Title :-

"Intelligent Facial Recognition IoT Access Management System"

By :

Kewal Nanavati-22070122099

Manan Bhimjiyani-22070122107

Megha Beria-22070122117

Supervised by :

Chandan Chaubey

(Faculty of sensors and microcontrollers)

Abstract:

The presented project embodies a multifaceted security system that harmonizes computer vision, Internet of Things (IoT), and instant messaging platforms to create an intelligent, interactive, and responsive security solution. This system capitalizes on a fusion of technologies to offer advanced access control.

At its core, the system utilizes a camera and OpenCV's robust facial recognition capabilities, enabling it to identify and detect human faces in real-time. Once a face is detected within the camera's frame, the system captures an image of the individual and initiates a two-way communication process. This image, along with an access request, is seamlessly transmitted to a preconfigured Telegram chat for immediate user interaction. The user is prompted to respond with a simple "yes" or "no" through the Telegram chat interface, effectively granting or denying access to the secured area.

The Python script orchestrates the processing of these responses and subsequently communicates with an Arduino microcontroller through a serial connection. The Arduino is equipped to act as the physical access controller, making it capable of operating various mechanisms such as electronic door locks or barriers. When a positive response ("yes") is received from the user via Telegram, the Python script sends an "O" command to the Arduino, triggering the opening of the door. Conversely, a negative response ("no") results in the transmission of a "C" command, instructing the Arduino to keep the access point secured.

This innovative amalgamation of computer vision, real-time communication, and IoT technology creates a dynamic and user-friendly security system. The approach offers real-time visual feedback by capturing and transmitting facial images, coupled with user authorization via messaging. This paradigm shift in security system design enhances both the convenience and responsiveness of access control, catering to a broad spectrum of applications ranging from home security to industrial access management. In a world increasingly focused on smart and interconnected solutions, this project represents a testament to the potential of marrying diverse technologies to address real-world security challenges.

Introduction

Introduction:

In an era characterized by a growing need for both security and convenience, innovative solutions at the intersection of computer vision, IoT, and instant messaging have taken center stage. This project seeks to address the challenges associated with traditional access control systems by introducing a multifaceted, intelligent, and interactive security solution.

A key component of contemporary security measures, access control is essential in a variety of contexts, including commercial, industrial, and residential ones. Even if they work, traditional approaches like keycards and passcodes don't provide the responsiveness and involvement that modern needs need. Traditional access systems frequently require human verification, which is inconvenient and causes delays. Given these constraints, we have developed a sophisticated method that leverages instant messaging and computer vision to transform access management.

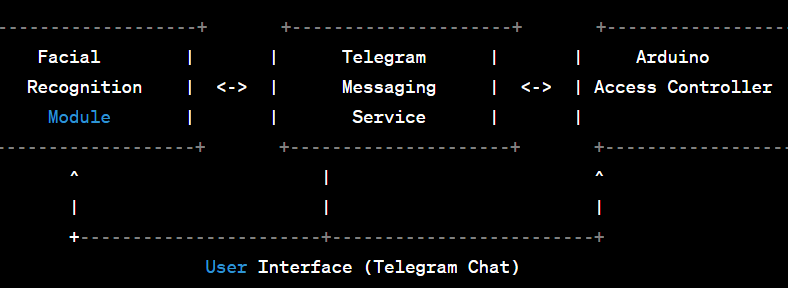
The core problem we aim to address is the need for a more agile, user-friendly, and responsive access control system. To this end, we propose an integrated system that employs facial recognition technology to detect and identify individuals seeking access. This system captures and transmits facial images to a designated user through a popular messaging platform, Telegram. The user is then empowered to grant or deny access in real-time through simple "yes" or "no" responses. This innovative approach eliminates the need for physical keys, cards, or complex verification processes, thereby enhancing convenience.

The innovative value of this project lies in the fusion of technologies to create an integrated security solution that is not only responsive but also user-centric. By integrating computer vision, IoT, and messaging, this project represents a leap forward in the evolution of access control. The new-found synergy between these technologies not only improves user experience but also ensures that access control is more intelligent, efficient, and adaptable to contemporary needs. In the following sections, we will delve into the technical aspects of the system, exploring its development, components, and the underlying principles that make it an exemplar of innovative security solutions.

Methodology:

Block Diagram:

The project's methodology involves the integration of hardware components, software, and messaging platforms to create an intelligent access control system. Below is a block diagram illustrating the key components and their interactions in the system:



Components Diagram:

1. Facial Recognition Module:

- Utilizes OpenCV and a pre-trained Haar Cascade classifier for real-time face detection.

- Captures facial images when a face is detected and saves them for communication.

