VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANASANGAMA, BELAGAVI - 590018

A Project Report

on

SMART TICKET SYSTEM FOR METRO TRAIN

Submitted in partial fulfillment for the award of degree of

Bachelor of Engineering

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Sagar G U (1BG19CS088)

Under the Guidance of

Dr. Jalaja G
Professor., Dept. of CSE
BNMIT

Vidyayāmruthamashnuthe

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Approved by AICTE, Accredited as grade A Institution by NAAC. All eligible branches – CSE, ECE, EEE, ISE & Mech. Engg. are Accredited by NBA for academic years 2018-19 to 2024-25 & valid upto 30.06.2025
URL: www.bnmit.org

Department of Computer Science and Engineering

2022-2023

B.N.M. Institute of Technology

Approved by AICTE, Affiliated to VTU and Accredited as Grade A Institution by NAAC.
All UG Branches- CSE, ECE, EEE, ISE & Mech. E Accredited by NBA for Academic Years 2018-19 to 2020-21 & valid up to 30.06.2021
Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070, INDIA
Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in www. bnmit.org

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the project work entitled **'Smart ticket system for metro train'** carried out by **Mr. Sagar G U (1BG19CS088),** bonafide students of VIII Semester, **B.N.M. Institute of Technology** in partial fulfillment for the award of Bachelor of Engineering in COMPUTER SCIENCE AND ENGINEERING of **Visvesvaraya Technological University**, Belagavi during the year 2022-23. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report. The project phase III report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said Degree.

Dr. Jalaja G **Professor** **Department of CSE** **BNMIT, Bengaluru**	**Dr. Chayadevi M L** **Professor & HOD** **Department of CSE** **BNMIT, Bengaluru**	**Dr. Krishnamurthy G.N.** **Principal** **BNMIT, Bengaluru**

Name	**Signature with Date**
Examiner 1:	
Examiner 2:	

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned my effort with success.

We would like to sincerely thank **Sri. Narayan Rao R. Maanay,** Secretary, BNMIT, Bengaluru for providing excellent academic environment in the college.

We would like to sincerely thank **Prof. T. J. Rama Murthy**, Director, BNMIT, Bengaluru for having extended his support and encouraging us during the course of the work.

We would like to sincerely thank **Prof. Eishwar N Maanay**, Dean, BNMIT, Bengaluru for having extended his support and encouraging us during the course of the work.

We would like to sincerely thank **Dr. Krishnamurthy G.N**, Principal, BNMIT, Bengaluru for his constant encouragement.

We would like to sincerely thank **Mrs. Chayadevi M L,** Professor, Head, Department of Computer Science and Engineering, who has shared her opinions and thoughts which helped us in the completion of our Project Phase III successfully.

We would like to thank our project coordinators **Dr. Anitha Mohan**, Professor, Department of Computer Science and Engineering, BNMIT Department of Computer Science and Engineering, BNMIT for guiding us in a systematic manner.

We consider it is a privilege to express gratitude to our guide **Dr. Jalaja G**, Professor, BNMIT who has given us all the support and guidance in completing the Project Phase III successfully.

Sagar G U (1BG19CS088)

TABLE OF CONTENT

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Today, everything in the world is smart and digitalized. Many advances have been made in the transportation sector. Public transportation in India has always been an area where such new advances have turned their faces out. Even the remotest regions of the country are well connected by public transport system in India. Moving from one location to another isn't an issue within the country. Railways, road transport and air transport are the main public transportation available in India. In addition to, the country has a developed waterway network as well. Whilst the first two are affordable for people in general, the air transport has limited availability because of its airports and cost barrier. Common man doesn't find air travelling affordable. Public transport facilities are meant for carrying goods as well as people. Metro train service has been one of the newest public transportation mode that has improved the travel conditions in the country.

There has been a significant increase in ridership in Metro train services. Millions of people take the metro rail in cities, using a cheap, safe and convenient mode of transport that makes their bearable in otherwise barely liveable urban India. More than Rs 1 trillion was invested in metro rail projects in 2018-19, and the amount is expected to peak to Rs 1.8 trillion by 2021, according to a report by India Infrastructure Research. All government-owned metro systems run as corporations and have devised business plans to generate non-fare box revenue to be financially viable. Some of them have used trains to target customers with advertisements and digital content through tie-ups with some of the biggest technology companies of the world.

In the past few years, face recognition owned significant consideration and appreciated as one of the most promising applications in the field of image analysis. Face recognition is a type of biometric application that can identify individual in a digital image by analysing and comparing patterns. Biometric technology is the technology which helps in identifying an individual by using some fixed statistical techniques. These techniques are based on the physiological or behavioural traits. There are different techniques which are supported by the biometric such as iris recognition, finger print recognition, gait recognition, ear pattern, face recognition and many more. Every technique has its own advantages and disadvantages. The research on which focus is sorely based on the face biometric. There are different biometrics such as iris scan, finger scan, palm/hand print, gait, ear pattern, face recognition, many more.

Face biometric offers the possibility of identifying an individual, without any human assistance and does not require an expert for interpreting the identification correlation results.

1.2. MOTIVATION

Metro rail service has been widely accepted as a solution for most of the traffic and environmental pollution related problems which major cities throughout the world are facing now. Metro rail construction activities are being undertaken in a big way in India, existing metro rail network of the city of Kolkata, Delhi, Bengaluru, Chennai, Mumbai and Hyderabad are being expanded. Two of the significant problems associated with the ticket part of the service, is that it is inefficient, inflexible and time consuming. The current system doesn't account various problems such as security and expansion to other transportation services. The security lacks as there is almost zero tracking of metro users done which presents risks of violence and other such threats. Human beings always distinguish and identify other people by observing and comparing faces in the daily life. The recognition method is very common. Face recognition system is very essential and important for providing security, mug shot matching, law enforcement applications, user verification, user access control, etc and is mostly used for recognition for various applications. These all applications require an efficient Face recognition system. Integrating the ideas of Metro ticketing and Biometric Authentication using Face recognition, new Smart Ticket Technology is provided.

1.3. PROBLEM STATEMENT:

1. To build a smart ticket system for metro train services using image recognition and artificial intelligence models.
2. To build a reliable verification and identification system using biometric method for metro train commuters.

The system should be able to help provide a Smart ticket solution by solving above problems.

1.4. OBJECTIVES:

1. The main objective of the proposed system is to provide a Smart ticketing system for Metro commuters using Face Biometric Authentication.
2. To provide a secure ticketing system for Metro commuters that is efficient, flexible and fool-proof.
3. The main aim is to reduce the time spent in getting the tickets and check in and out through Automatic Fare Collection System gates.
4. The proposed work can be implemented based on different machine learning approaches for Face Biometric Authentication of Metro commuters.
5. To provide an online platform for payments and account management of users, employees and administrators.

1.5. SUMMARY

The proposed system addresses three key challenges in the field of ticket system:

1. Reducing time spent during the ticket process.
2. Hands free ticket system, as it allows simple walk in walk out system.
3. The cost problem is challenged as it is a one-time investment and doesn't require any sort additional requirements.

These challenges were addressed by introducing remote mounted camera systems connected to centralized servers using trained machine learning models to detect and recognize users and bill them accordingly.

CHAPTER 2

LITERATURE SURVEY

2.1. INTRODCUTION

A literature survey in a project report represents the study done to assist in the completion of a project. A literature survey also describes a survey of the previous existing material on a topic of the report.

The focus of a literature survey in on the following and in this order:

1. Existing theories on a topic with universal acceptance across the board
2. Books on the subject acting as a reference for the concepts that project uses whether they are specific or generic.
3. Current research concerning the field of the project from the oldest to latest. Research papers might be a reference for theories nut most cases require a critical comparison to establish the purpose of the project and improvement
4. You may also include another project report and what helped you
5. Challenges for the project and by ongoing work if it is available

Literature surveys provide brief overviews or a summary of the current research on topics. The structure written requires to be in a way that it seemed logical. It needs to chronologically represent a development of the ideas in the field that is being researched. The length of a literature survey depends much on whether the purpose of the project report is to complete a college assignment or submitting for journal publication. It can review a few research papers on a topic or be a full-length discussion on the significant work in the field until that date.

2.2. LITERATURE SURVEY

In [1], Manoj Panwar et al. proposes that urban systems are the driving force of economic growth. Urbanization contributes significantly to economic, social, and physical transformation. Infrastructure services are supporting the transformation. Delhi City's pace of urbanization and industrialization has not only burdened the transportation system, but has also adversely affected human development, the environment, and the urban ecosystem. The basic concept of co-benefits

is applied in order to understand more than its expected benefit the multiple aspect of transport policy. Researchers quantified the benefits and co-benefits of various enacted policies and development initiatives carried out by the government from 2001 to 2015 in Delhi, India's capital city, and observed that emissions from all vehicles continued to rise despite sincere efforts by the planners due to an increase in the number of vehicles in the city.

METHODOLOGY: Research.

ADVANTAGES: Detailed research done on different pollutants caused by different modes of transportation and highlights the requirement of Metro as the main public transport mode.

LIMITATION: No Technical Solution for the proposed problem.

In [2], Zhang Rui et al. proposes Biometric authentication has been widely studied and has attracted particular attention in both academia and industry in order to overcome the difficulty of password management and improve the usability of authentication systems. Many biometric authentication systems, especially for mobile devices, have been researched and developed. However, there are still defects in the existing biometric authentication systems. Some biological characteristics were not investigated in depth. Existing systems may be vulnerable to attacks such as replay attacks and suffer from invasion of user privacy, which significantly inhibit end-user adoption. For the purpose of safe and privacy-preserving identification, the literature still lacks a thorough review of recent advances in biometric authentication. In the above paper, by focusing on security and privacy solutions, identify and thoroughly review the current biometric authentication systems. They examine the risks of biometric authentication and suggest a number of criteria for authentication that is safe and confidential. By analysing their differences and summarizing the advantages and disadvantages of each based on the proposed criteria, further review the existing works of biometric authentication. In particular, in biometric authentication, the paper discusses the problems of detecting aliveness and protecting privacy. Based on the survey, they identified a number of open research issues and specify a number of important research directions worthy of special efforts in future research.

METHODOLOGY: Detailed research on finding the better biometric authentication system.

ADVANTAGES: Complete survey on various parameters of biometric authentication techniques.

LIMITATION: No standard method for privacy and No technical solutions are proposed.

In [3], Varun Kaushik et al. states that the world moves in the direction of automation. Conductor's work is one of many that due to automation is on the wane. RFID is the most popular form of automatic ticketing. RFID-based ticketing system is cost-effective and operationally efficient. The system has been successfully implemented in metro trains by cities around the world and some cities like Singapore have also successfully implemented it in the buses. A fool proof system is the void that lies in all of the above-mentioned places. The systems rely on passengers' promptitude to ensure ticket-free travel. The paper presents a method to avoid ticketless travel by implementing a two-way authentication for the head-count using RFID and Face Detection. A mismatch in the outputs of both would imply ticketless travel.

ALGORITHMS: Single Shot Detection Algorithm.

METHODOLOGY: Ticket System using RFID and Face Detection.

ADVANTAGES: Fool proof ticketing system.

LIMITATION: Implementing two different technologies makes proposed method inflexible.

In [4], Nataliya Boyko et al. proposes the overview and study the time complexity of face recognition computer vision algorithms. The main idea of the article is to compare two popular computer vision libraries objects, they are OpenCV and Dlib, explore features, analyse for and against each of them, and understand the situation in which each of them suits the best method. Computer vision technologies that are used for face recognition have been developed. Research has been conducted on two prominent computer vision libraries. Evaluate their characteristics and measure the advantages and disadvantages of each of them. Examples of building recognition application based on histogram-oriented gradients for face finding, face landmark estimation for face orientation, and deep convolutionary neural network to compare with known faces. The article makes the concept of face recognition popular. It described the scientific basis for facial recognition and building a complete system of recognition. The fundamental principles of face recognition programs are formulated. A comparative study of the efficiency of both libraries in relation to the time of execution was provided in relation to the number of iterations of the algorithms used. Also built two simple applications for face recognition based on these libraries and comparing their performance.

METHODOLOGY: Face Detection and Face Recognition.

ALGORITHMS: OpenCV-HAAR, Dlib-HOG, DCNN128-encoding algorithms.

ADVANTAGES: comparison of Dlib and OpenCV libraries.

LIMITATION: All different techniques of OpenCV and Dlib are not considered**.**

In [5], Neel Ramakant Borkar et al. proposes that Face Recognition is the ability by their facial characteristics to detect and recognize an individual. Face is a multidimensional, requiring many mathematical computations. Face recognition system is very important and essential for providing security, matching mug shot, law enforcement applications, user verification, user access control, etc. and is mostly used for recognition for different applications. All of these applications need an efficient system for face recognition. Several approaches are already being developed and have low recognition power, high false alarm frequency. The main task of the research is therefore to develop a face recognition system with better accuracy and improvement. The paper proposes a hybrid face recognition algorithm by combining two face recognition techniques by integrating (PCA) principle Component Analysis, (LDA) Linear Discriminant Analysis. Jacobi method is used to compute Eigen vector that are necessary for PCA and LDA algorithms. Face Recognition system will be implemented on Embedded system based Raspberry pi 3 board.

ALGORITHMS: PCA, LDA, PCA+LDA

METHODOLOGY: Comparing and Combining algorithms.

ADVANTAGES: Real time hardware implementation is proposed.

LIMITATION: No real time implementation over custom dataset.

In [6], Kewen Yan et al. proposes that face recognition method is presented in the above paper based on the Convolution Neural Network (CNN). The network consists of three layers of convolution, two layers of pooling, two layers of full connection, and one layer of regression Soft-max. Stochastic gradient descent algorithm is used to train both the extractor feature and the classifier that can extract and classify facial features automatically. To solve the over-fitting problem, the Dropout method is used. The Convolution Architecture for Feature Extraction framework (Caffe) is used during the training and testing process. The face recognition rate of the ORL face database and AR face database based on the network is 99.82% and 99.78%.

ALGORITHMS: CNN, Stochastic gradient descent algorithm.

METHODOLOGY: CNN based Face Recognition.

ADVANTAGES: Improved accuracy, preprocessing of images can be eliminated.

LIMITATIONS: Processing and Memory Requirement are high.

