

Facial Recognition for Surveillance

Team Argus

November 2022



Facial Recognition for Surveillance

**Project report in partial fulfillment of the requirement for the
award of the degree of Bachelor of Technology**

In

Computer Science and Engineering (I.O.T)

submitted by

Deeptangshu Saha	Enrollment No. 12021002029109
Bhaskar Das	Enrollment No. 12021002029110
Nilay Mukherjee	Enrollment No. 12021002029112
Himadri Dutta	Enrollment No. 12021002029104
Ankit Kumar	Enrollment No. 12021002029111
Shubhayu Mandal	Enrollment No. 12021002029154
Rupam Ghosh	Enrollment No. 12021002029113
Shantanu Bera	Enrollment No. 12021002029141
Naimur Hossain Sk	Enrollment No. 12021002029080
Sampad Narayan Bhattacharjee	Enrollment No. 12021002029106

Under the guidance of

Prof. Sandip Mandal

&

Prof. Debaleena Datta

Department of Computer Science and Engineering



UNIVERSITY OF ENGINEERING & MANAGEMENT, KOLKATA
University Area, Plot No. III – B/5, New Town, Action Area – III, Kolkata –
700160.

certificate

This is to certify that the project titled Facial Recognition For Surveillance submitted by Deeptangshu Saha (University Roll No. 17 Sec. C), Bhaskar Das (University Roll No. ,Sec A), Nilay Mukherjee (University Roll No. ,Sec A), Himadri Dutta (University Roll No.), Ankit Kumar (University Roll No.), Shubhayu Mandal (University Roll No.), Rupam Ghosh (University Roll No. ,Sec B), Shantanu Bera (University Roll No.), Naimur Hossain Sk (University Roll No.), and Sampad Narayan Bhattacharjee (University of Roll No.) students of UNIVERSITY OF ENGINEERING & MANAGEMENT, KOLKATA, in partial fulfillment of requirement for the degree of Bachelor of Computer Science and Engineering, is a bonafide work carried out by them under the supervision and guidance of Prof. Sandip Mandal & Prof. Debaleena Datta during 3rd Semester of academic session of 2022 - 2023. The content of this report has not been submitted to any other university or institute. I am glad to inform that the work is entirely original and its performance is found to be quite satisfactory.

Signature of Guide

Signature of Guide

Signature of Head of the Department

Acknowledgement

We would like to take this opportunity to thank everyone whose cooperation and encouragement throughout the ongoing course of this project remains invaluable to us. We are sincerely grateful to our guide Prof. Sandip Mandal and Prof. Debaleena Datta of the Department of Computer Science and Engineering, UEM, Kolkata, for their wisdom, guidance and inspiration that helped us to go through with this project and take it to where it stands now.

Last but not the least, we would like to extend our warm regards to our families and peers who have kept supporting us and always had faith in our work.

Deeptangshu Saha
Bhaskar Das
Nilay Mukherjee
Himadri Dutta
Ankit Kumar
Shubhayu Mandal
Rupam Ghosh
Shantanu Bera
Naimur Hossain Sk
Sampad Narayan Bhattacharjee

Contents

1	Abstract	6
2	Introduction	7
3	Literature Survey	8
3.1	Eigen faces algorithm with PCA	8
3.1.1	Theory	8
3.1.2	Drawbacks	8
3.2	Haar-Cascade Classifiers Viola Jones	8
3.2.1	Theory	8
3.2.2	Drawbacks	9
3.3	Support Vector Machine (SVM)	9
3.3.1	Theory	9
3.3.2	Drawbacks	9
4	Problem Statement	11
5	Proposed Solution	12
6	Experiment Setup & Result Analysis	13
7	Conclusion & Future Scope	15
	References	16

1 Abstract

The facial Recognition is a vast field and to be on top of it one has to master the algorithms. In this project we master one of those algorithms.

