University Institute of Technology

Rajiv Gandhi Proudyogiki Vishwavidyalaya

Automated Attendance System Using Face Recognition
Minor Project Report

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

University Institute of Technology– RGPV

Submitted by:

Divyanshu Dev Awasthi (0101CS181045)

Anuja Deharia (0101CS181045)

Faiz Muhhammad (0101CS181046)

Jatin Vishwakarma (0101CS181045)

DECLARATION

I declare that the project work with the title **"Automated Attendance System Using Face Recognition**" is my original work done under **Prof. SHIKHA AGRAWAL** and **Prof. BHAWNA KOL,** University Institute of Technology - RGPV, Bhopal. I have learned and followed all the rules and regulations provided by the Institute while writing this report. I have tested the plagiarism, which is below 30 percent in similarity. This project work is being submitted as the MINOR PROJECT in the fulfillment of the requirements for the degree of Bachelor of Technology in Computer Science Engineering at University Institute of Technology – RGPV Bhopal for the academic session 2018– 2022.

Divyanshu Dev Awasthi (0101CS181045)

Jatin Vishwakarma (0101CS181055)

Faiz Muhhammad (0101CS181046)

Anuja Deharia (0101EC181024)

University Institute of Technology

Rajiv Gandhi Proudyogiki Vishwavidyalaya

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project titled "**AUTOMATED ATTENDANCE SYSTEM USING FACE RECOGNITION**" submitted by Divyanshu Dev Awasthi (Roll no: 0101CS181045) in partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Computer Science engineering at University Institute of Technology – RGPV, Bhopal, is an authentic work carried out by him under my supervision and guidance.

Prof. Bhawna Kol
DoCSE
UIT-RGPV, Bhopal

Prof. Shikha Agrawal
DoCSE
UIT-RGPV, Bhopal

ACKNOWLEDGEMENT

On the submission of my thesis report on “Automated Attendance System Using Face Recognition”, I would like to express my indebted gratitude and special thanks to my supervisor Prof. Shikha Agrawal & Prof. Bhawna Kol, Department of Computer Science Engineering, UIT-RGPV, Bhopal, who in spite of being extraordinarily busy, spare time for guidance and keep me on the correct path and allowed me to carry out my work in the previous year. I truthfully appreciate and value her admired supervision and support from the start to the end of this project. I am obliged to her for having helped me shape the trouble and providing insights towards the way out.

I would like to offer my special thanks to all the other Professors from the Department of Computer Science Engineering, UIT-RGPV, Bhopal for providing a solid backdrop for our studies and exploration afterward. They have been a great sources of motivation for me and I thank them from the core of my heart.

Last but not the least I would like to thank each and every person who is involved directly or indirectly to make this project successful.

Divyanshu Dev Awasthi (0101CS181045)

Jatin Vishwakarma (0101CS181055)

Faiz Muhhammad (0101CS181046)

Anuja Deharia (0101EC181024)

Contents

Figures

ABSTRACT

Uniqueness or individuality of an individual is his face. In this project face of an individual is used for the purpose of marking their attendance automatically. Attendance of the students is very important for every college, universities and school. Conventional methodology for taking attendance is by calling the name or roll number of the student and the attendance is manually recorded on the attendance register. Time consumption for this purpose is an important point of concern. Assume that the duration for one subject is around 60 minutes or 1 hour & to record attendance takes 5 to 10 minutes. For every tutor this is consumption of time. To stay away from these losses, an automatic process is used in the form of this project which is based on image processing. In this project face detection and face recognition is used. Face detection is used to locate the position of face region and face recognition is used for marking the understudy's attendance. The database of all the students in the class is stored and when the face of the individual student matches with one of the faces stored in the database then the attendance is recorded.

INTRODUCTION

Attendance is prime important for both the teacher and student of an educational organization. So it is very important to keep record of the attendance. The problem arises when we think about the traditional process of taking attendance in class room. Calling name or roll number of the student for attendance is not only a problem of time consumption but also it needs energy. So an automatic attendance system can solve all the above problems.

There are some automatic attendances marking system which are currently used by many institutions. One of such system is biometric technique. Although it is automatic and a step ahead of traditional method it fails to meet the time constraint. The student has to wait in queue for giving attendance, which is time taking.

This project introduces an involuntary attendance marking system, devoid of any kind of interference with the normal teaching procedure. The system can be also implemented during exam sessions or in other teaching activities where attendance is highly essential. This system eliminates classical student identification such as calling name of the student, or checking respective identification cards of the student, which can not only interfere with the ongoing teaching process, but also can be stressful for students during examination sessions.

Problem Statement

Traditional student attendance marking technique is often facing a lot of trouble. The face recognition based automated attendance system emphasizes its simplicity by eliminating classical student attendance marking technique such as calling student names or checking respective identification cards. There are not only disturbing the teaching process but also causes distraction for students during exam sessions. Apart from calling names, attendance sheet is passed around the classroom during the lecture sessions. The lecture class especially the class with a large number of students might find it difficult to have the attendance sheet being passed around the class. Thus, face recognition student attendance system is proposed in order to replace the manual signing of the presence of students which are burdensome and causes students get distracted in order to sign for their attendance. Furthermore, the face recognition based automated student attendance system able to overcome the problem of fraudulent approach and lecturers does not have to count the number of students several times to ensure the presence of the students.

