Intelligent Access Control System For Safety Critical Areas In Industries

Abstract:

This model uses smart IOT technology ,it is developed to ensure the safety of workers near the construction sites.

At construction sites helmets play a very important role for the safety of the workers.Presence of a person to check whether all the workers are wearing helmet is quite difficult.Hence by using this smart access system it makes the work less complicated. This is done by using video and image capturing process. Access is denied to the workers who do not meet the requirements. It is always said that *it is better to be safe than sorry.*

Problem Statement :

Intelligent Access Control System For Safety Critical Areas In Industries

Project Working :

Entry of every worker is restricted if they don't meet the specified requirements. Using a Smart Access System we can monitor workers safety. Using the webcam or any other wireless camera, video is recorded at the entrance. The system recognises the worker and detects if the worker is wearing shoes and helmet or not. Only the ones wearing helmet and shoes are allowed to enter the site whereas the access is denied for the rest.

Basic steps include:

- Live video streaming at the entrance.
- Recognises the face of the worker and captures the image.
- Detects the presence of a helmet and shoes.
  - If the result is positive; the door is automatically unlocked using a smart lock.
  - If the result is negative; the access is denied to that worker.

Components:

Hardware Components:

1) ServoMotor-SG90
2) Node MCU- ESP8266
3) Camera - hp webCam

Software Components:

1) Python idle 3.7.3
2) Clarifai

Working Principle

Servo motor:

A **servo motor** is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through **servo mechanism**. If motor is used is DC powered then it is called DC servo motor, and if it is AC powered motor then it is called AC servo motor. We can get a very high torque servo motor in a small and lightweight packages. Doe to these features they are being used in many applications like toy cars , RC helicopters and planes, Robotics, Machine etc.

NodeMCU :

The **ESP8266 can** be controlled from your local Wi-Fi network or from the internet (after port forwarding). The ESP-01 module has GPIO pins that **can** be programmed to turn an LED or a relay ON/OFF through the internet. The module **can** be programmed using an Arduino/USB-to-TTL converter through the serial pins (RX,TX).

CODE:

Code to capture worker's image and detect the safety requirements(helmet and shoes)

```
import pyttsx3
import requests
import numpy as np
import cv2
import time
from datetime import datetime

from clarifai.rest import ClarifaiApp
from clarifai.rest import Image as ClImage
face_cascade = cv2.CascadeClassifier('haar-face.xml')
app = ClarifaiApp(api_key='e04e2dab8dac410aa2dd0e4fa01136b5')
cap = cv2.VideoCapture(0)
print ('camera is initialized')
y=0
x=0
engine = pyttsx3.init()
data1 = " "
while True:
     ret, img = cap.read()
     if ret:
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

        faces = face_cascade.detectMultiScale(gray, 1.3, 5)

        for (x,y,w,h) in faces:
            cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
            roi_gray = gray[y:y+h, x:x+w]
            roi_color = img[y:y+h, x:x+w]
            picname = datetime.now().strftime("%y-%m-%d-%H-%M")
            picname = picname+'.jpg'
            cv2.imwrite(picname,img)
            print ("Saving Photo")
            model = app.models.get('General')
            pic2='C:/Python/Python37-32/'
            pic=pic2+picname
            #pic='C:\Python\Python37-32\man.jpg'
            image = ClImage(file_obj=open(pic, 'rb'))
            response=model.predict([image])
            data1 = response['outputs'][0]['data']['concepts']
            print ("Updated String :- ",data1)
            for row in data1:
                 if row['name'] == 'helmet':
                      if row['value']>= 0.9993858933448792:
                         x=1
                 if row['name'] == 'talent':
                      if row['value']>=  0.6682344079017639:
                         x=1

            #print ("Updated String :- ",data1)
                 if row['name'] == 'shoe':
                      if row['value']>= 0.9948883056640625:
                         y=1
     if x == 0:
          engine.say(' please wear your helmet and then enter.')
          engine.runAndWait()
     if y == 0:
          engine.say(' please wear your shoes and then enter.')
          engine.runAndWait()
     if x==1 & y==1:
          engine.say('you can enter inside.')
          engine.runAndWait()
          print ("Updated String :- ",data1)
                                                                url       =
'https://sirnamengji.eu-gb.mybluemix.net/command?message=dooropen'
          response = requests.get(url)
          print(response.status_code)
          print(response.text)

     cv2.imshow('img',img)
     time.sleep(0.1)
     k = cv2.waitKey(30) & 0xff
     if k == 27:
        break

cap.release()
cv2.destroyAllWindows()
```