2 Introduction

Facial Recognition is a biometric security measure which is used a lot by both regular people as well as some of the most important and secure places in the world. This raises the question that computers which is used to make us humans' life easier can do this work for us? Now this is a question which many different researchers tried to answer. This how all the different facial recognition algorithms came to be.

The automatic facial recognition system involves the application of an intelligent artificial system to recognise the human faces under any circumstances. Today the study of facial recognition has involved a keen interest in pattern recognition, computer vision and other related fields. Camera is the only device for face recognition system. Face recognition provides an inexpensive and reliable personal identification which is applicable in many fields [1].

Other factors that affect the accuracy are face shape, texture, specs, hair, illumination etc. There are several external uncontrollable factors that affect the accuracy of the image recognition system. However there is publication of face recognition algorithms which reveals that each algorithm has certain characteristics and provides good accuracy in different aspects. This research studies about some well know face recognition algorithms and makes a comparison of their recognition accuracies both on train and test set. Eigen faces, SVM, KNN and CNN are chosen in this experimental study. A variation of face viewpoints is the factor that has been used in the experiment to study the effect of recognition accuracy. In this way the advantages and disadvantages of different algorithms can be studied. Consequently it will help the developers to choose the best facial recognition algorithm in their field of implementation[2].

3 Literature Survey

Some of the facial recognition systems are

- Eigen faces algorithm with PCA
- Haar-Cascade Classifiers Viola Jones
- Support Vector Machine (SVM)

3.1 Eigen faces algorithm with PCA

3.1.1 Theory

Eigenfaces are a set of eigenvectors used in the computer vision problem of human face recognition. Eigenfaces assume ghastry appearance. They refer to an appearance-based approach to face recognition that seeks to capture the variation in a collection of face images and use this information to encode and compare images of individual faces in a holistic manner. Specifically, the eigenfaces are the principal components of a distribution of faces, or equivalently, the eigenvectors of the covariance matrix of the set of face images, where an image with $N \times N$ pixels is considered a point (or vector) in N^2 -dimensional space. The idea of using principal components to represent human faces was developed by Sirovich and Kirby[3].

3.1.2 Drawbacks

The tests conducted on various subjects in different environments shows that this approach has limitations over the variations in light, size and in the head orientation, nevertheless this method showed very good classifications of faces(85% success rate). A good recognition system should have the ability to adapt over time. Reasoning about images in face space provides a means to learn and subsequently recognize new faces in an unsupervised manner. When an image is sufficiently close to face-space (i.e., it is face-like) but is not classified as one of the familiar faces, it is initially labeled as "unknown" . The computer stores the pattern vector and the corresponding unknown image. If a collection of "unknown" pattern vectors cluster in the pattern space, the presence of a new but unidentified face is postulated. A noisy image or partially occluded face would cause recognition performance to degrade. The eigenface approach does provide a practical solution that is well fitted to the problem of face recognition. It is fast, relatively simple, and has been shown to work well in constrained environment[4].

3.2 Haar-Cascade Classifiers Viola Jones

3.2.1 Theory

Motivated by the challenge of face detection,P. Viola and M. Jones proposed an object detector framework using Haar-like features, which has been widely used by other works not only for face detection, but also for object locations. Thanks to the Open Computer Vision Library implementation [17], the general object detector framework has become popular and motivated the community

to generate their own object classifiers. These classifiers use haar-like features that are applied over the image. Only those image regions, called sub-windows, that pass through all the stages of the detector are considered to contain the target object. Fig. 3 shows the detection cascade schematic with N stages. The detection cascade is designed to eliminate a large number of negative examples with a little processing.[5]

3.2.2 Drawbacks

Based on the experiment before the system can classify and detect the face in many cases and conditions. With six types of condition and five times of trial, obtain the accuracy by 75,33% by using V-J and 80,22% by using HOG. V-J algorithm can detect frontal face very well in images, regarding of their scale, pose, makeup, expression, and illumination, but rather difficult to detect the face who have occlusions like using helm, eyeglass, and mask. The V-J algorithm can perform in real-time on many applications and hardware, the main problem with Haar cascades is in the parameter called detect multiscale and scale factor. If the scale factor is too low, many pyramid layers will be evaluated, this will help to detect more than one faces in images, but the detection process will be slower and increases the false-positive detection rate. On the other hand, if scale factor is too large, it cannot detect the face in small pixel. The recommended size for datasets at least above 250*250 pixels. The HOGs more accurate than V-J for face detection, it can represent local appearance very well[6].