2. Telegram Messaging Service:

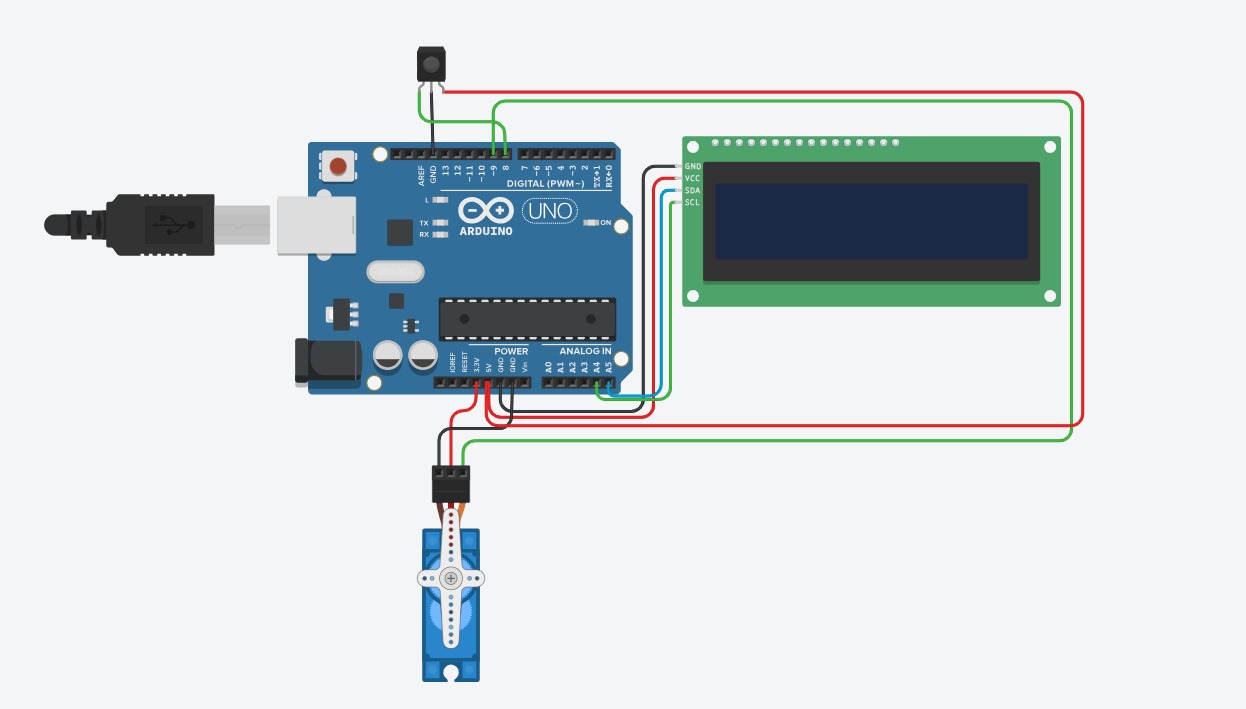
- Utilizes the Telepot library for Python to interact with the Telegram Bot API.

- Sends and receives messages and images to and from the user via a Telegram chat.

3. Arduino Access Controller:

- Connects to the computer via a USB serial connection (e.g., COM7).

- Receives commands from the Python script to control access mechanisms.



Circuit Diagram:

The circuit for this project involves a camera, computer, and Arduino:

1. Camera: Captures video frames for facial recognition.

2. Facial Recognition Software: Detects faces using OpenCV and a Haar Cascade classifier.

3. Computer: Runs the Python script, manages facial recognition, and communicates with Telegram.

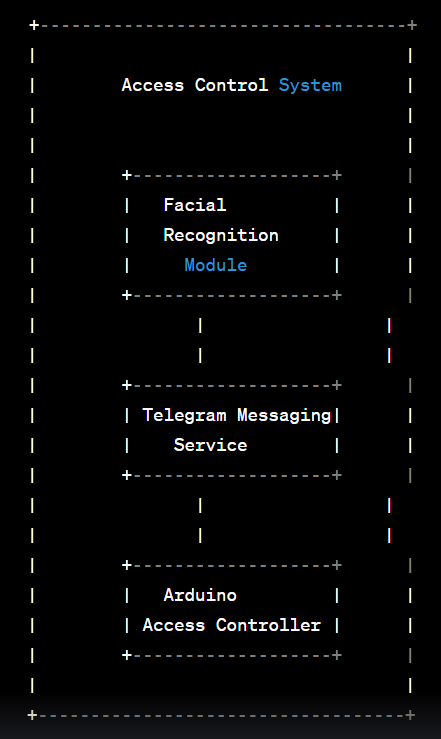
4. Telegram Messaging Service: Sends access requests and receives user responses.

5. Arduino: Connects to the computer via USB serial and controls the physical access mechanism.

6. Access Mechanism: The physical door lock or barrier controlled by the Arduino.

This system combines computer vision, messaging, and IoT technologies to create a responsive access control system.

Final Project Diagram:



The facial recognition module captures facial images and sends them to the Telegram messaging service. The messaging service interacts with the user via a Telegram chat, providing access requests and receiving "yes" or "no" responses. Based on the user's response, the Python script sends commands to the Arduino access controller, which controls the physical access mechanism, such as a door lock.

This integrated approach combines computer vision, messaging, and IoT technologies to create a responsive and user-centric access control system, offering improved security and convenience.

