**FACE RECOGNITION ATTENDANCE**

**SYSTEM**

**PROJECT SYNOPSIS**

OF MAJOR PROJECT

**BACHELOR OF TECHNOLOGY**

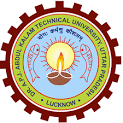
COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY:-

VINAY PRATYUSH

1907860100107

OCTOBER 2022



**MANGALMAY INSTITUTE of ENGINEERING & TECHNOLOGY**

**8, KNOWLEDGE PARK – II**

**GREATER NOIDA - 201308**

**Table of contents**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Contents** | **Page no.** |
| 1. | Introduction | 1. |
| 2. | Rationale | 2. |
| 3. | Objectives | 3. |
| 4. | Literature Survey | 4. |
| 5. | Feasibility study | 8. |
| 6. | Methodology | 9. |
| 7. | Facilities required for proposed work | 18. |
| 8. | Expected outcome | 19. |
| 9. | References | 20. |
| 10. | End |  |

**1.** **Introduction**

Old practices of attendance are not quite efficient now a days for keeping track on student’s attendance. Student enrolment in schools and colleges is increasing every year and taking each student attendance can be a time consuming task and also wasteful. However, it is vital to maintain records of a student’s appearance in class So, it is necessary to discuss the effective system which records attendance of student automatically.

Maintaining the attendance is very important for school and colleges to assess the performance of a student. All school/colleges have their own way of taking attendance. Most are taking attendance manually using attendance registers, marking attendance sheet or file-based approach. These methods are cost-effective and can work if number of students are low. But when scale is increased, number files can increase significantly and keeping track can become harder manually. It also consumes unnecessarily a lot of time which can be avoided if the system is automated. Many organizations are using biometrics system to take attendance. But when applied on bigger scale like schools and colleges where number of students are very high in comparison to employees of some organization, large ques can form which will again be time consuming and the problem will not be solved. On top of that it also costs more than taking attendance manually.

Taking consideration of all the above-mentioned points we have introduced a new way of tracking attendance of students. We have used face recognition technology to recognize students based on recordings from camera and photos of students, provided to school, collage or organization’s administration. Instead of manually marking the attendance of students by roll-call, we will recognize a student while they enter the class by cameras and automatically mark the attendance of the student. It can be done by comparing the student’s camera recording by the photo provided in collage database. The name of student if present in collage database will automatically be written in an excel sheet along with the time they have entered the class. This will encourage students to attend classes and also be punctual.

The project is made using python and face recognition framework. Face recognition framework has an already trained model for different tasks required face detection. The framework allows us to detect faces find how similar the faces are and can also tell us if person is same or not with 98 percent accuracy which is very close to human level accuracy in detecting faces. We have used open-cv library to capture the image and for general interaction with camera. It is crucial for using camera for recording and processing images. We have used readily available os library of python to read and write our excel sheet.

The face recognition framework uses HOG algorithm for face detection. Dlib library is used for finding landmarks. Main landmarks in face are found to figure out pose. Then using the landmarks image is warped in a way that eyes and mouth are centred. The cantered image is passed through a neural network to get 128 measurements. Based on these measurements images are compared to find most similar image.

1.

**2. Rationale**

Taking attendance in institutions like school and collages manually can be tedious and time consuming. It is specially tough when there are a large number of students in any school or a collage. The system of manual entry for the students faces issue of wastage of time and also becomes complicated on a large scale.

Whenever we have to measure the performance of students, finding and calculating the average of each enrolled student is also very complicated task. The human effort is more here. The retrieval of information is not easy as records are maintained in a handwritten register. This existing system requires correct feed from input into respective field. Therefore we are in need for an automated system for making and maintaining the attendance of students.

Traditional student attendance marking technique is facing a lot of trouble. Face recognition attendance system simplify the overly complicated tradition student attendance system. The classical student attendance marking system like calling student names or checking identification card of each student not only disturbs the teaching process but also causes distraction to students during exam sessions. Apart from calling names, student attendance sheet is usually passed around the classroom during lecture sessions. The lecture, specially if the class has a very large number of students, is disturbed in the process and many times some attendances are not marked and some students who do not appear in class can mark their attendance with help of a friend who is attending the class. The system is prone to many human errors and these errors are not easily discovered and fixed. Thus face recognition is proposed to replace the manual signing of replace the burdensome and error-prone task of manual signing of attendance by students. The face recognition attendance system will also help to eliminate the fraudulent signing of attendance of a student by some other student. It will also ensure that if a student have attended a class his attendance is not missed because of some error.

It is needed to develop a real time operating student 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 change in background, illumination and expression. High accuracy and fast computation time will be the evaluation points of the performance.