3.3 Support Vector Machine (SVM)

3.3.1 Theory

Support vector machine is a supervised learning model with associated learning algorithms, it analyses data for classification and regression analysis. SVM training algorithm differentiates categories making it non probabilistic binary linear classifier. SVM separates categories by a clear gap as wide as possible also known as margin. SVM also performs non-linear classification using Kernel trick. SVM is capable of delivering higher classification accuracy. SVM can be used to detect text, digit, image classification and object detection. SVM constructs hyper-plane between 2 or more clusters; hyper-plane can be used to detect outliers among data. To achieve optimal parameter setting SVM requires extensive cross validation commonly known as model selection. The choice of a kernel function, the standard deviation of the Gaussian kernel, training data, relative weights of slack variable impacts the overall results. SVM minimizes the empirical classification error and maximizes the geometric margin. SVM is based on Structural Risk Minimization (SRM). SVM maps input vector to a higher dimensional space with maximal separating hyper plane.[2]

3.3.2 Drawbacks

In the previous discussion, we learned that the superiority of SVM made it a significant development in the fields of modern machining, predicting protein and face detection. We also learned that the model of SVM is only determined only by the support vectors, so it is very convenient for us to train. But SVM also has some drawbacks[6], for instance, the training efficiency of the existing

SVM for large-scale sample data sets of practical problems cannot reach the ideal training efficiency. Therefore the future development of SVM may concentrate more on how to further improve the SVM algorithm. In addition, despite the fact that SVM has outstanding advantages in theory, compared with theoretical research, applied research on application still lags behind. Therefore, how to apply SVM more in people's daily life and explore new application areas of SVM will be the emphasis of future research.^[7]

4 Problem Statement

It is clear from the title that we are going to use facial recognition in helping us to conduct surveillance. Now 'Surveillance' is a rather sensitive topic because people always co-relate surveillance with unlawful monitoring as well as breach of privacy. So for now in this document by surveillance we will mean strictly the public streets where CCTV for monitoring purposes are accepted by all as lawful and ethical surveillance. We are going to use our product 'Sharingan' duly meaning the 'copy eye' as it's in the initial stages we are using it to identify the limited people in the database thus just copying the face and matching it with the database to identify whether they are in the database and identify if present in the database.

5 Proposed Solution

We are using the OpenCV library in python and using it for both computer vision as well as facial recognition. It uses CNN for training its models and recognizing faces. We used the CNN algorithm as because the size of the database as well as the speed of the algorithm for less data is quite high which in turn speeds up the process by a lot. We also made sure to lessen the time taken by creating pre encode pictures so that it does not do it everytime the program runs.