In [7], Radhika C Damale et al. proposes that the identification of the person is referred to as face recognition from the facial features. In the various computer vision algorithms such as face detection, expression detection and many applications for video surveillance, a facial feature can be used. Face recognition technologies have recently drawn researchers to it. Three different methods such as SVM, MLP and CNN were presented in the above approach. DNN is used to detect faces. The features were extracted using PCA and LDA function extraction algorithms for SVM and MLP-based approaches. The images are fed directly to the CNN unit as a function vector in the CNN-based approach. The proposed approach shows the good recognition accuracy for CNN based approach. The SVM, MLP and CNN achieves the testing accuracy around 87%, 86.5% and 98% on self-generated database respectively.

ALGORITHMS: PCA, LDA, CNN, Classifiers- SVM, MLP.

METHODOLOGY: Feature Extraction and Classification of Face.

ADVANTAGES: Different Face Recognition algorithms comparisons along with classifiers.

LIMITATION: Real time hardware implementation is not proposed.

From this literature Survey, detailed analysis has been done on different modes of public transportation and the importance of Metro rail services to urban transport and pollution has been duly noted as presented by Manoj Panwar, Deepak Kumar Singh, and Veruval Devdas et al. [1]. Different modes of biometric techniques have been analysed and the required technique has been selected. The existing system has been thoroughly analysed and the problems would be addressed in the proposed system in the presented by Zhang Rui, Zheng Yan et al. [2]. The process of execution for the proposed system has been understood from various papers, and various algorithms such as PCA, LDA, CNN using Classifiers such as SVM and MLP have been thoroughly analysed and compared presented by Neel Ramakant Borkar et al. [5]. Kewen Yan et al. [6] presents datasets such as AT&T, BioID, Georgia Tech face and Yale Face Databases.

2.3. METHODOLOGY:

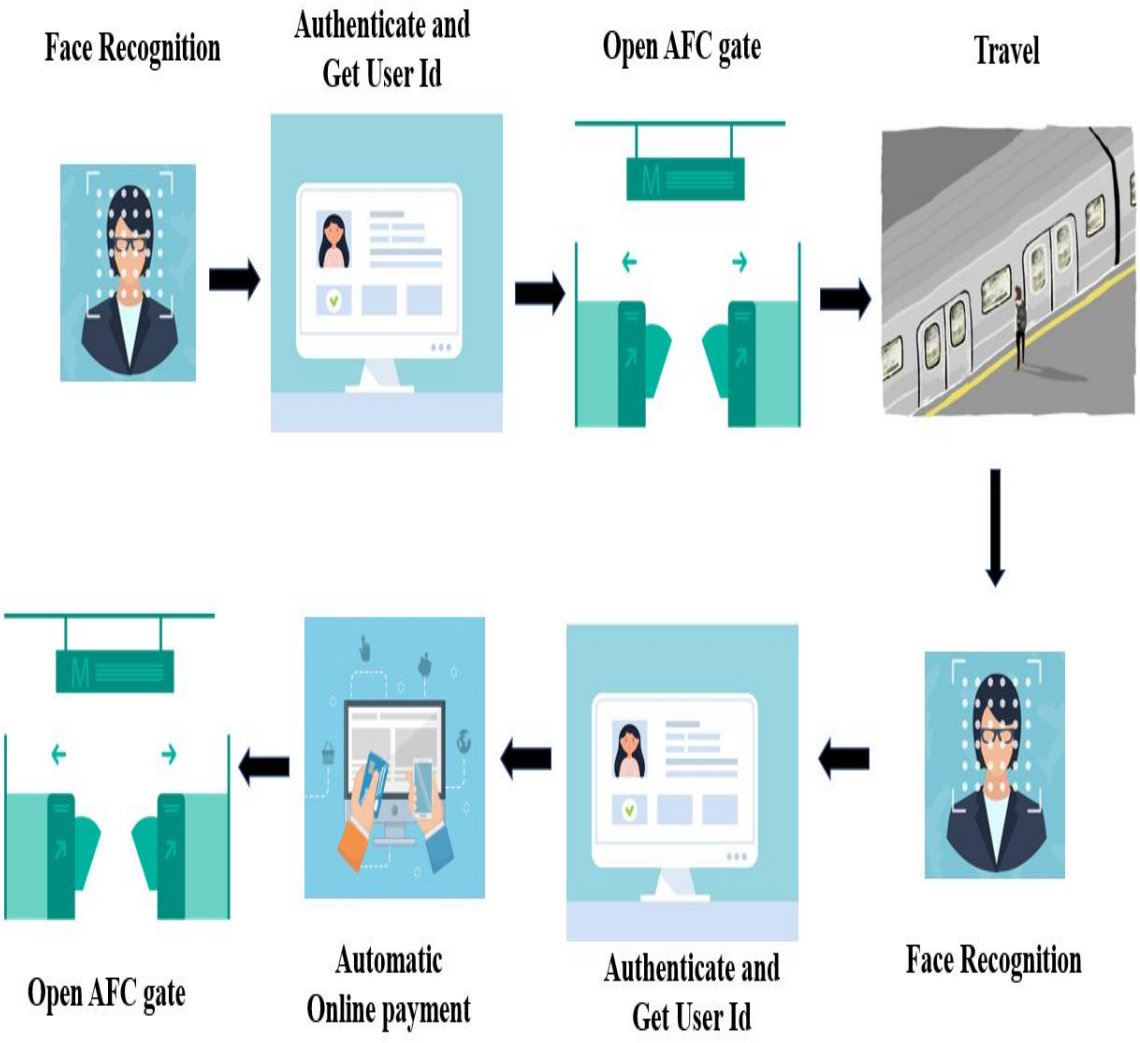


Figure 2.1 Proposed System

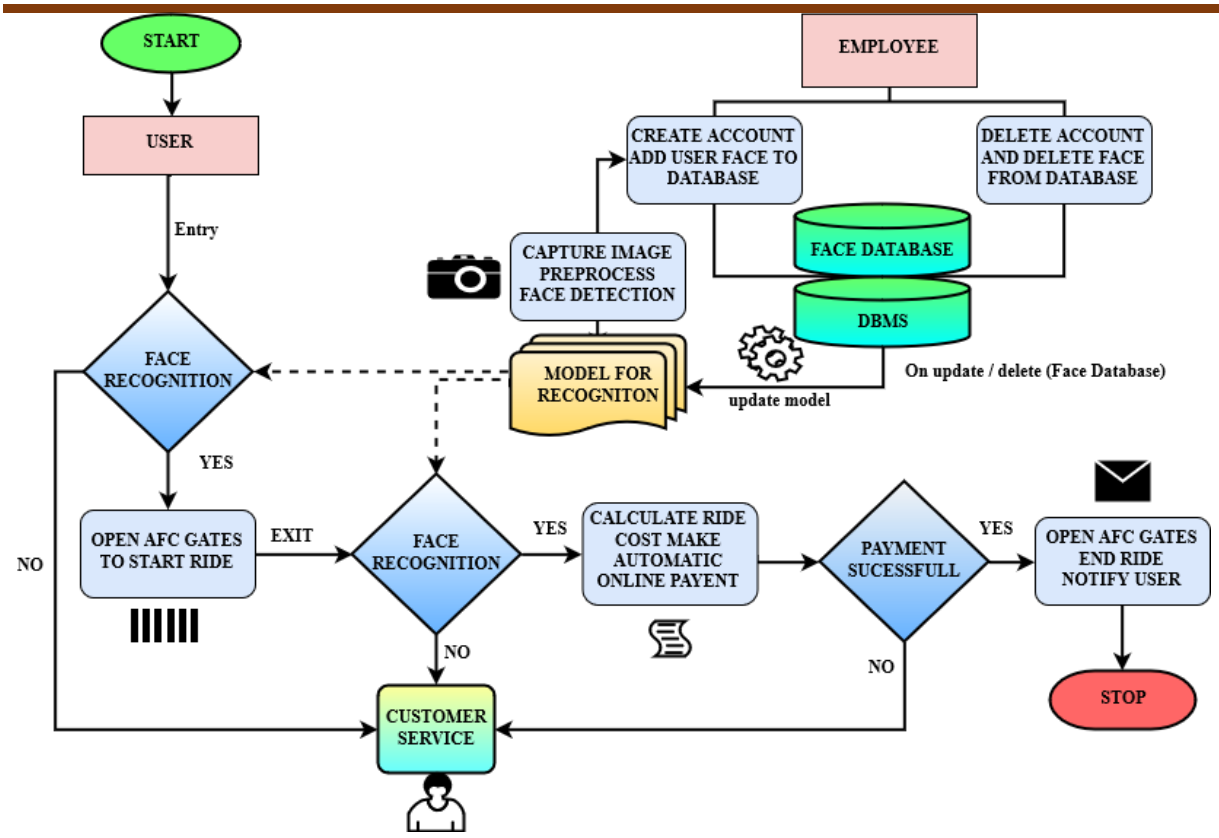


Figure 2.2 Methodology

The figure 2.1 illustrates the overview of the proposed system.

1. An online portal will be provided to metro commuters, employees and administrators. User can create his Smart Ticket System account through portal or employee assistance in the station by providing personal details and face images. Portal provides set of utilities to commuters, employees and administrators such as creating/deleting accounts, fetching station details and ride cost, adding money to commuter wallet.
2. On update or delete of each user to/from the database face recognition model gets updated by retraining the model. Here Face detection and image pre-processing before training the model is performed.
3. Having a Smart Ticket System account when commuter walks towards the entry system, near the Automatic Fare Collecting machine a camera captures frame performs face detection, image pre-processing and face recognition for the captured face using the trained model to fetch user id. If the user has threshold balance in online wallet, system stores entry details of user, display ride details on a display and AFC Gate opens.
4. User can travel to which ever destination. Upon reaching the destination exiting system again performs capturing a frame, face detection, image pre-processing and face recognition using the trained model to fetch user id. System will automatically calculate the cost from source

station to destination station for recognized user id and automatically deduct amount from online wallet. Upon successful completion of payment ride details will be displayed and AFC gates opens user can end the ride.

5. User will be notified about ride details, cost and time on completion of each ride. User can also check details about his previous rides in the portal.

2.4. SUMMARY

An effective implementation of smart ticketing system has been a requirement to complement the smart fast trains and public transportation technologies evolving over the time to address issues of traffic congestion and pollution in cities. Smart Ticket System for metro provides a solution for many key factors of public transportation such as User friendly, Fool proof, Secure Authentication, Data analytics for business optimization of public transportation. Using the Smart Ticket System, a metro commuter can travel easily. No need of cash, material-based ticket, waiting for ticket in a queue. All the user needs is to have a Smart Ticket System account. Just walk out technology makes public transportation smart, user friendly and environment friendly.

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 INTRODUCTION

System requirements are the configuration that a system must have in order for a hardware or software application to run smoothly and efficiently. Failure to meet these requirements can result in installation problems or performance problems. The former may prevent a device or application from getting installed, whereas the latter may cause a product to malfunction or perform below expectation or even to hang or crash.

System requirements are also known as minimum system requirements. The broad classification of system requirements are functional requirements, data requirements, quality requirements and constraints. System requirements often indicate the minimum and the recommended configuration. The former is the most basic requirement, enough for a product to install or run, but performance is not guaranteed to be optimal. The latter ensures a smooth operation.

The hardware/software manufacturers provide an upgrade assistant program that users can download and run to determine whether their system meets a product's requirements. The requirements at the system level that describe the functions which the system as a whole should fulfil to satisfy the stakeholder needs and requirements, and is expressed in an appropriate combination of textual statements, views, and non-functional requirements. The latter expressing the levels of safety, security, reliability, etc., that will be necessary.

System requirements play major roles in systems engineering, as they form the basis of system architecture, design activities, system integration and verification activities. System requirements acts as reference for validation and stakeholder acceptance and provide a means of communication between the various technical staff that interact throughout the project.

Elicitation of stakeholder requirements starts in Concept Definition, and will be initially developed though interview and mission analysis. System requirements are considered in detail during System Definition. Neither can be considered complete until consistency between the two has been achieved, as demonstrated by traceability, for which a number of iterations may be needed.

A requirement is a statement that identifies a product or processes operational, functional, or design characteristic or constraint, which is unambiguous, testable, or measurable and necessary for product or process acceptability. The role the requirement plays in the definition process; for instance, its position in the system block (e.g. translated, derived, satisfied) or its state of agreement (e.g. proposed, approved, cancelled).

System requirements refer to the hardware and software components of a computer system that are required to install and use software efficiently. The software manufacturer will list the system requirements on the software package. If the computer system does not meet the system requirements then the software may not work correctly after installation. System requirements for operating systems will be hardware components, while other application software will list both hardware and operating system requirements.

System requirements are most commonly seen listed as the required minimum and recommended requirements. The minimum system requirements need to be met for the software to run at all on your system, and the recommended system requirements, if met, will offer better software usability.

The requirements are crucial when starting a project if you want to achieve a desirable result at the finish line. They should include all the features and function a product should have. Requirements determination is a critical stage in any software development process due to its fundamental importance to making sure the project ultimately solves the right problem in the right way.

3.2 SOFTWARE / HARDWARE REQUIREMENTS

HARDWARE REQUIREMENTS:

• Processor : Intel Pentium IV or more.

• RAM : 2 GB or more.

• Cache : 1MB.

• Hard Disk : 10 GB recommended.

SOFTWARE REQUIREMENTS:

• Development Platform : Windows 10

• Language : Python, HTML, PHP

• Software : Jupyter NoteBook

3.2.1 Functional Requirements

A functional requirement defines a function of a system or its component, where a function is described as a specification of behaviour between outputs and inputs. As defined in requirements engineering, functional requirements specify particular results of a system.

• Scikit-Learn

It is a Machine learning library to be used in python. It is a simple and efficient tool for data mining and data analysis. It is open source, commercially usable and reusable in various contexts. Scikit-Learn contains various supervised and unsupervised learning. algorithms for data pre-processing, feature extraction, feature selection, classification, regression, clustering, association rules, and visualization.

• Matplotlib

It is a library that basically provides the 2D/3D plotting functions for python programming language in a similar manner as that of MATLAB. We can generate plots, histograms, power spectra, bar charts, scatterplots, boxplots etc. with just a few lines of code. It provides easy and quick way to visualize data from python. It gives a wide range of object - oriented API so as to combine plots into applications.

• Tensor Flow

Tensor flow is an open source library which was developed by researchers and engineers at Google's AI organization. It first defines model in abstract, and then we start the session to execute it. It is mainly used for doing high performance numerical computation like for neural networks using data flow graphs. Stateful dataflow graphs are used to represent the tensor flow computations.

• NumPy

It is an open source powerful package for implementing N-dimensional array object and large matrices of numeric data. It helps in doing scientific computation on the array objects. It contains various mathematical and numerical routines implemented in python to operate on the large arrays. It takes as its input the files in CSV (Comma Separated Values) format. If we use NumPy in python then it gives the functionality on the array objects which is similar to that of MATLAB. The reason being that they both are interpreted.

