IT204: ATTENDANCE SYSTEM USING FACE RECOGNITION

CHARAN R - 191IT212
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: charankavitha1978@gmail.com

V N KARTHIK - 191IT154
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: v.n.karthik7@gmail.com

RAKESH KALLIMANI - 191IT244
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email: rakeshkallimani123@gmail.com

RAHUL NAYAKA - 191IT243
Information Technology
National Institute of Technology Karnataka
Surathkal, India 575025
Email:rahulnayaka16@gmail.com

***Abstract*—Taking attendance in a classroom, especially in large college classrooms, is a tedious, time-consuming task that is generally error-prone and inconvenient during the class. The Face Recognition Attendance System presents a merger of the novel technology of 'Facial Recognition' to help alleviate the overhead of this mundane task. This face recognition implementation aims to complete the entire task of taking attendance and updating the attendance in an excel sheet using a webcam. The Face Recognition Attendance System uses Python's programming language using the OpenCV package for the main image processing task.**

I. Introduction

With the furtherance of new technologies each and every day, humanity is slowly going towards wireless technology. It is quite discernible that the future ahead of us will become so much advance that maybe 90 percent of things that we are doing right now will be either automated or become contact less. One such progress is the facial recognition technology.Traditional method of marking attendance is a monotonous task in many schools and colleges. It is also an additional burden to faculties who mark attendance by manually calling the names of students which might take about 10 minutes of entire session. This is time consuming. There are some chances of proxy attendance. Many institutions started deploying many other techniques for recording attendance like use of iris recognition, fingerprint recognition, Radio Frequency Identification (RFID), and so on. However, these systems consume more time and are obstrusive in nature. Face recognition is an important biometric feature, which can be easily acquirable and is non-intrusive.The task is quite difficult as the real time background subtraction in an image is still a challenge Face recognition based systems are relatively heedless to various facial expression. Face recognition system embodies of two categories: verification and face identification.This project is being carried out due to the concerns that have been highlighted on the methods which professors use to take attendance during lectures. Manually writing down names on a sheet of paper as a method to track student attendance has effectuated this project to be carried out. In this project we tried to develop a prototype that will capture attendance by face detection and recognition of students faces in a digital image taken by a webcam.

In This system we tried to build an effective class attendance system using face recognition techniques. The proposed system will be able to mark the attendance via webcam. It will detect faces and then its made to recognize the faces. After recognition, it will update the attendance in the excel sheet.

II. Literature Survey

Over the many years, many researchers have proposed different face recognition techniques, motivated by the increased number of real world applications requiring the recognition of human faces. There are several problems that make automatic face recognition a very difficult task. The importance of automatic face recognition must be cope with numerous variations of images of the same face due to changes in the various parameters such as pose,illumination,facial expression,motion,glasses,facial hair and background

Since the 1970s there was many techniques and algorithms developed for machine to learn to recognize known faces. Most of the recent techniques involve at least three steps: • Face detection: • Face preprocessing: • Face recognition.
There are a few face detection methods that the previous researchers have worked on. However, most of them used frontal upright facial images which consist of only one face. The face region is fully exposed without obstacles and free from the spectacles.

Viola-Jones algorithm is suggested for face detection for student attendance system. most concluded that out of methods such as face geometry- based methods, Feature Invariant methods and Machine learning based methods, Viola-Jones algorithm is not only fast and robust, but gives high detection rate and perform better in different lighting condition.Viola-Jones algorithm gives better performance in different lighting condition. The limitations of viola jones algorithm is that it takes long time to train images, limited head pose, difficult to detect in dark environment.
BUT in this project we are using a method invented in 2005, technique called HOG, history of oriented gradients. we are implementing the paper by dalal and trigs.

1. Face Detection E-attendance System. International Journal of Computer trends and technology (IJCTT) – Volume 27 No.3 – Sept, 2015.An application which helps in capturing the images of the students in a class and mark their attendance according to their presence. This is a time consuming system and also there possibility of faulty attendance

2. Face Detection and Recognition for Automatic Attendance System. International Journal of Computer Science and Mobile Computing - April, 2006.Viola Detector and Haar – cascade extracts features quickly and it is easy to use. The compilation time of viola detector is a little time-taking. This system uses only haar cascade. We have used Local Binary Patterns with Histograms of Oriented Gradients for better accuracy and also compilation time is greatly reduced.

