International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 11, ISSN No. 2455-2143, Pages 94-100
Published Online March 2023 in IJEAST (http://www.ijeast.com)

DIGITAL AUTHENTICATION USING FACE RECOGNITION

Ms. R. Sandhiya
Department of Computer Science and Engineering
R.M.K College of Engineering and Technology, Chennai, Tamil Nadu, India

Sai Siddhardha Narisetty
Department of Computer Science and Engineering
R.M.K College of Engineering and Technology, Chennai, Tamil Nadu, India

Svs Dheeraj Kumar
Department of Computer Science and Engineering
R.M.K College of Engineering and Technology, Chennai, Tamil Nadu, India

Ujjini Vijay Kumar
Department of Computer Science and Engineering
R.M.K College of Engineering and Technology, Chennai, Tamil Nadu, India

Tella Vedith
Department of Computer Science and Engineering
R.M.K College of Engineering and Technology,
Chennai, Tamil Nadu, India

Abstract – **Security is the main concern in any of the web application or apps. To ensure security, biometricsystem is used for higher secure system. Usually manyof the security based devices uses fingerprint authentication for access. Finger Print recognition based devices have a large demand for security concerns. On March 11, 2020 the World Health Organization has declared the situation with a worldwide pandemic of Corona virus (COVID-19). The World Health Organization characterizes this sickness as a pandemic since all residents of the world are possibly presented to COVID-19 disease. With thefoundation of the worldwide pandemic status, WHO additionally affirmed that COVID-19 was a global crisis. The pattern of digitalization is turning into another business pattern to create and make due amidst an emergency because of this pandemic.**

However some portals and software system requires the direct contact of fingerprint or entering the password for authentication. Authentication is a significant issue in system control. It renders in the high security of any software system. Due to pandemic most of us are away from the working or educational places where we personally use the fingerprint for the authentication or attendance. Many of the security based devices are fingerprint recognition based but the COVID-19 has left a drawback with this fingerprint based secure systems as the people are not physically available for the authentication. In order to overcome this we can use the contactless biometric system methods like face recognition system. In this paper we give a solution of face recognition for login system to access the information where usually fingerprint isused for authentication.

Keywords **– Security, Authentication, facial recognition, COVID-19.**

I. INTRODUCTION

An emerging direction for authenticating people is the adoption of biometric authentication systems. Biometric credentials are becoming increasingly popular as a means of authenticating people due to the wide range of advantages that they provide with respect to classical authentication methods (e.g., password-based authentication). The most characteristic feature of this authentication method is the naturally strong bond between a user and her biometric credentials. This very same advantageous property, however, raises serious security and privacy concerns in case the biometric trait gets compromised.

Biometric authentication is a quick, accurate, and user-friendly tool that offers an efficient and reliable solution in multiple access control systems. A typical example of biometric authentication systems (BAS) is access control systems equipped with sensors (e.g., for iris or fingerprint scans). In this case, the sensor captures the biometric trait of the person who requests access, while access is granted only after the person has been recognised as an authorised user of the system. One of the mainadvantages of biometrics is that they do not require to memorise complicated passwords or carry tokensalong since they cannot be forgotten or lost.

While BAS provide important usability advantages, they are susceptible to threats, like any other security system. For biometric authentication, however, a successful attack can have severe implications in theusers' lives and privacy. Unlike passwords or tokens, biometric credentials cannot be kept secret or hidden, and stolen biometrics cannot be revoked as easily. Thus, the risk of them being compromised (i.e., captured, cloned, or forged) is high and may lead to identity theft or individual profiling andtracking in case the templates are used and cross- matched in different biometric databases. In addition, stolen biometrics can be used to learn sensitive information about their owners, such as ethnic group, genetic information, and medical diseases, or even to perform illegal activities by compromising health records.

Fig 1 shows the issues and security of a area specifically where it concerns with data privacy, internet, applications, personal devices and common configuration enumeration (CCE).

II. SYSTEM ARCHITECTURE

The proposed system is mainly based on face detection and face recognition. Among the many possible approaches, we have decided to use Haar- Like features algorithm for face detection part and local binary pattern approach for face recognition part.

Fig 2 shows the steps of recognition system application steps which is a basic idea for any kind of application used for recognition.