The identification process in the project is done in seconds. The extracted features to determine the identity of student are accurate as it is done by processing landmarks and comparing measure of different features of student on camera and photo of students in database. The model trained to consistent with many scenarios (changes in background, illumination etc.). The model if highly accurate. It is 98% accurate which is very close to human level accuracy in distinguishing between images. We have simple excel sheet to mark attendance and records relates to it. Which makes it simple to access and use by school and collage staff with varying level of qualifications in many different fields. The face recognition attendance system will help education institute and many other organisations to be more efficient and effective by automation tedious and time consuming work.

2

**3. Objectives**

* The purpose of this project is to simplify the ordeal of taking and maintaining the attendance record of students in a school or college with large with large number of students by automating the whole process. It will also eliminate many human errors that can happen in the process as machines are less prone to error. It will save substantial amount of time and eliminate extra work done by collage staffs.
* The project will help students by reducing unnecessary distractions during exam sessions and prevent fraudulent signing of attendance sheets. It will avoid disruptions caused during lectures due to passing of attendance sheet and time wasted on taking attendance orally by teacher. Lectures will be able to devote their full time to lectures instead of wasting valuable class time on taking attendance.
* The project tries to solve many problems that occurs when using face recognition model for taking attendance of students. The project is made keeping in mind that background of images captured through camera may vary and there can be multiple students in the camera. The system will mark attendance students whose image are in database of organization and is able to distinguish a person in database from a person who is not in database. So, it is less likely to be erroneous.
* The objective of project is also to minimize the cost by using softwares which are already or freely available to school or college. The only cost in this project is only installation of cameras which can be scaled according to comfort of organization. It can easily be managed by collage staff of any qualification. The attendance is automatically entered into excel sheet and lectures and provides flexibility to be edited if required. It provides valuable service to college staffs, lecturers and students without need of significant investment by organization.
* The overall objective of project is to help administration of an organisation to be more efficient and effective by automating the necessary but time-consuming work of taking attendance and maintaining a record of attendance.

3

**4. Literature Review**

**4.1 Project Background**

In the face detection and recognition system, the process flow is initiated by being able to detect the facial features from a camera or a picture store in a memory. The algorithm processes the image captured and identifies the number of faces in the image by analysing from the learned pattern and compare them to filter out the rest. This image processing uses multiple algorithm that takes facial features and compare them with known database.

The motivation behind this project is to simplify the means by which attendance is taken during lectures and how much time it takes. The use of ID cards or manually calling out attendance and writing it down on sheets is not productive and efficient. This system will detect the number of faces on the class and will also identify them from the store database. With the face detection and recognition system in place, it will be easy to tell if a student is actually present in the classroom or not.

**4.2 Previous Work**

**Project-1**

This is a project done by students as a final year project at Kingston University London in 2018.

The system will be presented an image either via camera or from memory and it must detect the number of faces on it automatically. After identifying faces, the system should crop the faces from the image and store them in memory for image recognition which will be done in the second step. The system should be able to automatically count the number of faces detected on the image.

The second step will be the recognition part where the system will be able to match faces from the stored dataset and compare it to the input data from the first step. A software will be used for this system which automatically sorts out the faces. The software will be inter-active so to facilitate interaction between multiple tasks as required. Because the system has two steps, the second phase of the system will involve the training of images on a dataset that are to be used for recognition.

**Technology Used**

The key algorithms are Viola-Jones for face detection and Hidden Markov Model with SVD.

* The implementation of The Viola-Jones algorithm is available on softwares like MATLAB, OpenCV and Web Browsers (using adobe flash).
* The existing implementation of the Hidden Markov Model with SVD for face recognition are available on MATLAB, C++ and OpenCV libraries.

4

**Project – 2**

This is a project done by students as a final year project at University of Nairobi in 2012.

The system will comprise of two modules. The first module also known as face detector is a mobile component, which is basically a camera that captures student faces and stores them in a file using computer vision face detection algorithms and face extraction techniques. The second module is a desktop application that does face recognition of the captured images (faces) in the file, marks the students register and then stores the results in a database for future analysis.

**Technology Used**

The following tools will be used in the implementation of the designed system. They’ve been divided in to two categories; Mobile and Desktop tools.

* **Mobile Tools:-** The face detection module will use OpenCV library for implementation by use of the frontal Haar Cascade face detector in either Android studio.

