RESEARCH ARTICLE   OPEN ACCESS

Face Recognition Based Door Lock System Using Opencv and C# with Remote Access and Security Features

Prathamesh Timse[**],Pranav Aggarwal[**], Prakhar Sinha[**],Neel Vora[**] ,
 **Student, Department of Electronics and Telecommunication Engineering, K.J Somaiya College Of Engineering, Vidyanagar, Vidyavihar(E), Mumbai - 400 077, Maharashtra, India.

ABSTRACT
This paper investigates the accuracy and effectiveness of the face detection and recognition algorithms using OpenCV and C# computer language. The adaboost algorithm [2] is used for face detection and PCA algorithm[1] is used for face recognition. This paper also investigates the robustness of the face recognition system when an unknown person is being detected, wherein the system will send an email to the owner of the system using SMTP [7]. The door lock can also be accessed remotely from any part of the world by using a Dropbox [8] account.
Keywords – Recognition, Detection, OpenCV, C#, Adaboost [2], PCA [1], SMTP [7] , Dropbox [8] .

I.   INTRODUCTION

Human beings are recognized by their unique facial characteristics. In the face recognition approach, a given face is compared with the faces stored in the database in order to identify the person. The purpose is to find a face in the database, which has the highest similarity with the given face. In the field of biometrics, facial recognition technology is one of the fastest growing fields.

The recent interest in face recognition can be attributed to the increase of commercial interest and the development of feasible technologies to support the development of face recognition. Major areas of commercial interest include biometrics, law enforcement and surveillance, human-computer interaction, multimedia management (for example, automatic tagging of a particular individual within a collection of digital photographs) smart cards, passport check, Criminal investigations, access control.

However, face detection is more challenging because of some unstable characteristics, for example, glasses and beard will impact the detecting effectiveness. Moreover, different kinds and angles of lighting will make detecting face generate uneven brightness on the face, which will have influence on the detection process.

To overcome these problems, the system used adaboost algorithm [2] implemented using Haar classifiers for face detection and PCA algorithm [1] for face recognition implemented using face recognizer function of OpenCV.

Rest of the paper is organized as follows: Section II describes the proposed face recognition system. Face detection mechanism is explained in Section III. Section IV describes the face recognition mechanism, Section V states the Remote access and security features of the system, Section VI provides the results and observations and the last section gives the conclusion.

II.   SETUP OF FACE RECOGNITION SYSTEM
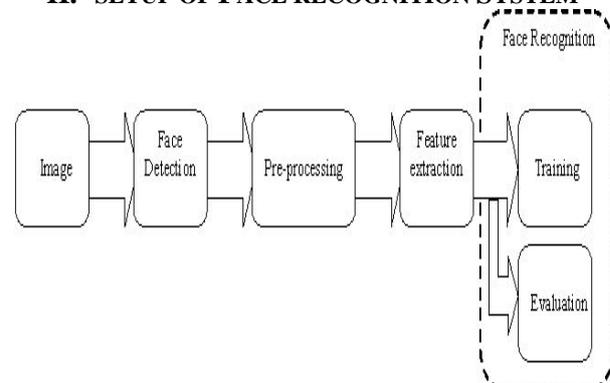
III. FACE DETECTION BY HAAR CASCADED CLASSIFIER USING VIOLA JONES METHOD

Viola and Jones devised an algorithm, called Haar Classifiers, to rapidly detect any object, including human faces, using Adaboost [2]classifier cascades that are based on Haar-like features and not pixels.[4]

Open CV uses Viola Jones method published in 2001, to detect faces using 4 key concepts

- Simple rectangular features called haar features
- An integral image for rapid face detection
- The adaboost machine learing method
- A cascaded classifier to combine many classifiers efficiently[6]