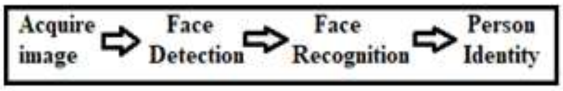


Fig 2: Steps of face recognition system applications

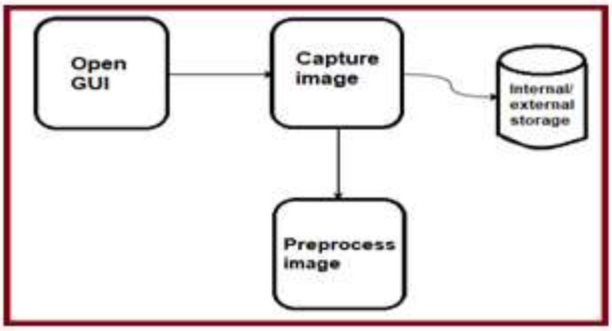
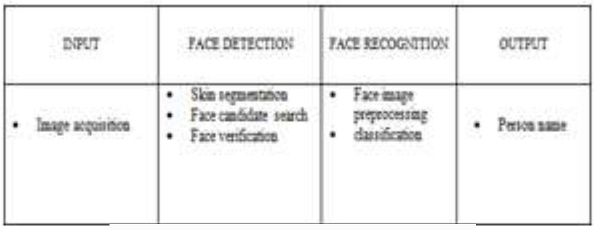


Fig 3: System architecture for Registering Face

Fig 3 shows the GUI process. When user opens GUI, there is button to capture images where the user captures images from camera. After image capturing, pre-processing of that image is done and then face detection and recognition is accomplished and result is shown to the user.



INPUT	FACE DETECTION	FACE RECOGNITION	OUTPUT
• Image acquisition	• Skin segmentation • Face candidate search • Face verification	• Face image preprocessing • classification	• Person name

Fig 4: Face Recognition Approach

Fig 4 discuss about the sub process in facial recognition whereas Fig 16, Face detection is the process of identifying human face in an image. Theprocess of face detection is given below:

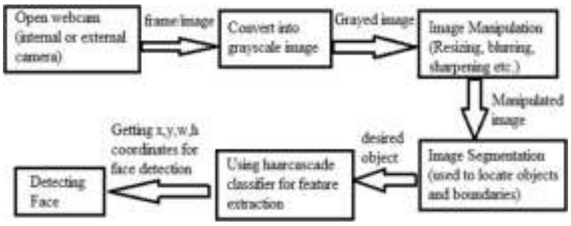

Open webcam (internal or external camera)
frame/image
Convert into grayscale image
Grayed image
Image Manipulation (Resizing, blurring, sharpening etc.)
Manipulated image
Image Segmentation (used to locate objects and boundaries)
desired object
Using haarcascade classifier for feature extraction
Getting x,y,w,h coordinates for face detection
Detecting Face

Fig 5: The framework of face detection process

Fig 5, Firstly the image is imported by providing the location of the image. Then the picture is transformed from RGB to Grayscale because it is easy to detect faces in the grayscale. After that, the image manipulation used, in which the resizing, cropping, blurring and sharpening of the images done if needed. The next step is image segmentation, which is used for contour detection or segments the multiple objects in a single image so that the classifier can quickly detect the objects and faces in the picture. The next step is to use Haar-Like features algorithm, which is proposed by Voila and Jones for face detection. This algorithm used for finding the location of the human faces in a frame or image. All human faces share some universal properties of the human face like the eye's region is darker than its neighbour pixels and nose region is brighter than eye region.

The haar-like algorithm is also used for feature selection or feature extraction for an object in an image, with the help of edge detection, line detection, center detection for detecting eyes, nose, mouth, etc. in the picture. It is used to select the essential features in an image and extract these features for face detection. The next step is to give the coordinates of x, y, w, h which makes a rectangle box in the picture to show the location of the face or we can say that to show the region of interest in the image. After this, it can make a rectangle box in the area of interest where it detects the face.