Hence, there is a need to develop a real time operating automated attendance system which means the identification process must be done within defined time constraints to prevent omission. The extracted features from facial images which represent the identity of the students have to be consistent towards a change in background, illumination, pose and expression. High accuracy and fast computation time will be the evaluation points of the performance

Aim and Objectives

The objective of this project is to develop automated attendance system based on face recognition. Expected achievements in order to fulfill the objectives are:

- To detect the face segment from the video frame.
- To extract the useful features from the face detected.
- To classify the features in order to recognize the face detected.
- To record the attendance of the identified student.
- To mail the attendance to the concerned party.

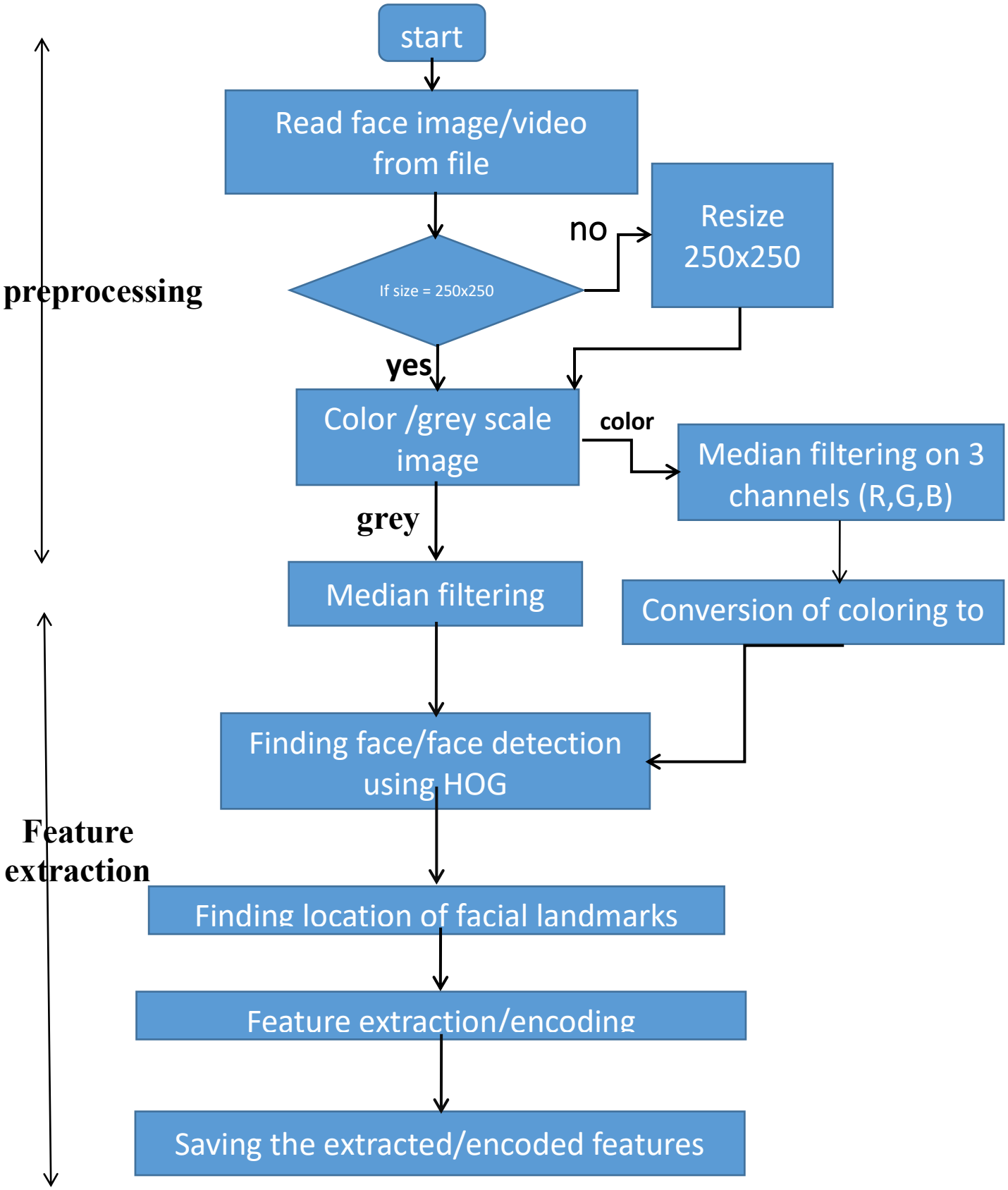
General Framework

1. Image Acquisition from video frame
2. Face detection
3. Feature extraction
4. Face recognition
5. Attendance
6. Mailing the Attendance