Extended Haar-like features are chosen to enhance detecting precision, which is divided into edge feature, linear feature and center surround feature. And the above features comprise feature module, in which there are two kinds of rectangle, white and black, as in Fig 1[3]
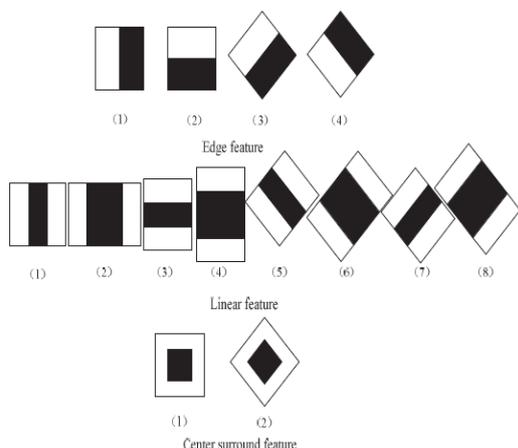

Fig 1. Extended Harr-like features

Integral image is a method used to quickly calculate the characteristic value: the difference between the sum of white pixels and the sum of black pixels. In Fig 2, it shows that the characteristic value composed by II and IV is the variation between the sum of IV pixels and the sum of II pixels. The former is the difference between the sum of integral image value of A and D and the sum of integral image value of B and C. The later is the sum of integral image value of B and integral image value of A.
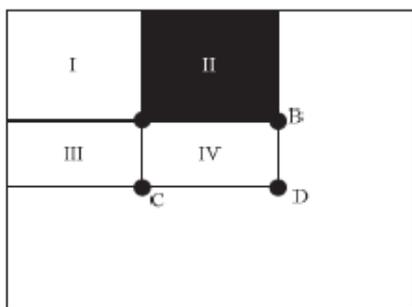

Fig 2. Four arrays figure

The cascaded classifier used by OpenCV is Haar cascade classifier which is trained on thousands of human faces. The training data is stored in an xml file which is later used by the classifier to detect faces. This paper uses the Haar cascade classifier haarcascade_frontalface_alt_tree.xml created by Rainer Lienhart.

Adaboost [2]is from high dimensional space and large amounts of data to train a strong classifier.

Strong classifier is composed by multiple weak classifiers, the bias direction (j p), the threshold (θ j) and characteristic function j f are consist of a weak classifier. The Binary weak classifier as follows:

$$h_j(x) = \begin{cases} 1 & p_j f_j(x) < p_j \theta_j \\ -1 & otherwise \end{cases}$$

The steps of Strong classifier training algorithm are as follows:

(1) About the samples (x1,y1) ,(x2,y2)…(xn,yn) and ,yi=-1 and yi=1 are corresponding to the counter-samples and positive-samples

(2)The Initialize the sample weights about yi=-1 and yi=1 are $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$,

m and l is the number about counter-samples and positive-samples;

(3) Iterate T times: t=1,2,…T :

1. Normalized the sample right $w_{t,i} = \dfrac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$;

2. The weight of error rate w_t for each characteristic to train a classifier h_j $\varepsilon_j = \sum_{i=1}^{n} w_i \left| h_j(x_i - y_i) \right|$;

3. Select the smallest error rate ε_t about the classifier h_t ;

4. Update the weight $w_{t+1,i} = w_{t,i}\beta_t^{1-e_i}$, if the sample x_i is classified correctly, thus, $e_i = 0$,contrarily, $e_i = 1$ .but $\beta_t = \dfrac{\varepsilon_t}{1 - \varepsilon_t}$.

(4) Training the strong classifier is:

$$h(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq 0 \\ -1 & otherwise \end{cases} \quad And \quad \alpha_t = \log\frac{1}{\beta_t}$$

Strong classifier can be achieved arbitrarily low error rate for training data. When the training samples are representative comparison, the strong classifier can be also achieved the ideal true error rate.[5]

The detection process was completed by us with an accuracy of 95%. The system could also detect multiple faces as is evident from the following figure:
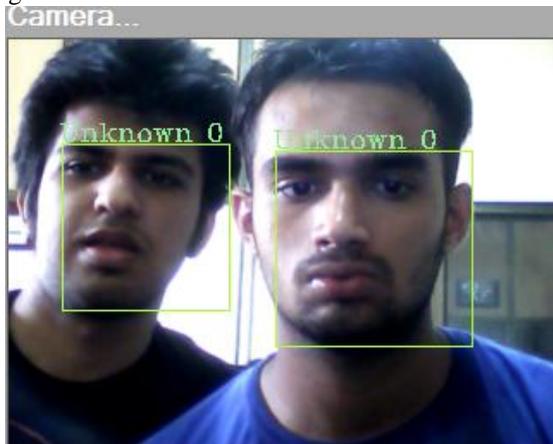
IV. FACE RECOGNITION