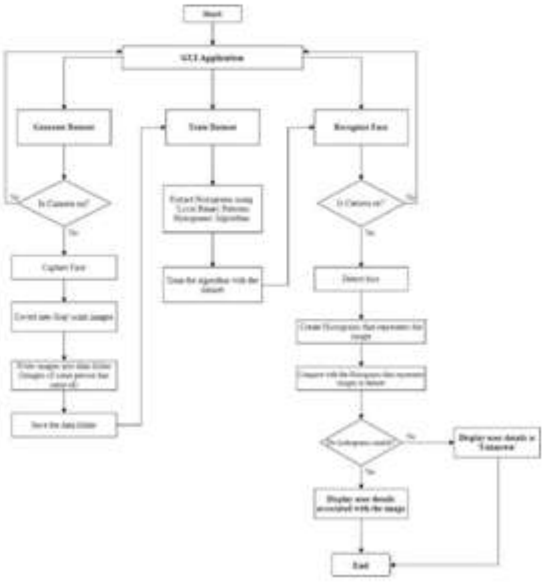

Start
GUI Application
Generate Dataset
Train Dataset
Recognize Face
Is Camera on?
No
Yes
Capture Face
Convert into Gray scale images
Save the data folder
Extract Histograms using Local Binary Patterns Histograms Algorithm
Train the algorithm with the dataset
Is Camera on?
Detect face
Create Histogram that represents the image
End

Fig 6: Proposed System Architecture of application and it's work flow

IV. CONCLUSION

The implementation phase involves putting the project plan into action. This phase is started when the majority of the code for the program is written. It includes the translation of the requirements specified in requirement phase into a logical structure that can be implemented in a programming language. The code is written in Jupyter Notebook as an integrated development environment (IDE). This emulator helped to implement the project in real-like environment.

A. Tools Used:

1. Camera Integrated System: We can use either internal or external camera for this project. The internal camera can be our webcam that is alreadypresented in our system and the external camera can be joined externally to our system (e.g. Logitech webcam).
2. Jupyter Notebook: We used Jupyter notebook as an integrated development environment (IDE). It is very easy to use and also flexible because as its namesuggest, we can write notes and execute code in thesame IDE. Other IDE that can be used are sublime text, PyCharm and so on.

B. Libraries Used:

OpenCV: OpenCV is an open source computer vision and machine learning free software library This library has more than 2500 optimized algorithms, which can be used to detect and recognize faces and to generate dataset.

1. TensorFlow: TensorFlow is Google's Open-Source Machine Learning Framework for dataflow programming across a range of tasks. This library can be used with TFlearn to train the model and to make predictions.
2. TFlearn**:** TFlearn is a modular library in python that is built on top of core TensorFlow. It is a deep learning library. We can use either Keras or TFlearn as a TensorFlow framework.

C. Algorithm Used:

□ Local Binary Pattern Histogram:

LBPH (Local Binary Pattern Histogram) is a Face-Recognition algorithm it is used to recognize the face of a person. It is known for its performance and how it is able to recognize the face of a person from both front face and side face.

□ Steps of the Algorithm:

Parameters: The LBPH uses 4 parameters:

□ Radius:

The radius is used to build the circular local binary pattern and represents the radius around the central pixel. It is usually set to 1. Between two circles we take this as scaling factor.

□ Neighbors:

The number of sample points to build the circular local binary pattern. Keep in mind: themore sample points you include, the higher thecomputational cost. It is usually set to 8.

□ Grid X:

The number of cells in the horizontal direction.The more cells, the finer the grid, the higher thedimensionality of the resulting feature vector. Itis usually set to 8.

□ Grid Y:

The number of cells in the vertical direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

□ Extracting the Histograms:

Now, using the image generated in the last step, wecan use the Grid X and Grid Y parameters to divide the image into multiple grids.

As we have an image in grayscale, each histogram (from each grid) will contain only 256 positions (0~255) representing the occurrences of each pixel intensity. Then, we need to concatenate each histogram to create a new and bigger histogram. Supposing we have 8x8 grids, we will have 8x8x256=16.384 positions in the final histogram. The final histogram represents the characteristics of the image original image.

□ Performing the face recognition:

In this step, the algorithm is already trained. Each histogram created is used to represent each image from the training dataset. So, given an input image, we perform the steps again for this new image and creates a histogram which represents the image. So to find the image that matches the input image we just need to compare two histograms and return the image with the closest histogram. We can use various approaches to compare the histograms (calculate the distance between two histograms), for example: euclidean distance**,** chi-square**,** absolutevalue, etc.

The algorithm output is the ID from the image withthe closest histogram. The algorithm should also return the calculated distance, which can be used asa 'confidence' measurement.

Often in LBPH algorithm we get fooled about the 'confidence' name, as lower confidences are better because it means the distance between the two histograms is closer. We can then use a threshold and the 'confidence' to automatically estimate if thealgorithm has correctly recognized the image.

We can assume that the algorithm has successfully recognized if the confidence is lower than the threshold defined.

