Automatic Missing-Child Recovery System using Eigenfaces

Chapter · May 2020

DOI: 10.1007/978-981-15-3607-6_11

2 authors:

Nuruzzaman Faruqui
Daffodil International University

Mohammad Abu Yousuf
Jahangirnagar University

1st International Conference on Advances in Science, Engineering and Robotics Technology 2019 (ICASERT 2019)

Automatic Examinee Validation System using Eigenfaces

Nuruzzaman Faruqui
Institute of Information Technology
Jahangirnagar University
Dhaka, Bangladesh
faruquizaman27@gmail.com

Mohammad Abu Yousuf
Institute of Information Technology
Jahangirnagar University
Dhaka, Bangladesh
yousuf@juniv.edu

Md. Fazlul Karim Patwary
Institute of Information Technology
Jahangirnagar University
Dhaka, Bangladesh
patwary@juniv.edu

***Abstract*—In most of the developing countries the seats of educational institutions are limited and barley 30% of the total eligible candidates get the opportunity to get admitted after facing competitive admission test. The scenario is worse in the job market. The growing graduation to unemployment ratio has become a major concern in developing and underdeveloped countries. As a result, job recruitment tests have become more competitive than ever. In order to survive in these types of competitive examinations, many candidates try to take unethical approaches. Because of large number of candidates, exam administrators cannot validate each and every candidate one by one. Moreover, modified image in registration paper makes it difficult to identify valid examinee. This is the weakest point of examinee validation system of competitive examination and significant number of candidates try to pass the examination by exploiting this weakness. The easiest way to exploit this weakness is to hire and send proxy examinee who has higher possibility of passing the exam instead of actual examinee. In this paper, we are proposing an automatic, fast, efficient and accurate yet simple to implement examinee validation system using eigenfaces. Our system will make it almost impossible for a proxy examinee to enter the exam-center.**

Keywords—examinee validation, face recognition, eigenface, segmentation, innovative registration.

I. INTRODUCTION

The rate of cheating in the competitive examination through proxy examinee is rising at an alarming rate [1]. It deprives the meritorious candidates from getting the positions they deserve. And if the cheaters do not get caught, they get employed in important positions. In the long run the productivity of the organizations they join, educational institutions they get admitted in end up in poor productivity. The purpose of competitive exams isto find out the appropriate candidates. If wrong person sits in important position through unethical means, they will not perform their responsibility as the proper candidate would have done. It is very dangerous for a nation, especially for developing countries. In developing countries excessive population creates extra pressure on both educational institutions and job market. Because of having limited seats, institutions can barely take 30% of total eligible candidates. The condition in job market is worse than this. The competition has become so tough thatmore candidates are being engaged in cheating in competitive examinations. At the same time, it is not possible for exam-administrators to verify every individual because of large number of candidates. This is where the weak point is and tendency of getting in exploiting this weak point is becoming acuter day by day.

Automatic examinee validation system is the most effective solution to proxy examinee problem. However, implementing such system is time consuming and challenging task. Using convolutional neural network, it is possible to validate examinee with 96.2% accuracy [2]. To gain such accuracy five images are needed of each individual. In the year 2017 against 2024 vacancies 343,532candidates attended examination conducted by Bangladesh Public Service Commission (BPSC)[3]. In similar situation, taking five images of each candidate for training will yield 1,717,660 training images. Processing this large volume of image data and training the network using themis not a feasible solution. However, through linear segmentation of training data according to the number of exam-center, it is possible to distribute the computational task among multiple systems. In this approach the training time can be reduced significantly. Although it sounds feasible yet distributing the trained network, permanently setting up validation equipments in exam-centers, training new network before every examination make it too complex and expensive to implement.

We need a system that works in real-time and does not require training like deep neural network or convolutional neural network. The system has to be very simple, fast, accurate and instantly deployable. We are proposing an automatic examinee validation system using eigenfaces which is very simple, easy to implement and works very fast in real-time. Our system does not require any training. Theoretically, our system will work fast enough even for image database larger than 5×10^5 images. No matter how many images are there in the database, our system will always compare the test image with 10 training images and will yield over 97.88% accurate result.

The remaining part of the paper is as follows -relevant works are reviewed in section 2. The proposed method has been presented in section 3. In section 4, two ways have been shown which can be used to physically implement the validation system. The experimental result of the proposed method has been demonstrated in section 5. Finally, we concluded the paper in section 6.