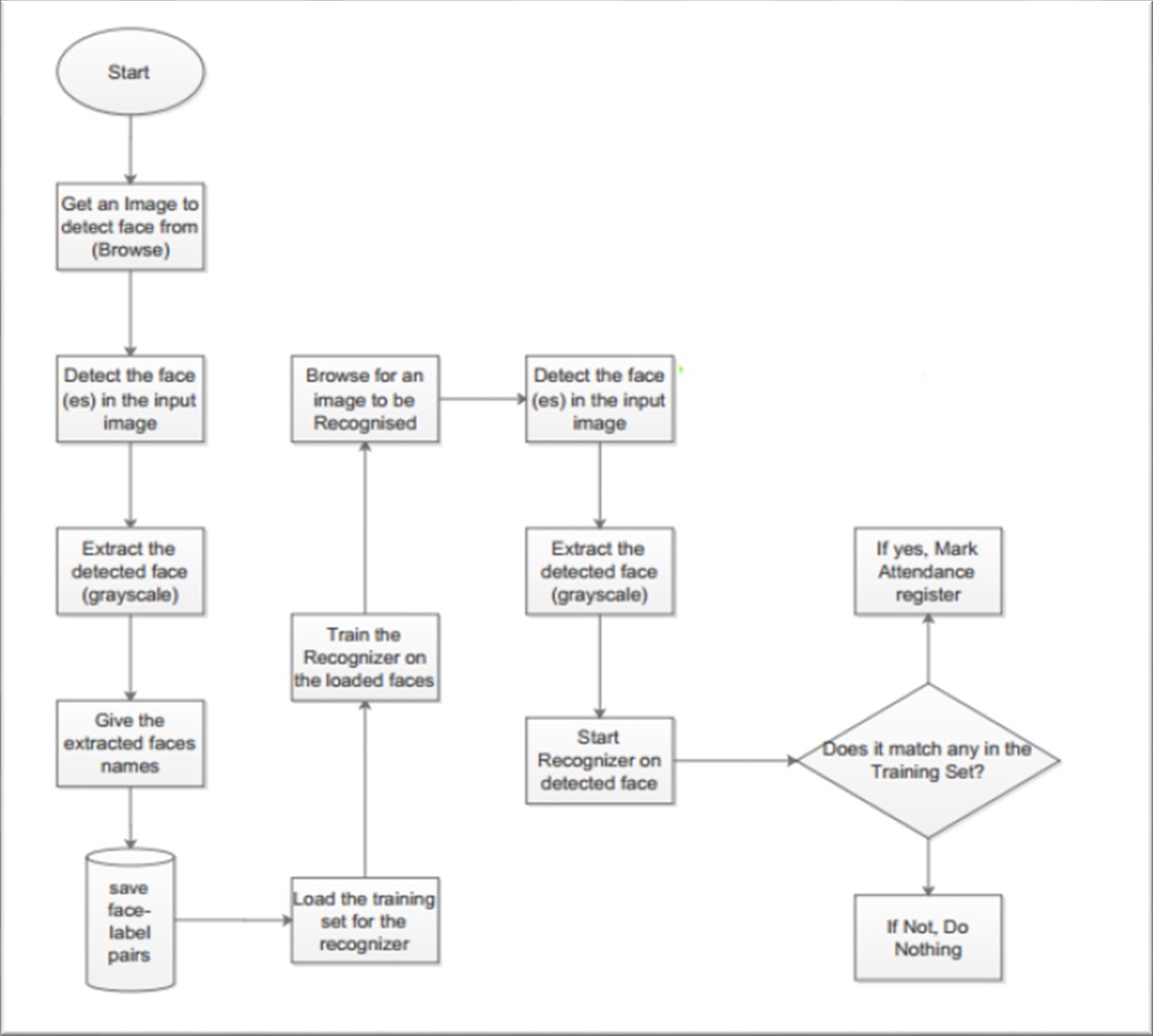
**OpenCV for Android Library** - (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision.

**Android Studio/ Eclipse IDE** - Android Studio is the official IDE for Android application development, based on IntelliJ IDEA.

* **Desktop Tools**:- **EmguCV Library -** EmguCV is a cross platform .Net wrapper to the OpenCV image processing library. OpenCV/EmguCV uses a type of face detector called a Haar Cascade. The Haar Cascade is a classifier (detector) trained on thousands of human faces.

