Attendance application

CMPN-446: Image Processing And Computer Vision

10/01/2022

—

Names:

Aya Mahmoud Abdelfattah	1170049
Aya Magdy Essam	1170018
Monica Ehab	1170038
Ranime Hossam Eldine	1170039

Idea

Remote Attendance System for corporates using facial recognition.

This is a better replacement for fingerprint-based systems since it requires touching a screen , hence being more hygienic especially since the outbreak of COVID-19.
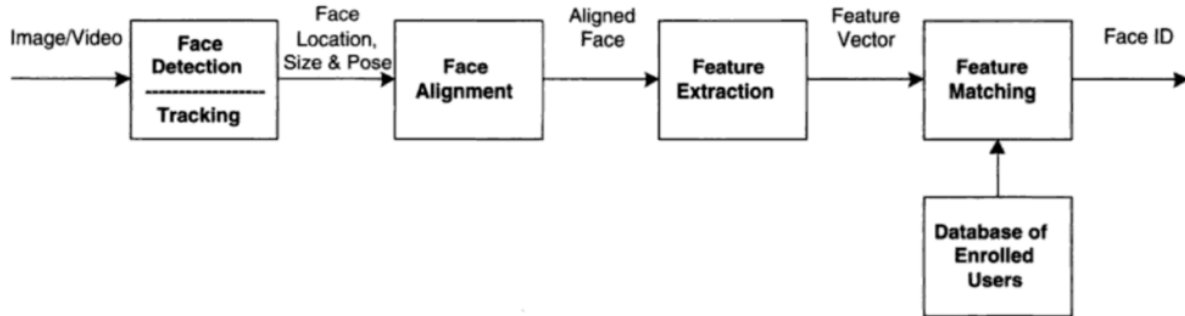
Modules/Block Diagram:
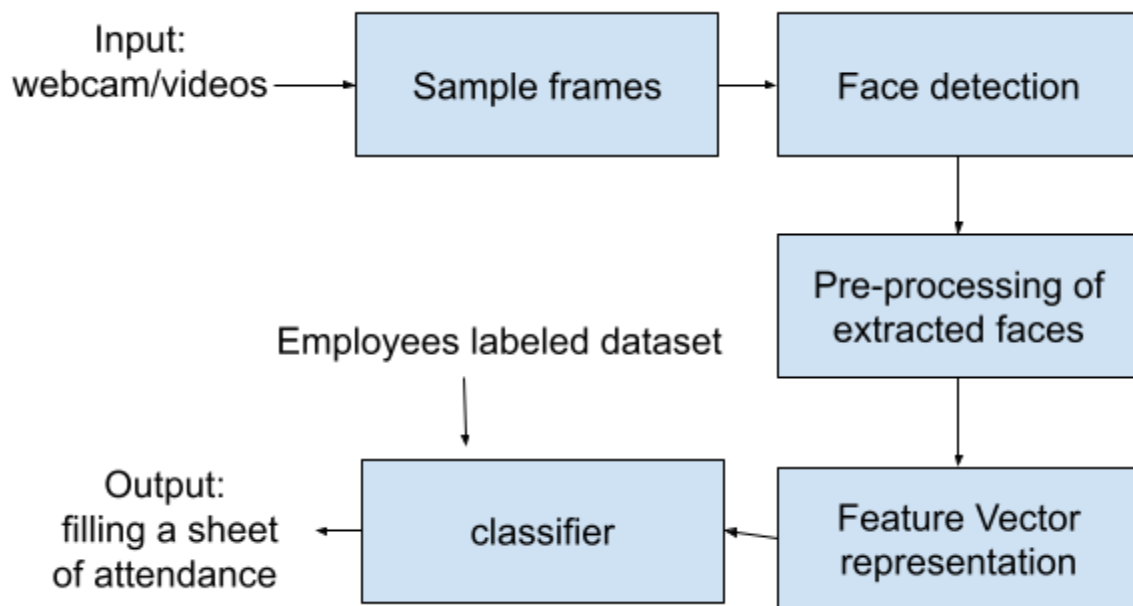
General pipeline

Fig. 1.2. Face recognition processing flow.