II. BACKGROUND

The normalized distance between facial features and ratios among the features can be used for automatic face recognition. Bledsoe [4] and Kanade [5]have used eyes, nose, mouth and head outline as facial features. The existence of facial features, their relative positions and size are used to classify face and non-face classes. By comparing the features of images belong to face class, new faces can be

978-1-7281-3445-1/19/$31.00 ©2019 IEEE

recognized if they are already in existing face class. However, this approach cannot be extended to multi-view face recognition. Moreover, Diamond and Carey [6]have shown that individual facial features are inefficient for adult human face recognition. In addition, the feature points are not prominent for all races. As a result, there are possibilities of having face images in non-face class.

Instead of classifying images based on individual facial features, face images can be considered as 2D matrix where 2D characteristics represent faces. Turk and Pentland [7]have developedeigenvector-based face detection and recognition system which can learn to recognize new faces in an unsupervised manner. Diverting from geometry or high-level visual characteristics, they have taken information theory-based approach. In this approach they have transformed images into characteristic feature images named eigenfaces. These features are the principle components of training set of face images. Then, projecting new images on the eigenfaces and comparing the position of new eigenface with existing eigenfaces, they have performed the recognition. This approach remains effective for multi-views such as profile face, view from 45° angle and straight view. Moreover, this recognition method is fast, simple and robust.

However, the size of the face space imposes challenges on eigenface construction. Sirovich and Kirby have represented their idea of reducing face space using principle component representation [8]. They have said face images can be reconstructed by storing small collection of weights of each face. However, a set of standards which will uniquely define how the weights will be distributed are required as well. Combining the weights and the standards, it is possible to reconstruct the actual face image from small image space. They have also suggested that if a face can be reconstructed from the weights and standards, a face can be recognized using the same principle as well. If a face image gives similar weights and standards as another face image, then both of the face images will be the same. In other words, weights and standards of one face image can be used to recognize the same face image. If the face image of the same person varies in dimension, the standards which are actually vectors adjust the weights proportionally. As a result, the size variation cannot affect the recognition accuracy.

Simple implementation, fast recognition and robustness of face recognition method of Turk and Pentland paved the way of using eigenface based system in application level. Wagh, Thakare, Chaudhari, and Patilhave shown an application of detecting attendance fraudulent activities in class room using eigenfaces[9]. To meet the acceptance level at application level, only eigenfaces are not enough. Wagh et al. in their paper have combined luminance variant reduction technique to reduce impact of light variation among testing and training image. They have also used Viola-Jones algorithm to detect the face in order to get specific face space. After primary recognition a cross checking by students' image database have enhanced the accuracy of their system.

Eigenface-based face verification systems are enough secured for non-financial and non-confidential national defense related application. J. Galbally, J. Fierrez, J. Ortega-Garcia, C. McCool and S. Marcel have shown security vulnerabilities of Eigen-face based face verification system by hill-climbing attack [10]. The success rate of hill-climbing attack is more than 85%. It does not require a large number of training image to initiate the attack. It clearly suggests that precaution must have to be taken in eigenface-based verification system. However, hill-climb attack cannot impact effectively in real-time validation system. Which leaves a scope of implementing real-time automatic eigenface based validation system without the risk of having hill-climb or brute-force attack on the security system.

III. Proposed Method

Automatic examinee validation system using eigenface is a two-step process. The first step is related to registration system and the second step is the validation system. With the existing examination registration system, it is not possible to effectively validate examinees automatically. The training image required for validation must be acquired during registration process. In existing system, candidates register through online registration form for examinations and upload scanned copies of their photos.With a little modification of scanned images using photo editing software like Photoshop, it is very easy to alter the images in registration card to send proxy examinee instead of real examinee with very little possibility of getting caught. To leave no scope of sending proxy examinee we have to change the registration system first.