**Visual Studio -** Visual Studio is able to build and run the solution examples after a proper configuration of EmguCV. The desktop software will implement the two sub-systems (Training set manager and Face recognizer) together with face detector in windows form

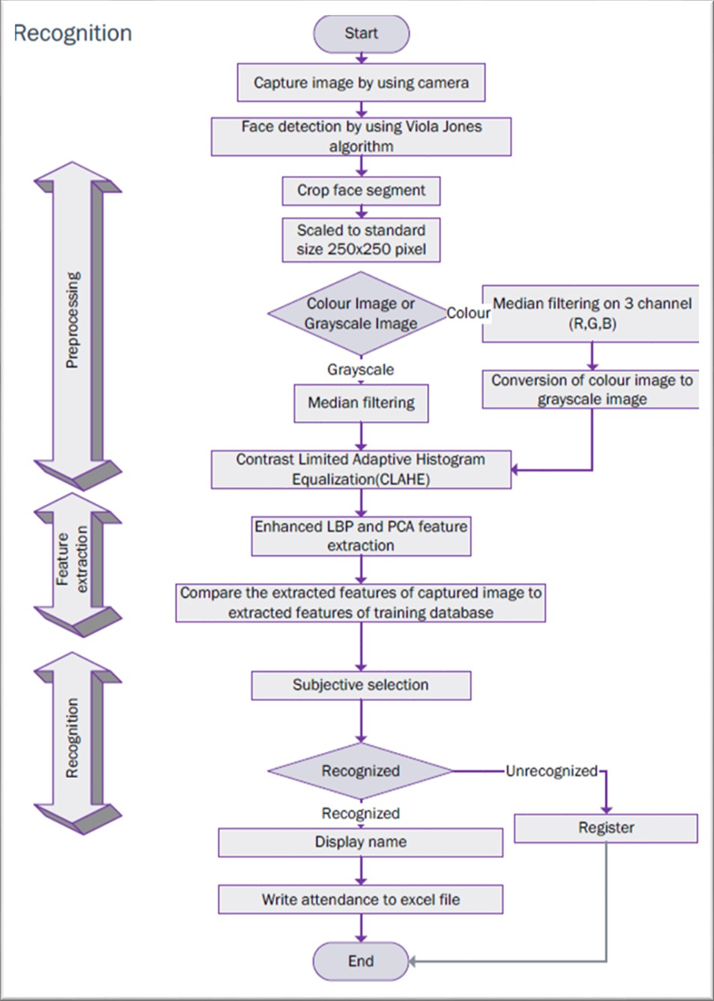
Following flowchart explains flow of information throughout process

 5.

**Project – 3**

This is a project done by students as a final year project at Universiti Tunku in 2018

The approach performs face recognition-based student attendance system. This method is also similar to others and begins with the input of an image either loaded from memory or from camera. Then it pre-processes the facial features and extracts it followed by subjective selecting and then the recognition of the facial images from known database. Both LBP and PCA feature extraction methods are studied in detail and computed in this approach in order to make comparisons. LBP is enhanced in this approach to reduce the illumination effect. An algorithm to combine enhanced LBP and PCA is also designed for subjective selection in order to increase the accuracy.



The project is completely built in MATLAB with OpenCV libraries implemented in it.

6.

**Project – 4**

This project was made by students of Institute of Engineering and Technology, Lucknow.

The project is LBPH algorithm. In this project first the images are collected from database and human face is detected among many objects using Haar Cascade. Haar Cascade is a pretrained classifier available in OpenCV library of python. Other classifiers like Local Binary Pattern and Principal Component analysis is also pretrained and readily available in OpenCV library of python language. Haar Cascade and LBP classifiers are used in yhis project to detect and identify human faces.

The project is divided into two parts: -

1. **Face Identification: -** Given a face image that belongs to a person in a database and we need to tell whose image it is or specifically recognize a face in an image and give decision whether the face is correctly recognize or not.
2. **Face Verification: -** Given Face image that might not belong to database and we need to authenticate whether a correct face is subjected to the database or not.

The flow chart explains how the project works:-



7.

**5. Feasibility Study**

**5.1 Operational Feasibility**

This system can be implemented in organization because there is adequate support from user and management. The results obtained by system will be affected only when a student goes through significant change in appearance. However new photo of student can be taken and uploaded in database. It is being developed in python so necessary actions are carried out automatically.

**5.2 Technical feasibility**

The necessary technology to implement this project exists and is easily available. And all the equipments have technical capacity to be used within the system. It is guaranteed that the almost all the results will be reliable with minimal error chance the accuracy of algorithm used in the project is 98%. The project allows flexibility in use of data obtained by deploying the project.

This project is being developed in intel i5 11th generation processor with the RAM of 8 GB. The environment required for deployment of project is any windows platform higher than XP. The language used in the project is Python version 3.10.4.

**5.3 Financial Feasibility**

The system developed and installed will be good benefit to organization. The system will be developed and operated in python 3.10.4 with OpenCV, facerecognition and numpy libraries. All the other libraries are already available in python. The facerecognition, numpy and OpenCV libraries are open source and free. Python language interpreter can also be downloaded for free from its official website. The only cost will be installation of CCTV cameras which can be scaled according to wish and budget of the organization.

8.

**6. Methodology**

The project is implemented in five steps: -

1. Finding all the faces in camera.
2. Posing and projecting faces.
3. Encoding faces.
4. Finding person’s name from encoding in our database.
5. Updating the attendance sheet.