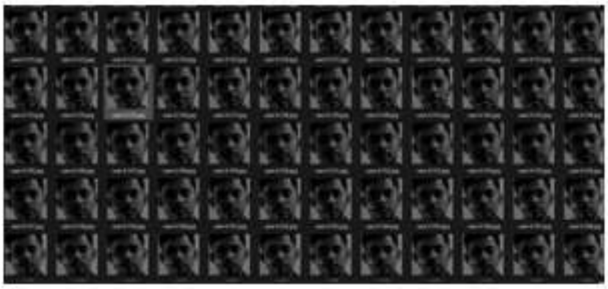
D. Data Collection:

In our paper, the data is collected in the form of Images. For data collection, we used the python openCV. It combines the best qualities of OpenCV, C++, API and Python language. OpenCV supports a lot of algorithms related to Computer Vision and Machine Learning. We used the haarcascade_frontalface_default.xml file for detecting the frontal faces and cropping it to the sizewe required and saved in the folder which is used later on to train the model. Detection=cv2.CascadeClassifier("haarcascade_fro ntalface_default.xml")

E. IMPLEMENTATION RESULTS:

- Generation of Dataset:

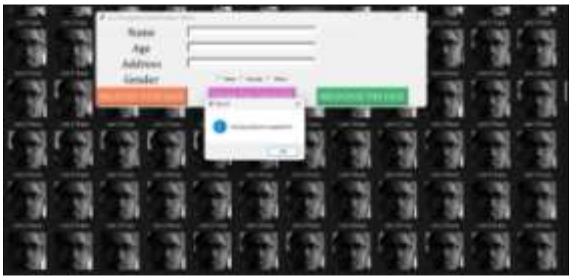
In generation of dataset, the data is collected through the integrated camera. It coverts the live RGB images to Gray scale while capturing. The capturing stops when the images collected are 200 and these 200 images are saved in the data folder.



Output Figure 1: Saving the data collected.

- Training the dataset:

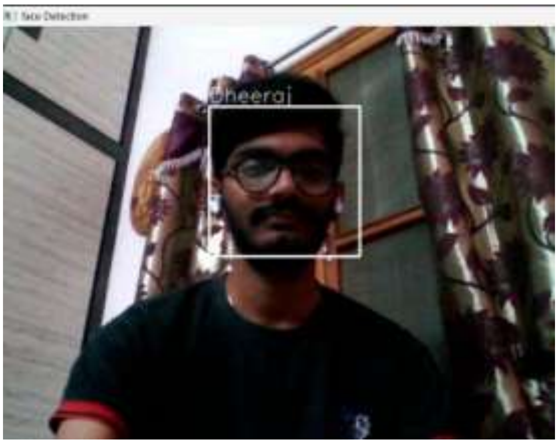
In training of the dataset, the images which are collected through the generation of dataset are trained and stored in a XML file. After training the message box displays with the "Training dataset complete!!!" through which we can understand that our data is successfully trained and saved the resultsin xml file.



Output Figure 2: Training the collected data

- Facial Recognition of authorized user:

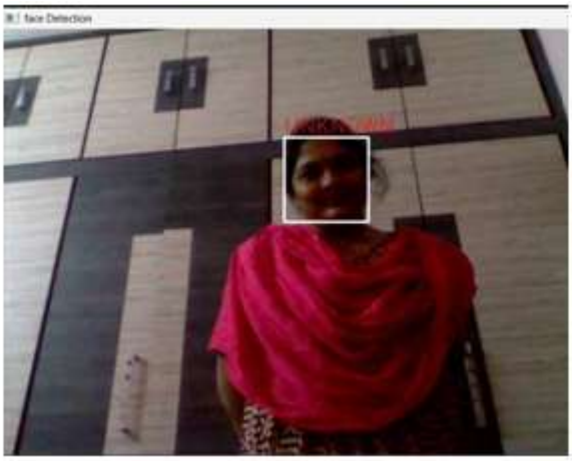
In case of an authorized user, the facial recognition is done and checks for the image and id of the user in the database for validation. If the image match with the data (id and user image) it prints the "User Name" on top of the rectangular box of the face recognition in a window.