A. Proposed Registration System

With the existing online registration system, we propose an additional module which is a real-time image capturing unit. Instead of uploading the scanned copy of photo, the candidates will capture 10 photos according to given instructions. The device using which the candidates will

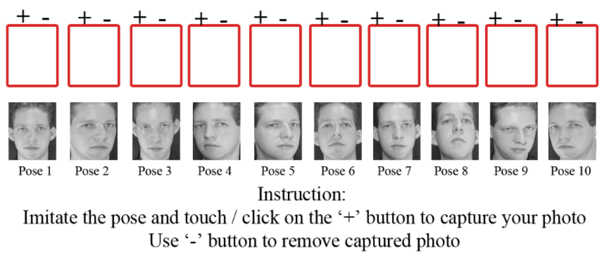

Fig. 1. Photo Capturing During Registration.

register must have to have webcam for this registration process.Fig. 1 shows the interface of registration process. Candidates have to take 10 pictures imitating the poses recommended in the instruction. The image capturing module uses Viola-Jones face detention algorithm[12]to detect the faces and crop the face rectangle by 92×112 pixel. The images are labeled from 1 to 10. Once the candidate confirms registration, a directory is created based on choice of exam center (e.g. Dhaka) and a subdirectory is created under this newly created directory named according to the registration number of the candidate. In this way, candidates of different exam centers will have different directories. In those directories, the subdirectories will contain the images of the candidate. Each subdirectory represents individual registered candidate. As each registration number is unique, each candidate will have their images in specific subdirectory. These images are the training image of our system.

B. Examinee Validation

We have used eigenfaces for examinee validation. Even though convolutional neural network is much better at image recognition, we cannot use it in the validation system we are proposing. Because the dataset our system has to handle is very large, which requires huge amount of time to train the network. And for every new examination, new network has to be trained. Which makes convolutional neural network unrealistic for the system we are building. Our system is based on the work ofTurk and Pentland [7]. They have shown how eigenfaces can be used for face recognition. We calculated the eigenfaces in exactly the same way they did.However, we have modified the recognition phase to meet the requirement of our system. We have reduced the size of the face images by calculating and removing the mean. Instead of simply classifying as known and unknown, we have added an access method for validated candidate. The method proposed in [7] was near-real-time whereas our implementation works in real-time.

1) Prioritizing Eigenface for Recognition

Predefined measurement of facial features is not sufficient for face recognition. Instead of looking at face images from high level visual features it is much efficient in focusing from information theory perspective. Eyes, nose, lips, and hair are global feature of a face. From information theory perspective they carry very small amount of information as the probability of having these features is very high. Our purpose is to recognize faces. That is why, the variations among testing face and training faces are the major concern. The face image is a 2D matrix that carries information. From information theory perspective, our task is to extract the information from the 2D matrix and encode it efficiently. Then we will encode the training image in the same way and later we will compare among encoded pieces of information.

We have computed the variations among the face images stored in database. We are not considering those images as face images but 2D matrix those carry information. From these perspective, facial features described in [5] are simply information. There is no distinction among them from high level visual perspective (i.e. nose is not nose but some pixel variations). Once the information of the 2D matrices are collected we compare among them and identify them based on the variation of information they present. According to [8] the maximum varying regions are the principal component. They call it eigenvectors of the covariance matrix. When we represent these eigenvectors as images, it produces face images, indifferent to human eyes, but meaningful in computation in terms of variations among multiple faces.

All of the faces stored in the registration database can be represented in terms of linear combination of the eigenfaces. We have 10 images for each individual. That means there are 10 eigenfaces for each registered candidate. For our application, it is not mandatory to match all 10 of the eigenfaces with 100% accuracy with the testing image. Out of 10 eigenfaces, we compute the best eigenface based on the information variation of that face with other faces in the same directory (e.g. Dhaka). The number of training data is large even though we have classified them according to location of exam-center during registration process. As a result, reducing computational time is mandatory. Instead of taking all 10 of the eigenfaces, taking the eigenface that exhibit the highest variation when compared with other candidates' eigenface is maintained with highest priority for recognition.

