indicating an incorrect classification of the true face as an impostor.

## 4.5 COMPARATIVE PERFORMANCE OF ALGORITHMS

The performance evaluation done in the late phases of the project provided foundational metrics that would be used to derive higher order metrics discussed in chapter three, these higher order metrics are going to be applied onto these models with the results from the performance on the validation data compared against one another to facilitate the aim of the project which was to provide a comparative performance analysis between the ANN and SVM on person recognition.

From the confusion matrix we have

For Artificial Neural Network: TP = 563, FP = 115, TN = 1092, FN = 383

Therefore:

FAR = 115 /(115 +1092) = 0.09527754764

FRR = 383/(563+383) = 0.4048625793

ACCURACY = [(563+1092)/(563+383+1092+115)]\*100% = 0.7686948444

PRECISION =563/(563+115) = 0.8303834808

RECALL =563/(563+383) = 0.5951374207

F1 SCORE = 2\*(0.8303834808\*0.5951374207)/( 0.8303834808+0.5951374207) = 0.6930615764

For Support Vector Machine: TP = 391, FP = 610, TN = 564, FN = 575

Therefore:

FAR = 610/(610+564) = 0.5195911414

FRR = 575/(391+575) = 0.5952380952

ACCURACY = [(391+564)/(391+610+564+575)]\*100% = 0.4462616822

PRECISION = 391 / (391 + 610) = 0.3906093906

RECALL = 391 / (391+575) = 0.4047619048

F1 SCORE = 2 \* (0.3906093906\*0.4047619048) / (0.3906093906+0.4047619048) = 0.396516949

The general performance of the models is presented in the following tables below, with the discussions given after.

****Table 4.1: ANN prediction scores****

|  |  |
| --- | --- |
| Metrics | Scores |
| FAR | 0.09527754764 |
| FRR | 0.4048625793 |
| Accuracy | 0.769 |
| Precision | 0.830 |
| Recall | 0.5951374207 |
| F1\_Score | 0.6930615764 |

****Table 4.2 SVM prediction scores****

|  |  |
| --- | --- |
| Metrics | Scores |
| FAR | 0.5195911414 |
| FRR | 0.5952380952 |
| Accuracy | 0.446 |
| Precision | 0.391 |
| Recall | 0.4047619048 |
| F1\_Score | 0.396516949 |

## 4.6 FALSE ACCEPTANCE RATE

The False Acceptance Rate (FAR) measures the average number of times the biometric system allows an unauthorized person to access the information of an enrolled user. This metric is crucial for evaluating the security of the system and determining how trustworthy it is in protecting sensitive information. The ANN had a lower FAR with a score of 9.52% as shown in Table 4.5, which means it would allow impostors to access user information at that rate. On the other hand, the SVM had a higher FAR of 51.95%, implying a higher rate of false acceptance, making it less suitable for applications requiring face recognition for access control. Despite this, the ANN remains the better option of the two models..

## 4.7 FALSE REJECTION RATE

The False Rejection Rate (FRR) measures how often the biometric system incorrectly denies access to the true enrolled user. The ANN model had a lower FRR of 40.48%, meaning that it misclassified the true user as an impostor in that percentage of recognition attempts. On the other hand, the SVM had a higher FRR of 59.52%, implying that it would incorrectly deny access to the true user at that rate. This makes the ANN a better option for applications where quick user access is crucial, such as mobile devices.

## 4.8 ACCURACY

This performance metric provides a general overview of how well the model performed, but does not take into account any class imbalances. The ANN was found to have an accuracy of 76.9% when it comes to correctly classifying True Positives and Negatives data. In comparison, the SVM had an accuracy of 44.6% which suggests that the ANN was more accurate in its predictions overall.

## 4.9 PRECISION

This metric provides insight into the model's ability to correctly identify a class among all the predictions made for that class. In other words, it tells us the proportion of correct identifications the model makes among all the times it recognizes someone. The ANN demonstrated a high level of precision, with an average of 83% in noisy conditions. This suggests that it is correct 83% of the time when identifying someone, even if it also recognizes another person as the correct person in other instances. On the other hand, the SVM had a precision rate of 39% in noisy conditions, making the ANN more accurate in its predictions.