3. Attendance Monitoring System using Face Recognition – International Journal of Emerging Research in Management and Technology Volume 5, Issue 5 – May 2016. In this paper, the problem was discussed by applying two algorithms such as Viola Jones for Face Detection and Principal Component Analysis for Face Recognition. In our system, LBP algorithm is combined with Haar – cascade which is used for detection purpose and when the LBP algorithm is applied it creates a Histogram of that image which gives minute details of each face so that it can recognize quickly and give results in less time with more accuracy.

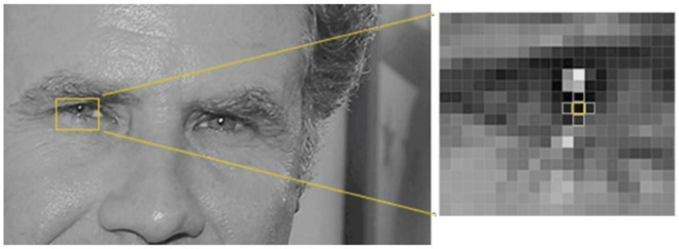
III. Problem Statement

Generally,Attendance system has been taken manually which cause time waste, paper word besides it is not accurate as there is a chance of proxy.Face recognition technology can be utilized to build an automated attendance system that makes counting and identifying students much easier and convenient.

IV. Methodology

HOG: Histogram of Oriented Gradient
The histogram of oriented gradients (HOG). It is a feature descriptor used in image processing and computer vision and also the purpose of object detection. This technique counts the occurrences of gradient orientation in the image. This method is similar to edge orientation histograms, shape contexts and scale-invariant feature transform descriptors, but differs in the aspect that it is computed on a dense grid of evenly spaced cells and It uses an overlapping local contrast normalization for improved accuracy. Single angle or the distribution of the local intensity gradients in orientation can be represented without accurate knowledge of the gradient. The histogram gives invariance details of the image because it is robust to lightning changes. Histogram is calculated quickly because it deals less time complexity. The image is divided into cells of 7x15=105 blocks in total for a 64x128 image, each block has 2x2 cells with size 8x8. Then the image is divided into 16x16 blocks of 50overlap. Interpolation is done to find better histograms, this can be achieved by tri-linear interpolation. As each block has 2x2 cells with size 8x8 the gradient orientation is divided into 9 bins as shown in figure 4.2.2. The bins have the angles 0,20,40. . . .180 , 0,20,40, ... 180. The final vector will combine all the 105 (cells) histograms each one 9 dimension will become 3780 dimension feature vector.

Step 1: Finding all the faces
face detection is the first step in our pipeline . To find faces in an image, we convert our image black and white. One at a time we'll look at every single pixel in our image. we will look at the pixels that directly surrounding the current pixel.



Our objective is to find out how dark the current pixel is compared to the pixels surrounding it.

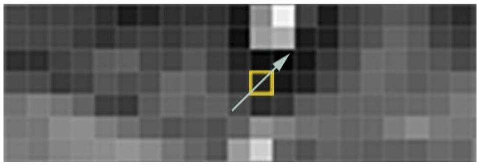


Fig. 1.

If we repeat this process for every single pixel in the image, we end up with an arrow replacing the current pixel. The arrows are called gradients and they indicate the flow from light to dark across the entire image.

Consider looking at the pixels directly,we will have a problem of identifying dark images and light images of the same person because they will have different pixel values. But

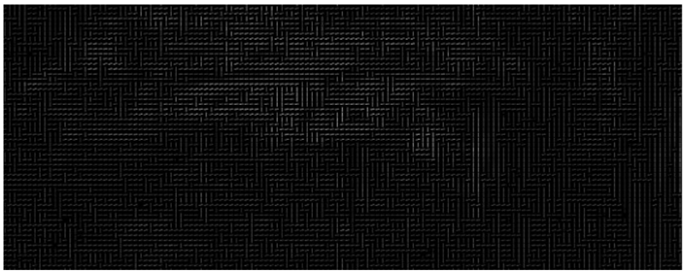


Fig. 2.

just by only considering the direction the brightness changes, both dark images and bright images will end up with the same representation. we will then break the image into small squares of 16x16 pixels each. In each square, we'll be counting how many gradients point in each direction (how many point up, point up-right, point right, etc. . .). Then we'll replace the square in the image with arrow directions that were the strongest. The end result is that we will be converting the original image into a very basic simple representation that will be able to capture the basic structure of the face in a simple way.