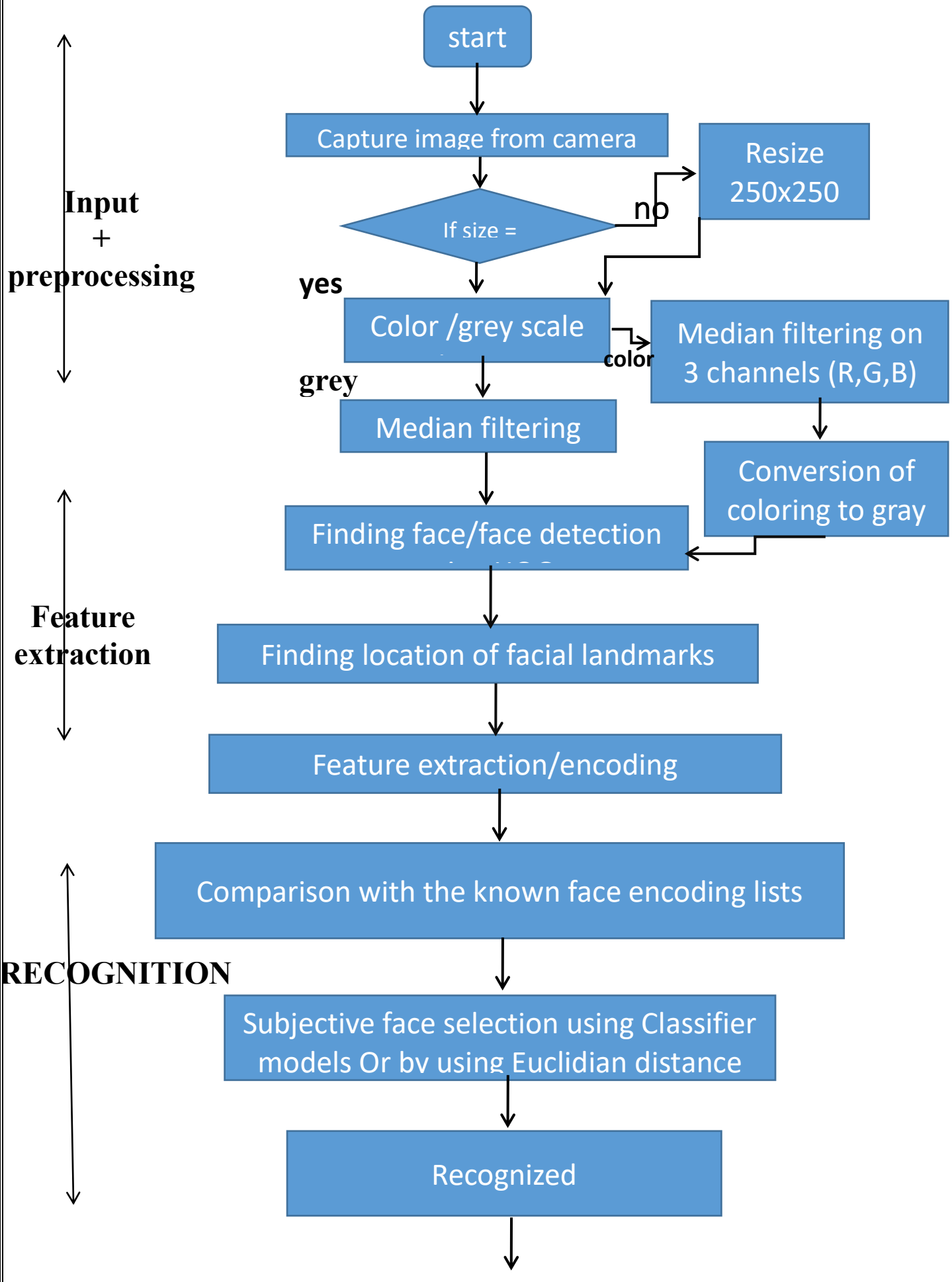
Face recognition is a series of several related problems, which we want to solve:

1. Finding all the faces in a picture.
2. Understanding that even in different circumstances and environment who's the same person.
3. Picking out unique features.
4. Comparing the unique features of that face to all the people you already know to determine the person's name.

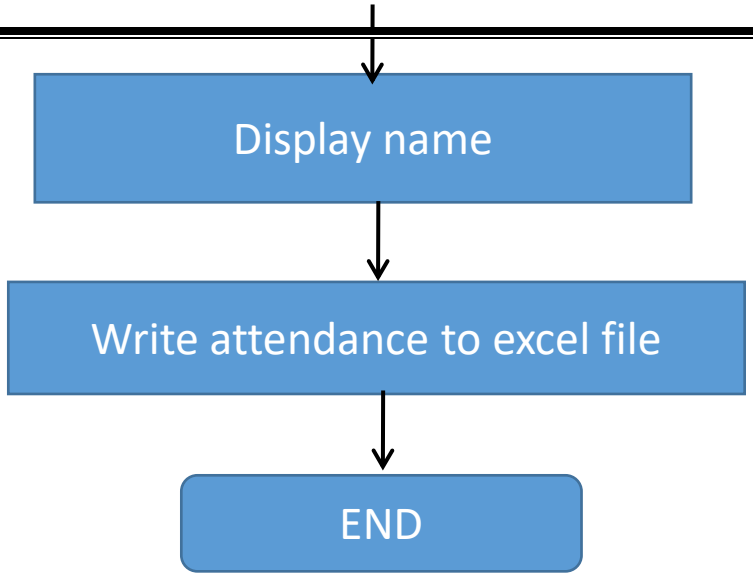
Flow of the proposed approach (Training part)



Flow of the proposed approach (Recognition Part)



Attendance Registering / Record Updating



Chapter 1

Face detection

1.1 Introduction

Face detection is defined as finding the position of the face of an individual. In other word it can be defined as locating the face region in an image. After detecting the face of human its facial features is extracted and has wide range of application like facial expression recognition, face recognition, observation systems, human PC interface and so forth…Detecting face in an image of single person is easy but when we consider a group image of an image containing multiple faces, the task becomes difficult.

For the application of face recognition, detection of face is very important and the first step. Only after detecting the faces the face recognition algorithm can be functional. Face detection itself involves some complexities for example surroundings, postures, enlightenment etc.

There are some existing methodologies for detection of face. Some of them are skin color based, characteristic or feature based (feature like mouth, nose and eyes) and neural network based.