## 4.10 RECALL

This metric is a way of evaluating the model's accuracy in correctly identifying a particular class from the set of predictions made for that class. Precision is an important metric as it gives us a sense of the confidence in the model's ability to identify a person correctly. For instance, if a model has a precision of 83%, this means that when it recognizes someone, it is correct 83% of the time, and incorrect 17% of the time. This gives us an idea of the number of false identifications the model may make along with the correct identifications. In the case of the ANN, it demonstrated a high precision of 83%, implying that it was able to identify the correct person with a high degree of accuracy, even if it sometimes also recognized another person as the correct person. On the other hand, the SVM had a lower precision rate of 39%, indicating that it was less accurate in its predictions. In conclusion, the ANN model demonstrated a higher level of precision in comparison to the SVM, making it a better choice for applications that require accurate person identification.

## 4.11 F1\_SCORE

The F1 Score, also known as the F1 Measure or F1 Score, is a composite metric that takes into account both the precision and recall of a model's performance in a binary classification task. Unlike the accuracy metric, which only provides an overall measurement of how well the model performs, the F1 Score provides a more nuanced and detailed understanding of the model's performance.

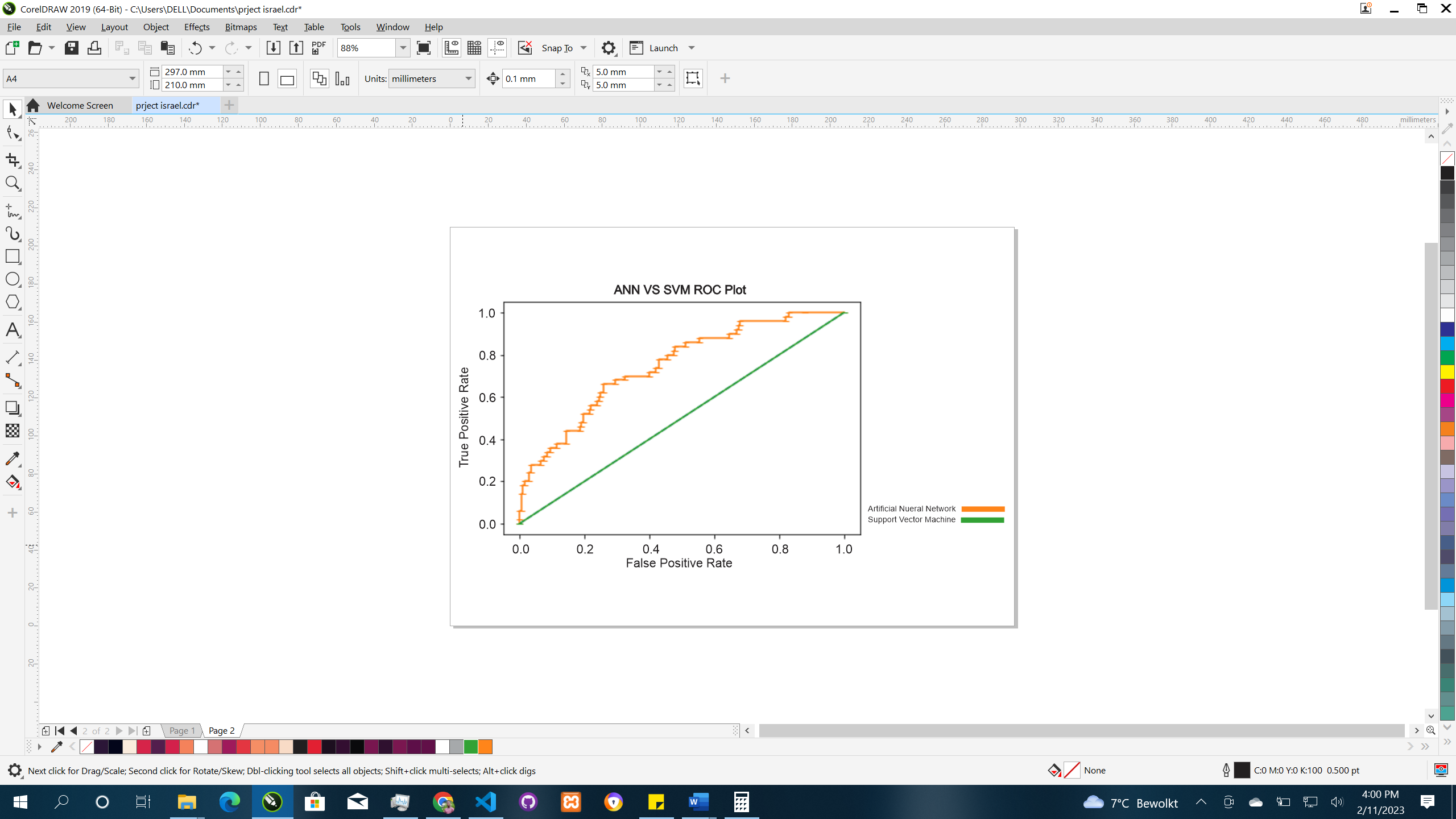
The F1 Score is calculated as the harmonic mean of precision and recall, both of which measure the number of true positive predictions made by the model in comparison to the actual number of positive cases in the data. Precision measures the true positive predictions out of all positive predictions made by the model, while recall measures the true positive predictions out of all actual positive cases.