• **SciPy**

Free and open source python library used for scientific and technical computing. The SciPy (Scientific Python) package incorporates a large collection of useful algorithms like minimization, Fourier transformation, regression, contains modules for optimization, linear algebra, interpolation, special functions, signal and image processing and other applied mathematical techniques common in science and engineering in addition to NumPy.

• **PHP**

PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites. It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.Other Hardware based functional requirements that aid in the production ready model of the proposed project include.

• **Arduino Uno**

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery. The USB connection with the PC is necessary to program the board and not just to power it up. The Uno automatically draw power from either the USB or an external power supply.

Figure 3.1 Arduino Uno board

3.2.2 Non - functional Requirements

Non-functional Requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs. Also known as system qualities, non-functional requirements are just as critical as functional Epics, Capabilities, Features, and Stories. They ensure the usability and effectiveness of the entire system. Failing to meet any one of them can result in systems that fail to satisfy internal business, user, or market needs, or that do not fulfil mandatory requirements imposed by regulatory or standards agencies.

- **Performance**

  The speed at which the computer operates or the total effectiveness of a computer system including throughput, response time and availability. It can also be referred by counting operations or instructions performed during a benchmark test.

- **Reliability**

  It is the probability of failure-free software operation for a specified period of time in a specified environment. It is the availability of the system to perform its intended functions and operations in an environment without experiencing failure.

- **Flexibility**

  It is the capacity of a framework to adjust to changing situations and circumstances, and to adapt to changes to business approaches and rules.

- **Maintainability**

  The ease with which a software system or component can be modified to correct faults, improve performance or adapt to a changed environment. It measures the speed with which a system can be restored to operational status after a failure.

- **Security**

  It is an idea implemented to protect software against malicious attack and other hacker risks so that the software continues to function correctly under such potential risks. Security is necessary to provide integrity, authentication and availability.

- **Scalability**

  It is the capability of a system, network or process to handle a growing amount of work, or its potential to be enlarged to accommodate that growth.

3.3 SUMMARY

In the software development process, requirement phase is the first software engineering activity. The phase described is a user-dominated phase and translates the ideas or views into a requirements document. Note that defining and documenting the user requirements in a concise and unambiguous manner is the first major step to achieve a high-quality product.

The programming language used for the implementation of the proposed model is Python. Python is a “batteries included” computer programming language. More concretely, Python is a programming language that, in contrast to other programming languages such as C, Fortran or Java, allows users to more readily focus and solve domain problems instead of dealing with the complexity of how a computer operates.

Software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide to the user for perfect interaction. Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. Used appropriately, software requirements

specifications can help prevent software project failure. The software requirements specification document lists sufficient and necessary requirements for the project development. To derive the requirements, the developer needs to have clear and thorough understanding of the products under development. It is achieved through detailed and continuous communications with the project team and customer throughout the software development process.

CHAPTER 4

PROPOSED SYSTEM

4.1 INTRODUCTION

System design is the process in which we design the elements of a system such as the architecture of the system, modules of the system and variety of components used the different interfaces of those components and the data that goes through that system. Systems design could be seen as the application of systems theory to product development. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system. Software design is the process of implementing software solutions to one or more sets of problems. One of the main components of software design is Software Requirements Analysis (SRA). They are created in order to meet the needs of the users. They are not only intended to solve the existing problems, but they also come up with acceptable solutions to the problems that may arise in the future.

The systems design approach first appeared right before World War II, when engineers were trying to solve complex control and communications problems. They needed to be able to standardize their work into a formal discipline with proper methods, especially for new fields like information theory, operations research and computer science in general.

- **Architectural Design**

  The process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.

- **Logical Design**

  A logical design is a conceptual, abstract design. The logical design of a system pertains to an abstract representation of the data flows, inputs and outputs of the system. This is often conducted via modelling, using an over-abstract (and sometimes graphical) model of the actual system. In the context of systems, designs are included. Logical design includes entity-relationship diagrams. You do not deal with the physical implementation details yet; you deal only with defining the types of information that you need.

- **Physical Design**

  The physical design relates to the actual input and output processes of the system. This is explained in terms of how data is input into a system, how it is verified or authenticated, how it is processed, and how it is displayed. In physical design, the following requirements about the system are decided.

1. Input requirement
2. Output requirement
3. Storage requirement
4. Processing requirement
5. System control and backup or recovery

   Put another way, the physical portion of system design can generally be broken down into three sub-tasks:

1. User Interface Design
2. Data Design
3. Process Design

User Interface Design is concerned with how users add information to the system and with how the system presents information back to them. Data Design is concerned with how the data is represented and stored within the system. Finally, Process Design is concerned with how data moves through the system, and with how and where it is validated, secured and/or transformed as it flows into, through and out of the system. At the end of the system design phase, documentation describing the three sub-tasks is produced and made available for use in the next phase.

Physical design, in this context, does not refer to the tangible physical design of an information system. To use an analogy, a personal computer's physical design involves input via a keyboard, processing within the CPU, and output via a monitor, printer, etc. It would not concern the actual layout of the tangible hardware, which for a PC would be a monitor, CPU, motherboard, hard drive, modems, video/graphics cards, USB slots, etc. It involves a detailed design of a user and a product database structure processor and a control processor. The H/S personal specification is developed for the proposed system.

4.2 PROPOSED MODEL

The smart ticket system for metro train services is a fully functional project that encompasses all the various required processes. These different functional and non-functional processes are all either dependent on others or aid in working of other processes. The model is proposed to be build in such a manner so that all the fields are being worked on as all these different functions are not been built and assembled in such a manner previously. This gives a very new system that can be experimented on and extended to other projects.

The proposed model includes various functionalities which can be separated into few different models:

- Data Acquisition and Preprocessing

- Data Preparation and Model construction
- Model training
- Model testing and evaluation

- **Data Acquisition and Preprocessing**

  Image pre-processing is the study of algorithm that takes an image as input and outputs vector(features) or image(matrix) as output. For preprocessing the data is taken from the online source. The pre-processing steps can be resizing, normalization, scaling, mean, standard deviation, rotation, edge detection. Data resizing, or data scaling, is a geometric data transformation which modifies the data size based on a data interpolation algorithm. This data scaling process can increase or decrease the resolution of a target data so that the absolute size of image data is adjusted.

- **Data Preparation and Model construction**

  Data preparation is the phase where the data is put in a suitable place and prepare it to use in the network model. The data set needs to be split into two parts - Train and Test. After this, they keep aside the Test set, and randomly choose X% of their Train dataset to be the actual Train set which is the majority of the dataset and the remaining (100-X) % to be the Validation set, where X is a fixed number (say 80%). The second part is used for evaluating the trained model's performance. The same data that the model was trained on for evaluation is not used for testing.

  Model Construction first starts with model selection. Some models are very well suited for image data, others for sequences (like text, or music), some for numerical data, others for text-based data. So, based on the requirements and preprocessing method a model is chosen. For the chosen project Convolutional Neural Networks works best for processing the image to classify the expression and detection emotion.

- **Model training**

  After model construction comes model training**.** Training is performed using a "labeled" dataset of inputs in a wide assortment of representative input patterns that are tagged with their intended output response. Training uses general-purpose methods to iteratively determine the weights for intermediate and final feature neurons. The Convolution Neural Network model chosen is trained with one of the few standard datasets that are provided in the referenced papers in this phase.

- **Model Testing and Evaluation**

  Once the model has been trained it is possible to carry out model testing. During this phase a test set of data is loaded. This data set has never been seen by the model and therefore its true

accuracy will be verified. Finally, the saved model can be used in the real world. Also, the name model evaluation phase means that the model can be used to evaluate new data.

Other functional modules involving various features are:

- **User end module:**

  As the project is user oriented, we need to involve the user are almost all the steps of the project. This means that starting from login to working of the project the user is a important part. This starts with the user logging in and then being a new user having to provide all details along with photos for image recognition. Online payments systems are also enabled to ease the usability.

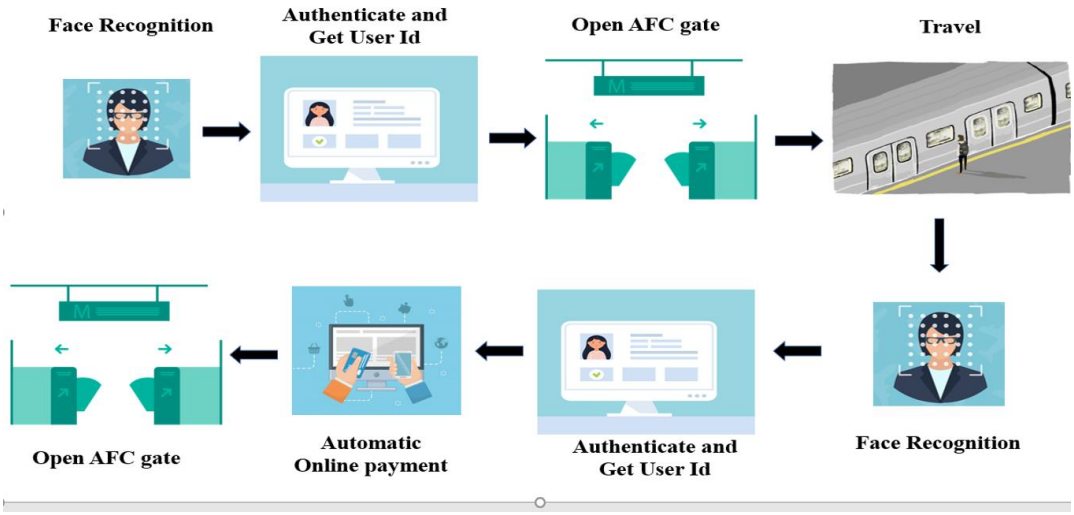


Figure 4.1 Proposed Model

- **Logical design:**

  The main function which is working on logical part are the processes after image training, image recognition during working of the project and also the post metro train ride functionalities.

  The process of image training is not under the logical system design as it is a predefined method and not something that needs to be tweaked for the proposed project of smart ticket system. Whereas the process after that is tailor made for our needs. For which the logic is defined specifically. The images that are to be trained are separated into folders named using the unique id that is generated. Which thereafter allows easy identification. The image recognition's aftermath process benefits with these logic also. The naming convention is extended and used.

- **Physical models**

  The requirements for physical structures arise due to the fact that the project as a real time project is expected to function with humans. The important physical structures are the AFC gates and the camera system.

4.3 DATA FLOW DIAGRAM

4.3.1 DESCRIPTION

A data flow (DFD) is a graphical representation of the "flow" of data through information through information system, modeling its prospects. DFDs are also used for the visualization of data processing. A DFD shows that what kind of information will be input to the system and output from the system, where the data will come from and go to, and where the data will be stored. It does not show about the timing of the processors, or information about whether processes will operate in sequence or in parallel.

Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation.

Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow.

4.3.2 USES OF DFD's

- Data flow diagram is useful for establishing the boundary of the business or system domain (the sphere of analysis activity) under investigation.
- It identifies external entities along with their data interfaces that interact with the processes of interest.
- A DFD can be useful tool for helping secure stake holder agreement (sign–off) on the project scope. It is also a useful tool for breaking down a process into its sub processes for closer analysis.
- It helps in the logical information flow of the system.
- It determines of physical system construction requirements.
- Simplicity of notation.
- It establishes the manual and automated systems requirements.

4.3.3 APPLICATIONS

A data flow diagram can be modeled in the early requirements elicitation process of the analysis phase within the system development life cycle (SDLC) to define the project scope. A data flow diagram can also be created throughout the system development life cycle that investigates an aspect of the system

if necessary, each process under study within a DFD can be broken down into its sub-process on a new data flow diagram to show more details. A sub-process in turn can be broken down further to reveal its sub-process on a new data flow diagram, and so until sufficient analysis is reached. The activity of drilling down the data flow diagram level is called "functional decomposition" with the resulting new data flow diagram referred to as level DFDs.

4.3.4 COMPONENTS

A DFD can be assembled from the four components

1. **Process**: A process is a logical activity that transforms many floods within the incoming domain. A process can be regarded as a (black box) is to it receive input and produce output. A rounded rectangle (or circle) represents a process under a study. Each process is a labeled inside into rectangle describes its function is proposed as shown in the figure 4.1.

Figure 4.2: Process

2. **External entity:** An entity is referred to as external source or sink (destination)for data flowing in and of domain as shown in the figure 4.2. A rectangle defines an external and is labeled with a noun phrase its rectangle to an organization, process, machine or person (i.e., a thing) is outside the domain under analysis.

Figure 4.3: External Entity

3. **Dataflow:** A data represents the path of data moving is thought the under analysis. A data flow shows the movement of data between a process and an external entity. An arrow is the symbol used a process with other DFD components as shown in the figure 4.3.

Figure 4.4: Data Flow

4. **Data store:** Figure 4.4 shows the data store, represents a logical data repository accessible within the domain under study a data store can be a place where data is created, read, changed and stored temporality or permanently by a process. A thin rectangle with the store side open shows a data store an mislabeled with a noun phrase inside its rectangle to describe the data store.