Open CV (Open Source Computer Vision) is a popular computer vision library. The Eigenfaces method takes a holistic approach to face recognition: A facial image is a point from a high-dimensional image space and a lower-dimensional representation is found, where classification becomes easy. The lower-dimensional subspace is found with Principal Component Analysis, which identifies the axes with maximum variance. A class-specific projection with a Linear Discriminant Analysis is applied to face recognition . The basic idea is to minimize the variance within a class, while maximizing the variance between the classes at the same time. A database of 25 persons was created for research purpose. After acquiring the data the images were read from a very simple CSV file because it is a simple platform-independent approach. The problem with the image representation is its high dimensionality. Two-dimensional pxq grayscale images span a m=pq –dimensional vector space, so an image with 100X100 pixels lies in a 10,000-dimensional image space already. Only a decision if there's any variance in data can be made, so there's a need to look for the components that account for most of the information. The Principal Component Analysis (PCA) [1] turns a set of possibly correlated variables into a smaller set of uncorrelated variables. The idea is, that a high-dimensional dataset is often described by correlated variables and therefore only a few meaningful dimensions account for most of the information. The PCA method [1] finds the directions with the greatest variance in the data, called principal components.

Algorithm Description

Let X = {$x_1, x_2, \ldots, x_n$} be a random vector with observations $x_i \in R^d$.

1. Compute the mean μ

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

2. Compute the Covariance Matrix S

$$S = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T$$

3. Compute the eigenvalues λ_i and eigenvectors v_i of S

$$Sv_i = \lambda_i v_i, i = 1, 2, \ldots, n$$

4. Order the eigenvectors descending by their eigenvalue. The k principal components are the eigenvectors corresponding to the k largest eigenvalues.

The k principal components of the observed vector x are then given by:

$$y = W^T(x - \mu)$$

where $W = (v_1, v_2, \ldots, v_k)$.

The reconstruction from the PCA basis is given by:

$$x = Wy + \mu$$

where $W = (v_1, v_2, \ldots, v_k)$.

The Eigenfaces method then performs face recognition by:

- Projecting all training samples into the PCA subspace.
- Projecting the query image into the PCA subspace.
- Finding the nearest neighbor between the projected training images and the projected query image.

There emerges a problem. Suppose there are 400 images sized 100X100 pixel. The Principal Component Analysis solves the covariance matrix $S = XX^T$ where $size(X) = 10000 \times 400$.

We would end up with a 10000×10000 matrix, roughly $0.8GB$. Solving this problem isn't feasible, so there's a need to apply a trick. From the linear algebra lessons we know that a $M \times N$ matrix with $M > N$ can only have $N-1$ non-zero eigenvalues. So it's possible to take the eigenvalue decomposition $S = X^T X$ of size $N \times N$ instead:
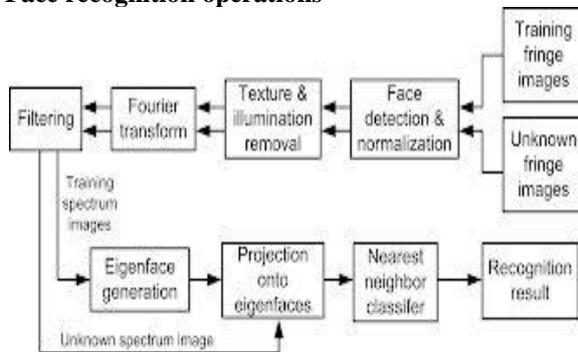
$$X^T X v_i = \lambda_i vi$$

and get the original eigenvectors of $S = X^T X$

with a left multiplication of the data matrix:

$$XX^T(Xv_i) = \lambda_i(Xv_i)$$

The resulting eigenvectors are orthogonal, to get orthonormal eigenvectors they need to be normalized to unit length.