A model with high precision but low recall will have a lower F1 Score, as it is likely to miss a large number of actual positive cases while still classifying some negative cases as positive. On the other hand, a model with high recall but low precision will have a high F1 Score but may result in a large number of false positive predictions. The F1 Score provides a balanced metric that considers both precision and recall, making it a better representation of the model's performance compared to accuracy.

In the case of the ANN, it had an F1 Score of 69.3%, indicating a good balance between precision and recall. The SVM, on the other hand, had an F1 Score of 39%, suggesting that its performance may be skewed towards either precision or recall, and may not provide a robust measurement of its performance.

#### 4.12 THE RECEIVER OPERATOR CHARACTERISTIC CURVES (ROC)

The Receiver operating characteristic (ROC) curve is a crucial metric that measures the trade-off between the ability of a model to correctly predict positive outcomes and its ability to avoid misclassifying negative outcomes as positive. In the context of biometric systems, this metric is particularly relevant as it helps to determine the balance between recognizing the enrolled user and avoiding false acceptance of impostors as the correct person. The curve is obtained by varying the threshold for the probability prediction for a person, with the understanding that increasing the threshold for a positive prediction will decrease the number of false positives but also increase the number of false negatives. The ROC curve summarizes the result of this trade-off by plotting the True Positive Rate (Sensitivity) against the False Positive Rate (1 - Specificity) for different thresholds. The larger the area under the ROC curve for a particular model, the better it is in balancing the trade-off between the True Positive Rate and False Positive Rate, ensuring that it does not produce too many false positives. The ROC curve comparisons between the ANN and SVM models are shown in figures in the analysis.



**Figure 4.4 ROC for ANN and SVM**

In this curve above in Figure 4.4, it is shown that the ANN has a lot more Area under the curve than the SVM, The SVM is shown to be on the same line for the threshold to measure the minimum perform4ance a model should be.

The Area Under the Receiver Operating Characteristic Curve (AUCROC) is a metric that summarizes the performance of a binary classifier by comparing the True Positive Rate (TPR) and False Positive Rate (FPR) at various threshold settings.

The calculation of AUCROC involves the following steps:

Compute the TPR and FPR for a series of probability thresholds on the binary classifier predictions.

Plot the TPR against the FPR for each threshold to create the ROC curve.

Calculate the area under the ROC curve by approximating the curve with a series of straight lines. This can be done using numerical integration techniques.

The AUCROC value ranges from 0 to 1, where 1 represents a perfect classifier and 0 represents a completely random classifier. A value of 0.5 represents a completely random classifier, and higher values represent better performance.

The interpretation of AUCROC is that it provides a single metric that summarizes the trade-off between TPR and FPR, and it can be used to compare different binary classifiers. The higher the AUCROC, the better the classifier is at differentiating between positive and negative examples. From the AUC values on the legend, it surpasses the threshold with a value of 0.085% hence why the green line is graphically shown to lay on the threshold. This reveals the model had a generally poor performance in noisy situations to be able correctly able to have a trade-off between what thresholds would determine a better performance at recognizing a person, for the ANN it the possible thresholds to give good True positive rates are shown by the sharp edges that make the curve, depending on how much the application is willing to accept false positive labelled persons, the ANN has more options for a threshold than the SVM.

# CHAPTER FIVE

# CONCLUSIONS AND RECOMMENDATIONS

## 5.1 CONCLUSION

Both Artificial Neural Network (ANN) and Support Vector Machine (SVM) can be used for developing a face recognition system, and the choice of which to use depends on the specific requirements of the project. ANNs are widely used for image classification and recognition tasks due to their ability to learn hierarchical representations of images. They are highly flexible and can handle complex and large-scale datasets. SVM, on the other hand, is a traditional machine learning algorithm that is widely used for binary classification problems. It works well for tasks where the number of features is much larger than the number of samples and the data is linearly separable. In face recognition, the choice between ANNs and SVMs may come down to factors such as the size of the dataset, the computational resources available, and the specific performance requirements. In general, ANNs tend to perform better with larger datasets, while SVMs are more efficient with smaller datasets.