Algorithm 1 Face recognition algorithm

Input: X: Exam Center; Y: Registration Number; Z = Input Image
1: **if** X = true **then** access X
else terminate
2: **if** Y = true **then** access Y
Image Array ← $Load\ training\ images$
else terminate
3: **for** i = true
4: Processed Image[i] ← $Image\ Array[i] - mean(Image\ Array[i])$
5: Eigenface Array← $Eigenface\ Function\ (Processed\ Image)$
6: **end for**
7: Processed Input ← $Eigenface\ Function(Z - mean(Z))$
8: **for** i = true
9: Euclidean Distance ← $Processed\ Input - Eigenface\ Array\ [i]$
10: **if** Euclidean Distance > threshold **then** allow access to exam-center
else deny access to exam-center

2) Algorithm for Face Recognition and Validation

The procedure or the face recognition and validation are shown in algorithm 1. As stated earlier, our system is built on [7]. We modified the approach taken in [7] and converted themethod into an applicable validation system. The algorithm is initiated by user inputs which are actually the information of candidates. The candidates select the names of exam-centers, registration numbers and the system captures their frontal faces. If the inputs are correct, the algorithm executes the further instructions and return the result (line 10 of algorithm 1). It allows only valid candidates to enter the exam-center and prevent others from getting accesses.

3) Calculating Eigenfaces

Training images are 2D matrix consisted of 2-dimensional array having 92 and 112 values respectively. Each of the matrix has 10304 elements. The entire space does not represent the face. Moreover, we do not need the entire face. We are looking at the image from information theory perspective. To reduce the size of the image, we need to remove the information which will not be used in face recognition. By calculating and removing the mean from each image, we can reduce the size of the image. From matrix perspective, the value of common element will be replaced by zero. From visual perspective, the images will carry information that distinguishes them from one another.

Let the training set of face images of one candidate isbe $F_1, F_2, F_3, \ldots F_{10,}$. To further reduce the size of face space, we calculate the average of the faces.

$$\text{Average face, } A = \frac{1}{10}\sum_{n=1}^{10} F_n \quad (1)$$

Now we have the average face. Using the average face, we can calculate the deviation matrix of other faces of the candidate by subtracting the A from each face.

$$\text{Deviation, } \phi_i = F_i - A \tag{2}$$

Then using principle component analysis (PCA) we find a set of vectors and values which form the eigenfaces. We followed the method of using PCA of[7] to calculate the eigenfaces. According to Turk and Pentland the eigenvalues (λ_k) and eigenvectors (μ_k) together form the covariance matrix,

$$C = \frac{1}{10}\sum_{n=1}^{10} \phi_n \phi_n^T = II^T \tag{3}$$

Here I is the array of deviation matrix. The covariance matrix is still large as it is consisted of 2 arrays containing 92 and 112 elements respectively. Computing over this large matrix is expensive and time consuming. However, [7] shows that eigenvectors can be determined my solving smaller M by M matrix and then combining them linearly. This makes the calculation much faster. At the end, the background is removed which gives us the eigenfaces only.

4) Validation using Euclidean Distance

Eigenfaces contain the pattern that distinguishes faces from one another. They are spanned in M' dimensional subspace where the original image is an 92×112 image space. The eigenvector M' which is associated with largest eigenvalues are chosen as the eigenface. Now for a new image, the face is detected using Viola-Jones algorithm [11] and cropped with a rectangle of 92×112 pixel. Then it is transformed into eigenface consisted of an eigenvector and eigenvalue. The weights of eigenfaces form vectors. These vectors represent the face images. The Euclidean distance between the weights of vectors of input image and training image describes if the new image belongs to the training image set or not. If the new face belongs to the training image located inside subdirectory named after the candidate's registration number, we consider the person as valid candidate. Otherwise the candidate is considered a proxy candidate.

IV. Implementation

The performance of the validation system depends implementation architecture. Here arises the issue of performance vs implementation cost trade-off, which leads to two different architecture – (i) Single Validation Unit (SVU) architecture and (ii) Parallel Validation Unit (PVU) architecture. The SVU architecture is preferred where implementation cost is the major concern. And the PVU architecture is preferred where performance is the major concern. One advantage of PVU architecture is here the implementation cost and performance can be adjusted by changing the number of units. On the other hand SVU architecture does not offer the flexibility of adjusting the performance vs cost parameter.

A. SVU Architecture

In SVU architecture there is one validation unit at the entrance of the exam perimeter (fig. 2). The examinees are validated one by one. Valid examinees get the access to the exam perimeter. The suspected examinees generate notifications in invigilators' office with their names and registration numbers. These examinees do not get the access to exam perimeters. Instead they are directed to invigilators' office for further investigation.