Code:

Arduino :

#include <Servo.h>

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

const int iPin = 8;

Servo servo;

LiquidCrystal\_I2C lcd(0x27, 16, 2);

void setup() {

  servo.attach(9);

  pinMode(iPin,INPUT);

  lcd.init();

  lcd.backlight();

  Serial.begin(9600);

}

void loop() {

  if (Serial.available() > 0) {

    char command = Serial.read();

    if (command == 'O') {

      // Open the door (adjust servo position)

      servo.write(90); // You may need to adjust this angle

      lcd.clear();

      lcd.setCursor(0, 0);

      lcd.print("Access Granted");

      delay(3000);

      int ir\_out;

  ir\_out=digitalRead(iPin);

      if (ir\_out==HIGH)

  {

    lcd.clear();

      lcd.setCursor(0, 0);

      lcd.print("Door closing");

          lcd.clear();

      lcd.setCursor(0, 0);

      lcd.print("Thanks");

  }

  delay(5000);

    } else if (command == 'C') {

      // Close the door (reset servo position)

      servo.write(0);

      lcd.clear();

      lcd.setCursor(0, 0);

      lcd.print("Access Denied");

      delay(5000);

      lcd.clear();

      lcd.setCursor(0, 0);

      lcd.print("Thanks");

    }

  }

}

Python:

import cv2

import numpy as np

import telepot

import time

import serial # Import the serial library

# Telegram Bot Token and Chat ID (Replace with your bot token and chat ID)

BOT\_TOKEN = '6729207909:AAF99rowjcsw697wluLnK-QXdq5FGjS\_7Ro'

chat\_id = '1923768228'

CHAT\_ID = '1923768228'

# Initialize the Telegram bot

bot = telepot.Bot(BOT\_TOKEN)

def handle(msg):

chat\_id = msg['chat']['id']

content\_type, chat\_type, chat\_id = telepot.glance(msg)

if content\_type == 'text':

message = msg['text']

if message.lower() == 'yes':

# Send a message to Arduino for access granted

ser.write(b'O') # Send 'O' to Arduino to open the door

send\_telegram\_message('face\_detected.jpg', 'Access granted.')

elif message.lower() == 'no':

# Send a message to Arduino for access denied

ser.write(b'C') # Send 'C' to Arduino to keep the door closed

send\_telegram\_message('face\_detected.jpg', 'Access denied.')

# Function to send a message with a photo to Telegram

def send\_telegram\_message(image\_path, message):

with open(image\_path, 'rb') as photo:

bot.sendPhoto(CHAT\_ID, photo, caption=message)

# Initialize the face detection model (you may need to install OpenCV and face detection model)

face\_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_frontalface\_default.xml')

# Open the camera (you may need to change the camera index)

cap = cv2.VideoCapture(0)

# Initialize the serial connection to the Arduino (change the port name)

ser = serial.Serial('COM7', 9600) # Replace 'COM3' with your Arduino's port

while True:

ret, frame = cap.read()

if not ret:

continue

# Convert the frame to grayscale for face detection

gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

# Detect faces

faces = face\_cascade.detectMultiScale(gray, scaleFactor=1.3, minNeighbors=5)

if len(faces) > 0:

# Face detected

cv2.imwrite('face\_detected.jpg', frame) # Save the image with the face

send\_telegram\_message('face\_detected.jpg', 'Face detected. Access request received.')

break

# Prompt for access approval

'''access\_approval = input('Access granted? (yes/no): ')

if access\_approval.lower() == 'yes':

# Send a message to Arduino for access granted

ser.write(b'O') # Send 'O' to Arduino to open the door

else:

# Send a message to Arduino for access denied

ser.write(b'C') # Send 'C' to Arduino to keep the door closed'''

# Display the frame with rectangles around faces (optional)

for (x, y, w, h) in faces:

cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)

cv2.imshow('Face Detection', frame)

# Press 'q' to exit

if cv2.waitKey(1) & 0xFF == ord('q'):

break

bot.message\_loop(handle)

cap.release()

cv2.destroyAllWindows()

Conclusion:

The project's methodology is in perfect alignment with the goals outlined in the introduction. A user-centered and responsive access management system has been effectively implemented through the integration of instant messaging, facial recognition, and Internet of Things technologies.

The project overcomes the crucial issue of slow and cumbersome traditional access control methods, as was hinted to in the introduction. The clever method, which depends on facial recognition, gives the security system a flexible element. The technology bridges the gap between security and convenience by allowing users to participate in real-time access authorization through Telegram messaging.

The introduction's concept of this harmonic confluence of technology results in an approach that is revolutionary. The system's installation validates this project's creative worth. In addition to being a thoughtful and adaptable solution, it also meets the needs of the modern, globally interconnected world.

To sum up, the project's technique yields a concrete outcome that is consistent with the main goal stated in the introduction. Facial recognition, instant messaging, and the Internet of Things drive an inventive approach to access control that redefines security paradigms and provides a useful, user-friendly security solution.