**Step 1 Finding all the faces in camera.**

First, we need to detect all the faces in camera before telling them apart.

We are going to use a method called histogram of oriented gradients or HOG in short.

**Histogram of Oriented Gradients**

Histogram of Oriented Gradients, also known as HOG, is a feature descriptor like the Canny Edge Detector, SIFT (Scale Invariant and Feature Transform). It is used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in the localized portion of an image. This method is quite similar to Edge Orientation Histograms and Scale Invariant a Feature Transformation (SIFT). The HOG descriptor focuses on the structure or the shape of an object. It is better than any edge descriptor as it uses magnitude as well as angle of the gradient to compute the features. For the regions of the image, it generates histograms using the magnitude and orientations of the gradient.

**Steps to calculate HOG Features**

1. Take the input image you want to calculate HOG features of. Resize the image into an image of 128x64 pixels (128 pixels height and 64 width). This dimension was used in the paper and was suggested by the authors as their primary aim with this type of detection was to obtain better results on the task of pedestrian detection. As the authors of this paper were obtaining exceptionally perfect results on the MIT pedestrian database, they decided to produce a new and significantly more challenging dataset called the ‘INRIA’ dataset (http://pascal.inrialpes.fr/data/human/), containing 1805 (128x64) images of humans cropped from a varied set of personal photos.

9.

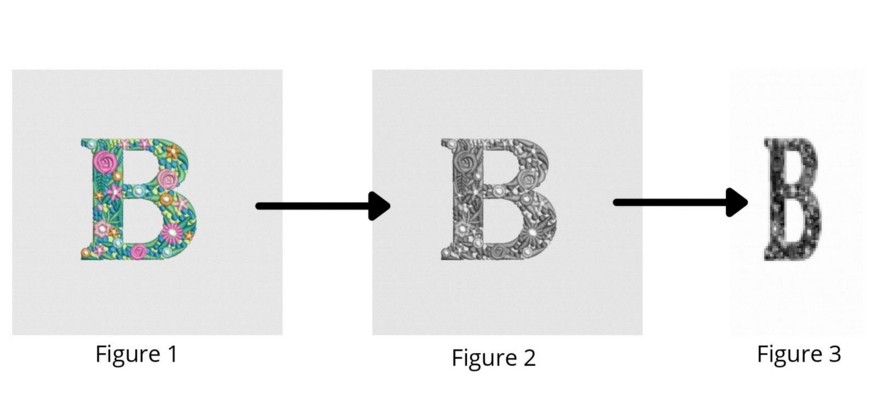


Figure 1: The image imported to get HOG features of.

Figure 2: The imported image grayscale for the process.

Figure 3: Resized and grayscale image of the imported image.

2. The gradient of the image is calculated. The gradient is obtained by combining magnitude and angle from the image. Considering a block of 3x3 pixels, first Gx and Gy is calculated for each pixel. First Gx and Gy is calculated using the formulae below for each pixel value .



where r, c refer to rows and columns respectively.

After calculating Gx and , magnitude and angle of each pixel is calculated using the formulae mentioned below.



10.

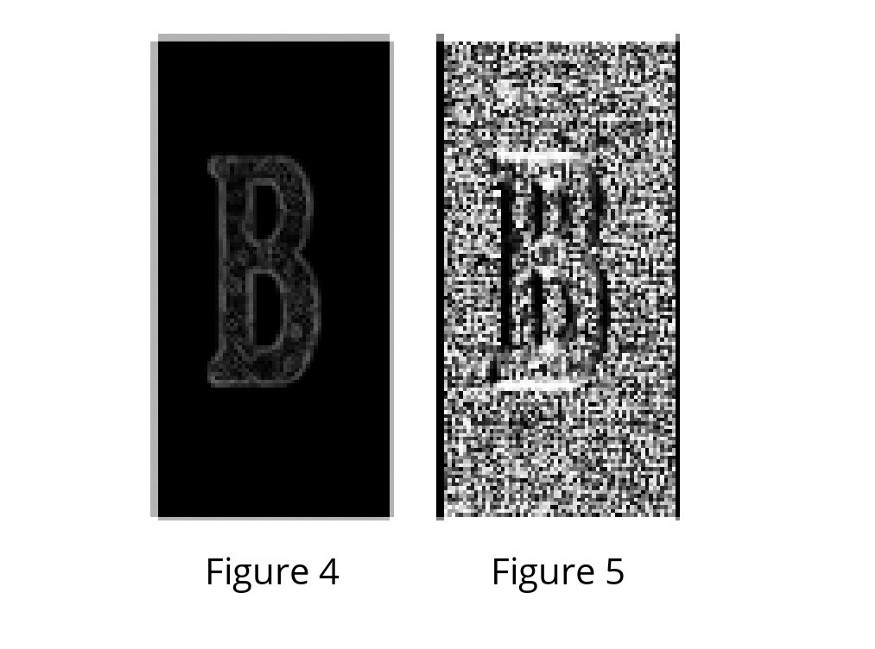


Figure 4: Visualization of magnitude of the image.