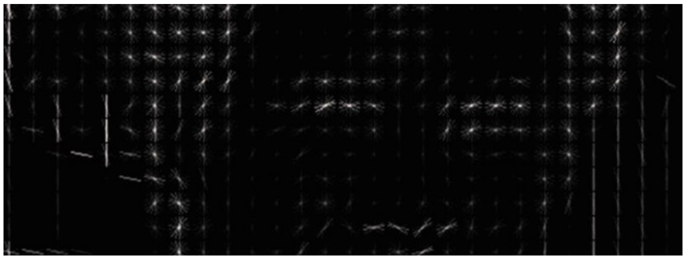


Fig. 3.

To find faces in the HOG image, we need to find the part of our image that looks the most similar to the known HOG pattern that was generated from the other cluster of other training faces:

Using this technique, we can easily find faces in any image:

Step 2: Face landmark estimation

We are using this algorithm so that the machine can identify side view of a person also .The basic outline of the idea is we will generate 68 specific points also called as landmarks that exist the face — the outside edge of each eye, the inner edge of each eyebrow,the top of the chin, etc.Then we will use a machine learning algorithm for training to find these 68 specific points on the face.With knowing these different points the machine will be able to locate the position of nose, mouth,eyes etc.Then,we will be using the basic tranformations like rotations and scaling that preserve parallel lines.This is also called as affine tranformations.

Step 3: ENOCDING FACES

We will be Passing the centered face image obtained in second step through a neural network that will know to measure features of the face. We are going to use a pre trained Deep Convolutional Neural Network to generate 128 measurements for each face like the size of each ear, the spacing between the eyes, the length of the nose, etc. They generally call these 128 measurements as embedding. we are using a pre trained network provided by open cv package in python to convert the image into 128 different values.

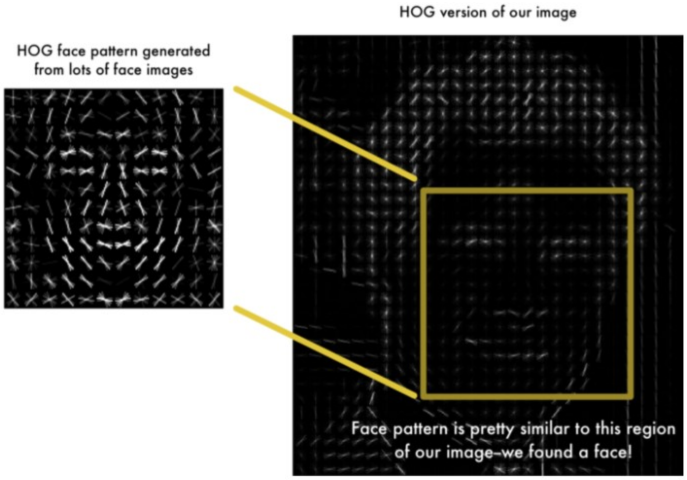

HOG version of our image
HOG face pattern generated from lots of face images
Face pattern is pretty similar to this region of our image–we found a face!

Fig. 4.

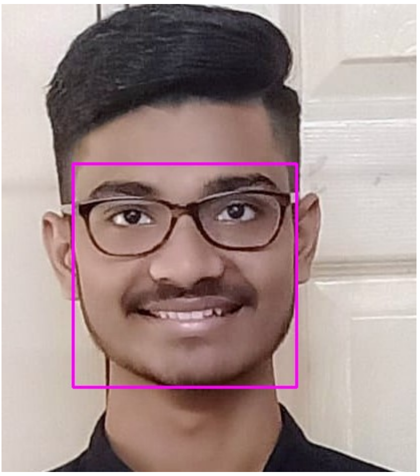


Fig. 5.

Last step: Finding the persons name from the database Finding the persons name from the database We will find the person in our database of the registered people who has the closest measurements to our test image. We will be using a Basic machine learning classification algorithm known as linear SVM classifier.we are using a simple linear svm classifer. Now we need to train a classifier that takes in these measurements from the new test image and it tells you which known person is the closest match. After doing the face recognition, only one thing is left for us to do now. That is to update the name which is recognized from the face recognition model and add it to our excel sheet. we are doing this by simply using python commands as open cv allows this to happen.

