A

**Project Report**

On

**AttendEase**

Submitted during 6th semester in partial fulfilment of the requirements for the award of degree of

# Bachelor of Technology

in

# Electronics and Computer Engineering

by

**Sakshi Mittal (22001015055)**

Under supervision of

# Dr. Anu Sharma

****

**Department of Electronics and Computer Engineering**

**J.C. Bose University of Science & Technology, YMCA Faridabad – 12100**

# Table of Contents

Candidate’s Declaration ..................................................... i

Certificate ..................................................... ii

Acknowledgement ..................................................... iii

Chapter 1: Introduction

1.1 Project Overview ..................................................... 1

1.2 Motivation and Problem Statement ..................................................... 2

1.3 Technologies Used ..................................................... 3

1.4 System Highlights ..................................................... 4

Chapter 2: Implementation and Result Analysis

2.1 Problem Formulation ..................................................... 5

2.2 Objectives ..................................................... 6

2.3 Methodology ..................................................... 7

2.4 Application Scripts ..................................................... 10

2.5 Performance Evaluation ..................................................... 11

Chapter 3: Limitations and Their Solutions

3.1 Limitations ..................................................... 13

Chapter 4: Applications and Scope

4.1 Applications ..................................................... 15

4.2 Future Scope ..................................................... 16

References ..................................................... 17

# CANDIDATE’S DECLARATION

I hereby certify that the work which is being presented in this Project titled “AttendEase” in fulfilment of the requirements for the degree of Bachelor of Technology in Electronics and Computer Engineering and submitted to “J.C. Bose University of Science and Technology, YMCA, Faridabad”, is an authentic record of my own work carried out under the supervision of Dr. Anu Sharma.  
The work contained in this project has not been submitted to any other University or institute for the award of any degree or diploma by me.  
  
Student’s Name  
Sakshi Mittal (22001015055)

# CERTIFICATE

It is hereby certified that the project titled “AttendEase” submitted by Sakshi Mittal (22001015055) of 3rd Year (6th Semester) to “J.C. Bose University of Science and Technology, YMCA, Faridabad” for the award of the degree of Bachelor of Technology in Electronics and Computer Engineering is a record of a bonafide work carried out by them in the Project Workshop – J.C. Bose University of Science and Technology, YMCA, Faridabad under my supervision as mentor for May 2025 examination.  
In my opinion, the project has reached the standards of fulfilling the requirements of the regulations to the degree.  
  
Dr. Anu Sharma  
Faculty, Department of Electronics and Computer Engineering

# ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals.  
I would like to extend my sincere thanks to all of them. I am highly indebted to my mentor Dr. Anu Sharma for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.  
I would like to express my gratitude towards my parents & members of J.C. Bose University of Science and Technology for their kind cooperation and encouragement which helped in the completion of this project. I would like to express my special gratitude and thanks to our teachers as well for their guidance.  
  
Student’s Name  
Sakshi Mittal (22001015055)

# CHAPTER 1: INTRODUCTION

## 1.1 Project Overview

AttendEase is an intelligent attendance management system that harnesses the power of machine learning and computer vision to automate traditional attendance-taking processes. The system eliminates manual and biometric errors by introducing real-time facial recognition, offering a seamless, contactless, and efficient method of tracking presence in institutional or corporate environments.  
  
Traditional systems often suffer from issues such as proxy attendance, data manipulation, and delays in record maintenance. AttendEase addresses these challenges by using facial features as a unique identifier, thereby improving both security and reliability.  
  
The solution is divided into three core modules:  
1. Face Data Acquisition & Storage – Users input their name and the system collects 100 facial images using a webcam. Images are processed and stored using OpenCV and Pickle.  
2. Real-Time Face Recognition & Attendance Logging – Either a K-Nearest Neighbors (KNN) or a CNN model is used to recognize faces in live video streams. Attendance is logged with timestamp and confirmed via speech feedback.  
3. Interactive Attendance Dashboard – Built with Streamlit, this displays attendance logs in real-time using an auto-refresh mechanism.

## 1.2 Motivation and Problem Statement

In modern educational and workplace settings, the need for non-intrusive, accurate, and automated attendance systems is critical. Manual systems are tedious, time-consuming, and prone to human error, while biometric systems still require physical interaction. AttendEase offers:  
  
- Contactless recognition  
- Reduced administrative workload  
- Fool-proof attendance logging  
- User-friendly dashboard for monitoring

## 1.3 Technologies Used

|  |  |
| --- | --- |
| Category | Tools/Technologies |
| Programming Language | Python |
| Computer Vision | OpenCV |
| Machine Learning | KNN (Scikit-learn), CNN (Keras) |
| Web Interface | Streamlit |
| Face Detection Model | Haar Cascade Classifier |
| Speech Feedback | win32com.client (TTS) |
| Data Handling | NumPy, Pandas, CSV |
| Model Serialization | Pickle |
| Auto-Refresh Utility | streamlit\_autorefresh |
| Dashboard Display | Streamlit DataFrames |

## 1.4 System Highlights

- Modular Design: Easily switchable between KNN and CNN models.  
- Speech Feedback: Verbal confirmation of attendance.  
- Background Overlay: Uses a pre-designed background image for improved aesthetics during recognition.  
- Data Privacy: All facial data is locally stored in serialized files.