Figure 5: Visualization of angle of the image.

3. After obtaining the gradient of each pixel, the gradient matrices (magnitude and angle matrix) are divided into 8x8 cells to form a block. For each block, a 9-point histogram is calculated. A 9-point histogram develops a histogram with 9 bins and each bin has an angle range of 20 degrees. Figure 8 represents a 9 bin histogram in which the values are allocated after calculations. Each of these 9-point histograms can be plotted as histograms with bins outputting the intensity of the gradient in that bin. As a block contains 64 different values, for all 64 values of magnitude and gradient the following calculation is performed. As we are using 9 point histograms, hence :



Each Jth bin, bin will have boundaries from:



Value of the centre of each bin will be:



11.

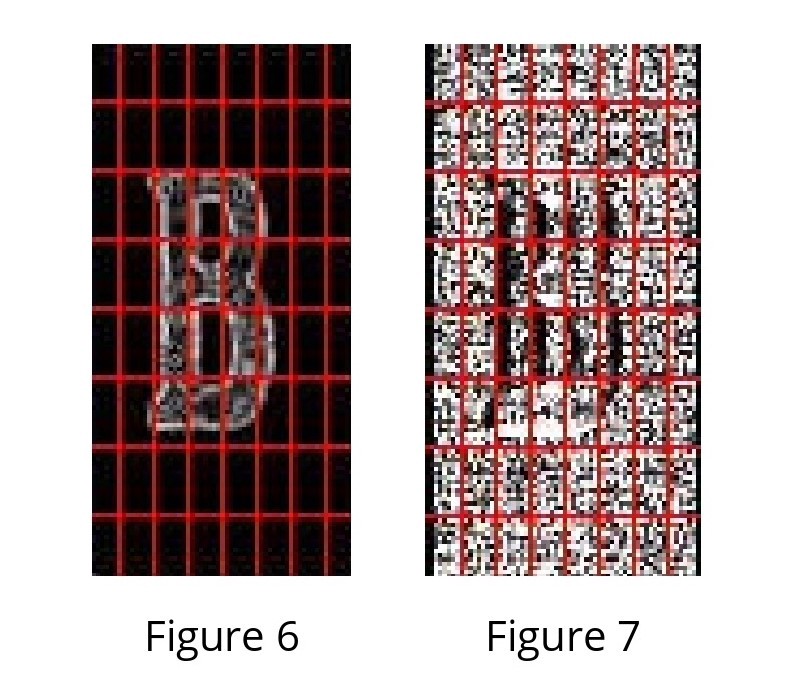


Figure 6: 8x8 blocks on the magnitude image.

Figure 7 : 8x8 blocks on an angle image.

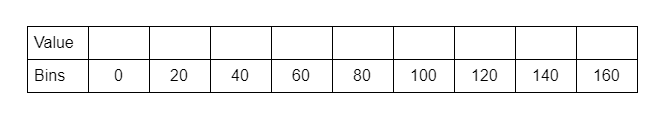
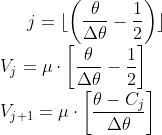


Figure 8: Representation of a 9 bin histogram. This one single histogram will be unique for one 8x8 block made up of 64 cells. All 64 cells will add their Vj and Vj+1 value to the jth and (j+1)th index of the array respectively. (Image by author)

4. For each cell in a block, we will first calculate the jth bin and then the value that will be provided to the jth and (j+1)th bin respectively. The value is given by the following formulae:

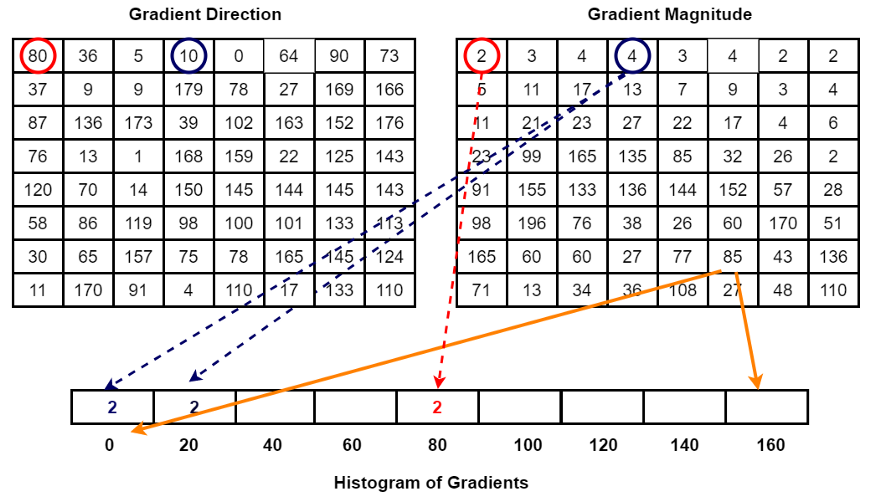
12.