Lets just have a quick review of what we are doing:
1)We are creating a simplified version of the image by using HOG Algorithm.By, using this simplified image we are going to look at the part of the image that most looks like generic HOG encoding of a face.

2)Use the face land mark estimation algorithm to find the pose of the face. once,we find those landmarks we use them to warp up the image so that the eyes and mouth are centrered.

3)Find the embedding of the image:That is, finding the 128 different measurements using a neural network.

4)Find the closest matching of the embedding and find the name of the person if match is found.

5)If match is found den save the person's name to the excel sheet.We, are even saving the time of the attendance when taken into the excel sheet

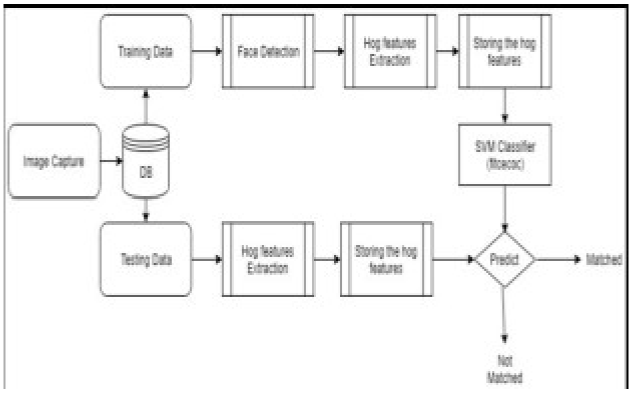


Fig. 6. Flowchart

V. System descriptions

- HARDWARE CONFIGURATIONS: This project is implemented in windows 10 version which has 8gb ram and 1tb hard disk
- SOFTWARE REQUIREMENTS:We have used python as our base language which was implemented in pycharm community edition 2020.2.2.we had to use the open cv package and dlib library for this project.OpenCV is known for its library of programming functions mainly aimed at real-time computer vision.dlib is a toolkit for making real world machine learning.
- DATASET: we have used around 30 people images in this project.The image from the webcam is taken and it is compared to the database of images if match is found then it displays the name of the person.

VI. Results and Analysis

- We first tried using the program for static images and the result was pretty good with accuracy above 97 percent

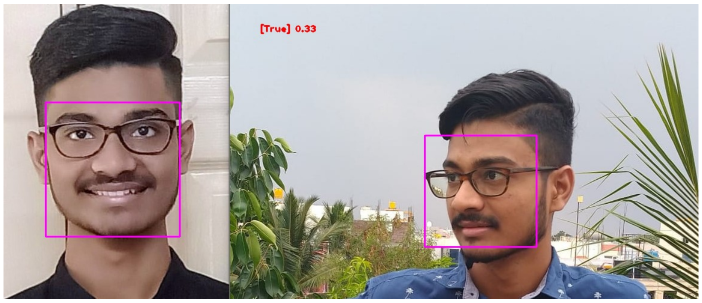


Fig. 7.

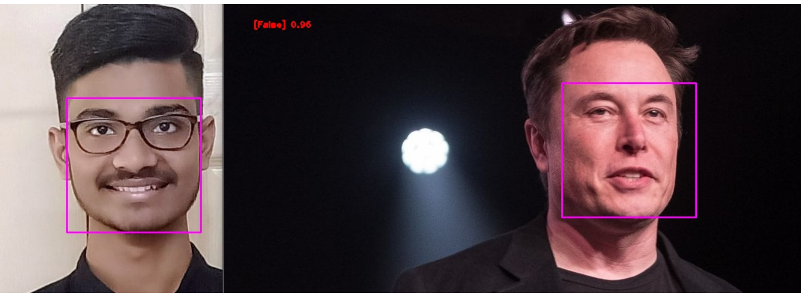


Fig. 8.