# CHAPTER 2: IMPLEMENTATION AND RESULT ANALYSIS

## 2.1 Problem Formulation

In both educational institutions and corporate settings, attendance management is an essential administrative task. Traditionally, this task has been executed using manual methods such as roll calls, paper registers, or biometric systems. These systems present multiple challenges:

* Time Inefficiency: Manual attendance takes up valuable time, especially in large groups.
* Error Prone: Human errors and proxy attendance can lead to inaccurate records.
* Health Concerns: Biometric systems require physical contact, posing hygiene concerns in a post-pandemic environment.
* Scalability Issues: As the number of users increases, managing and retrieving attendance data becomes more complex.

The problem, therefore, lies in the lack of an efficient, scalable, accurate, and hygienic attendance system. The goal is to formulate a solution that eliminates these inefficiencies while leveraging advancements in artificial intelligence and machine learning.

1. Problem Statement:

* To design and implement a real-time, contactless, and intelligent facial recognition-based attendance management system that automates attendance logging and provides an interactive interface for monitoring and analysis.

## 2.2 Objectives

The primary objective of the AttendEase project is to provide a smart, reliable, and automated alternative to traditional attendance systems using facial recognition technology. The project aims to meet the following objectives:

1. Primary Objectives:

* Automate the Attendance Process: Eliminate the need for manual marking by automating attendance using face recognition.
* Real-Time Recognition and Logging: Detect and recognize faces in real-time and log attendance dynamically with time-stamped entries.
* User-Friendly Interface: Provide an interactive dashboard for administrators to monitor attendance records live.
* Secure and Accurate Logging: Ensure data integrity through accurate face recognition and secure storage of attendance data.

## 2.3 Methodology

The AttendEase system is implemented through the integration of machine learning, computer vision, and web-based dashboard development. The methodology is divided into three primary modules:

### Module 1: Face Data Acquisition and Storage

* Face Detection:  
   - Utilizes OpenCV’s Haar Cascade Classifier (haarcascade\_frontalface\_default.xml) to detect faces in real-time via webcam.  
   - The webcam is accessed using cv2.VideoCapture(0).
* Data Sampling and Storage:  
   - Every 10th frame is captured to reduce data redundancy.  
   - Each face is converted to grayscale and resized to 50×50 pixels.  
   - 100 face samples are collected per user.  
   - Data is stored using Pickle in two files: faces\_data.pkl and names.pkl.

### Module 2a: KNN-Based Recognition and Attendance Logging

* Loading Trained Data:  
   - The stored .pkl files (faces\_data.pkl and names.pkl) are loaded to train a K-Nearest Neighbors (KNN) classifier.
* Model Training:  
   - A KNN classifier with k=5 is fit on flattened face vectors and labels.
* Live Recognition:  
   - Video frames are processed to detect faces with OpenCV.  
   - Each detected face is resized, flattened, and passed to the KNN model for prediction.
* Attendance Logging:  
   - Pressing the 'o' key writes the recognized name and timestamp to Attendance\_<date>.csv.  
   - Text-to-speech feedback confirms attendance capture.  
   - Pressing 'q' exits the loop.

### Module 2b: CNN-Based Recognition and Attendance Logging

To improve recognition accuracy and scalability, a Convolutional Neural Network (CNN) model was also implemented.

* Data Preparation:  
   - Face samples are reshaped to (-1, 50, 50, 3) and normalized (pixel values scaled to [0,1]).
* Label Encoding:  
   - Names labels are encoded into numeric form using scikit-learn’s LabelEncoder, and the encoder is saved as label\_encoder.pkl.
* Model Architecture:  
   - Three Conv2D layers (32, 64, 128 filters) each followed by MaxPooling2D.  
   - A Flatten layer and a Dense layer of 128 units (ReLU), ending with a softmax output matching the number of users.
* Training:  
   - Model compiled with Adam optimizer and sparse categorical cross-entropy loss.  
   - Trained for 20 epochs with a batch size of 32 and 20% validation split.  
   - Final model saved as face\_cnn\_model.h5.
* Live Recognition:  
   - The live video feed is processed frame-by-frame.  
   - Detected face regions are resized to (50,50,3), normalized, and passed through the CNN for prediction.  
   - Predicted class is mapped back to the user name via the saved LabelEncoder.
* Attendance Logging:  
   - Similar to KNN, pressing 'o' logs attendance with speech feedback, and 'q' exits.