The SVU architecture is simple and easy to implement. However, it is suitable for small number of students. When the number of students increases, the total validation time increases linearly. In this case total time to validate S_t number of students V_t will be,

$$V_t = S_t \times V_i \tag{4}$$

Here V_i is average time to validate each examinee. Equation 4 shows that the total time required to validate all of the examinee is directly proportional to the number of total students. That is why for public exams where large number of candidates attend exams, SVU architecture fails to exhibits satisfactory performance. However, from network security perspective this approach is much secured. Because of having single unit, the validation server and invigilators' office can be established nearby. As a result, a wired communication is possible between server and the validation unit. It ensures security from attacks like eavesdropping or data manipulation by man-in-the-middle attack.

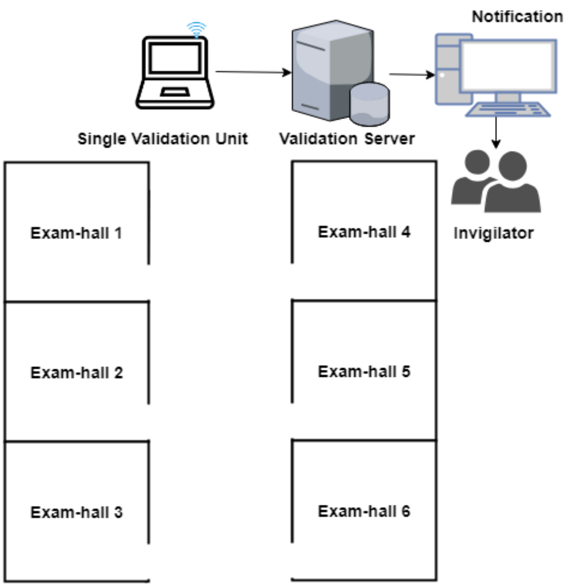

Fig.2. SVU architecture

B. PVU Architecture

In PVU architecture (fig. 3) multiple SVU work simultaneously. In this model, the examinees can enter the exam perimeter without any validation. However, before entering the exam-hall they have to go through the validation process. Here, the valid examinees get access to respective exam-halls. And the suspected candidates are directed to the invigilators' office. In this case with username and registration number, the exam-hall number are sent in the notification.

The implementation of PVU system is not as simple as SVU system. The complexity and cost increases with the increase of validation units. However, the total time required to validate all of the examinees reduces linearly with the increases of number of validation units. Here, the total time required to validate S_t number of students V_t will be,

$$V_t = \frac{1}{U} \times (S_t \times V_i) \quad (5)$$

Here U represents the total number of validation unit. The more the validation unit the less the time required to validate in PVU method. Equation 5 shows that it is possible to reduce the total validation time by increasing the validation unit. That means the performance of the system can be enhanced. However, because of having multiple units in different spatial locations, wireless communication among units and server is the most convenient communication method. In this case, the network security becomes a vital issue. Intruders can eavesdrop in the network to steal information. At the same time, they can alter the validating data by man-in-the-middle attack. Proper network security needs to be implemented to use PVU method.

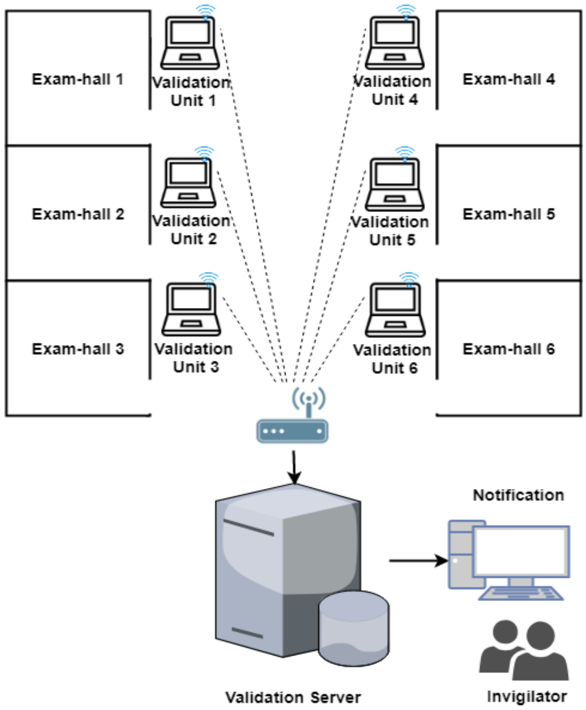

Fig. 3. PVU architecture

We prefer implementing the PVU architecture because of its performance and flexibility.