- As we can see from the above picture, the result displays true or false if the person is matched and also the number next to it represents the closeness of the images

  Later we tried implementing this in real time application using web cam.
  We tried implementing this for multiple images within the webcam and the results were pretty good. Now, as the names were recognised by the webcam this names would be stored in the excel sheet for maintaining the attendance. The below snapshot is from the python database where the names are stored in the .csv file
  Taking a look at the excel sheet.
  As we can see from the above snapshot the names of

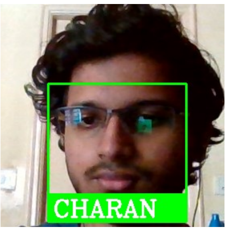


Fig. 9.

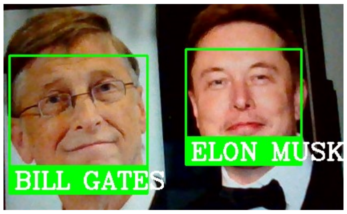
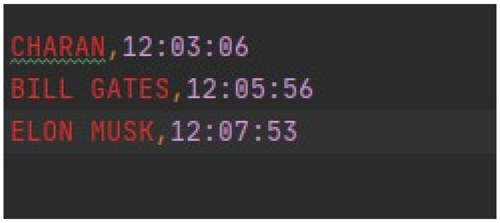
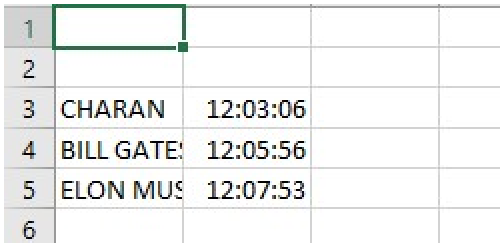


Fig. 10.



```
CHARAN,12:03:06
BILL GATES,12:05:56
ELON MUSK,12:07:53
```

Fig. 11.



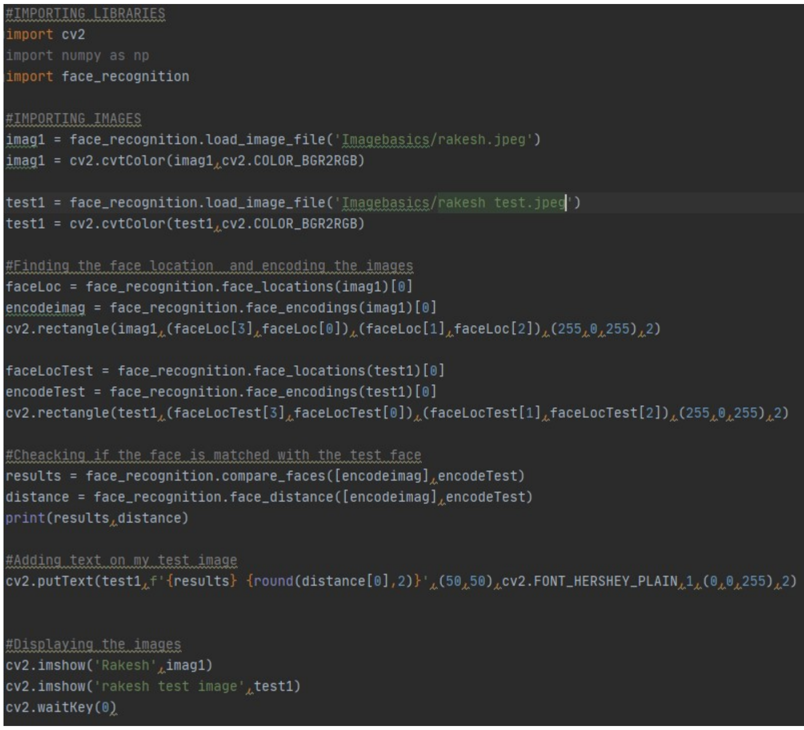
1			
2			
3	CHARAN	12:03:06	
4	BILL GATE!	12:05:56	
5	ELON MUS	12:07:53	
6			

Fig. 12.

the person and timings of their recognition are also considered.
This has been a challenging task to implement the system with face detection and recognition in real time detection.First we tried using CNN model but it required heavy hardware requirements.The process of training a convolutional neural network to produce the embeddings required a lot of data and computer power. Even with an expensive NVidia Telsa video card, it would takes more than 20 hours probably of continuous training to get good accuracy.

In our project "ATTENDANCE SYSTEM USING FACE RECOGNITION" gives us overall an efficiency of around 97 percent.