Output Figure 3: Authorized User 1

- Facial Recognition of unauthorized user:

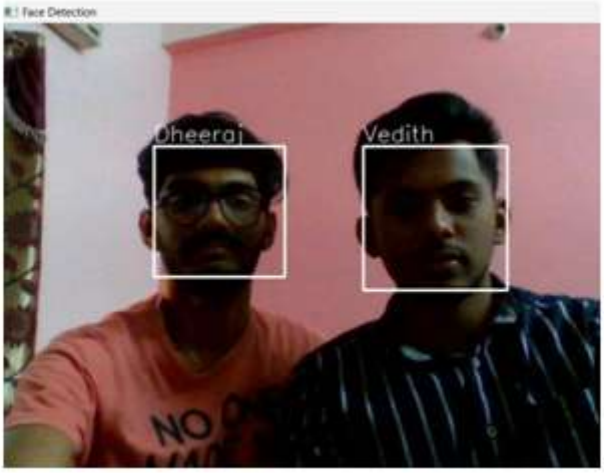
In case of an unauthorized user, the facial recognition is done and checks for the image and id of the user in the database (id and user image) for validation. The image doesn't match with the data so prints as "UNKNOWN" on top of the rectangular box of the face recognition in a window.



Output Figure 6: Unauthorized User

- Facial Recognition of two authorizedusers:

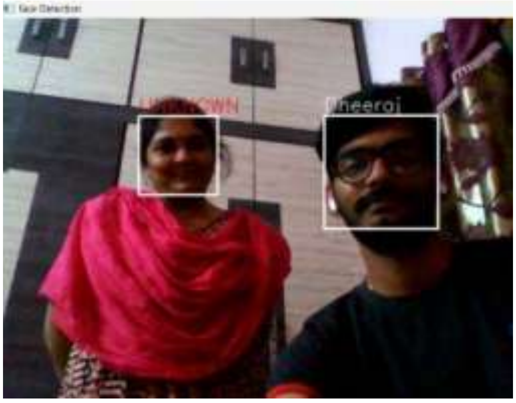
In case of two or more authorized users, the facial recognition is done and checks for the images and ids of the users in the database for validation. If the images match with the data (ids and user's images) it prints the respective "User's Name" on top of therectangular box of the face recognition in a window.



Output Figure 7: Two authorized users

- Facial Recognition of authorized and unauthorized user:

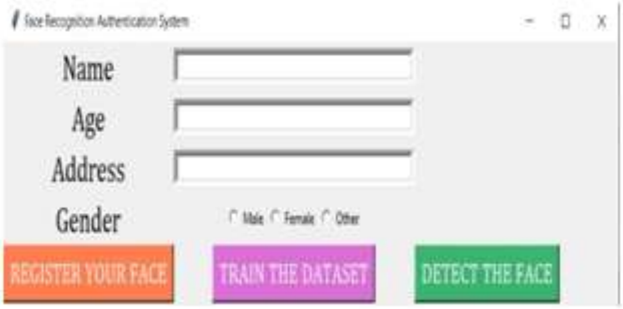
In case of authorized user and an unauthorized user, the facial recognition is done and checks for the images and ids of the users in the database for validation. If the images match with the data (ids anduser's images) it prints the respective "User's Name" or else it prints "UNKNOWN" on top of the rectangular box of the face recognition in a window.



Output Figure 8: An authorized user and an unauthorized user

- Application Window:

Our entire face recognition is converted as GUI application and further connected with database for validations of authorized users.



Output Figure 9: Entire Application Window

F. Performance Analysis:

In our proposed system, we used LBPH as a facial recognition algorithm in which the predictions of the algorithm generally given by "confidence". According to LBPH algorithm as lower confidencesthe better is the system because the distance betweenthe two histograms is closer when confidence is low.

We can then use a threshold and the 'confidence' to automatically estimate if the algorithm has correctlyrecognized the image. Thus said we can assume that the algorithm has successfully recognized if the confidence is lower than the threshold defined.

We used the threshold value as 85 in this paper and below are some of the test images where we performed the confidence predictions on the dataset, we divided the testing and training set and performed the analysis using LBPH predictions. The user id , test and confidence rate is shown in the below table:

TABLE I : PERFORMANCE ANALYSIS

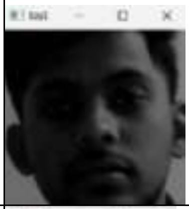
User ID	**Test Image**	**Confidence (Prediction in LBPH)**
1		40.38
2		55.97
3		62.49
4		70.66

Table II: Performance Analysis of test images

IV. COMPARATIVE ANALYSIS

A. Study on face recognition techniques and neural network

Algorithms used in this paper are PCA, MPCA and Neural Network. It showed the performance result as 66.67%. The main advantage of this proposed work in the paper is to reduce dimensionality and drawback is class separability remain same in this model.