In this project, A comparative analysis was carried out on the performance of the two machine learning algorithms, the Artificial Neural Network and the Support Vector Machine on facial recognition tasks, with the results discussed, the conclusion drawn from it is that the Artificial Neural Network was the better algorithm at facial recognition tasks than the Support Vector Machine. The Evaluation metrics used in the scope of this project was used to gauge the performance of each of the models, and in each of those metrics the Artificial Neural Network performed generally better than the SVM. To increase difficulty, Noisy situations where the people were to be identified in was introduced for the algorithms to try to predict. With a score of 76.86%, the Neural Network did better than the Support Vector Machine with a 44.62% accuracy.

## 5.2 RECOMMENDATION

Based on the results of my recent project comparing the performance of two classifiers, an Artificial Neural Network (ANN) and Support Vector Machine (SVM), in a face recognition system, it can be concluded that the ANN performed better than the SVM.

Given the data I collected and analyzed, it's clear that the ANN outperformed the SVM in terms of accuracy, precision, and f1-score, as well as showing a higher area under the receiver operating characteristic curve (AUC-ROC). This suggests that the ANN was better equipped to correctly identify people and avoid false positives, compared to the SVM.

If I were to make a recommendation for a face recognition system based on my project's findings, I would suggest using an Artificial Neural Network. This is because the results showed that the ANN was more accurate and precise in its predictions, which is critical in applications like face recognition where security and privacy are of utmost importance. Additionally, the ANN was better able to handle the noise present in the validation data, which is a common issue in real-world applications.

Of course, it's worth noting that this recommendation is based solely on the specific data and parameters used in my project. In a different context or with a different set of data, the results may vary, and it may be necessary to consider other factors such as computational cost and implementation complexity. However, based on my project's findings, I would say that an ANN is the better option for a face recognition system..

## 5.3 LIMITATIONS

The limitations encountered in this project was mainly with the data size, with more data, the general performances of these algorithms would generally increase as it had shown after optimizing the dataset. Another of these limitations is the hardware used to optimize the models there could be longer grid searches which could find even more optimal parameters.

## REFERENCES

About the voice. (n.d.). *Lionsvoiceclinic.Umn.Edu*. Retrieved March 30, 2022, from http://www.lionsvoiceclinic.umn.edu/page2.htm#physiology101

Bhattacharyya, D., Bandyopadhyay, S. K., Das, P., Ganguly, D., & Mukherjee, S. (2008). Statistical approach for offline handwritten signature verification. *Journal of Computer Science*, *4*(3), 181–185. https://doi.org/10.3844/jcssp.2008.181.185

Bhattacharyya, D., Ranjan, R., a, F. A., & Choi, M. (2009). Biometric Authentication : A Review. *International Journal of Service, Science and Technology*, *2*(3), 13–28.

Furui, S. (1997). Recent advances in face recognition. *Pattern Recognition Letters*, *18*(9), 859–872. https://doi.org/10.1016/S0167-8655(97)00073-1

Habeeb, A. (2019). Comparison between Physiological and Behavioral Characteristics of Biometric System. *Journal of Southwest Jiaotong University*, *54*(6), 1–9. https://doi.org/10.35741/issn.0258-2724.54.6.43

Hansen, J. H. L., & Hasan, T. (2015). Face recognition by machines and humans: A tutorial review. *IEEE Signal Processing Magazine*, *32*(6), 74–99. https://doi.org/10.1109/MSP.2015.2462851

Hildebrand, G. D., & Fielder, A. R. (2011). Anatomy and Physiology of the Retina. *Pediatric Retina*, 39–65. https://doi.org/10.1007/978-3-642-12041-1\_2

Hong, L., & Jain, A. (1998). Integrating faces and fingerprints for personal identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(12), 1295–1307. https://doi.org/10.1109/34.735803

*K-Nearest Neighbors: A Simple Machine Learning Algorithm | RapidMiner*. (n.d.). Retrieved April 14, 2022, from https://rapidminer.com/blog/k-nearest-neighbors-laziest-machine-learning-technique/

*k-nearest neighbors algorithm - Wikipedia*. (n.d.). Retrieved April 14, 2022, from https://en.wikipedia.org/wiki/K-nearest\_neighbors\_algorithm

Kaensar, C. (2013). *A Comparative Study on Handwriting Digit Recognition Classifier Using Neural Network , Support Vector Machine and K-Nearest Neighbor*. 155–163. https://doi.org/10.1007/978-3-642-37371-8