Snapshots of the code:



```
#IMPORTING LIBRARIES
import cv2
import numpy as np
import face_recognition

#IMPORTING IMAGES
imag1 = face_recognition.load_image_file('Imagebasics/rakesh.jpeg')
imag1 = cv2.cvtColor(imag1,cv2.COLOR_BGR2RGB)

test1 = face_recognition.load_image_file('Imagebasics/rakesh test.jpeg')
test1 = cv2.cvtColor(test1,cv2.COLOR_BGR2RGB)

#Finding the face location  and encoding the images
faceLoc = face_recognition.face_locations(imag1)[0]
encodeimag = face_recognition.face_encodings(imag1)[0]
cv2.rectangle(imag1,(faceLoc[3],faceLoc[0]),(faceLoc[1],faceLoc[2]),(255,0,255),2)

faceLocTest = face_recognition.face_locations(test1)[0]
encodeTest = face_recognition.face_encodings(test1)[0]
cv2.rectangle(test1,(faceLocTest[3],faceLocTest[0]),(faceLocTest[1],faceLocTest[2]),(255,0,255),2)

#Cheacking if the face is matched with the test face
results = face_recognition.compare_faces([encodeimag],encodeTest)
distance = face_recognition.face_distance([encodeimag],encodeTest)
print(results,distance)

#Adding text on my test image
cv2.putText(test1,f'{results} {round(distance[0],2)}',(50,50),cv2.FONT_HERSHEY_PLAIN,1,(0,0,255),2)


#Displaying the images
cv2.imshow('Rakesh',imag1)
cv2.imshow('rakesh test image',test1)
cv2.waitKey(0)
```

Fig. 13.

The above code is used for static images and for deploying in web cam we had to run a different but similar algorithm.
Draw backs of the system:

- Possibility of false positives
- The sensing of image may change overtime as this depends on the fps as well as the quality of the webcam
- There is possibility of data redundancy
- Identifying and differentiating twins is still a challenge for us

VII. Future work

As, we know that python is an emerging language, open cv package from python can be deployed in other platforms too. Here we are using laptop's inbuilt web cam, but we can deploy this model and connect it with to other devices like mobiles or cctv or any other device capable of recording an image. We will need a front end development to use this model effectively so that we can built a login portal for administrators to keep an eye on the attendance.It depends on the adaptability of the open cv package.We can deploy this model in android studio as well as there is open cv package.

VIII. Conclusion

Attendance system using face recognition is designed to solve the problems of the existing manual attendance system.We have used the emerging technology of face recognition to identify the person name if it is available in our database.If the identity of the person is recognised it displays the name and the name is recorded in the excel sheet of our attendance list.Even the time of the person attendance is recorded.The system efficiency is good for frontal faces but when there is disturbance in the fps of the webcam it slightly miscalculates the person's name.However i believe this problem can be solved by using higher resolution web cam and the system's efficiency can be improved by using higher specs system.

INDIVIDUAL CONTRIBUTION

Implementing HOG : CHARAN R
Implementing face landmark estimation algorithm: V N Karthik
Found the embedding algorithm to implement: Rakesh kallimani
Designing and time lime : Rahul nayaka

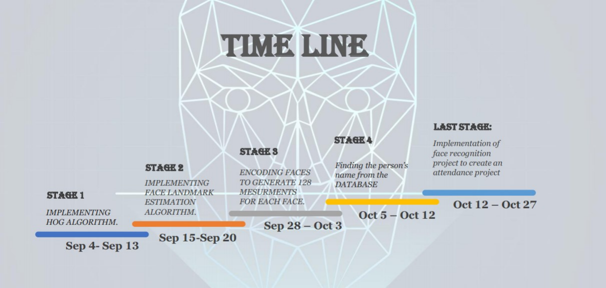

TIME LINE
STAGE 1
IMPLEMENTING HOG ALGORITHM.
Sep 4- Sep 13
STAGE 2
IMPLEMENTING FACE LANDMARK ESTIMATION ALGORITHM.
Sep 15-Sep 20
STAGE 3
ENCODING FACES TO GENERATE 128 MESURMENTS FOR EACH FACE.
Sep 28 – Oct 3
STAGE 4
Finding the person's name from the DATABASE
Oct 5 – Oct 12
LAST STAGE:
Implementation of face recognition project to create an attendance project
Oct 12 – Oct 27

Fig. 14.