## 2.4 Application Scripts

* app.py: Streamlit dashboard code for displaying and auto-refreshing attendance logs.
* test\_knn\_attendance.py: Real-time attendance capture using KNN.
* test\_cnn\_attendance.py: Real-time attendance capture using the trained CNN model.

## 2.5 Performance Evaluation

The AttendEase system was evaluated as a multiclass classification task. The performance of the KNN and CNN models is summarized below in terms of test accuracy and, for CNN, training and validation losses:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Test Accuracy | Training Loss | Validation Loss |
| KNN | 92.23% | N/A | N/A |
| CNN | 99.68% | 0.03 | 0.04 |

# CHAPTER 3: LIMITATIONS AND THEIR SOLUTIONS

## 3.1 Limitations

While AttendEase demonstrates the successful implementation of a real-time face recognition-based attendance system, it comes with certain limitations:

1. Manual Trigger for Attendance Logging

* Limitation:
* The user needs to press the ‘o’ key manually to log attendance after recognition, which defeats the purpose of full automation in real-world scenarios.
* Solution:
* Replace manual logging with automatic attendance logging once face recognition confidence crosses a defined threshold.

2. Limited User Interface and Reporting Features

* Limitation:
* The Streamlit dashboard, while effective for basic real-time monitoring, lacks advanced features like filtering, historical analysis, downloadable reports, and role-based access.
* Solution:
* Upgrade the dashboard using advanced frameworks like Dash, Flask + React, or Power BI integration, and add features such as:
* Date-wise filtering
* Export to Excel/PDF
* Admin vs. Viewer roles

3. No Handling of Fake Attendance

* Limitation:
* AttendEase lacks anti-spoofing measures, which means a photo or video could be used to trick the system into marking fake attendance.
* Solution:
* Implement liveness detection techniques using:
* Blink detection
* Head movement prompts
* 3D depth analysis using stereo or IR cameras

# CHAPTER 4: APPLICATIONS AND SCOPE

## Applications

1. Educational Institutions

* Schools, Colleges, and Universities can automate student attendance in classrooms, lectures, labs, and examinations.
* Reduces the workload of faculty members and minimizes human error.

1. Corporate Workspaces

* Used in offices and workplaces to track employee attendance and ensure punctuality.
* Integrated with HR and payroll systems to automate salary calculations based on attendance logs.

1. Government Offices and Public Institutions

* Streamlines attendance systems in government departments, improving accountability and transparency.
* Supports large-scale deployment in high-security environments due to its non-intrusive and accurate nature.

1. Events, Seminars, and Conferences

* Ensures secure and quick check-ins during events and large gatherings.
* Can be used for issuing digital badges and attendance certificates based on logs.

1. Healthcare Institutions

* Can be applied in hospitals and clinics to manage staff rosters without physical contact.
* Offers safe attendance tracking for both staff and visitors, especially in sensitive environments.

## 4.2 Future Scope

* Deep Learning Integration – Replace or augment the existing KNN algorithm with deep learning models such as FaceNet for improved recognition.
* Cloud-Based Architecture – Migrate the system to a cloud platform for centralized storage and real-time remote access.
* Mobile Application Development – Develop a mobile app interface for dashboard viewing, notifications, and remote attendance capture.
* Masked Face Recognition – Incorporate masked face detection techniques for environments with facial coverings.
* Multi-Camera and Multi-Angle Support – Enable multiple camera feeds to cover larger areas for scalability.
* Biometric Integration – Combine facial recognition with other biometric features (voice, fingerprint) for multi-factor authentication.

# REFERENCES

• OpenCV (Open Source Computer Vision Library): https://docs.opencv.org/ and https://learnopencv.com/

• Scikit-learn – K-Nearest Neighbors Algorithm: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

• Pickle Module in Python (Serialization): https://docs.python.org/3/library/pickle.html

• NumPy (Numerical Python): https://numpy.org/doc/

• Pandas (Data Manipulation and Analysis): https://pandas.pydata.org/docs/

• CSV Module in Python: https://docs.python.org/3/library/csv.html

• Streamlit (Web App Framework for Python): https://docs.streamlit.io/ and https://streamlit.io/

• Streamlit Autorefresh Component: https://docs.streamlit.io/library/api-reference/utilities/st.experimental\_rerun and https://pypi.org/project/streamlit-autorefresh/

• win32com.client – Text-to-Speech in Python: https://learn.microsoft.com/en-us/office/vba/api/overview/excel/win32com and https://stackoverflow.com/questions/56352430/how-to-use-win32com-client-in-python