1.2 Finding all the faces in a picture

For this we use a method called Histogram of Oriented Gradients (HOG)

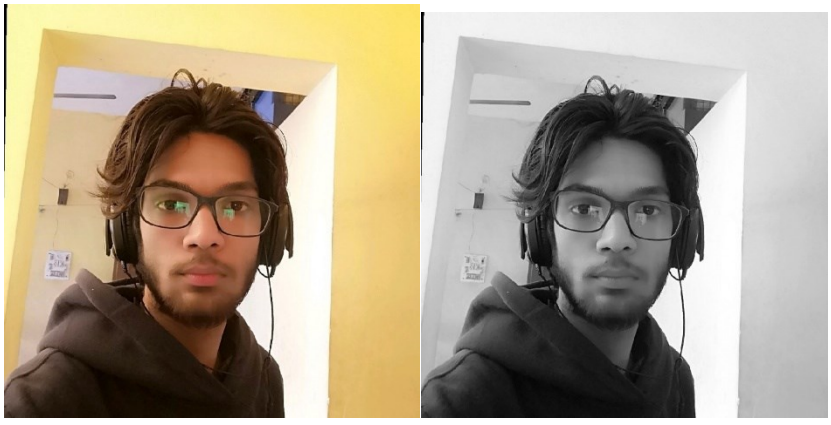
As we don't need colour to find faces in an image we'll take black and white image:

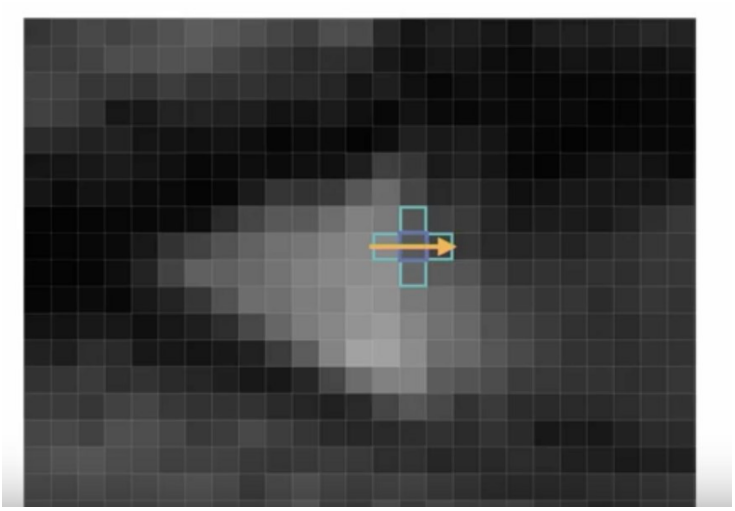
Then we'll look at every single pixel in our image one at a time and the pixels that directly surrounds it, and try to figure out how dark it is as compared to them so that we can draw an arrow towards the direction which is getting darker.

Eventually every pixel will be replaced by an arrow (gradients), as as it is too much of detail we'll break up the image into small square of 16x16 pixels each and reapply the above process, but this time on these blocks and replace them by arrows, resulting in basic structure of face in a simple way.

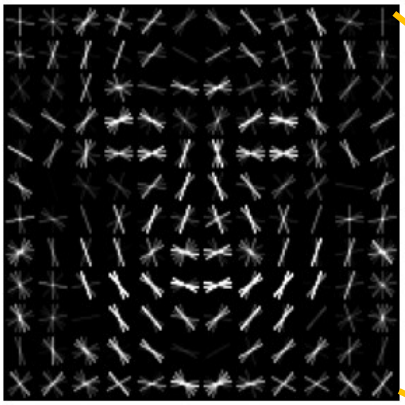
If we analyze pixels directly, really dark images and really light images of the same person will have totally different pixel values. But by doing above things, both images will end up with the exact same representation. That makes the problem a lot easier to solve!

To find faces in this HOG image, all we have to do is find the part of our image that looks the most similar to a known HOG pattern that was extracted from a bunch of other training faces:



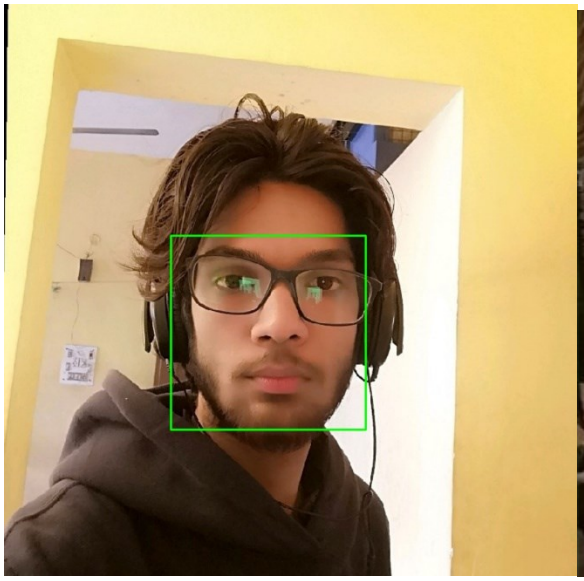


HOG face pattern generated from lots of face images



HOG version of our image

Face pattern is pretty similar to this region of our image – We found a face!



Chapter 2

Face Encodings Generation

2.1 Introduction

Face encoding generation is needed to identify a face by a computer. It is defined as identifying how a face of an individual is after its. In other word it can be defined as converting a face of an individual into a unique numerical data that specifies its face.

For the application of face recognition, encoding of face might be the most important step. Only after encoding the faces the face recognition algorithm can be functional. Face encodings itself involves some complexities for example postures and angels.