Figure 4.5: Data Store

4.3.4 LEVELS OF ABSTRACTION

Figure 4.6: Level 0 Data Flow Diagram

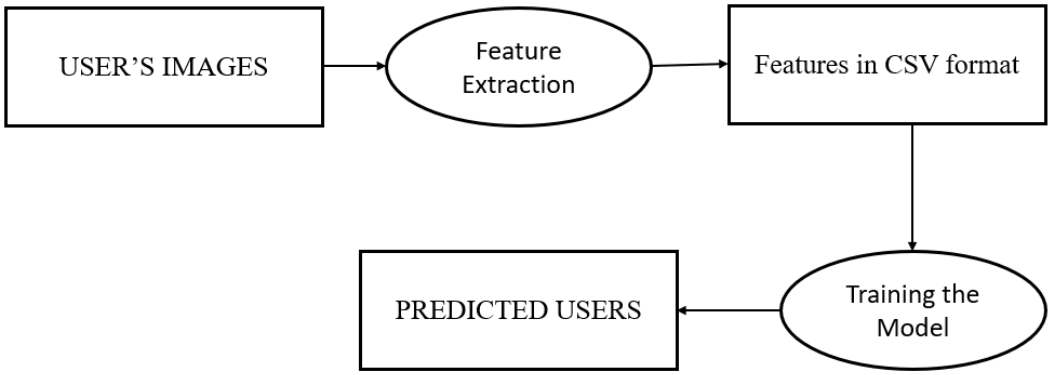


Figure 4.7: Level 1 Data Flow Diagram

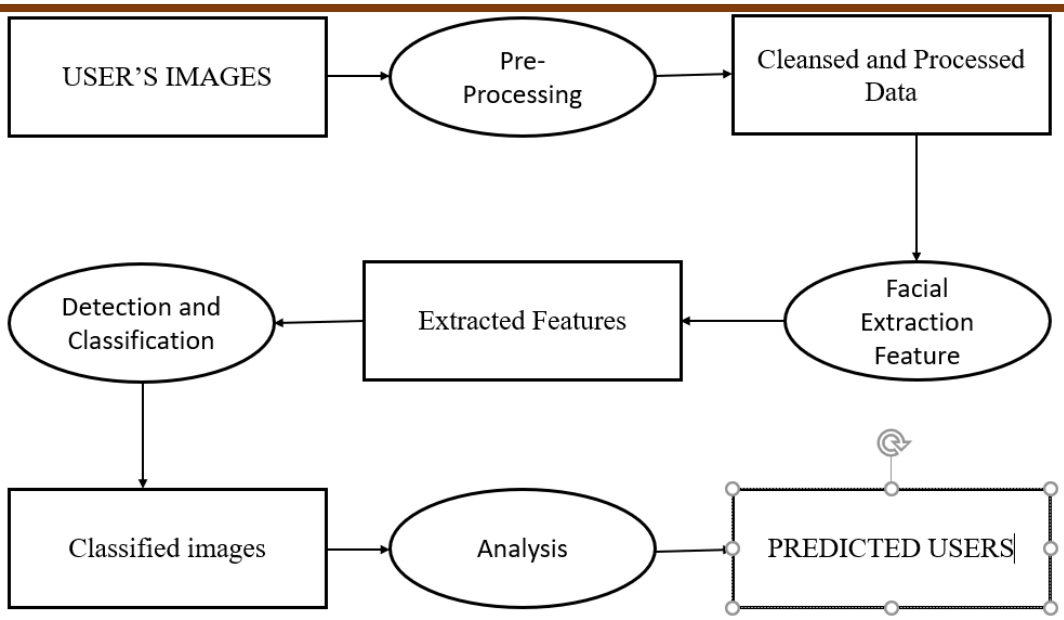


Figure 4.8: Level 2 Data Flow Diagram

Figure 4.6 shows the DFD Level 0 is also called a Context Diagram. It is a basic overview of the whole system or process being analyzed or modeled. It is designed to be an at-a-glance view showing the system as a single high-level process with its relationship to external entities.

This context-level data flow diagram (DFD) is next explored, to produce a level 1data flow diagram (DFD) that shows some of the detail of the system being modelled. It also identifies the internal data stores that must be presenting order for the system to do its job, and flow of data between the various inputs of the system. The level 1 data flow diagram (DFD) for the proposed system is shown in figure 4.7.

A level 2 data flow diagram (DFD) offers a more detailed look at the processes that make up an information system than a level 1 Data flow diagram does. It can be used to plan or record the specific makeup of a system. The Data flow diagram for the file uploading process in the proposed system is shown in Figure 4.8.

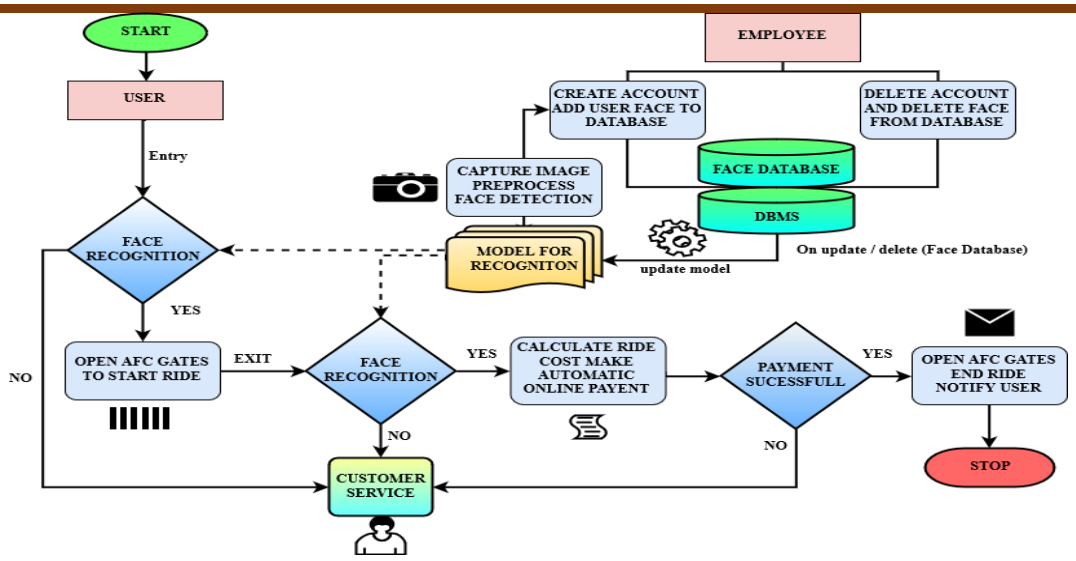


Figure 4.9: WORKFLOW DIAGRAM

Figure 4.9 is the final designed work flow diagram. The above diagram includes all the different features composing of all requirements. The diagram further shows how different individual smaller parts all come together to form the larger project. The face recognition is performed on the user and once detected the AFC gates are opened. The logical part of payment and such are handled and are mentioned in the low diagram. The employee gets to create and delete accounts of users and while doing so also do alterations to the Database as the Database keeps increasing and getting updated. The camera system takes photos and updates to the DB.

4.4 SUMMARY

This chapter gives a brief introduction to the system design process, its methodologies requires for developing a system. It deals with different types of design processes used in real world and system architectures. Proposed model mainly describes the how exactly the system works. Data flow diagram for the proposed model is designed in three different levels of abstraction. Dataflow diagram (DFD's) offers a graphical representation for summarizing the movement of data flows in the different levels of processes. It is mainly discussed about what is the proposed system that is implementing with that of the existing system. It describes the how the complexity is reduced, reduced in the cost and about the performance of the system.

The aim is to design and develop a web content filtering program using machine learning and data mining approaches. The model uses the input as the training and test data sets and effectively classifies the content into predefined classes. The comparison studies can also be carried out which deal with comparing results obtained from different algorithms with those obtained by the algorithm used in this project.

CHAPTER 5

IMPLEMENTATION

5.1 INTRODUCTION

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering. The architectural design of a system emphasizes the design of the system architecture that describes the structure, behavior and more views of that system and analysis. The logical design of a system pertains to an abstract representation of the data flows, inputs and outputs of the system. It is often conducted via modelling, using an over-abstract (and sometimes graphical) model of the actual system. In the context of systems, designs are included. Logical design includes entity-relationship diagrams (ER diagrams). The physical design relates to the actual input and output processes of the system. It is explained in terms of how data is input into a system, how it is verified or authenticated, how it is processed, and how it is displayed.

Implementation is the process of transforming a mental plan in familiar terms into one compatible with the computer. It is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy. Computer programming is a key element in the process of implementation which is the process of designing, writing, testing, debugging / troubleshooting, and maintaining the source code of computer programs. The source code is written in a programming language. The purpose of programming is to create a program that exhibits a certain desired behavior. The process of writing source code often requires expertise in many different subjects, including knowledge of the application domain, specialized algorithms and formal logic.

Metro smart ticket system is a project containing of both hardware and software components. The mentioned setup leads to it having different implementation and system design specifications. The hardware system design is made specifically by keeping in mind the special flow of data between these hardware devices and the software components. Which causes special requirements such as precision and time management. The software also needs to be highly accurate which requires a thorough study of various technologies that are available to make sure that the best technique is employed.

The other regions where the system design implementation is necessary is the front-end and back end. The frontend design must be user friendly and also efficient. Where as the back end must be quick as there is high level speed transactions required. All of the discussed technologies should also work when they are integrated together.

5.2 SYSTEM DESIGN

There are a wide range of methodologies that can be used in order to design a system. The most important aspects when it comes to selecting a methodology for a project are:

- The accuracy
- The complexity
- The ability to implement

These are the aspects that need to be taken into consideration while selecting an ideal way in which a project can be implemented.

The design of the system must be done in such a way that all the above mentioned conditions are taken into account and are used in the best possible way.

5.2.1 System Architecture

A system architecture diagram is used to show the relationship between different components. Usually they are created for systems which include hardware and software and these are represented in the diagram to show the interaction between them.

One can think of system architecture as a set of representations of an existing (or future) system. These representations initially describe a general, high-level functional organization, and are progressively refined to more detailed and concrete descriptions. System architecture conveys the informational content of the elements consisting of a system, the relationships among those elements, and the rules governing those relationships. The architectural components and set of relationships between these components that an architecture description may consist of hardware, software, documentation, facilities, manual procedures, or roles played by organizations or people. Figure 5.1 shows the system architecture for the proposed model.

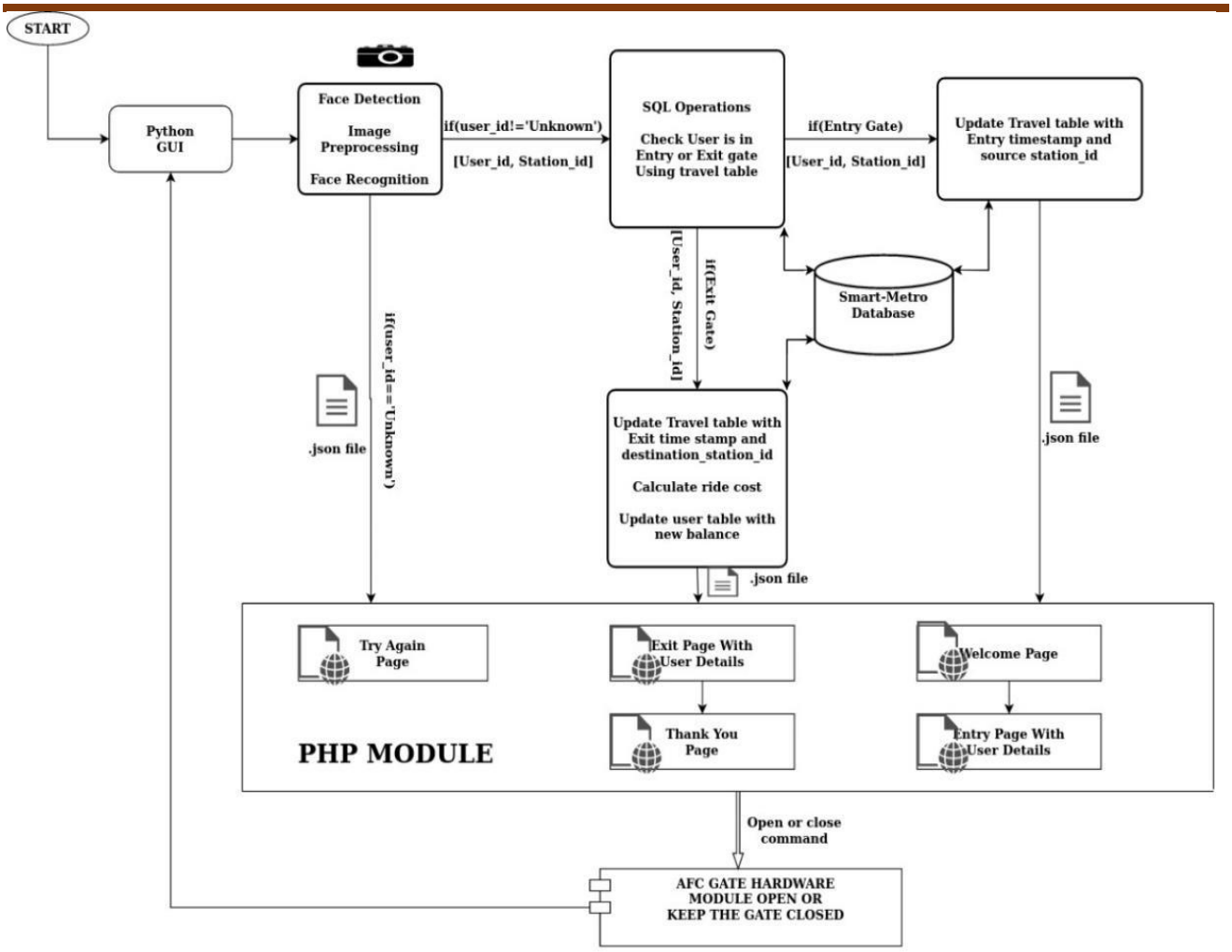


Figure 5.1 System Architecture

5.2.2 Flow Chart

A flowchart is one of the seven basic quality tools used in project management and it displays the actions that are necessary to meet the goals of a particular task in the most practical sequence. Also called as process maps, which type of tool displays a series of steps with branching possibilities that depict one or more inputs and transforms them to outputs.

The advantage of flowcharts is that they show the activities involved in a project including the decision points, parallel paths, branching loops as well as the overall sequence of processing through mapping the operational details within the horizontal value chain. Moreover, particular tool is very used in estimating and understanding the cost of quality for a particular process. Which is done by using the branching logic of the workflow and estimating the expected monetary returns. Figure 5.2 shows the flow chart for the proposed system.

