Face Recognition Attendance System – Setup Guide

1. Overview

This application uses a webcam to recognize faces, mark attendance in a MySQL database, and manage attendance records through a graphical interface built with Tkinter.

2. Requirements

Python 3.8 – 3.11 is recommended (avoid 3.12+). Required packages: opencv-python, opencv-contrib-python, pillow, numpy, face_recognition, mysql-connector-python, pandas, openpyxl.

3. Install Required Python Packages

Run the following commands:
pip install opencv-python
pip install opencv-contrib-python
pip install pillow
pip install numpy
pip install face_recognition
pip install mysql-connector-python
pip install pandas
pip install openpyxl

Note: face_recognition requires dlib which needs CMake and a C++ compiler.

4. MySQL Database Setup

1. Install MySQL.
2. Create database and table using:
CREATE DATABASE face_recognition;
USE face_recognition;
CREATE TABLE ATTENDANCE (
id INT AUTO_INCREMENT PRIMARY KEY,
name VARCHAR(255) NOT NULL,
time DATETIME NOT NULL,
date DATE NOT NULL
);

5. Folder Structure

project_folder/
■
■■■ Images/ (face images)
■■■ face.jpg (background)
■■■ face_recognition_app.py (script)
Images folder must exist and contain at least one image.

6. Running the Application

Open terminal in project folder and run:
python face_recognition_app.py
Buttons:
- REGISTER NEW FACE (admin password required)
- MARK ATTENDANCE
- SHOW ATTENDANCE
- CLEAR ATTENDANCE

7. Features

• Face detection & recognition
• MySQL attendance storage
• Tkinter GUI
• Export to Excel
• Admin authentication

8. Hardware & Environment

• Webcam
• Good lighting
• Minimum 4GB RAM

9. Troubleshooting

ModuleNotFoundError for face_recognition → Install CMake + compiler
No face detected → Ensure good lighting
MySQL connection error → Check credentials
Slow detection → Lower resolution

10. Script Configuration

Update MySQL credentials:
mydb = sql.connect(
host='localhost',
user='root',
password='your_password',
database='face_recognition'
)