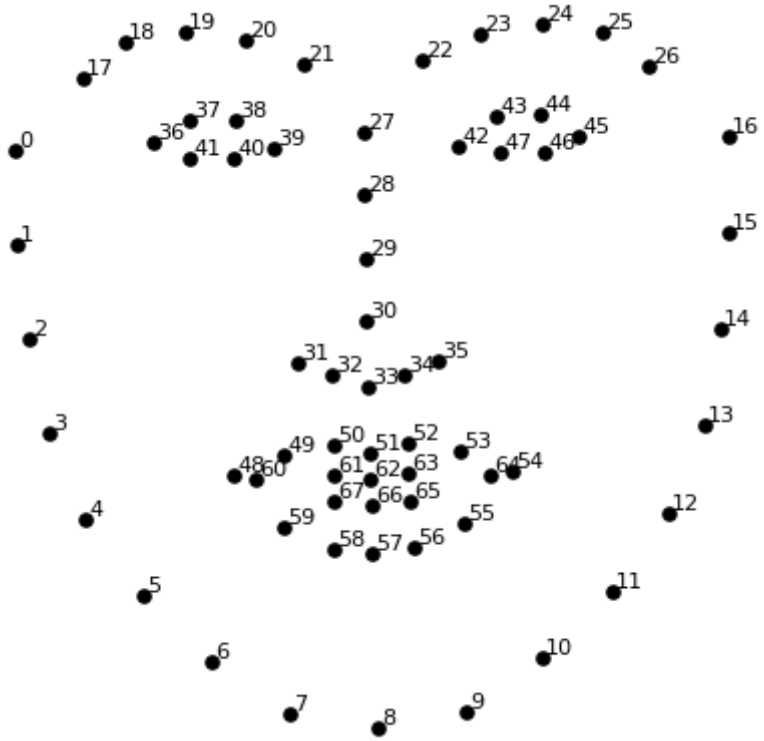
2.2 Posing and Projecting Faces

What if the faces aren't facing the camera and are looking in different directions, can computer recognise them as same person? NO

To solve this problem we can wrap the faces so that eyes and lips are always in the sample place of the image.

To do this, we use an algorithm called **face landmark estimation**.

The basic idea is we will come up with 68 specific points (called *landmarks*) that exist on every face — the top of the chin, the outside edge of each eye, the inner edge of each eyebrow, etc. Then we will train a machine learning algorithm to be able to find these 68 specific points on any face:



2.3 Encoding Faces

We humans can tell the faces apart but how would a computer do the same thing?

By encoding the features of a face, converting it into numerical data.

So what should be the features to encode?? (not from the human point of view)

We leave that task to computer itself, as deep learning does a better job than humans at figuring out which parts of a face are important to measure.

So we train a Deep Convolutional Neural Network to generate 128 measurements for each face.

The training process works by looking at 3 face images at a time:

1. Load a training face image of a known person
2. Load another picture of the same known person
3. Load a picture of a totally different person

Then the algorithm looks at the measurements it is currently generating for each of those three images. It then tweaks the neural network slightly so that it makes sure the measurements it generates for #1 and #2 are slightly closer while making sure the measurements for #2 and #3 are slightly further apart:

After repeating this step millions of times for millions of images of thousands of different people, the neural network learns to reliably generate 128 measurements for each person. Any ten different pictures of the same person should give roughly the same measurements.