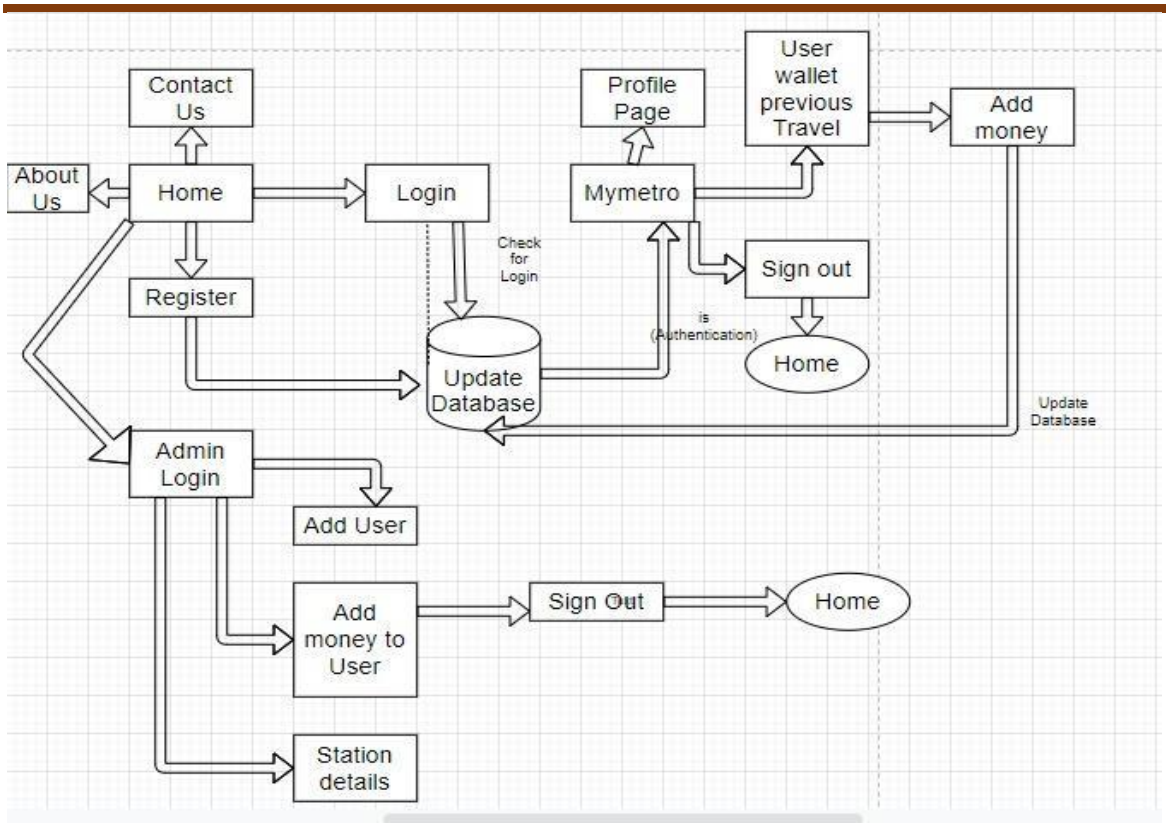


Figure 5.2 Flow Chart

5.2.2 Data Collection

Figure 5.4 AT&T and Georgia Tech dataset images

The fig 5.4 shows the collection of images used in training the network using the standard dataset of AT&T dataset and Georgia Tech dataset. These datasets are standard datasets are obtained from

the various referenced papers that are used to complete the project. The datasets include images of people in different angles and different light intensity. The quality of images used are also of different variations. The other dataset used is made specially for following project using standard camera system and including about 10 images in various orientations. The folders are named with labels that specify their identity.

5.2.3 Preprocessing

The preprocessing techniques are applied on the raw images from the dataset. The images are first made in greyscale. From the image can be used for detection of face. The detection of face in image is important as the region only includes the pixels that are important. Face detection went mainstream in the early 2000's when Paul Viola and Michael Jones invented a way to detect faces that was fast enough to run on cheap cameras. However, much more reliable solutions exist now. We're going to use a method that was invented in 2005 called Histogram of Oriented Gradients — or just *HOG* for short. Looking at every single pixel in the images one at a time and for every single pixel, we will have to look at the pixels that directly surrounding it. The goal is to figure out how dark the current pixel is compared to the pixels directly

surrounding it. Then draw an arrow showing in which direction the image is getting darker. After repeating the process for every single pixel in the image, we end up with every pixel being replaced by an arrow. These arrows are called *gradients* and they show the flow from light to dark across the entire image.

The above process is really good for replacing the pixels with gradients. If we analyze pixels directly, really dark images and really light images of the same person will have totally different pixel values. But by only considering the direction that brightness changes, both really dark images and really bright images will end up with the same exact representation. That makes the problem a lot easier to solve. But saving the gradient for every single pixel gives us way too much detail. It would be better if we could just see the basic flow of lightness/darkness at a higher level so we could see the basic pattern of the image. To do which, we'll break up the image into small squares of 16x16 pixels each. In each square, we'll count up how many gradients point in each major direction (how many point up, point up-right, point right, etc…). Then we'll replace that square in the image with the arrow directions that were the strongest. The end result is we turn the original image into a very simple representation that captures the basic structure of a face in a simple way. Figure 5.5 describes the final HOG processed image. To find faces in HOG image, all we have to do is find the part of our image that looks the most similar to a known HOG pattern that was extracted from a bunch of other training faces.

The images might have photos of people in different angle and their faces might not be rightly oriented. Which means we need to do something called warping. Warping is done with respect to eyes and lips. These are the general face landmarks. Which will make it a lot easier for us to compare faces in the next steps. To do which, we are going to use an algorithm called **face landmark estimation**. The basic idea is we will come up with 68 specific points (called *landmarks*) that exist on every face — the top of the chin, the outside edge of each eye, the inner edge of each eyebrow, etc. Then we will train a machine learning algorithm to be able to find these 68 specific points on any face. Once we know where the eyes and mouth are, we'll simply rotate, scale and shear the image so that the eyes and mouth are centered as best as possible. It doesn't have to do any fancy 3d warps because that would introduce distortions into the image. We are only going to use basic image transformations like rotation and scale that preserve parallel lines called affine transformations.

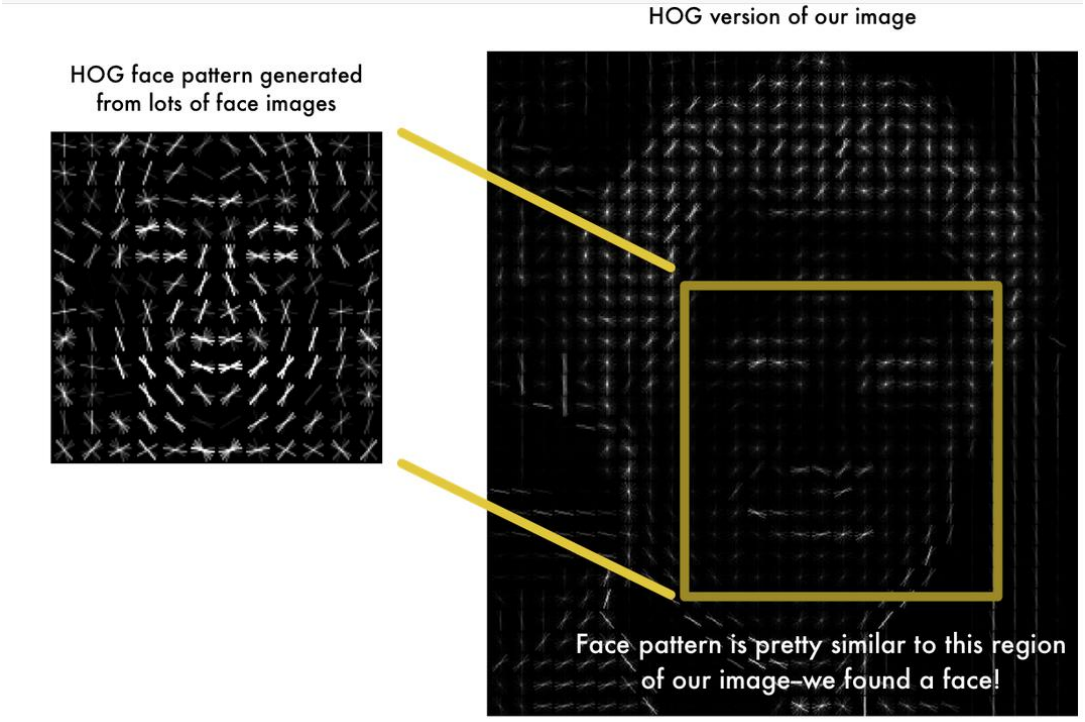


Figure 5.5 HOG processing

5.2.4 Algorithm

After we have pre processed our images we can start with employing our algorithm to help with face recognition. There are various methods available to perform facial recognition. The goal is to find the ones with most accuracy and time management. With reduced computation requirement.

PCA Algorithm:

PCA is a statistical approach used for reducing the number of variables in face recognition. In PCA, every image in the training set is represented as a linear combination of weighted eigenvectors called eigenfaces. These eigenvectors are obtained from covariance matrix of a training image set. The main idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of many variables correlated with each other, either heavily or lightly, while retaining the variation present in the dataset, up to the maximum extent. The same is done by transforming the variables to a new set of variables, which are known as the principal components (or simply, the PCs) and are orthogonal, ordered such that the retention of variation present in the original variables decreases as we move down in the order. So, in a way, the 1st principal component retains maximum variation that was present in the original components. The principal components are the eigenvectors of a covariance matrix, and hence they are orthogonal.

In the above model, we use PCA for testing and compared to our model, PCA has less accuracy. In PCA, we find the Linear combination of features which maximizes the total variance in data. It is a powerful way of representing the data but it doesn't consider any classes and so a lot of discriminative into may be lost when throwing the components away. It turns a set of possibly correlated variables into smaller set of uncorrelated variables. The High-dimensionality dataset is often described by correlated variables and therefore only a few meaningful dimensions account for most of the information. That might be the reason for less accuracy in our model. The variances in our data is generated by external features like Light, composure of the image, etc. Here, the components containing discriminative information at all . so the projected samples are smeared together and classification becomes a bit difficult.

LDA Algorithm:

Linear Discriminant Analysis (LDA) is a popular feature extraction technique for face image recognition and retrieval. However, It often suffers from the small sample size problem when dealing with the high dimensional face data.

It has been demonstrated that the Linear Discriminant Analysis (LDA) approach outperforms the Principal Component Analysis (PCA) approach in face recognition tasks. Due to the high dimensionality of a image space, many LDA based approaches, however, first use the PCA to project an image into a lower dimensional space or so-called face space, and then perform the LDA to maximize the discriminatory power. It is proposed a new, unified LDA/PCA algorithm for face recognition. The new algorithm maximizes the LDA criterion directly without a separate PCA step. Which eliminates the possibility of losing discriminative information due to a separate PCA step. We discuss the connection between the new algorithm and the traditional PCA+LDA approach. We

also prove that the new algorithm is equivalent to the eigenface (PCA) approach in a special case, where each person has only one sample in the training set.

LDA performs class specific dimensionality reduction in our model. In order to find the combination of features that are separates the best between the classes. LDA maximizes the ratio between class to within class over overall scatter. The idea being very simple, Same class should cluster tightly together. While, different classes are as far away as possible from each other in the lower dimensional representation.

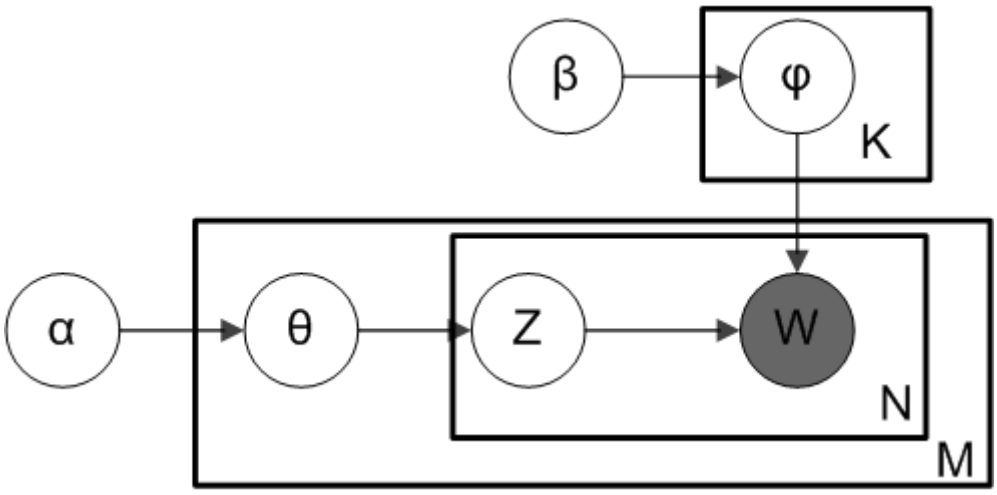


Figure 5.6 Plate Diagram of LDA Model

Figure 5.5 shows the plate diagram of LDA.

α is the per-document topic distributions,
β is the per-topic word distribution,
θ is the topic distribution for document *m,*
φ is the word distribution for topic *k,*
z is the topic for the *n*-th word in document *m*, and
w is the specific word.

FACE-RECOGNITION:

All the above algorithms use image objects to work with face recognition. The better way to work with is to create lists of points. One such method employs the use of 128 different point defining each face. Using deep learning method we can bring about a type of encoding that generates 128 different values each defining a image. The above process is shown in Figure 5.7.

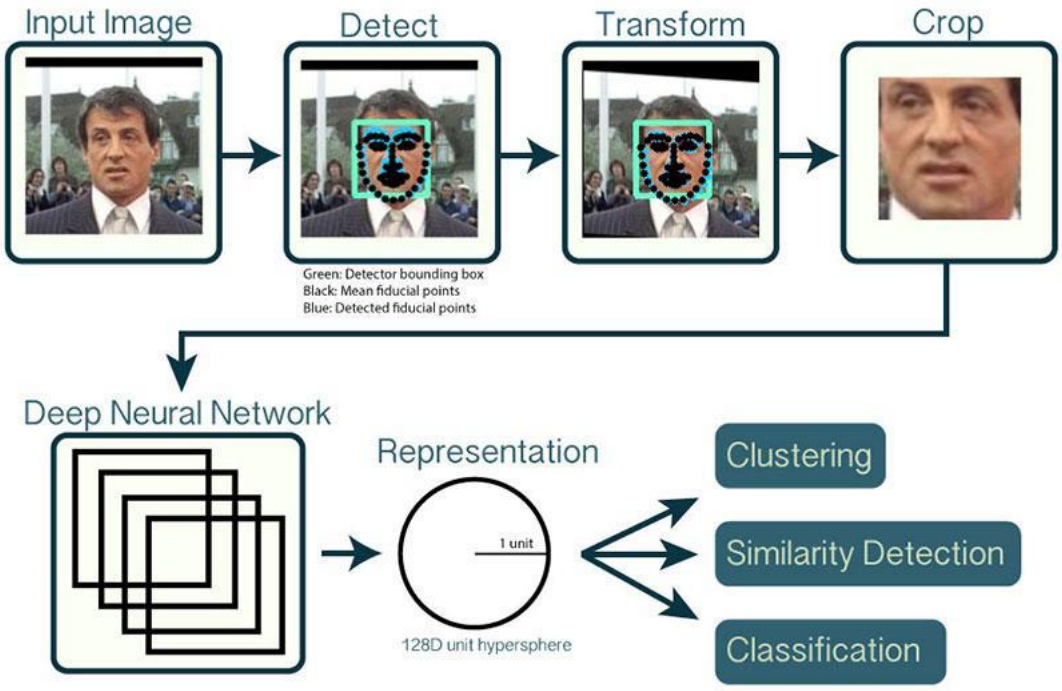


Figure 5.7 OpenCV Face Recognition

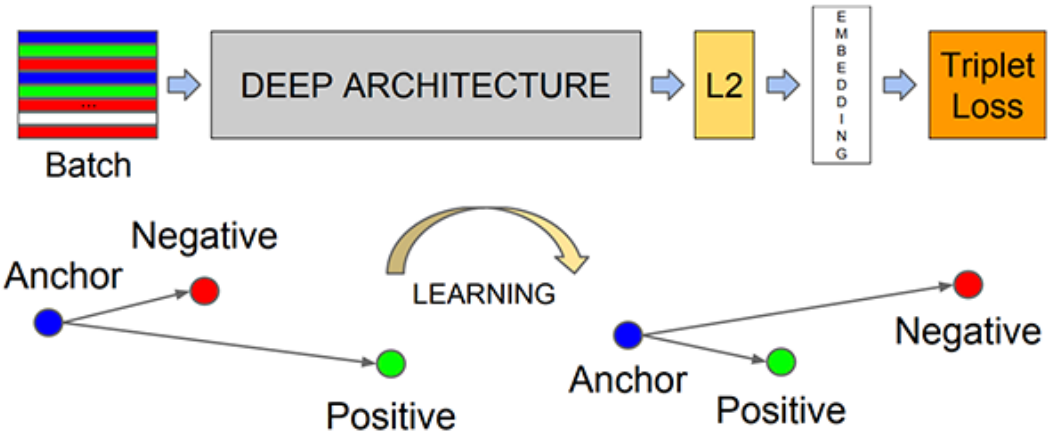


Figure 5.8 128D value calculation

The Figure 5.8 shows the working of **triplet loss function.** The triplet loss function takes three images. The main image, secondary image and a negative image. The deep learning method employed will generate 128 values such that they are closer to the values of images of same person and as far as possible to other persons. The weights of **CNN** network are tweaked in order to generate such possible numbers.