5. An array is taken as a bin for a block and values of Vj and Vj+1 is appended in the array at the index of jth and (j+1)th bin calculated for each pixel.

6. The resultant matrix after the above calculations will have the shape of 16x8x9.

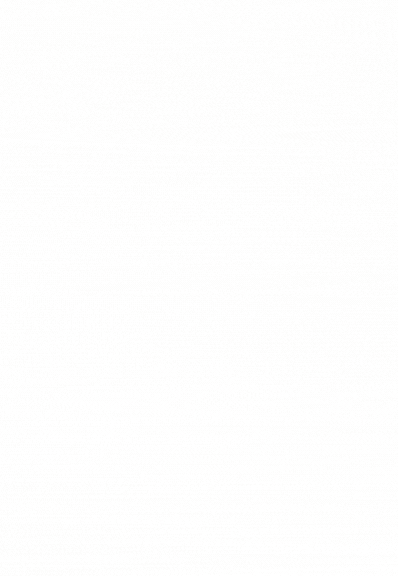
7. Once histogram computation is over for all blocks, 4 blocks from the 9 point histogram matrix are clubbed together to form a new block (2x2). This clubbing is done in an overlapping manner with a stride of 8 pixels. For all 4 cells in a block, we concatenate all the 9 point histograms for each constituent cell to form a 36 feature vector.



Method for calculation of 9 bin histograms is illustrated in the above image. Inspired by [<https://www.programmersought.com/article/42554276349/>]



13.



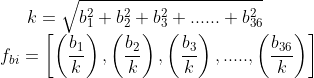
Traversing of 2x2 grid box around the image in order to make a combined fbi from 4 blocks. (Image by author)

8. Values of fb for each block is normalized by the L2 norm :



Where ε is a small value added to the square of fb in order to avoid zero division error. In code value taken of is 1e-05. (Image by author)

9. To normalize, the value of k is first calculated by the following formulae :



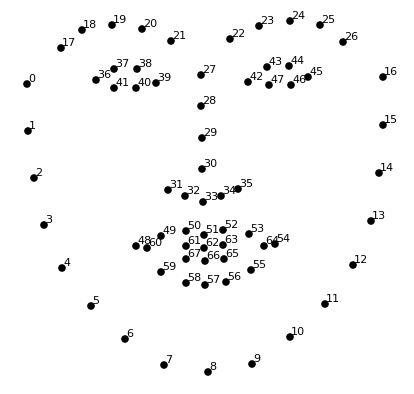
13.

10. This normalization is done to reduce the effect of changes in contrast between images of the same object. From each block. A 36 point feature vector is collected. In the horizontal direction there are 7 blocks and in the vertical direction there are 15 blocks. So the total length of HOG features will be : 7 x 15 x 36 = 3780. HOG features of the selected image are obtained.

**Step 2 Posing and Projecting Faces.**

Now we have to deal with the problem that faces turned different directions look totally different to a computer. To account for this, we will try to warp each picture so that the eyes and lips are always in the sample place in the image. This will make it a lot easier for us to compare faces in the next steps. To do this, we are going to 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:



Now that we know where the eyes and mouth are, we’ll simply rotate, scale and shear the image so that the eyes and mouth are centered as best as possible. We won’t do any fancy 3d warps because that would introduce distortions into the image. We are only going to use basic image transformations like rotation and scale that preserve parallel lines.

14.

**Step 3 Encoding Faces.**

Now we are to the meat of the problem — actually telling faces apart.

The simplest approach to face recognition is to directly compare the unknown face we found in Step 2 with all the pictures we have of people in database. There’s actually a huge problem with that approach. It is a very time-consuming process and we want to give output as soon as possible. What we need is a way to extract a few basic measurements from each face. Then we could measure our unknown face the same way and find the known face with the closest measurements.

The face detection framework has used OpenFace’s Deep Convolutional Neural Networks 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.