And now our model can generate measurements for any face, so we run it on our student's faces and save the encodings.

```
[-0.1534045   0.06742257  0.10010329  0.0224112  -0.01631372 -0.12036555
 -0.00271989 -0.03382826  0.11123551 -0.0388266   0.28441167 -0.02358414
 -0.15523493 -0.14264797  0.00064788  0.11415445 -0.15994549 -0.08783647
 -0.00083037 -0.07554685  0.06694273 -0.00797964 -0.03031057  0.10923384
 -0.09528444 -0.365749   -0.09523882 -0.158163    0.06083713 -0.0355465
  0.00284566  0.09772179 -0.17882326 -0.05268389 -0.01560916  0.08920108
  0.06949284 -0.0237027   0.19224562 -0.07223796 -0.12398335 -0.06266483
  0.06236489  0.27043861  0.10529109 -0.03216873  0.05715149  0.01525116
  0.08437168 -0.1705351   0.06652839  0.10013241  0.15761557  0.11829457
  0.04481268 -0.19547726  0.0060476   0.08484095 -0.24009177  0.0479988
 -0.05977572 -0.07879025 -0.04080552 -0.04001265  0.28817013  0.11674345
 -0.14721826 -0.14816414  0.14078428 -0.17626134 -0.01953826  0.07207128
 -0.09518676 -0.1369703  -0.3545824   0.12577577  0.30411801  0.12682906
 -0.24497914  0.08529955 -0.13297141 -0.07454696  0.02984791  0.0875254
 -0.06449533  0.12697943 -0.12520993  0.00437515  0.11625271  0.04341982
 -0.03890661  0.22273076 -0.02121942  0.04575346  0.05350915  0.05089413
 -0.07699683 -0.04400311 -0.11078162  0.01608595  0.05422406 -0.08579422
 -0.04437164  0.0525056  -0.20644605  0.11587299  0.0088893   0.00195532
 -0.06432797  0.02569891 -0.13976113 -0.03409303  0.14195023 -0.29628614
  0.21152617  0.10798469  0.03808979  0.23912078  0.07883725  0.05459694
 -0.04184894 -0.1299406  -0.09697782 -0.02669389  0.10433658 -0.04887332
  0.05869493  0.05613514]
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
2	0	-0.1102	0.04947	0.116701	-0.14625	-0.13738	-0.07407	-0.07983	-0.15741	0.192452	-0.12623	0.117552	-0.09327	-0.28198	0.028469	-0.1074	0.149111	-0.164	-0.23953	-0.00103	-0.07368	0.050208	0.034813
3	1	-0.03582	0.158331	0.10453	-0.02674	-0.13543	0.074238	-0.08425	-0.05082	0.128352	-0.0532	0.234383	0.003397	-0.18912	-0.09151	-0.09906	0.07892	-0.04752	-0.11893	-0.07327	-0.00085	-0.01563	0.088995
4	2	-0.14049	0.037297	0.049047	-0.07728	-0.06252	-0.08918	-0.09589	-0.07887	0.188309	-0.07366	0.126197	-0.04191	-0.20664	-0.08146	-0.05756	0.147403	-0.16483	-0.16887	-0.01686	-0.08294	0.126958	0.051851
5	3	-0.1534	0.067423	0.100103	0.022411	-0.01631	-0.12037	-0.00272	-0.03383	0.111236	-0.03883	0.284412	-0.02358	-0.15523	-0.14265	0.000648	0.114154	-0.15995	-0.08784	-0.00083	-0.07555	0.066943	-0.00798
6	4	-0.12869	0.117392	0.100653	-0.06088	-0.11162	-0.00701	-0.03189	-0.06989	0.112623	-0.0262	0.129453	-0.02189	-0.25774	0.101844	-0.02202	0.063705	-0.19215	-0.09527	-0.11558	-0.04169	0.015672	0.114216
7	5	-0.20128	0.180923	0.006539	-0.03984	-0.05957	-0.07622	-0.0274	-0.06228	0.194284	-0.06761	0.130368	0.002116	-0.16402	-0.11503	-0.08717	0.081383	-0.16547	-0.13198	-0.04453	-0.02363	0.14328	-0.00462
8	6	-0.14627	0.139517	0.039301	-0.06635	0.042696	-0.08528	-0.06522	-0.05491	0.108965	0.005218	0.221307	-0.02049	-0.19562	-0.11905	-0.01948	0.067663	-0.14067	-0.13705	-0.01302	-0.09966	0.042449	0.018503
9	7	-0.17673	0.068747	0.019823	-0.08026	0.039806	-0.08153	0.025188	-0.06219	0.182649	-0.00204	0.254384	0.02516	-0.23288	-0.09765	0.008868	0.051861	-0.18386	-0.11624	-0.10455	-0.10765	0.067843	-0.01874
10	8	-0.01668	0.117027	0.058027	-0.05669	-0.05058	-0.00036	-0.06772	-0.05184	0.117166	0.001003	0.16806	0.04281	-0.22093	0.139603	-0.01679	0.064642	-0.12286	-0.13348	-0.13009	-0.01726	0.026296	0.070326
11																							
12																							
13																							
14																							
15																							
16																							
17																							
18																							
19																							
20																							

	DC	DD	DE	DF	DG	DH	DI	DJ	DK	DL	DM	DN	DO	DP	DQ	DR	DS	DT	DU	DV	DW	DX	DY
1	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
2	0.049218	0.004914	-0.02831	-0.04783	0.075486	-0.09192	-0.02875	0.160429	-0.26951	0.237419	0.154991	0.053652	0.090448	0.012637	0.055446	0.053289	-0.09112	-0.07095	-0.10121	0.037457	-0.0892	-0.02666	0.087558