Code to detect whether the given image has met the requirements:

```
import pyttsx3
import requests
import numpy as np
import cv2
import time
```

```
from datetime import datetime
engine = pyttsx3.init()
from clarifai.rest import ClarifaiApp
from clarifai.rest import Image as ClImage

app = ClarifaiApp(api_key='e04e2dab8dac410aa2dd0e4fa01136b5')

while True:
            print ("Saving Photo")
            model = app.models.get('General')
            pic='C:\Python\Python37-32\man.jpg'
            image = ClImage(file_obj=open(pic, 'rb'))
            response=model.predict([image])
            data1 = response['outputs'][0]['data']['concepts']
            for row in data1:
                if row['name'] == 'helmet':
                    if row['value']>= 0.9993858933448792:
                        x=1
                    else:
                        print ('please wear the helmet')
                        engine.say(' please wear your helmet.')
                        engine.runAndWait()
             #print ("Updated String :- ",data1)
                if row['name'] == 'shoe':
                    if row['value']>= 0.9948883056640625:
                        x=1
                    else:
                        print ('please wear the shoe')
                        engine.say(' please wear your shoes.')
                        engine.runAndWait()

                else:
                    engine.say('you can enter inside.')
                    engine.runAndWait()
                    print ("Updated String :- ",data1)
                                                                   url   =
'https://sirnamengji.eu-gb.mybluemix.net/command?message=dooropen'
                    response = requests.get(url)
                    print(response.status_code)
                    print(response.text)



            print ("Updated String :- ",data1)
```

Program for the servo motor to rotate :

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <Servo.h>
const char* ssid = "Swathi Mengji";
const char* password = "asdsp0706";

#define ORG "gfzlqp"
#define DEVICE_TYPE "GNITS"
#define DEVICE_ID "1006"
#define TOKEN "987654321"
String command;
Servo myservo;
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char topic[] = "iot-2/cmd/home/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
//Serial.println(clientID);
void callback(char* topic, byte* payload, unsigned int payloadLength);

WiFiClient wifiClient;
PubSubClient client(server, 1883, callback, wifiClient);
void setup()
{
 Serial.begin(115200);
 Serial.println();
 myservo.attach(D2);
 pinMode(D1,OUTPUT);
 wifiConnect();
 mqttConnect();
}

void loop()
{
 if (!client.loop())
 {
  mqttConnect();
 }
delay(100);
}

void wifiConnect()
{
 Serial.print("Connecting to "); Serial.print(ssid);
 WiFi.begin(ssid, password);
 while (WiFi.status() != WL_CONNECTED)
 {
  delay(500);
  Serial.print(".");
 }
 Serial.print("nWiFi connected, IP address: "); Serial.println(WiFi.localIP());
}

void mqttConnect()
{
 if (!client.connected())
 {
  Serial.print("Reconnecting MQTT client to "); Serial.println(server);
  while (!client.connect(clientId, authMethod, token))
  {
   Serial.print(".");
   delay(500);
  }
  initManagedDevice();
  Serial.println();
 }
}
void initManagedDevice()
{
 if (client.subscribe(topic))
 {
  Serial.println("subscribe to cmd OK");
 }
 else
 {
  Serial.println("subscribe to cmd FAILED");
 }
}

void callback(char* topic, byte* payload, unsigned int payloadLength)
{
 Serial.print("callback invoked for topic: "); Serial.println(topic);

 for (int i = 0; i < payloadLength; i++)
 {
  //Serial.println((char)payload[i]);
  command += (char)payload[i];
 }
Serial.println(command);
if(command == "dooropen")//command given in node red
 {
 //digitalWrite(D1,HIGH);
 myservo.write(135);
 delay(2000);
 myservo.write(0);
 Serial.println("door is open");
 }
else if(command == "doorclose")
{
 //digitalWrite(D1,LOW);
 myservo.write(0);
 Serial.println("door is closed");
 }
command ="";// after giving command we are emptying it/ erasing the memory basically
}
```

Installations:

1) Python 3.7.3
2) Pip
3) Image
4) opencv
5) pyttsx3
6) requests
7) numpy as np
8) PIL
9) cv2

Team SQUADRON
Kruthika kanduri

Gandra Rashmi
Swathi Mengji
Sirnam Shruthika