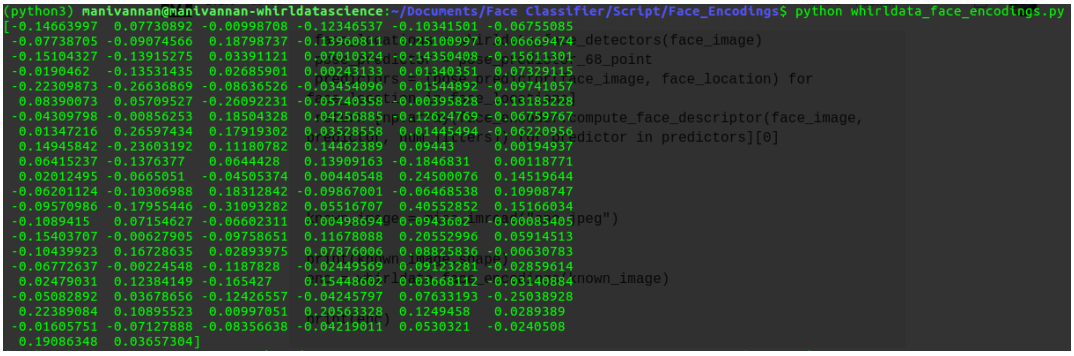


Figure 5.9 128D sample values

Once we have values like the one in Figure 5.9 we can add classifiers like the **SVM** classifier to obtain the prediction of test image. The label is given as the result.

Language and GUI: Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures combined with dynamic typing and dynamic binding make it very attractive for Rapid Application Development as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed. Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes it simple approach very effective.

5.3 SUMMARY

The complete system architecture and implementation has and been shown very detailly and decrepitly. With the help of inbuilt high level neural networks API and the libraries present in python eases the development of the model. Also, the optimizations required for a model can be performed in simple steps. The confusion matrix generated can be used to see the performance of the model. Finally, the prediction of the emotional state can be believed and predicted if the employee needs care or is he happy with his work.

The system has been designed in such a way that their will be one process of representing the system. The system is designed in the simple way the user can easily use the system according to their convince. The system is simple, cost effective and more reliable to use. The user uses system can be reduce the cost, the user need not be wait for the doctor their will be an reduce in the time of waiting. The different facilities has been provided to the user in the system where their will not be present in the other system

CHAPTER 6

RESULT ANALYSIS

6.1 INTRODUCTION

Analysis is the process of considering something carefully or using statistical methods in order to understand it or explain it. Here, the results that are obtained after execution of the various algorithms and software are subject to analysis. Which is further used in drawing conclusions related to the various parameters of the project like accuracy, complexity, usability, etc. The results which are obtained help the user in understanding the true purpose of the project and the need for its implementation.

Further, they are analyzed for any imperfections if present, and later the results are made available to the user. Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. Although crucial to software quality and widely deployed by programmers and testers, software testing still remains an art, due to limited understanding of the principles of software. The difficulty in software testing stems from the complexity of software: we cannot completely test a program with moderate complexity. Testing is more than just debugging. The purpose of testing can be quality assurance, verification and validation, or reliability estimation. Testing can be used as a generic metric as well. Correctness testing and reliability testing are two major areas of testing. Software testing is a trade-off between budget, time and quality.

6.2 TEST CASES

Testing is the process of trying to discover every conceivable fault or weakness in a work product. Testing provides a way to check the functionality of components, subassemblies, assemblies and a finished product. Testing is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fall in an unacceptable manner. If testing is done successfully it will remove all the errors from the software. There are various types of test. Each type addresses a specific testing environment.

Principles of Testing:

1. **Testing shows the presence of defect:** Every application or product is released into production after a sufficient amount of testing by different teams or passes through different phases like System Integration Testing, User Acceptance Testing, and Beta Testing etc.

2. **Early Testing:** Testers need to get involved at an early stage of the Software Development Life Cycle (SDLC). Thus the defects during the requirement analysis phase or any documentation defects can be identified. The cost involved in fixing such defects is very less when compared to those that are found during the later stages of testing.

3. **Exhaustive Testing is not possible:** Exhaustive testing takes unlimited efforts and most of those efforts are ineffective. Also, the project timelines do not allow testing of so many number of combinations. Hence it is recommended to test input data using different methods like Equivalence Partitioning and Boundary Value Analysis.

4. **Testing is Context- Dependent:** There are several domains available in the market like Banking, Insurance, Medical, Travel, Advertisement etc., and each domain has a number of applications. Also for each domain, their applications have different requirements, functions, different testing purpose, risk, techniques etc. Different domains are tested differently, thus testing is purely based on the context of the domain or application.

5. **Defect Clustering:** During testing, it may happen that most of the defects found are related to a small number of modules. There might be multiple reasons like the modules may be complex, coding related to such modules may be complicated etc. The Pareto Principle of software testing where 80% of the problems are found in 20% of the modules.

6. **Pesticide Paradox:** Pesticide Paradox principle says that if the same set of test cases are executed again and again over the period of time then these set of tests are not capable enough to identify new defects in the system. In order to overcome “Pesticide Paradox”, the set of test cases needs to be regularly reviewed and revised. If required a new set of test cases can be added and the existing test cases can be deleted if they are not able to find any more defects from the system.

7. **Absence of Error:** If the software is tested fully and if no defects are found before release, then we can say that the software is 99% defect free. But what if the software is tested against wrong requirements? In such cases, even finding defects and fixing them on time would not help as testing is performed on wrong requirements which are not as per needs of the end user.

Types of Testing:

1. Unit Testing

It focuses on smallest unit of software design. It is used to test an individual unit or group of inter related units. It is often done by programmer by using sample input and observing its corresponding outputs.

2. Integration Testing

The objective is to take unit tested components and build a program structure that has been dictated by design. Integration testing is testing in which a group of components are combined to produce output.

Integration testing are of two types: (i) Top down (ii) Bottom up

3. Regression Testing

Every time new module is added leads to changes in program. Regression testing makes sure the whole component works properly even after adding components.

4. Smoke Testing

The test is done to make sure that software under testing is ready or stable for further testing. It is called smoke test as testing initial pass is done to check if it did not catch the fire or smoked in the initial switch on.

5. Alpha Testing

It is a type of validation testing. It is a type of acceptance testing also which is done before the product is released to customers. It is typically done by QA people.

6. Beta Testing

The beta test is conducted at one or more customer sites by the end-user of the software. The version released is for the limited number of users for testing in real time environment.

7. System Testing

In the software is tested such that it works fine for different operating system. It is covered under the black box testing technique. It focuses on required input and output without focusing on internal working. It has security testing, recovery testing, stress testing and performance testing.

8. Stress Testing

It gives unfavourable conditions to the system and checks how they perform in those conditions.

9. Performance Testing

It is designed to test the run-time performance of software within the context of an integrated system. It is used to test speed and effectiveness of program.

Test Cases

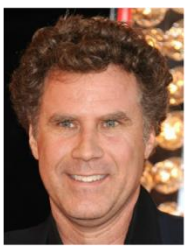
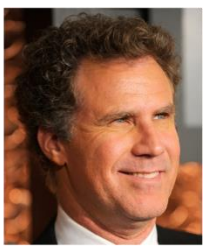
Test case 1:

Test Case ID	Test 1
Test Description	Testing normal functioning with group
Input	4 images of a 4 different persons in a group
Expected Output	All the person's name is recognized and updated in db
Actual Output	The system correctly predicts the name and updates db
Test Result	PASS

Test case 2:

Test Case ID	Test 2
Test Description	Testing normal function with non-aligned face
Input	1 images of a single person 
Expected Output	The person's name is recognized and updated
Actual Output	The system correctly predicts the name and enters in db
Test Result	PASS

Test case 3

Test Case ID	Test 3
Test Description	Testing normal function with identical people(ex: twins)
Input	2 images of a different persons  
Expected Output	The persons name is recognized precisely

Actual Output	The system correctly predicts each name
Test Result	PASS

Test case 4

Test Case ID	Test 4
Test Description	Testing working of billing between stations
Input	2 different input stations for ride
Expected Output	The cost should be calculated and deleted from db.
Actual Output	The system correctly calculates and shows new balance.
Test Result	PASS

Test case 5

Test Case ID	Test 5
Test Description	Entry and Exit Displays with correct name
Input	The user logged in is session and input also.
Expected Output	The screens should show the user details.
Actual Output	The system correctly forms and displays the screens.
Test Result	PASS

Test case 6

Test Case ID	Test 6
Test Description	Working of AFC gate simulator
Input	Input from python code as binary form.
Expected Output	The gate should open and shut in time period.
Actual Output	The system correctly opens and shuts the gate.

Test Result	PASS
	 

Test case 7

Test Case ID	Test 7
Test Description	Testing working of new user addition
Input	User details and images.
Expected Output	The user details should be added to db and image processed and kept in db labelled.
Actual Output	The system correctly stores the data.
Test Result	PASS

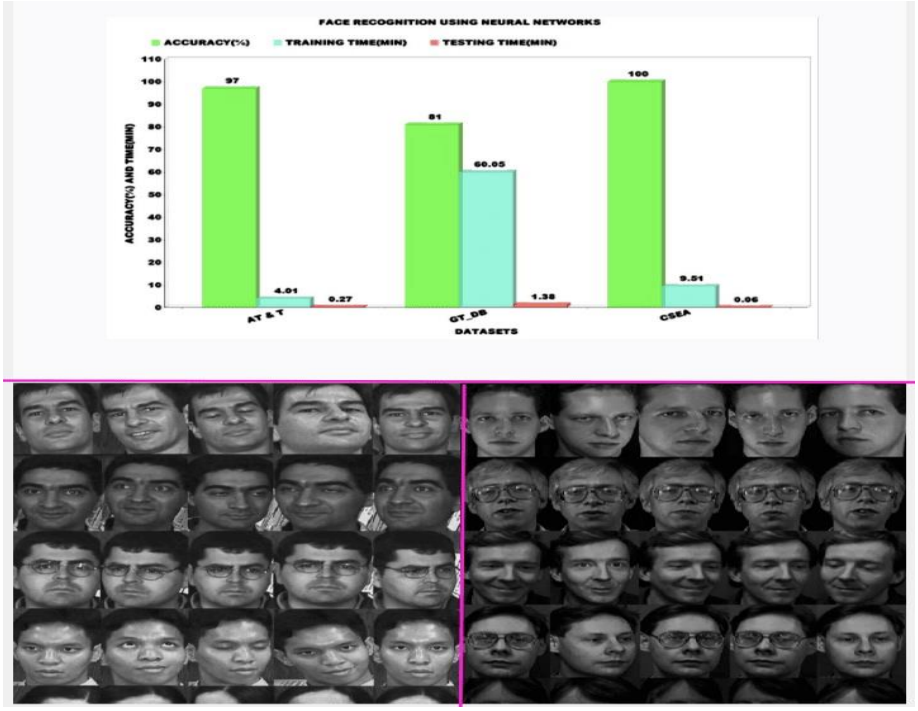
Test case 8

Test Case ID	Test 8
Test Description	Testing working of facial recognition on speed
Input	1 image of the person required.
Expected Output	The image must be recognized at fast pace.
Actual Output	The system correctly recognizes the pers
Test Result	PASS

6.3 RESULTS

Figure 6.1 : Application Home Page

Figure 6.1 show the GUI developed to use the Smart Metro Application. It has easy access to website, code, documentation, backend etc.



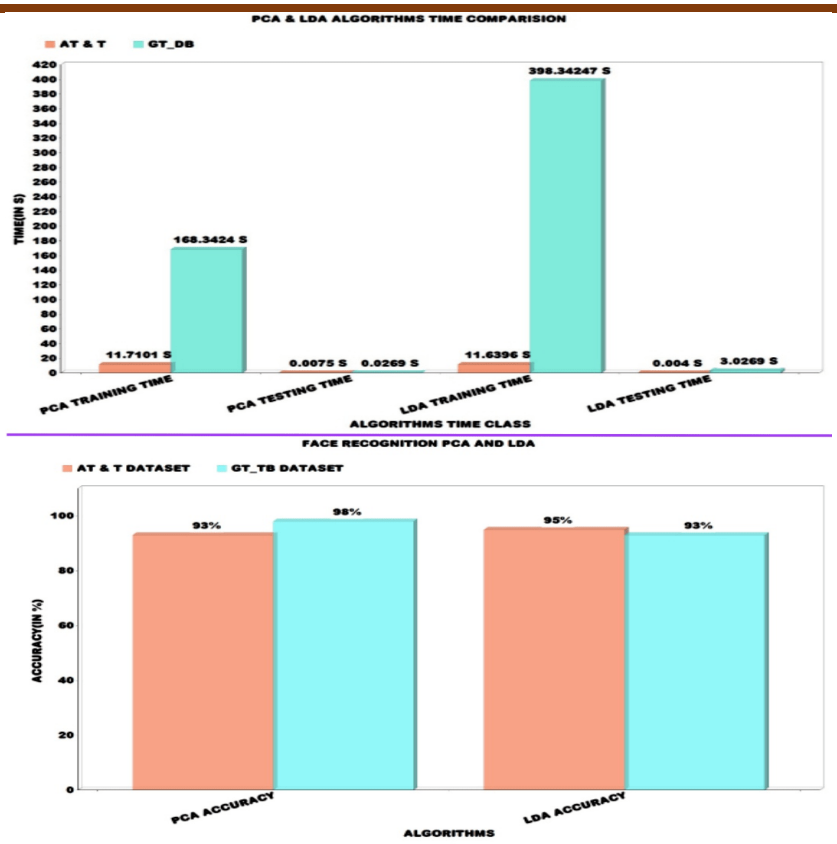


Figure 6.2 : Application Results

Figure 6.2 shows the quick results generated on the working of the code and is always updated.