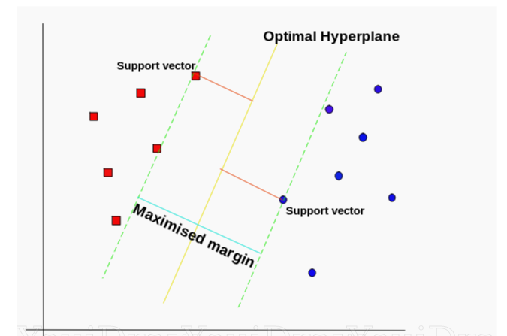
**Step 4 Finding a Person’s name from encoding.**

Now we have to find the person in our database of known people who has the closest measurements to our test image**.** We have used SVM classifier for it.

**SVM Classifier.**

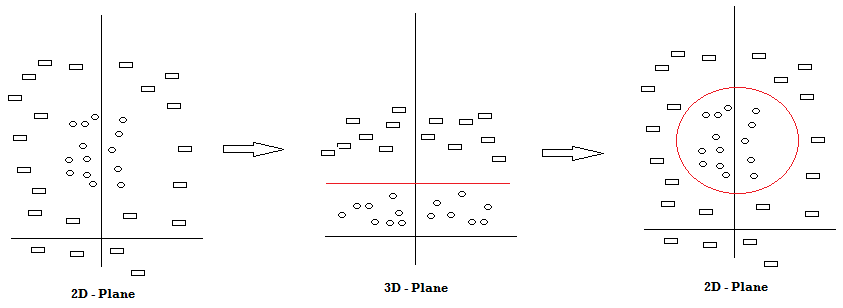
The Support Vector Machine (SVM) Classification is similar to the SVR. In SVM, the line that is used to separate the classes is referred to as **hyperplane**. The data points on either side of the hyperplane that are closest to the hyperplane are called **Support Vectors** which is used to plot the boundary line.

15.



In SVM Classification, the data can be either linear or non-linear. There are different kernels that can be set in an SVM Classifier. For a linear dataset, we can set the kernel as ‘*linear*’.

On the other hand, for a non-linear dataset, there are two kernels, namely ‘*rbf*’ and ‘*polynomial*’. In this, the data is mapped to a higher dimension which makes it easier to draw the hyperplane. Afterwards, it is brought down to the lower dimension.



SVM Mechanism (Source — Self)

From the above diagram, we can see that there are two classes of shapes, rectangle and circle. As it is difficult to draw a SVM line in the 2D Plane, we map the data points to a higher dimension (3D Plane) and then draw the hyperplane. It is then brought down to the original plane with the SVM Classifier drawn in red colour.

In this way, the SVM Classifier can be used to classify a data point to which class it belongs from the given dataset. Let us use this algorithm to solve a real-world problem.

16.

**Step 5 Updating the Database.**

This is the easiest step of the process in tis step we have used python to insert name of student in our excel sheet when they appear on CCTV camera. It has been taken into account that a student can come in CCTV camera multiple times. The excel sheet will be updated for each student only once. Here is the code of function used for it:-

def markAttendence(name):

    with open("C:\\Users\\DELL\\Desktop\\ML Project 7th semester\\face recognition attendance\\attendance.csv",'r+') as f:

        myDataList = f.readlines()

        nameList = []

        for line in myDataList:

            entry = line.split(',')

            nameList.append(entry[0])

        if name not in nameList:

            now = datetime.now()

            dtstring = now.strftime('%H:%M:%S')

            f.writelines(f'\n{name},{dtstring}')

17.

**7. Facilities required for proposed work.**

The project requires following hardware:

1. CCTV cameras.
2. A system with intel i5 processor, at least 4 GB RAM, Windows 7 or above.

The project has following software requirements:

1. Python at least version 3.8.
2. Face\_recognition library.
3. OpenCV library.
4. Dlib for using face\_recognition library.
5. Numpy latest version.
6. MS excel.

18.

**8. Expected outcomes.**

The expected outcomes of the project are: -

* Reduction in time spent in attendance marking.
* Reduction in human error done during attendance.
* More accurate attendance.
* Ease in analysing performance of students and ease in finding the average of student attendance.
* Less distractions during exam sessions.
* Judging the faces of students in database even if there are multiple faces of many people who are not in database.
* Giving same performance in all circumstances like different background and different lighting.
* Reducing fraud attendances.

19.

**9. References**

The project has been made after reading following articals.

* Machine learning is fun part 4: Modern face recognition with deep learning.<https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78>
* Machine learning basics: Support Vector Machine(SVM) classifier.<https://towardsdatascience.com/machine-learning-basics-support-vector-machine-svm-classification-205ecd28a09d>
* HOG(Histogram of Oriented Gradients): An overview.<https://towardsdatascience.com/hog-histogram-of-oriented-gradients-67ecd887675f>

20.

End