6 Experiment Setup & Result Analysis

```
import cv2
import face_recognition
import numpy as np
# from simple_facerec import SimpleFacerec

# Encode faces from a folder
# sfr = SimpleFacerec()
# sfr.load_encoding_images("images/")

frame_resizing = 0.25

def detect_known_faces(frame):
    small_frame = cv2.resize(frame, (0, 0), fx=
        frame_resizing, fy=frame_resizing)
    # Find all the faces and face encodings in the
    # current frame of video
    # Convert the image from BGR color (which OpenCV uses
    # ) to RGB color (which face_recognition uses)
    rgb_small_frame = cv2.cvtColor(small_frame, cv2.
        COLOR_BGR2RGB)
    face_locations = face_recognition.face_locations(
        rgb_small_frame)
    face_encodings = face_recognition.face_encodings(
        rgb_small_frame, face_locations)
    known_face_encodings = np.load('ImageEncoding.npy')
    image_names = open("ImageNames.txt", "r")
    known_face_names = []
    for known_face_names_var in image_names:
        known_face_names.append(known_face_names_var)
    face_names = []

    for face_encoding in face_encodings:
        # See if the face is a match for the known face(s
        # )
        matches = face_recognition.compare_faces(
            known_face_encodings, face_encoding)
        name = "Unknown"

        ## If a match was found in known_face_encodings,
        # just use the first one.
        # if True in matches:
        #     first_match_index = matches.index(True)
        #     name = known_face_names[first_match_index]
        # Or instead, use the known face with the
        # smallest distance to the new face
```

```
        face_distances = face_recognition.face_distance(
            known_face_encodings, face_encoding)
        best_match_index = np.argmin(face_distances)
        if matches[best_match_index]:
            name = known_face_names[best_match_index]
            face_names.append(name[:-1])

    # Convert to numpy array to adjust coordinates with
    # frame resizing quickly
    face_locations = np.array(face_locations)
    face_locations = face_locations / frame_resizing
    return face_locations.astype(int), face_names

# Load Camera
cap = cv2.VideoCapture(0)

while True:
    ret, frame = cap.read()

    # Detect Faces
    face_locations, face_names = detect_known_faces(frame)
    for face_loc, name in zip(face_locations, face_names):
        :
        y1, x2, y2, x1 = face_loc[0], face_loc[1],
            face_loc[2], face_loc[3]

        cv2.putText(frame, name, (x1, y1 - 10), cv2.
            FONT_HERSHEY_DUPLEX, 1, (0, 0, 200), 2)
        cv2.rectangle(frame, (x1, y1), (x2, y2), (200, 0,
            0), 4)

    cv2.imshow("Frame", frame)

    key = cv2.waitKey(1)
    if key == 27:
        break
cap.release()
cv2.destroyAllWindows()
```

The above is the driver code and works as expected. Though the code works as expected it also identifies the faces which are in the database with an accuracy of more than 99%. The only drawback being that the face size needs to be of a certain size or it won't detect the face at all. However the current distance is not so ideal and thus it needs to be improved further.

7 Conclusion & Future Scope

The immediate improvements that are to be done is fixing the minimum distance issue as the ideal distance is not so practical in a real world situation. And further down the road we will be seeing not only an advanced facial recognition surveillance system but for other purposes such as crime identification as well as possible criminal intent predictions.

References

- [1] Manop Phankokkrud and Phichaya Jaturawat. Influence of facial expression and viewpoint variations on face recognition accuracy by different face recognition algorithms. In *2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pages 231–237. IEEE, 2017.
- [2] Sanmoy Paul and Sameer Kumar Acharya. A comparative study on facial recognition algorithms. In *e-journal-First Pan IIT International Management Conference-2018*, 2020.
- [3] Lawrence Sirovich and Michael Kirby. Low-dimensional procedure for the characterization of human faces. *Josa a*, 4(3):519–524, 1987.
- [4] Y Vijaya Lata, Chandra Kiran Bharadwaj Tungathurthi, H Ram Mohan Rao, A Govardhan, and LP Reddy. Facial recognition using eigenfaces by pca. *International Journal of Recent Trends in Engineering*, 1(1):587, 2009.
- [5] Rafael Padilla, CFF Costa Filho, and MGF Costa. Evaluation of haar cascade classifiers designed for face detection. *World Academy of Science, Engineering and Technology*, 64:362–365, 2012.
- [6] C Rahmad, R An Asmara, DRH Putra, I Dharma, H Darmono, and I Muhiqqin. Comparison of viola-jones haar cascade classifier and histogram of oriented gradients (hog) for face detection. In *IOP conference series: materials science and engineering*, volume 732, page 012038. IOP Publishing, 2020.
- [7] Zhao Jun. The development and application of support vector machine. In *Journal of Physics: Conference Series*, volume 1748, page 052006. IOP Publishing, 2021.