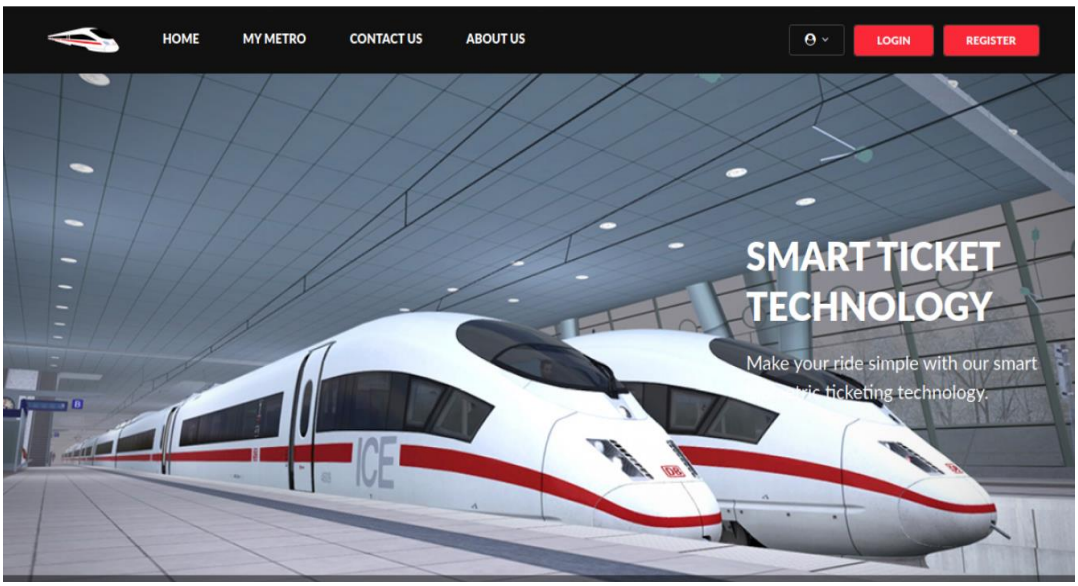


Figure 6.3 : Website home display

Figure 6.3 shows the Website for the online access of Smart metro ticket system. It has all the features of admin, user and employee.

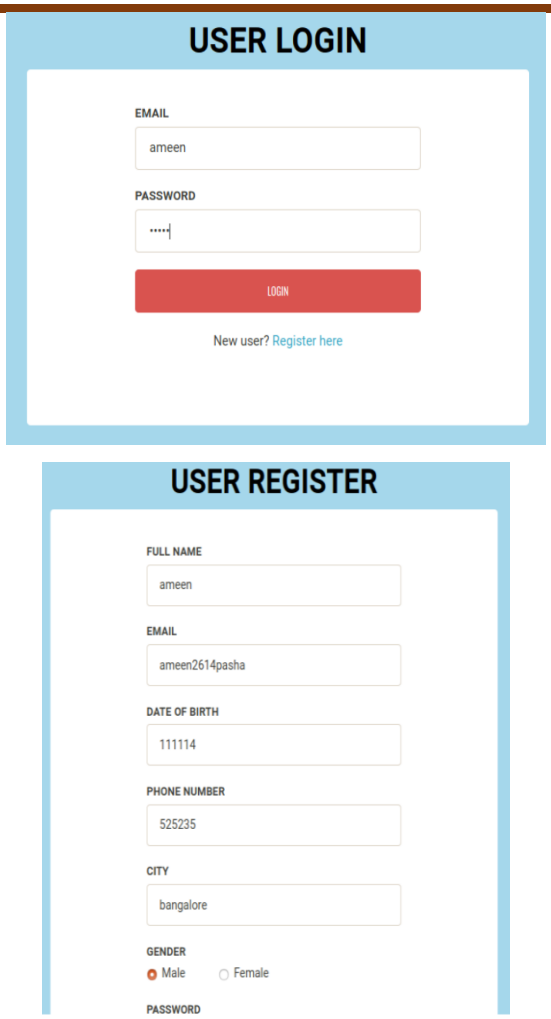


Figure 6.4 : Login and Registration

Figure 6.4 shows the Login and Registration pages for users. Employees and admin get similar pages. Admins can not be registered.

Figure 6.5 : Metro Entry display

Figure 6.5 shows the entry page once the person is authenticated. The page displays the predicted name and wallet and starting station in which the customer has visited.

Figure 6.6 : Metro Exit display

Figure 6.6 shows the exit page once the person is authenticated. The page displays the predicted name and wallet and starting station in which the customer has visited. This page also shows the exit station and the cost of the ride. The balance amount is updated and shown.

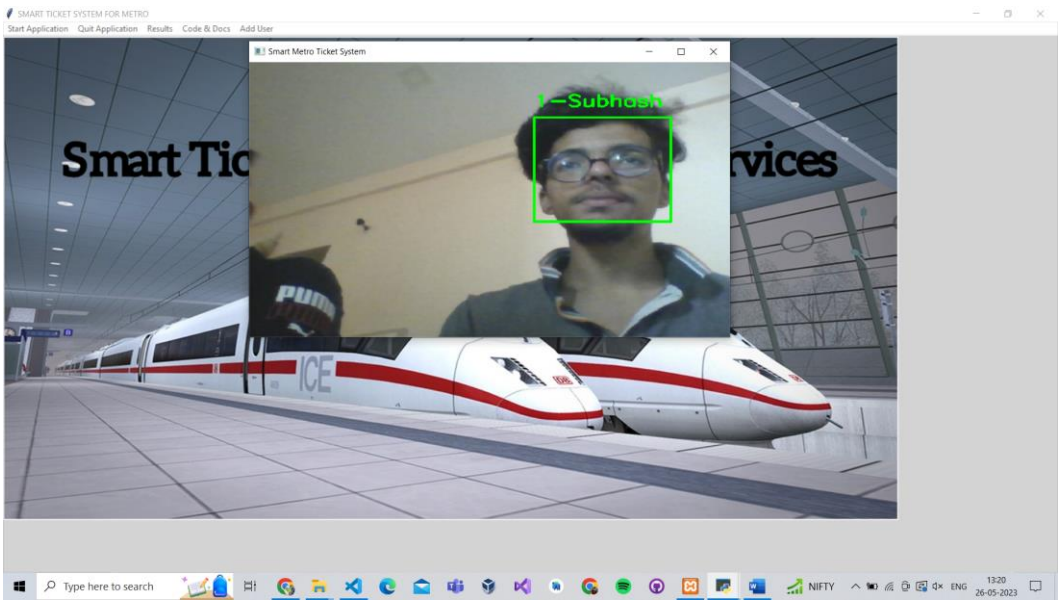


Figure 6.7 : Metro Ride Facial Recognition

Figure 6.7 shows the page during the person is authenticated. The page displays the predicted name and after this the entry page is opened if the entry station is empty or the exit station is opened.

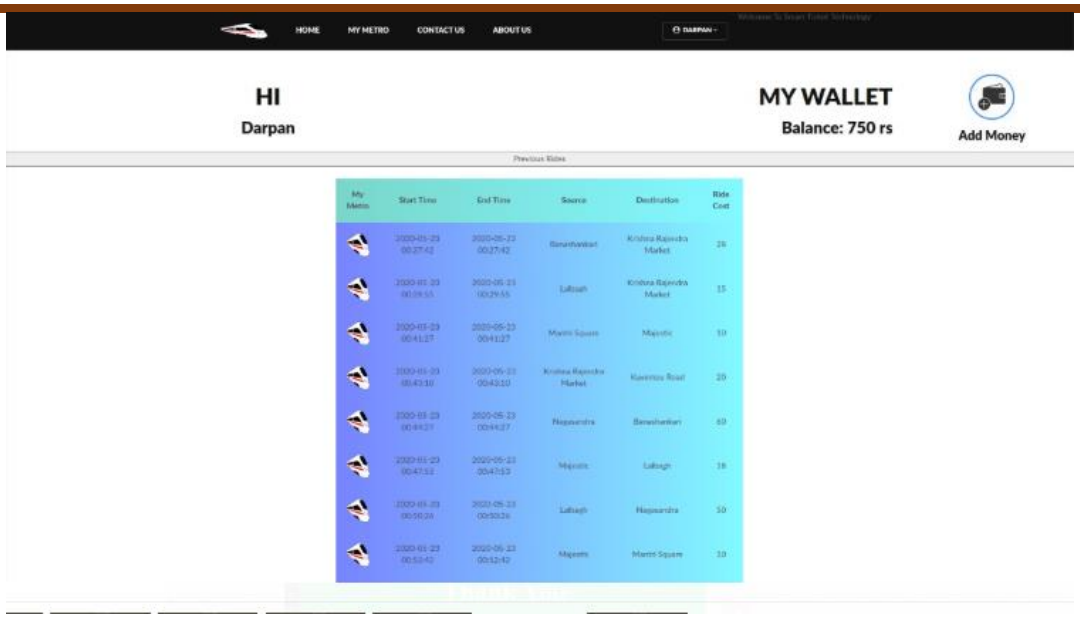


Figure 6.8 : User Ride Information

Figure 6.8 shows the User ride history page once the person is logged in. The page displays the session name and wallet and starting station in which the customer has visited and destination station. The ride cost is also mentioned along with date and time.

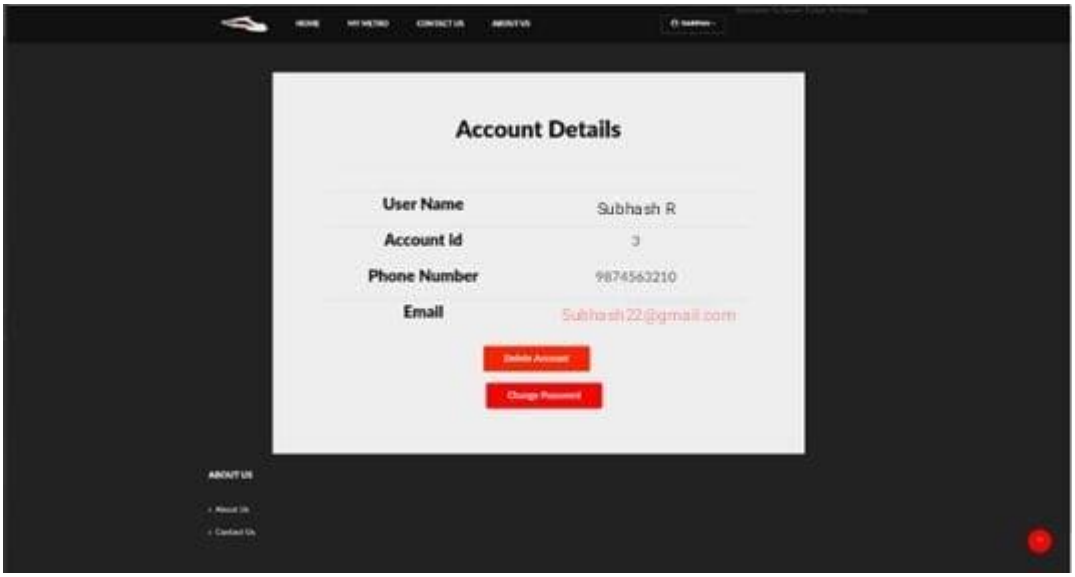


Figure 6.9 : User Personal Information

Figure 6.10 shows the User personal information page once the person is logged in. The page displays the shows all the data from database and also gives option for user to delete account or change password.

My Metro	Station Id	Station Name	Station Phone	Station Adress	Super Visor
	1	Yelechenahalli	1800-425-12345	Yelechenahalli, Bengaluru	emp1
	2	J P Nagar	1800-425-12345	J P Nagar, Bengaluru	emp1
	3	Banashankari	1800-425-12345	Banashankari, Bengaluru	emp1
	5	Rastriya Vidyalaya Road	1800-425-12345	Rastriya Vidyalaya Road, Bengaluru	emp1
	6	Jayanagar	1800-425-12345	Jayanagar, Bengaluru	emp1
	7	South End Circle	1800-425-12345	South End Circle, Bengaluru	emp1
	8	Lalbagh	1800-425-12345	Lalbagh, Bengaluru	emp1

Figure 6.10 : Admin Metro View

Figure 6.11 shows the admin's view of metro database in online portal

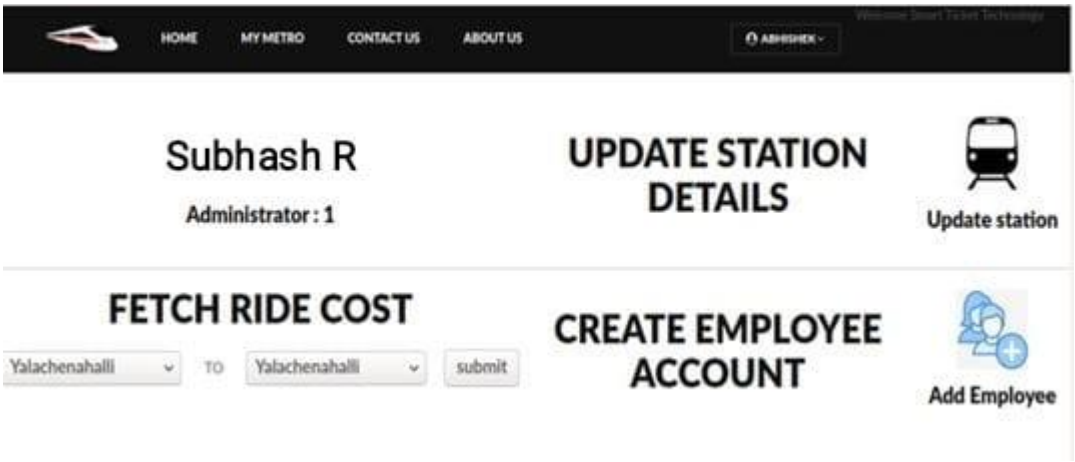


Figure 6.11 : Admin Metro Operations

Figure 6.11 displays the admin operation along with the admin's name and ride cost calculator. Admin can update station details and create employee account from here.