Blocks Description:

We will choose the more applicable approach during implementation.

We have ordered approaches in terms of priority.

1. Face Detection:

Input:

Whole image

Output:

Cropped image.

Approach:

Hog with SVM (sliding window)

2-Feature Extraction:

Input:

Aligned image.

Output:

Feature vector.

Approach:

Using PCA to extract eigen faces, calculate weights of eigen faces and obtain feature vectors to be used as input to classifier

3-Classification:

Input:

Feature vector.

Output:

Correct person from database.

Approach:

SVM (Support vector machine)

Libraries:

joblib==1.1.0
matplotlib==3.5.1
numpy==1.22.0
opencv_python==4.5.3.56
Pillow==9.0.0
PyQt5==5.15.6
scikit_image==0.19.1
scikit_learn==1.0.2
scipy==1.7.3
skimage==0.0

Datasets:

For face detection, data was generated from the sklearn package.

For face recognition, LFW Face Database : Main was used.

Experiment results and analysis

We tried experimenting with our program by just running it . Most times it will recognize the face right, we tried also experimenting with showing the program an image on the phone. But there are some cases when it recognizes the face wrongly. The program is somewhat fast. It takes on average 1.5 second to run the pipeline.The cases of wrong recognition mostly happen when the person is not looking directly to the camera.
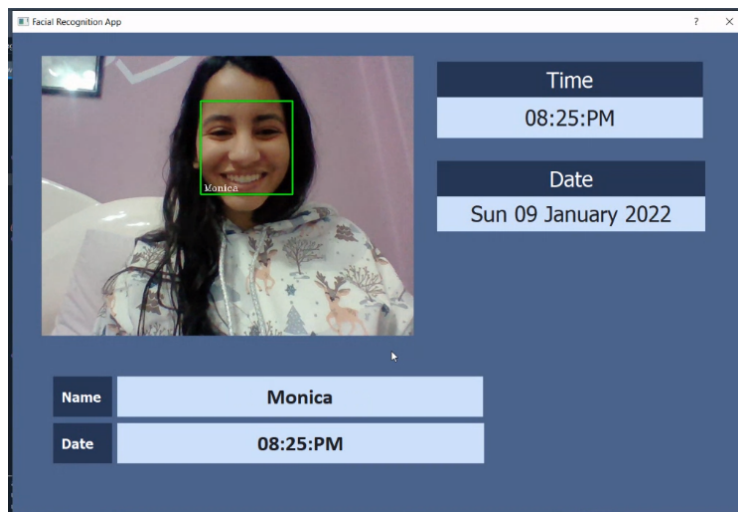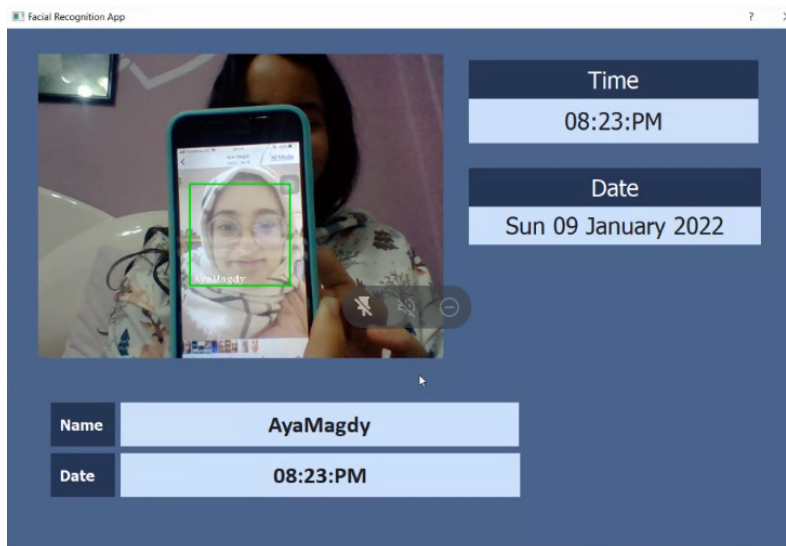