Face recognition operations

V. REMOTE ACCESS AND SECURITY FEATURES

This paper also brings out the possibility for remote access and sending out an email to the user in case of an intruder trying to access the door.

The remote access is done by using Dropbox [8] (service) operated by Dropbox, Inc.

The user has a predefined text file which he can upload to his Dropbox folder[8]. The system periodically reads the Dropbox folder [8] for that text file and if it's found the system allows the door lock to be opened.

In case of an intruder trying to access the door, the unsuccessful attempts made by the intruder is taken count of and his/her image is stored in a folder. These images are then sent to the user by email using SMTP server [7]. The mechanism is explained below:

The .Net and .Mail libraries are to be added in C#. This will allow us to access SMTPClient, MailMessage [7] and NetworkCredentials. A valid email account (with password) is needed to give the program a place to send the mail from. To send an email first a new mailMessage (an email) to send is to be created. To include an attachment with an e-mail message, first create the attachment by using the Attachment class, and then add it to the message by using the MailMessage attachments property. The attachment in our case is the image of the unauthorized user. The information that has to be specified is the SMTP host server[7] that you use to send e-mail, credentials for authentication, if required by the SMTP server and the e-mail address of the sender. A new SMTP client[7] to send our email is to be created. We created a new client which has two parameters (Host, Port). The host is where we'll be sending the mail from; this should be the same as your email account. The only reason we use port 25 is because it's the default SMTP port, although another port may be used. Now we fill in the client

details and send the email. Enabling SSL (Secure Sockets Layer, encryption) is required by most email providers to send mail.

VI. RESULTS AND OBSERVATIONS

The parameters for face recognition system in OpenCV Haar classifier[3] and face recognizer functions are as follows:

1. **Scale increase rate**: This parameter in the call to DetectHaarCascade() specifies how quickly OpenCV should increase the scale for face detections with each pass it makes over an image. Setting this higher makes the detector run faster (by running fewer passes), but if it's too high, you may jump too quickly between scales and miss faces. The default in OpenCV is 1.1, in other words, scale increases by a factor of 1.1 (10%) each pass.

This parameter may have a value of 1.1 , 1.2 , 1.3 or 1.4. We have set it to 1.2, which means it will run the moderate number of passes, thus will accurate as well as fast. The lower the value, the more "thoroughly" Haar detector will check the image for the "face", but naturally will take more time.

2. **Minimum neighbors threshold:**
The next parameter in the call to DetectHaarCascade() is the 'The minimum-neighbors threshold' which sets the cutoff level for discarding or keeping rectangle groups as "face" or not, based on how many raw detections are in the group. This parameter's value ranges from 0 to 10.

We have used minimum neighbors = 10 i.e. we want only an object to be marked as a face if it has the highest probability and vote of being the "face". If we set minimum neighbor to a value n, then detector will mark an object as "face" in any image IF there is a group of n rectangles (hits) identifying it as a "face".

3. **Minimum Detection scale:**
The third parameter in the call to Detect Haar Cascade () is the size of the smallest face to search for. We can change the default for this by changing its value from the Haar cascade classifier xml file .We have set it to 25X25,which gives us the best results.

A good rule of thumb is to use some fraction of your input image's width or height as the minimum scale - for example, 1/4 of the image width. If you specify a minimum scale other than the default, be sure its aspect ratio (the ratio of width to height) is the same as the defaults. i.e., aspect ratio should be 1:1.

4. **Threshold:**

Threshold is the maximum value of the Euclidean distance between the database image and input image.

A database of 25 persons was created wherein 50 images of each person were stored.

For simulation we tested the face recognition system for 3 known persons and 1 unknown person.

The observations were carried out in artificial fluorescent lighting conditions.

We kept the scale increase rate at 1.2, minimum neighbors threshold at 10 and minimum detection scale at 25X25.