3	0.212513	-0.00352	-0.05013	-0.01354	-0.06062	-0.06628	0.024253	0.167082	-0.34908	0.243827	0.090535	0.011715	0.241884	0.102607	0.11031	-0.0465	-0.11931	-0.13068	-0.13321	0.10038	-0.077	0.033727	0.032355
4	0.037941	-0.05514	-0.08053	-0.10825	0.001761	-0.05998	-0.06751	0.121424	-0.29306	0.160309	0.220163	-0.03156	0.178187	-0.01422	0.095062	-0.00563	-0.11206	-0.12345	-0.10915	0.05037	-0.16293	0.058325	0.033868
5	0.115873	0.008889	0.001955	-0.06433	0.025699	-0.13976	-0.03409	0.14195	-0.29629	0.211526	0.107985	0.03809	0.239121	0.078837	0.054597	-0.04185	-0.12994	-0.09698	-0.02669	0.104337	-0.04887	0.058695	0.056135
6	0.108209	-0.05747	-0.00449	-0.02876	0.050723	-0.13531	-0.05497	0.172091	-0.23095	0.221649	0.180887	0.007776	0.059755	-0.01024	-0.00854	0.124021	0.077815	-0.14806	-0.12933	0.044571	-0.08291	-0.0037	0.114347
7	0.035746	-0.00222	-0.07112	-0.09214	0.078467	-0.18855	-0.07733	0.084292	-0.19202	0.260382	0.160126	0.032324	0.147277	0.081129	0.065512	0.027103	-0.02309	-0.07841	-0.10341	0.117003	-0.01026	0.101895	-0.01651
8	0.123028	-0.01498	-0.01468	-0.03339	0.15661	-0.0738	-0.06771	0.109573	-0.21475	0.185198	0.181032	0.061822	0.127679	0.066599	-0.01931	0.007579	-0.04501	-0.04578	-0.10544	0.056743	0.020463	0.035462	0.001764
9	0.106442	0.00754	-0.07969	-0.01547	0.009861	0.024784	-0.00941	0.10992	-0.254	0.245306	0.222146	-0.00795	0.203853	0.01839	0.097032	-0.1073	-0.03219	-0.17845	-0.06119	-0.0462	0.072925	0.032138	0.002537
10	0.197149	0.013196	-0.07395	0.0143	0.012895	-0.11624	-0.08281	0.186655	-0.21272	0.206302	0.183572	0.088728	0.030349	0.038225	0.074178	5.29E-05	0.055667	-0.26121	-0.10723	0.149122	-0.03776	-0.04866	0.034743
11																							
12																							
13																							
14																							
15																							
16																							
17																							
18																							
19																							
20																							

Chapter 3

Face Recognition

3.1 Introduction

Face Recognition is the final piece in this puzzle.

How do we recognise a person from its face?

Actually it's the same for computers as it is for humans, by comparing/recognising the features of a face of an individual, but the difference is the features recognised by computer and human.

Computer will use it numerically generated encodings as a face's features, and try to recognise the face of an individual.

3.2 Face Recognition

(Finding the person's name from the face image)

For this we will first generate the encodings for the face image and them we can easily compare it with our encoding database of known faces.

We can use classification algorithms or we can just measure their Euclidian distance.

Most successful classification algorithms are SVM (also most widely used) and KNN.

By experimenting I found out that KNN gives more accurate results as compared to some other algorithms including SVM, RandomTree, LogisticRegression, etc.

& KNN algorithm is essentially the Euclidian distance of the some of the nearest entries.

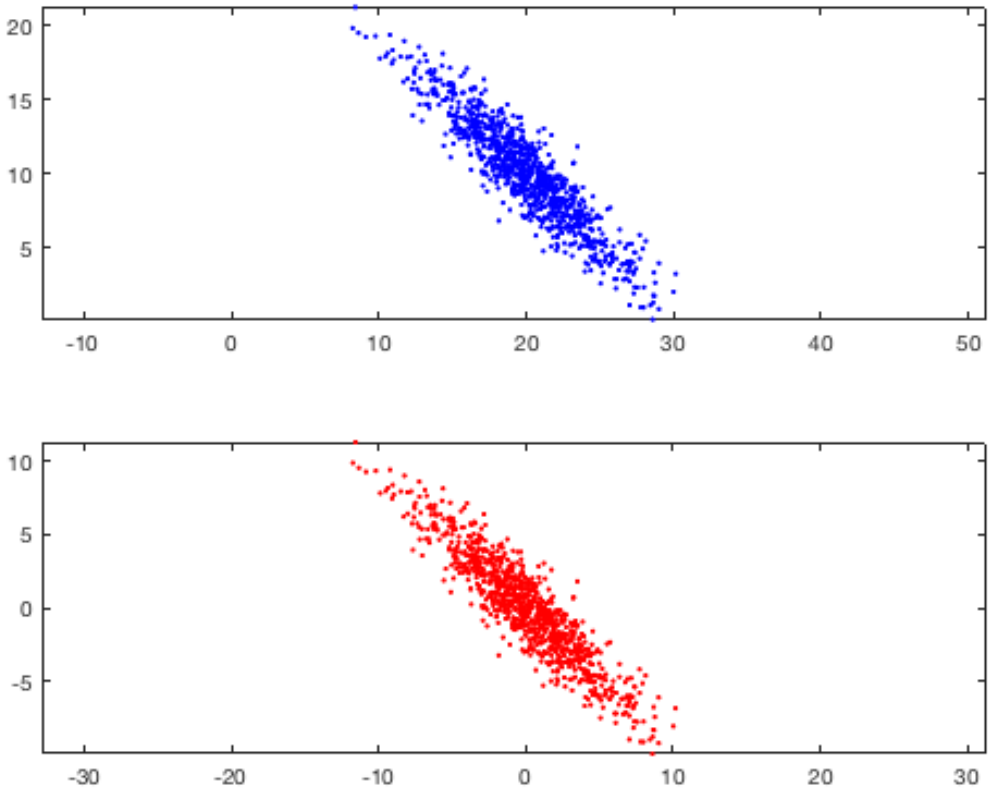
Using Euclidian distance directly goes fall in accuracy by a very little amount but it speed also improve by 3 to 5 times.

This doesn't really affect much early in the stage but as the no. of faces increases, this advantage slowly starts building up.

An example of comparing Euclidian Distance of two encodings-

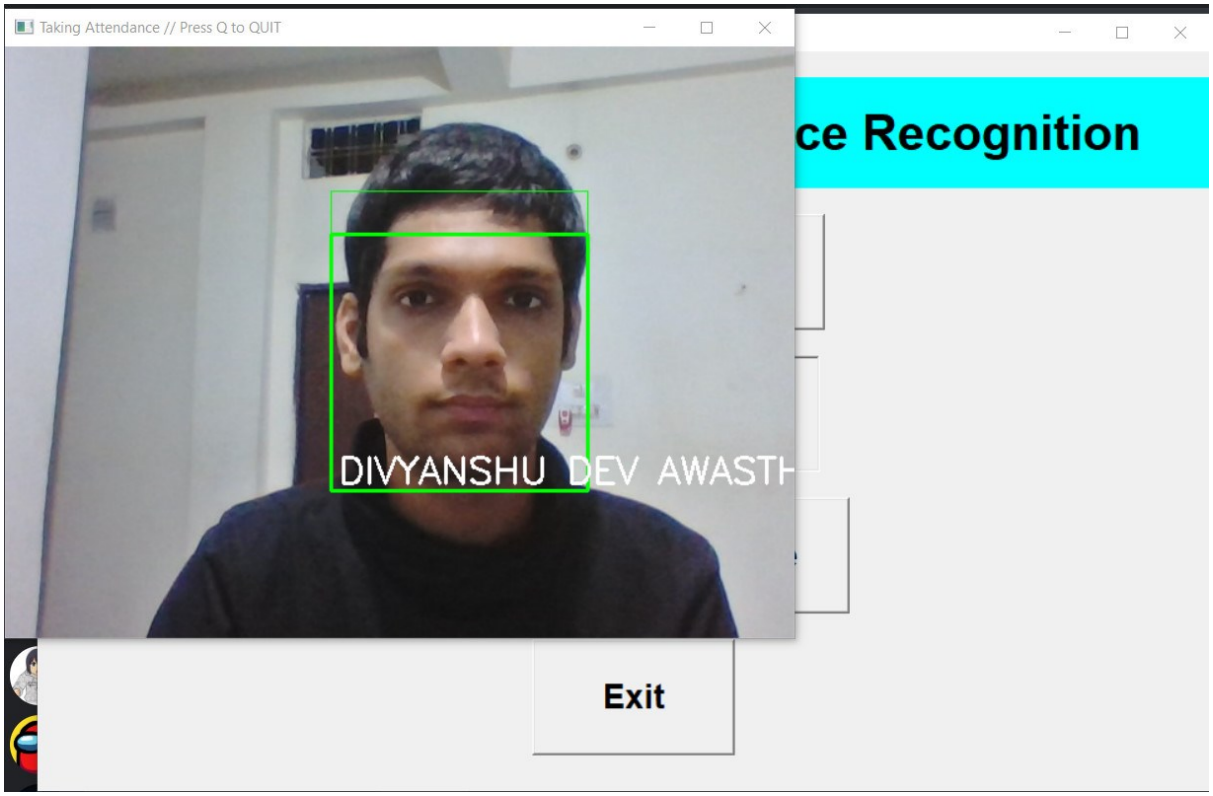
Blue Encoding is one of the encoding saved in our database.

Red encoding is the encoding of the face we have to recognise.



As we can see these two encodings are very close so they should be from a same face, and we know who the blue encoding belongs which means we now know what is the name behind the red encoding's face, which in turn means that we have successfully recognised the face of an individual.

Criteria for a face to be recognised is that, Euclidean distance must be less than 0.5 or 0.6


Taking Attendance // Press Q to QUIT
DIVYANSHU DEV AWASTH
ce Recognition
Exit

Chapter 4

Attendance Entry

4.1 Introduction