Correct recognition
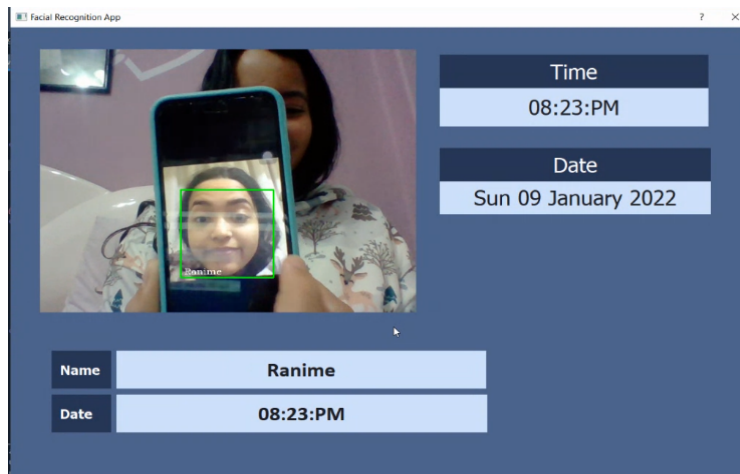
Correct recognition
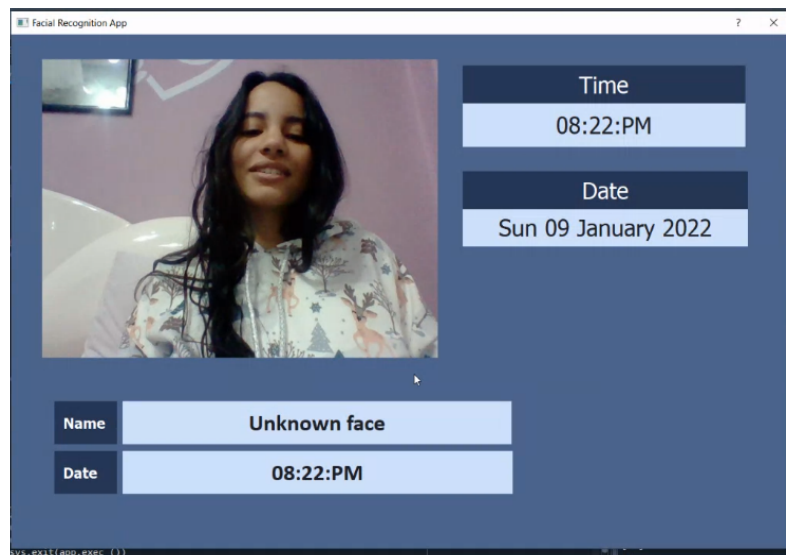
Wrong Recognition

Correct recognition

Work division between team members.

Face Detection Module
Aya Mahmoud   1170049
Aya Magdy   1170018

Face Recognition Module
Ranime Hossam   1170038
Monica Ehab   1170039

GUI and Integration:
All team members

References:

https://jakevdp.github.io/PythonDataScienceHandbook/05.14-image-features.html
A Jupyter Notebook that implements in Python 3 the Eigenfaces algorithm for face recognition
For the GUI implementation:
▶ Face Recognition Attendance GUI PyQt - 1 Hour Course | OpenCV Python | …

Related research papers:

Shih, F. Y., Cheng, S., Chuang, C. F., & Wang, P. S. (2008). Extracting faces and facial features from color images. International Journal of Pattern Recognition and Artificial Intelligence, 22(03), 515-534. Retrieved from:

https://www.cin.ufpe.br/~rps/Artigos/EXTRACTING%20FACES%20AND%20FACIAL%20FEATU RES%20FROM%20COLOR%20IMAGES.pdf

https://www.researchgate.net/profile/Vinh-Truong-Hoang/publication/335195714_Feature_ fusion_by_using_LBP_HOG_GIST_descriptors_and_Canonical_Correlation_Analysis_for_face_ recognition/links/5d9bf2aaa6fdccfd0e811077/Feature-fusion-by-using-LBP-HOG-GIST-descr iptors-and-Canonical-Correlation-Analysis-for-face-recognition.pdf