C. Validation System

The validation process starts with the user inputs. The examinee enters his username, registration number and password. The server validates the access credentials and if the credentials are valid, it serves the image capturing web application. At the same time, the server loads the image of the examinee those were taken during the registration process. In the next step, the examinee captures his image and send it to the server for validation. The server checks if recent image and previously saved images are matching using our proposed method. If the images match with acceptable rate, the candidate is declared as valid examinee. If the images do not match, then the server marks the candidate and generates an alarm at the invigilators' office. Concurrently, the candidate receives a message containing the instruction to visit the invigilators' office. The invigilators will further investigate the issue and will decide if the candidate is a valid candidate or proxy examinee.

V. Performance Analysis

We used Olivetti Research Laboratory (ORL) dataset of faces for performance analysis. Each of the images are in 8-bit grayscale. The images are in PGM format. This dataset contains 400 images of 40 person. There are 40 directories representing 40 individuals and each of those directories has 10 images of same person labeled from 1 to 10. To conduct the experiment, we used Windows 10, 64 bit operating system, with 8192 MB primary memory, powered by Intel(R) Core(TM) i3-6100 CPU, 3.70GHz (4 CPUs). All of the algorithms have been implemented using MATLAB R2017a.

Wehave evaluated the performance by the dataset loading time, recognition time and error rate.

A. Loading Time

The loading time maintains linear relationship with the number of images. We computed the loading time for 5, 10, 15, 20, 25, 30, 35 and 40 persons. Normally 35-40 candidates attend exam in a single room. That is why we considered it unnecessary to calculate the loading time for more than 40 candidates. To load dataset of 5 people, which is 50 images, our system took 0.098 seconds. The time was extended to 0.876 seconds for 40 candidates, which are 400 images.

B. Recognition Time

We have experimented with all 400 images. In none of the cases the recognition time exceeded 1.98 seconds. To calculate the average recognition time, we have calculated the average for every five individuals at a time. Finally, we calculated the average of these average times. And the average recognition time is 1.37 seconds. The recognition time includes the time to generate eigenfaces of training images and testing image, mean subtraction time, indexing time and Euclidean distance calculation time. Altogether, our algorithm took 1.37 seconds to successfully validate a face. However, we did not consider the time when the algorithm gave wrong output. The recognition time has been calculated for correct output only. In Fig. 4 the comparison

Table 1 Performance summary

Serial	Person	Loading Time (seconds)	Recognition Time (seconds)
1	5	0.098	1.12
2	10	0.211	1.36
3	15	0.309	1.6
4	20	0.429	1.98
5	25	0.487	1.18
6	30	0.547	1.22
7	35	0.656	1.21
8	40	0.876	1.31

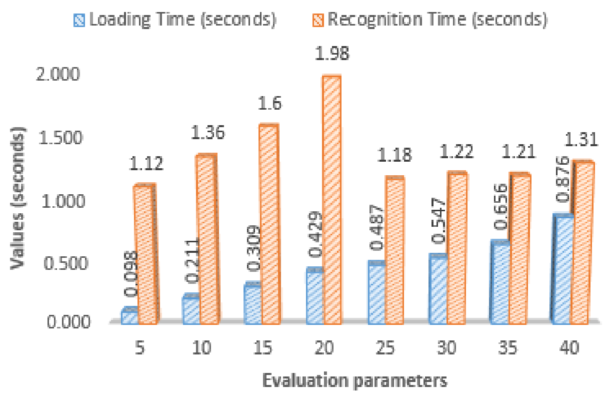

Fig. 4. Loading and recognition time.

between loading time and recognition time has been shown. In our system, the loading time is pre-validation process. It will be loaded before starting the validation process. The impact of recognition time is the major concern. Because recognition will be conducted in real-time and it determines the speed of the validation process.