IMPLEMENTED/BASE PAPER

HOG implementation: Histograms of Oriented Gradients for Human Detection by Navneet Dalal and Bill Triggs

Face landmark estimation : One Millisecond Face Alignment with an Ensemble of Regression Trees by Vahid Kazemi and Josephine Sullivan

Embedding algorithm: FaceNet: A Unified Embedding for Face Recognition and Clustering by Florian Schroff ,Dmitry Kalenichenko ,James Philbin

REFERENCES

[1] Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning by Adam Geitgey.
Website reference:https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78

[2] Histograms of Oriented Gradients for Human Detection by Navneet Dalal and Bill Triggs.

[3] One Millisecond Face Alignment with an Ensemble of Regression Trees by Vahid Kazemi and Josephine Sullivan KTH, Royal Institute of Technology Computer Vision and Active Perception Lab Teknikringen 14, Stockholm, Sweden

[4] FaceNet: A Unified Embedding for Face Recognition and Clustering by Florian Schroff Google Inc ,Dmitry Kalenichenko Google Inc, James Philbin Google Inc.

[5] Murtaza's Workshop - Robotics and AI

[6] Face Recognition Attendance System1 Palak M Jain;Shipra Singh; Nikita Singh Nancy Hablani; Dr. G. Paavai Anand

[7] Histogram of the Oriented Gradient for Face Recognition, TSINGHUA SCIENCE AND TECHNOLOGY ISSNll1007-0214ll15/15llpp216-224 Volume 16, Number 2, April 2011.

[8] Da Costa, Daniel M. M., Peres, S. M., Lima, C. A. M. and Mustaro, P. (2015) Face recognitionusing Support Vector Machine and multiscale directional image representation methods: Acomparative study. Killarney,Ireland. Neural Networks (IJCNN), 2015 International JointConference on: IEEE

[9] Dagher, I., Hassanieh, J. and Younes, A. (2013) Face recognition using voting technique for theGabor and LDP features. Dallas, TX, USA. Neural Networks (IJCNN), The 2013 InternationalJoint Conference on: IEEE

[10] Wenyi Zhao, and Rama Chellappa (2005) Face Processing: Advanced Modeling and Methods.Elsevier Science.

[11] Castrillón, M., Déniz, O., Hernández, D. and Lorenzo, J. (2011) 'A comparison of face and facialfeature detectors based on the Viola–Jones general object detection framework', Machine Visionand Applications, 22 (3), pp. 481-494.

[12] Alpaydin, E. (2014) Introduction to Machine Learning. 3rd ed edn. Cambridge: Cambridge: TheMIT Press.

[13] Belaroussi, R. and Milgram, M. (2012) 'A comparative study on face detection and trackingalgorithms', Expert Systems with Applications, 39 (8), pp. 7158-7164

[14] Smart Attendance Management System Using Face Recognition Kaneez Laila Bhatti1,*, Laraib Mughal1, Faheem Yar Khuhawar1, Sheeraz Ahmed Memon11Dept. of Telecommunication Engineering, MUET, Jamshoro, PK

[15] Linna, M., Kannala, J. and Rahtu, E. (2015) 'Online face recognition system based on local binarypatterns and facial landmark tracking', Lecture Notes in Computer Science (Including SubseriesLecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9386 pp. 403-414.

[16] Lenc, L. and Král, P. (2014) 'Automatic face recognition approaches', Journal of Theoretical andApplied Information Technology, 59 (3), pp. 759-769

[17] Dhanaseely, A. J., Himavathi, S. and Srinivasan, E. (2012) 'Performance comparison of cascadeand feed forward neural network for face recognition system', pp. 21.

[18] Castrillón, M., Déniz, O., Hernández, D. and Lorenzo, J. (2011) 'A comparison of face and facialfeature detectors based on the Viola–Jones general object detection framework', Machine Visionand Applications, 22 (3), pp. 481-494.

[19] Saxena, V.Grover, S. and Joshi, S. (2008) 'A real time face tracking system using rank deficientface detection and motion estimation' Cybernetic Intelligent Systems, 2008.CIS 2008.7th IEEEInternational Conference on, pp. 1-6. 10.1109/UKRICIS.2008.479895