The subjects were asked to show their face 10 times in the camera

The results obtained were as follows:

Known person 1:

Threshold	Correct recognition	False recognition	Not recognized
2000	8	0	2
2250	9	0	1
2500	9	0	1
2750	8	0	2
3000	10	0	0
3500	10	0	0
4000	9	1	0
4500	7	3	0
5000	5	5	0

It is observed that for the Known person1,the threshold value in the range 3000-3500 is highly accurate giving us a 100% accuracy under the conditions. Also as the threshold value is increased the false recognition of the person increases thus deteriorating the performance.

Known person 2:

Threshed	Correct recognition	False recognition	Not recognized
2000	0	0	10
2250	0	0	10
2500	0	0	10
2750	1	0	9
3000	6	1	3
3500	8	1	1
4000	5	4	1
4500	5	5	0
5000	4	6	0

For Known person2, initially there is no recognition in the threshold value up to 2500. The optimum performance is obtained at the value 3500 giving us 80% accuracy. Performance weakened for higher threshold values.

Known person 3:

Threshold	Correct recognition	False recognition	Not recognized
2000	0	0	10
2250	0	0	10
2500	2	0	8
2750	4	0	6
3000	7	0	3
3500	10	0	0
4000	5	4	1
4500	3	6	1
5000	2	8	0

Considering the results obtained for Known person3, the threshold value of 3500 gave an excellent result providing a 100% accuracy. Again the range 3000-3500 provided high efficiency. At further increase in the threshold values the subject was falsely recognized.

Unknown person:

Threshold	Correct recognition	False recognition
2000	10	0
2250	10	0
2500	10	0
2750	10	0
3000	10	0
3500	10	0
4000	10	0
4500	7	3
5000	2	8

As per the results, the subject was correctly recognized as unknown up till a threshold value of 4000 providing an accurate system of recognition. The image of the unknown person was sent as an attachment to the Gmail account through SMTP. However beyond the threshold of 4000 the subject was recognized as a known person degrading the performance.

VII. CONCLUSION

The paper suggests that the face detection process under controlled lighting conditions (fluorescent light) gives an accuracy of 95% with scale increase rate at 1.2 ,minimum neighbors

threshold at 10 and minimum detection scale at 25X25.

The recognition process is subject to the threshold value which is the maximum value of the Euclidean distances between the database images and input image. The threshold value was different for different persons ranging from 3000-3500.In this range the recognition rate is at 85%.

Also the SMTP and Dropbox service are used for added security and remote access features.

REFERENCES

[1] M. Turk and A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuro science, March 1991.

[2] P. Viola and Michael J. Jones. "Robust real-time face detection", International Journal of Computer Vision, 57(2):137-154, 2004.

[3] "The System of Face Detection Based on OpenCV" Xianghua Fan, Fuyou Zhang, Haixia Wang, Xiao Lu Key Laboratory for Robot & Intelligent Technology of Shandong Province, Shandong University of Science and Technology, Qingdao 266590

[4] "Facial feature detection using haar classifiers*"Phillip Ian Wilson Dr. John Fernandez Texas A&M University – Corpus Christi 6300 Ocean Dr. CI334, Corpus Christi, TX 78412 361-825-3622.

[5] " Miner Face Detection is Based on Improved AdaBoost Algorithm "Chao JIANG ,Gu-yong HAN ,Lei TIAN ,Song LU, Wei-xing HUANG Air Force Service College XuZhou JiangSu China.

[6] http://www.cognotics.com/opencv/servo_20 07_series/part_2/sidebar.html

[7] "Active e-mail system SMTP protocol monitoring algorithm", Sureswaran, R. Nat. Adv. IPv6 Centre (NAv6), Univ. Sains Malaysia, Minden, Malaysia Al Bazar, H. ; Abouabdalla, O. ; Manasrah, A.M. ; El-Taj, H.

[8] "On the impact of virtualization on Dropbox-like cloud file storage/ synchronization services", Haiyang Wang Sch. of Comput. Sci., Simon Fraser Univ., Burnaby, BC, Canada Shea, R. ; Feng Wang ; Jiangchuan Liu