Table 2 Accuracy and Error Calculation

Trial	Correct Recognition	Incorrect Recognition	%Accuracy	%Error
1	392	8	98	2
2	388	12	97	3
3	388	12	97	3
4	396	4	99	1
5	392	8	98	2
6	392	8	98	2
7	388	12	97	3
8	396	4	99	1

C. *Accuracy and Error Calculation*

We define the error as incorrect recognition. Our system recognizes faces from information theory perspective using eigenfaces. Although the probability of having same pattern for two different eigenfaces of two different persons is close to zero, sometimes the patterns demonstrate little difference which falls below our threshold level. As a result, there is possibility of having incorrect recognition. However, it is not feasible to employ additional algorithm to detect the error rate. That is why, we have taken human inspection approach to calculate the error of our validation system.

We have 400 images of 40 person. Each of them has 10 images with different poses. In order to find out the error rate, we loaded each of the images of every individual separately and ran the program to check if it can recognize the face correctly. We repeated the process for 8 times. In the first trial, out of 400 images 392 faces were recognized correctly, whereas 8 recognitions were incorrect. In the second trial 388 faces were successfully recognized. But 12 faces gave mismatched result. The result has been enlisted in table 2.

To calculate the percentage error, we used (6), where TI means ‘total image’ and CR means ‘correct recognition.

$$\%Error = \frac{TI-CR}{TI} \times 100 \qquad (6)$$

Equation (6) gives us the error. From the error we calculated the percentage accuracy. Fig. 5 shows the accuracy in percentage scale.

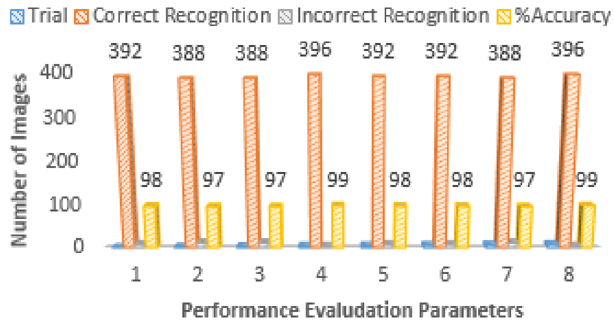

Fig. 5. Accuracy and error

The highest accuracy of our system was 99% where only 4 images were incorrectly recognized. The red bars represent the total number of correctly recognized image and the yellow bars represent the percentage accuracy.

Conclusion and Future Work

An effective, efficient, simple and inexpensive method has become essential to prevent proxy examinees from taking examinations. Manual process is no longer effective because of excessive and growing number of candidates in competitive tests. The automatic examinee validation system we have proposed has all of the features required to conduct a fair environment for competition. Our proposed method is fast and can recognize faces with above 97.88% accuracy. Moreover, this system does not require any training. The registration method itself processes the images for recognition system, which validates examinees before entering the exam-center. Although it adds few more steps in registration process, considering the performance, efficiency and accuracy the system we are proposing is an excellent solution to prevent proxy examinees and ensure that only the deserving candidates will survive in competitive examinations. In our future work, we will implement the validation system physically to conduct a dummy examination. If the system performs as good as the lab experiment, we will launch the validation system as an open source examinee validation system.

References

[1] “One-stop cheating service in exams”, The Daily Star, 2018, April 8, Retrievd from https://www.thedailystar.net/backpage/one-stop-cheating-service-exams-1559536

[2] S. Lawrence, C.L. Giles, A. C. Tsoi, A.D. Back, “Face recognition: a convolutional neural-network approach”, IEEE Transactions on Neural Networks, vol. 8 , pp. 89-113 , January, 1997.

[3] “38th BCS preliminary exam held”, Dhaka Tribune, December 29, 2017, Retrived from https://www.dhakatribune.com/bangladesh/20 17/12/29/38th-bcs-preliminary-exam

[4] W. W. Bledsoe, “The model method in facial reccognition,” Panoramic Research Inc., Palo Alto, CA, Ref. PRI:15, August 1966.

[5] T. Kanade, “Picture processing system by computer complex and recognition of human faces,” Department of Information Science, Kyoto University, Novermber 1973.

[6] S. Carey, R. Diamond, “From piecemeal to configurational representation of faces,” Science, Vol. 195, January 21, 1977, pp.312-313.

[7] M. A. Turk, A. P. Pentland, “Face Recognition Using Eigenfaces,” in Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Maui, HI, USA, USA, 3-6 June, 1991.

[8] L. Sirovich and M. Kirby, “Low-dimensional procedure for the characterization of human faces,” J. Opt. Soc. Am. A, Vol. 4, No. 3, March 1987, pp. 519-524.