Class attendance is very important aspects for the students studying in the colleges or schools. For an organization to be successful, it needs precise and quick method for recording the performance of the individuals inside this organization. Attendance gives the data of the individual whether that particular person was physically present or absent during the class. The traditional method of calling roll number or name of the student for marking attendance is time consuming or wastage of time and labor during the class hour.

4.2 Attendance Sheet

Our attendance sheet (an excel file) initially contains 2 columns – Name and Time.

	A	B	C
1	Name	Time	
2			
3			
4			

4.3 Marking Attendance of the Student

As soon as a face is recognised, the face and the time it is first recognised will be recorded in a CSV Sheet, effectively marking the attendance of the individual in the attendance sheet

	A	B
1	Name	Time
2	AISHWARYA RAI	02:21:40
3	DIVYANSHU DEV AWASTHI	02:21:47
4	MEHUL	11:03:11
5	MAYANK ATAMPOOJYA	02:11:32
6		
7		

Chapter 5

Mailing the Attendance Sheet

5.1 Introduction

A mailing function has been built in its GUI application so that after taking the Attendance you can simply mail it to yourself

Chapter 6

Softcopy & Source Code

It can be found here on Github -

https://github.com/divyanshuhub/Automated_attendance_system_using_face_recognition

Where you can

Simply run gui_final.py to open up a GUI

Or

In terminal, run - python setup.py build to generate its executable file

Chapter 7

Conclusion and Recommendations

7.1 Conclusion

In this approach, a face recognition based automated attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to change the image into a black and white and reduce the size of image for faster processing. Extraction of features from the facial image is performed by applying both HOG and CNN. The accuracy of this proposed approach is 100 % for high-quality images, and 95 % for low-quality images

As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting. In pre-processing stage, we converted the image into Black and white, then used HOG and then CNN and then we may use KNN Classification algorithm or Euclidian distance, KNN is the best approach here with best accurate results even for similar faces but for a very big crowd where results are required faster, Euclidian distance estimation might be better as it seldom gives wrong prediction and that to with very similar faces like those of twins.

7.2 Recommendations

In this proposed approach, there are a few limitations.

First, the doppelganger and twins might lead to False Recognition in Euclidean Distance estimation approach.

Second, every time a new face is added it takes a sometime as it generates encodings as the Training Image Dataset grows this time shall increase even more.

We can make it such as every time a new face is added only its encodings get generated and this might create problems such as false recognition due to name list not being in sync (in case of Euclidean Distance estimation approach). Here using KNN classification is much better as it will solve both the problems.

Chapter 8

Future Prospects

8.1 Future Prospects

Future Prospects of our projects are-

- To be integrated into an application for taking live classes such as Zoom, MS Teams, etc.
- To become a webapp for more accessibility.
- To become a mobile app for more accessibility and reach.
- To be used for security purposes.
- To be used for Data Collection.
- To be used for institutional surveillance
- Etc.

REFERENCES

[1] Adam Geitgey (2017). *Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning*

[online] Available at:

https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78

[2] Jesse Davis West. (2017). *History of Face Recognition - Facial recognition software*.

[online] Available at:

https://www.facefirst.com/blog/brief-history-of-face-recognition-software/

[3] deAgonia, M. (2017). *Apple's Face ID [The iPhone X's facial recognition tech explained]*.

[online] Computerworld. Available at:

https://www.computerworld.com/article/3235140/apple-ios/apples-face-id-the-iphone-xs-facial-recognition-tech-explained.html

[4] Sidney Fussell. (2018). *NEWS Facebook's New Face Recognition Features: What We Do (and Don't) Know*.

[online] Available at:

https://gizmodo.com/facebooks-new-face-recognition-features-what-we-do-an-1823359911