6.4 PERFORMANCE EVALUATION

To evaluate the algorithms and compare them, various parameters are considered like:

✓ Confusion Matrix

✓ Accuracy

✓ Precision

✓ Recall

✓ F1 - Score

1. CONFUSION MATRIX:

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. Which is the key to the confusion matrix.
The confusion matrix shows the ways in which your classification model is confused when it makes predictions.

	CLASS 1 Predicted	CLASS 2 Predicted
CLASS 1 Actual	TP	FN
CLASS 2 Actual	FP	TN

Here,

• Class 1 = Positive

• Class 2 = Negative

• Positive (P) = Observation is positive.

• Negative (N) = Observation is not positive.

• True Positive (TP) = Observation is positive, and is predicted to be positive.

• False Negative (FN) = Observation is positive, but is predicted negative.

• True Negative (TN) = Observation is negative, and is predicted to be negative.

• False Positive (FP) = Observation is negative, but is predicted positive.

2. ACCURACY:

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

Accuracy=Number of correct predictions/Total number of predictions

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

Accuracy=TP+TN/TP+TN+FP+FN

Where TP = True Positives,

TN = True Negatives,

FP = False Positives,

FN = False Negatives.

3. PRECISION:

The total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labeled as positive is indeed

positive (small number of FP).Precision is given by the relation:Precision= TP/TP+FPHigh recall, low precision: most of the positive examples are correctly recognized (low FN) but there are a lot of false positives. Low recall, high precision: positive examples (high FN) but those predicted as positive are indeed positive (low FP).

Precision is a good measure to determine, when the costs of False Positive is high. For instance, email spam detection. In email spam detection, a false positive means that an email that is non-spam (actual negative) has been identified as spam (predicted spam). The email user might lose important emails if the precision is not high for the spam detection model.

4. RECALL:

Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (small number of FN).

Recall is given by the relation:

Recall=TP/TP+FN

Recall is also referred to as the true positive rate or sensitivity.

5. F1-SCORE:

F1 is a function of Precision and Recall. A measure that combines precision and recall is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score:

F=2.precision.recall/precision+recall

measure uses Harmonic Mean in place of Arithmetic Mean as it punishes the extreme values more. Based on the value of these parameters, suitable graphs are drawn to give a clear understanding of how efficiently each algorithm works. Thus the algorithm which is the best fit can be concluded. The initial training is done on a static data set consisting of both legitimate and phishing URLs and URLs having other suspicious attacks. The testing is done on the data from real time also (dynamic data set).

The calculation of the above parameters where done and plots where made so that there is a better understanding of evaluation. The calculation was done across a huge range. The datasets used are basic dataset made and standard datasets like AT&T and Georgia Tech dataset. The algorithms tested are PCA, LDA and CNN algorithm. The figures 6.13, 6.14 and 6.15 show these plots.

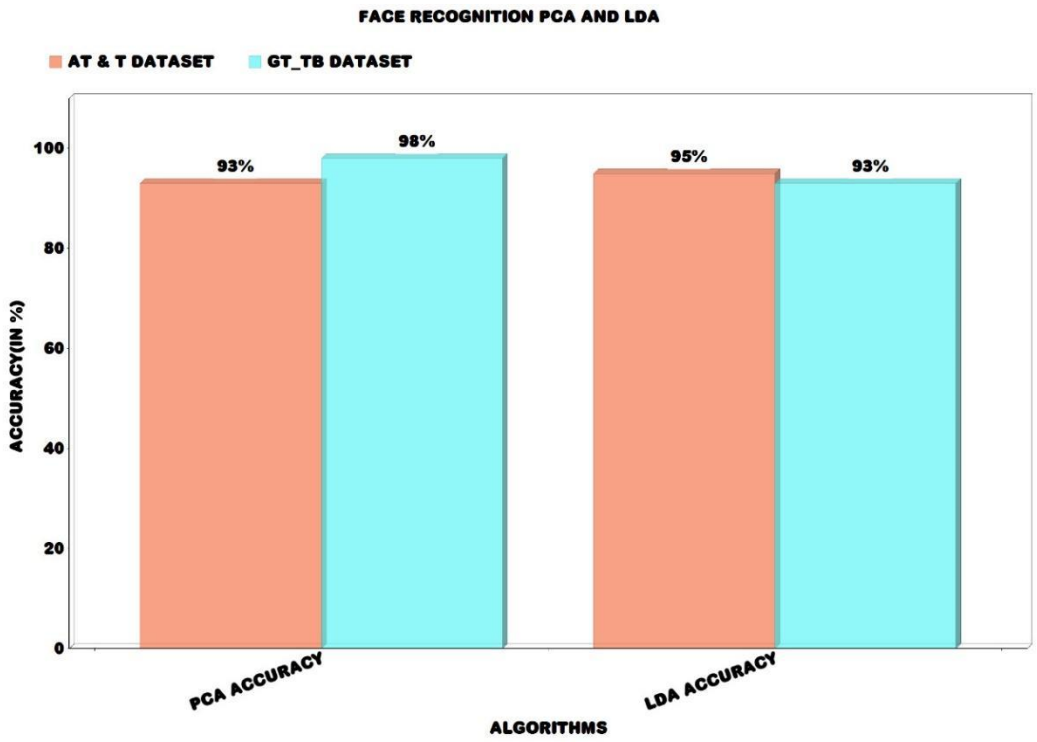


Figure 6.13 : PCA and LDA algorithm

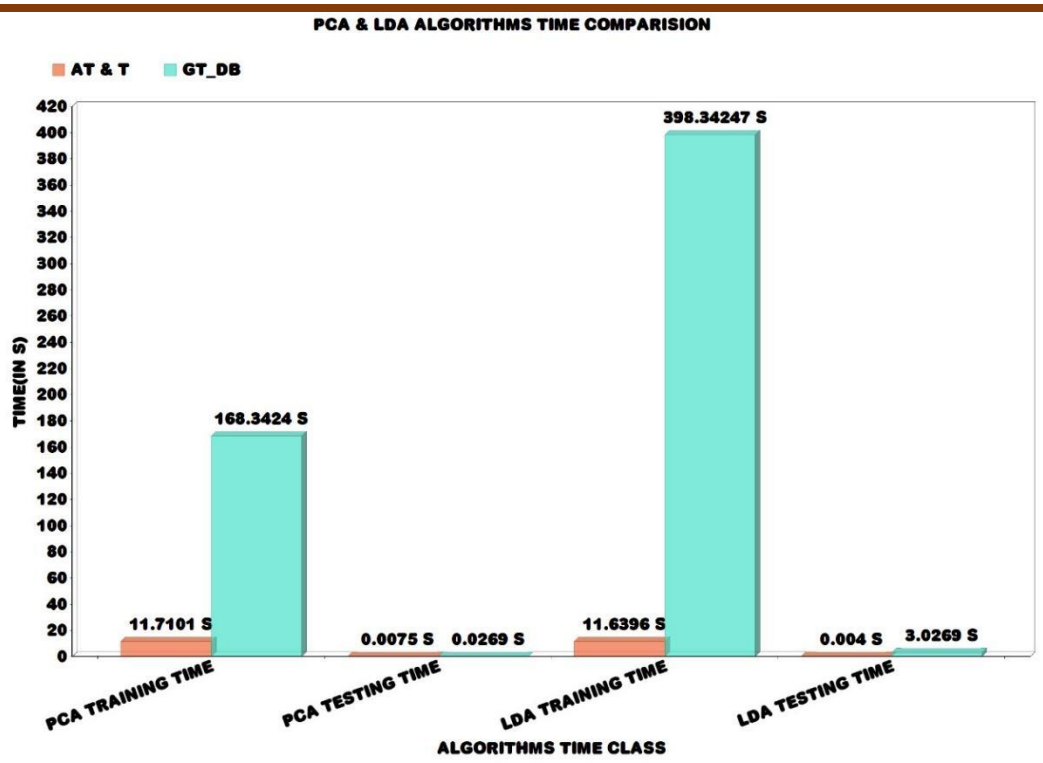


Figure 6.14 : PCA and LDA algorithm with Time

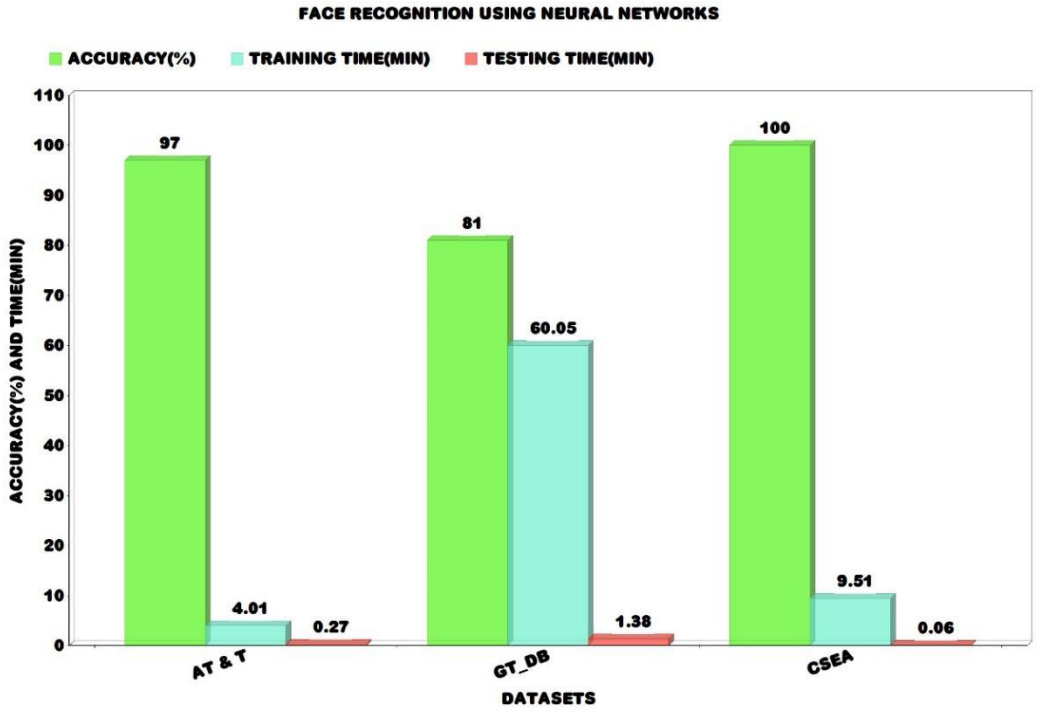


Figure 6.15 : Neural Network on different datasets.

6.5 SUMMARY

Smart Metro Ticket project is a system which makes use of SVMas a classifier. The preprocessing step makes use of two layers of HOG encoding which aids in extracting the features in order to improve the accuracy of the CNN model.

The model uses the input as the training and test data sets and effectively classifies the content into predefined classes of expressions.. The confusion matrix precision is used to describe the performance of the classification model on the set of test data for which the true values are known. It allows the visualization of the performance of an algorithm. The f1 score is used to measure the test's accuracy, and it balances the use of precision and recall to do it. The f1 score of the classifier is very high. The chapter provides an elaborate description of the results obtained for both training and testing dataset.

CONCLUSION AND FUTURE WORK

The Smart Metro Ticket System is project that includes both moving parts and software parts. The hardware uses Arduino board and different sensors for simulation of the AFC gate. The software components bring the user friendly website so that the users can access the portal and perform all the required operations. The GUI is built to show the various aspects of the project and this allows easy access. The goal was to provide a ticketing system that is easy to use and also very flexible. The system architecture and implementation was made by keeping this in mind. The testing performed shows the results that provide a visibly clear decision that the machine learning techniques that where finalized and employed are of the best and appropriate choices that could have been made. This system is developed using OpenCV and dlib libraries with python. The test results give good accuracy and the system meets all the functional requirements. Hopefully, it will be very useful for the organization to detect the job satisfaction of the employees and hence make the necessary changes to improve the work environment which would, in turn, raise the productivity of the metros.

The current project can be further enhanced by creating a better user-friendly GUI as application varies according to real time application; the model can be trained with images belonging to that particular locations in which it is used to improve the accuracy of prediction. Truly standalone systems with higher computation and realistic price limited camera system would hinder the capacity and the project might need a few changes to completely be reliable

REFERNCES

[1] Keerthi G S, and Usha C R: “Human face recognition in video using Convolutional Neural Network (CNN)”, International Research Journal of Engineering and Technology, 2019.

[2] Savitha G and Keerthi G S, “Video based Face Recognition Using Image Processing Techniques”, International Research Journal of Engineering and Technology, 2019

[3] Varun Kaushik, and Suhas P: “Fool Proof Ticketing System for Public Transport”, Proceedings of theInternational Conference on Communication and Electronics Systems (ICCES 2018), IEEE XplOre Part Number: CFP18AWO-ART; ISBN:978-1-5386-4765-3,2018.

[4]Nataliya Boyko, Oleg Basystiuk, and Nataliya Shakhovska : “Performance Evaluation Comparison Of Software for Face Recognition, based on Dlib and Opencv Library”, IEEE Second International Conference on Data Stream Mining & Processing, August 21-25,2018, Lviv, Ukraine.

[5] Neel Ramakant Borkar, and Sonia Kuwelkar : “ Real-Time Implementation Of Face Recognition System”, Proceedings of the IEEE 2017 International Conference on Computing Methodologies and Communication(ICCMC),ISBN: 978-1-5090-4890-8/17, ©2017 IEEE.

[6] Kewen Yan, Shaohui Huang, Yaoxian Song, Wei Liu, and Neng Fan: “Face Recognition Based on Convolution Neural Network”, Proceedings of the 36th Chinese Control Conference, July 26-28, 2017, Dalian, China.

[7] Radhika C Damale, and Prof. Bageshree V Phatak: “Face Recognition Based Attendance System Using Machine Learning Algorithms”, Proceedings of the second International Conference on Intelligent Computing and Control System (ICICCS 2018), IEEE Xplore Compliant Part Number:CFP18K74-ART; ISBN:978-1-5386-2842-3.