[9] P. Wagh, R. Thakare, J. Chaudhari, S. Patil, “Attendance system based on face recognition using eigen face and PCA algorithms,” in International Conference on Green Computing and Internet of Things (ICGCIoT), Noida, India, 8-10 October, 2015.

[10] J. Galbally, J. Fierrez, J. Ortega-Garcia, C. McCool, S. Marcel, “Hill-climbing attack to an Eigenface-based face verification system,” First IEEE International Conference on Biometrics, Identity and Security (BIdS), Tampa, FL, USA, 22-23 September, 2009.

[11] P. Viola, M. Jones, “Rapid object detection using a boosted cascade of simple features”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA, USA, 8-14 Dec. 2001.

[12] M. Turk, A. Pentland, “Eigenfaces for Recognition”, Journal of Congnitive Neuro-science, March 1991.

[13] K. Cheng, L. Xiang, T. Hirota and K. Ushijima, “Effective Teaching for Large Classes with Rental PCs by Web System WTS,” in Proc. Data Engineering Workshop 2005 (DEWS2005), 2005, 1D-d3.

[14] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition: A literature survey,” ACM Computing Surveys, 2003, vol. 35, no. 4, pp. 399-458.

[15] S. Nishiguchi, K. Higashi, Y. Kameda and M. Minoh, “A Sensor-fusion Method of Detecting A Speaking Student,” IEEE International Conference on Multimedia and Expo (ICME2003), 2003, vol. 2, pp. 677-680.

[16] R.E. Burkard and E. C¸ ela, “Linear Assignment Problems and Extensions”, In Handbook of Combinatorial Optimization, Du Z, Pardalos P (eds). Kluwer Academic Publishers: Dordreck, 1999, pp. 75-149.

[17] A. K. Jain, A. Ross, and S. Pankanti, “Biometrics: a tool for information security,” IEEE Trans. on Information Forensics and Security, vol. 1, no. 2, pp. 125–143, 2006.

[18] P. Mohanty, S. sarkar, and R. Kasturi, “From scores to face templates: a model-based approach,” Pattern Analysis and Machine Intelligence, vol. 29, pp. 2065–2078, 2007.

[19] C. Soutar, “Biometric system security. http://www.bioscrypt.com/assets/security soutar.pdf.”

[20] A. Adler, “Sample images can be independently restored from face recognition templates,” in Proc. CCECE, vol. 2, 2003, pp. 1163–1166.

[21] U. Uludag and A. K. Jain, “Attacks on biometric systems: a case study in fingerprints,” in Proc. SPIE-IE, vol. 5306, no. 4, 2004, pp. 622–633.

[22] F. S. Samaria and A. C. Harter, “Parameterization of a stochastic model for human face identification,” in Proc. 2nd IEEE Wkshp. Applicat. Comput. Vision, Sarasota, FL, 1994.

[23] K.-K. Sung and T. Poggio, “Learning human face detection in cluttered scenes,” in Computer Analysis of Images and Patterns, G. Goos, J. Hartmonis, and J. van Leeuwen, Eds. New York: Springer–Verlag, 1995, pp. 432–439.

[24] K. Sutherland, D. Renshaw, and P. B. Denyer, “Automatic face recognition,” in Proc. 1st Int. Conf. Intell. Syst. Eng. Piscataway, NJ: IEEE Press, 1992, pp. 29–34.

[25] D. L. Swets and J. J. Weng, “Using discriminant eigenfeatures for image retrieval,” IEEE Trans. Pattern Anal. Machine Intell., to appear, 1996.

[26] M. Turk and A. Pentland, “Eigenfaces for recognition,” J. Cognitive Neurosci., vol. 3, pp. 71–86, 1991.

[27] J. Weng, N. Ahuja, and T. S. Huang, “Learning recognition and segmentation of 3-D objects from 2-D images,” in Proc. Int. Conf. Comput. Vision, ICCV 93, 1993, pp. 121–128.

[28] L. Wiskott, J.-M. Fellous, N. Kr¨uger, and C. von der Malsburg, “Face recognition and gender determination,” in Proc. Int. Wkshp. Automat. Face Gesture Recognition, Z¨urich, Switzerland, 1995.