



---

# FACE RECOGNITION

---

Using ESP32 camera and ftdi module



**MONISHA A M**

MSC DFIS , II SEM

MOBILE AND NETWORK FORENSICS

SUBMITTED TO :

MR. N VISHNU VENKATESH

ASSISTANT PROFESSOR

DEPARTMENT OF FORENSIC SCIENCE

# INTRODUCTION

This project implements a face detection and recognition system using the ESP32-CAM module in combination with Python code running on a PC. The ESP32-CAM captures real-time images and streams them over Wi-Fi, while Python (with OpenCV and the face\_recognition library) processes the images to detect and recognize faces.

## INTRODUCTION TO COMPONENTS

1.ESP32-CAM Module: - An inexpensive development board featuring a OV2640 camera and a ESP32-S chip. Supports Wi-Fi, making it ideal for wireless image streaming. It can be integrated into IoT projects or function as a standalone camera server. Features GPIO pins for attaching external components, such as sensors or servos.

2.FTDI Programmer (Serial-to-USB Adapter):Because the ESP32-CAM does not have an integrated USB port, it is necessary to upload code to it. Transforms PC USB signals into ESP32 UART (serial) signals. Uses jumper wires to connect to the ESP32-CAM.

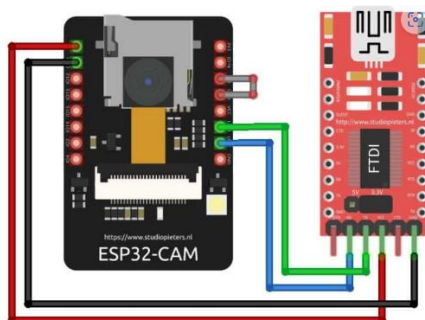
3. Jumper wires from one female to another: It is used to establish short-term, interchangeable connections between:ESP32-CAM and FTDI Programmer. External devices such as PIR sensors or servos are optional.

4.Arduino IDE: A platform for writing and uploading firmware to microcontrollers such as the ESP32-CAM.The CameraWebServer example, which enables live camera streaming via a local IP address, is uploaded here.

5.Visual Studio Code (VS Code): - A contemporary code editor that allows you to run Python scripts on your computer. Images streamed from the ESP32-CAM are used to perform face detection and recognition using Python.

6.Python and Libraries - Python: Manages facial recognition, image processing, and attendance recording. OpenCV (cv2): For displaying and manipulating images. face\_recognition: A robust library that uses deep learning to identify and detect faces.

## ESP32-CAM FTDI CONNECTION



ESP32-CAM	FTDI Programmer
GND	GND
5v	vcc(5v)
UoR	TX
UoT	RX
GPIO o	GND

## WORKING PRINCIPLE

An image is captured by the ESP32-CAM and streamed to a local IP address. Using the IP, a Python script retrieves and processes the image. Face\_recognition is used to detect faces. Preloaded trained images are compared to recognized faces. The system shows the name on the live feed if a match is discovered. A real-time OpenCV window displaying the processed image is shown.

## BENEFITS

No physical contact is required for contactless operation. Real-time monitoring features include instant facial recognition and a live camera feed. Low Cost: Open-source software and reasonably priced parts. Expandable: Servo motors for door locking and unlocking, Telegram bot for downloading the images or notifying intruders, and other features can be added. Educational Value: Excellent for teaching like attendance systems integration, and the Internet of Things.

## APPLICATIONS

- School/college attendance systems (via live recognition)
- Office and workplace access control
- Smart home automation
- Security surveillance
- Visitor identification systems

## CONCLUSION

For real-time identification, the Face Detection and Recognition system with ESP32-CAM and Python is a useful and effective solution. It combines the capabilities of artificial intelligence and the Internet of Things to produce a working prototype that can be developed into a comprehensive smart security or access control system.

## OUTPUTS/SNAPSHOTS

