Attendencer

(A Face-Recognition Attendance App)

(by Venus Agrawal)

Contents

- Introduction
- Concept Used_
- Procedure Of Application
- Coding Skills Used
- Where Are We Now/ What more to Accomplish
- Problems and Proposed Solutions
- Financial Outlook
- Conclusion

Introduction

Attendencer utilizes the concept of computer vision and creates a graphical interface unit to interact with its users and help them to mark their presence in a particular event.

I have created an app that can be utilized by modern institutions to keep track of their attendees without wasting any time on managing records.

This App can maintain , save and handle all the records regarding the attendance of any attendee on its own.

Concept Used

Computer Vision is a cutting edge field of computer science that aims to enable computers to understand what is being seen in an image.

Concept of Histogram of Oriented Gradients — or just HOG for short has been used.

Basic Pipeline of face-recognition includes:

1.Find faces in an image (Using HOG Algorithm)

2. Analyze facial features (Main Landmarks)

3. Compare against known faces(Deep Learning Neural Networks with 128 features)

4. Make a prediction(using classification machine learning algorithms)

HOG(Histogram Of Oriented Gradients)

Steps:

1. Convert images from coloured to black and white(in order to reduce the image storing and algorithm training time)

2.Look at each and every pixel in the image one by one.

3. draw arrows showing in which direction image is getting darker(comparing with surrounding pixels)

4. By only considering the direction in which the brightness changes(gradient), both very dark images and very bright images end up with the same exact representation.

1.Concept of **Affine Transformations** will be used to create different variations of a single image.

2.As We know only a few images will not help our pretrained model to predict effectively so we create different instances of the safe image.

3.This can be performed by scaling the image,changing its resolution,rotating the image to a certain angle etc.

4. Also image distortions can be created which will help the classifier train well.

Procedure Of Application

Step 1: Using the Histogram Of Oriented Gradients to **detect faces** in the image.

Step 2: **Posing And Projecting Faces**(This step includes creating multiple instances of the same input image to feed the classifier with more data.It includes scaling,rotating,and adding distortions to the root image)

Step 3: **Encoding Faces**: This section includes converting our image into numerical data which is used to distinguish mutiple faces from each other if our input image contains more than one face. This basically helps us in detecting mutiples faces in the same image and also classifying it.

Step 4: **Finding the persons name from the encoding**. To perform this we will just find the image in our database that has the closest distance with the image that we get from the webcam.

Coding Skills Used

I have used very basic programming language python to implement this project.

The deep learning face-recognitiong part includes the usage of the following libraries:

1. numpy
2. cmake
3. dlib
4. face-recognition
5. opencv-python
6. tensorflow

The GUI(graphical interface Unit) part is made by using the PyQt5 library

Where Are We Now/ What more to Accomplish

1. Currently I have been able to create a graphical interface unit that can take input of an image in case the user is new to our application.
   And then take the user to the main application where he can mark his attendance with the following features
   1. Date of Attendance
   2. Time at which he entered the attendance portal
   3. Time at which he leaves the attendance portal
   4. Total time elapsed at the portal.

2. **My future goals regarding this projects includes**:
   1. Increasing the accuracy of my face detection algorithm by changing or manipulating the tuning parameters.
   2. Provide more features to the users.
   3. Work on improving the existing features of the application.
   4. Also increasing the FPS(frames per second) of the video that was being displayed in my application.

Problems and Proposed Solutions

Problem 1: Slow Processing Time:
Solution: In order to improve the processing time of my application. I will host the application on a cloud server that has good specifications

Problem 2: Difficult to use
Solution: For now this application is a bit tricky to use but I will compile it further to make it easy to use.

Problem 3: Less Features
Solution: I have many more features in my arsenal to introduce in this project but in order to apply them I m learning more attributes of the pyqt5 library to implement them.

There are many more problems that i can think off. I would like to receive feedback from your side so that I can further improve this project.

Financial Outlook

1. A project is categorized as good only when it is cost efficient and practically applicable.

2. My project needs only a good cloud server to run on as it is using a heavy face-recognition algorithm.

3. Also when the Number of attendees become greater than 54, I have seen a decrement in the processing speed, So I need more cloud storage to store the data of new attendees.

Total Cost Per Month For Hosting This application will be 20 dollars/month

Conclusion

This project is just a small part of a more extended idea that I have in my mind. Attendance is not that big of an issue in our society but If used efficiently it will surely benefit our institutions.

Practically speaking if this application is applied to present day institutes than a lot of working effort will be saved and attedance accuracy will be improved.

Please provide feedback to this project so that I can make it better and more efficient in future.

Thank You.