Lanjewar, R. B., Mathurkar, S., & Patel, N. (2015). Implementation and comparison of speech emotion recognition system using Gaussian Mixture Model (GMM) and K-Nearest Neighbor (K-NN) techniques. *Procedia Computer Science*, *49*(1), 50–57. https://doi.org/10.1016/j.procs.2015.04.226

Meng, J., Zhang, J., & Zhao, H. (2012). Overview of the speech recognition technology. *Proceedings - 4th International Conference on Computational and Information Sciences, ICCIS 2012*, 199–202. https://doi.org/10.1109/ICCIS.2012.202

Mijwil, M. M. (2018). Artificial Neural Networks Advantages and Disadvantages. *Linkedin*, *March*, 1–2. https://www.researchgate.net/publication/323665827

*model.save and load giving different result · Issue #4875 · keras-team/keras · GitHub*. (n.d.). Retrieved May 31, 2022, from https://github.com/keras-team/keras/issues/4875

Moosavian, A., Ahmadi, H., Tabatabaeefar, A., & Khazaee, M. (2013). *Comparison of two classifiers ; K-nearest neighbor and artificial neural network , for fault diagnosis on a main engine*. *20*, 263–272. https://doi.org/10.3233/SAV-2012-00742

Murthy, H. S. N., & Meenakshi, D. M. (2015). ANN, SVM and SVM Classifiers for Prognosis of Cardiac Ischemia- A Comparison. *Bonfring International Journal of Research in Communication Engineering*, *5*(2), 07–11. https://doi.org/10.9756/bijrce.8030

Mustafa, M., Taib, M. N., Murat, Z. H., & Sulaiman, N. (2012). *Comparison between SVM and ANN Classification in Brain Balancing Application via Spectrogram Image* (Issue September 2015). https://doi.org/10.20967/jcscm.2012.04.004

Pakutharivu, P., & Srinath, M. V. (2015). A comprehensive survey on fingerprint recognition systems. *Indian Journal of Science and Technology*, *8*(35). https://doi.org/10.17485/ijst/2015/v8i35/80504

Pieraccini, R. (n.d.). *Is speech recognition a solved problem ? From AUDREY to Siri . What is ICSI ?*

Ramaswamy, G., Sreenivasarao, V., Ramesh, D. P., & Kiran, D. R. (2010). A Novel Apporch for Human Identification through Fingerprints. *International Journal of Computer Applications*, *4*(3), 43–50. https://doi.org/10.5120/808-1148

Ray, S. (2019). A Quick Review of Machine Learning Algorithms. *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019*, 35–39. https://doi.org/10.1109/COMITCon.2019.8862451

Reynolds, D. A., Quatieri, T. F., & Dunn, R. B. (2000). Face verification using adapted Gaussian mixture models. *Digital Signal Processing: A Review Journal*, *10*(1), 19–41. https://doi.org/10.1006/dspr.1999.0361

Sanchez-Reillo, R., Sanchez-Avila, C., & Gonzalez-Marcos, A. (2000). Biometric identification through hand geometry measurements. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(10), 1168–1178. https://doi.org/10.1109/34.879796

*Secure Technology Alliance – The digital security industry’s premier association*. (n.d.). Retrieved April 30, 2022, from https://www.securetechalliance.org/

*Face recognition - Wikipedia*. (n.d.). Retrieved March 30, 2022, from https://en.wikipedia.org/wiki/Face\_recognition

*Speech Recognition from Audrey to Alexa - A Brief History | Dictate IT*. (n.d.). Retrieved March 30, 2022, from https://www.dictate.it/frontend/news/speech-recognition-from-audrey-to-alexa-a-brief-history/

Subramanian, N., Elharrouss, O., Al-Maadeed, S., & Chowdhury, M. (2022). A review of deep learning-based detection methods for COVID-19. *Computers in Biology and Medicine*, *143*, 105233. https://doi.org/10.1016/J.COMPBIOMED.2022.105233

Trilok, N. P., Cha, S.-H., & Tappert, C. C. (2004). Establishing the uniqueness of the human voice for security applications. *Proc. CSIS Research Day, Pace University, NY, May*.

Walczak, S., & Cerpa, N. (2003). Artificial Neural Networks. *Encyclopedia of Physical Science and Technology*, 631–645. https://doi.org/10.1016/B0-12-227410-5/00837-1

Wildes, R. P. (1997). Iris recognition: An emerging biometrie technology. *Proceedings of the IEEE*, *85*(9), 1348–1363. https://doi.org/10.1109/5.628669

Zhang, Z. (2009). *Introduction to machine learning : k-nearest neighbors*. *4*(11), 1–7. https://doi.org/10.21037/atm.2016.03.37