B. Image-based Face Detection and Recognition: “State of the Art”

Algorithms used in this paper are Support Vector Machine and PCA which showed 71.15% as performance rate. The model is more effective in high dimensional spaces and is not suitable for largedatasets.

C. OpenFace: A general-purpose face recognition library with mobileapplications

Algorithm used in this paper is convolutional neural networks (CNN) resulted the performance rate of 70%. By this algorithm the memory requirements are reduced, and the number of parameters to be trained is correspondingly reduced. If the CNN has several layers then the training process takes a lot of time if the computer does not consist of good GPU which is the drawback in this model.

D. A Review of Face Recognition Technology

In this paper linear discriminate analysis (LDA) is used as algorithm and showed performance rate of 72%.It seeks directions that are efficient for discrimination between the data before use and within the class the scatter matrix is always single, since the number of pixels in images is larger than the number of images so it can increase detection of error rate if there is a variation in pose and lighting condition within same images which increases the error rate.

E. Study of Face Recognition Techniques: A Survey

MAHCOS Distance with the performance rate of 75% is proposed in the study. It exhibited highresolution but contains less information as side facesare considered.

F. Face Recognition System using Artificial Neural Networks Approach

Algorithm used in this paper is artificial neural networks (ANN). The performance rate is between 73-75% and it can handle large amount of data sets. It exhibits some of the drawbacks of hardware dependence and unexplained functioning of the network.

G. Deep facial recognition system using computational intelligent algorithms.

In this paper K-nearest neighbors is used and performance rate of 68% is showed. It stores the training dataset and learns from it only at the time of making real time predictions. This makes the KNN algorithm much faster than other algorithms that require training and accuracy depends on the quality of the data.

TABLE II: COMPARATIVE ANALYSIS

S.NO	**Algorithm**	**Performance**
1.	PCA, MPCA, Neural Network.	66.67%
2.	Support Vector Machine, PCA	71.15%
3.	CNN	70%
4.	Linear Discriminate Analysis (LDA)	72%
5.	MAHCOS Distance	75%
6.	Artificial neural network (ANN)	73-75%

7.	K-nearest neighbour	68%
8.	LBPH (our proposed algorithm)	76% to 85% for any type of data and 85% (for largerdata)

Table III: Comparative analysis of algorithms and their performance rate.

V. CONCLUSION

Our proposed system of facial recognition system using LBPH algorithm can be used as authenticationsystem which provides additional security to the system based on login. LBPH is one of the easiest face recognition algorithms and is provided by the OpenCV library which makes the implementation easy. Algorithm shows great results, mainly in a controlled environment. Through this paper we showed the performance results between 76% to 85. The proposed system is more suitable and shows better results of performance for larger data.

VI. REFERENCES

[1] Nasution M. I. P, Nurbaiti N, Nurlaila N, Rahma T. I. F and Kamilah K (2020) "Face Recognition Login Authentication for Digital Payment Solution at COVID-19 Pandemic," in (IC2IE),(pp.48-51)

[2] Kao Y, Gu H and Yuan S (2020) "Personal Based Authentication by Face Recognition," Fourth International Conference on Networked Computing and Advanced Information Management, (pp. 581-585)

[3] Pradhan M (2015), "Next Generation Secure Computing: Biometric in Secure E-transaction," International Journal of Advance Research in Computer Science and Management Studies, vol. 3, no. 4,(pp.473-489).

[4] Aayat Shdaifat, Randa Obeidallah, Ghadeer Ghazal, Alaa Abu Srhan Hashemite University, Zarqa, Jordan (2020) "A proposed Iris Recognition Model for Authentication in Mobile Exams."https://doi.org/10.3991/ijet.v15i12.13741

[5] A. Boehm et al. (2013), "SAFE: Secure authentication with Face and Eyes," 2013 International Conference on Privacy and Security in Mobile Systems (PRISMS), 2013,(pp.1-8) .

[6] White D, Dunn JD, Schmid AC, Kemp RI (2015), "Error rates in users of automatic face recognition software. PLoS One". doi: 10.1371/journal.pone.0139827.pmid:26465631

[7] Janelle Mason, Rushit Dave, Prosenjit Chatterjee, Ieschecia Graham-Allen, Albert Esterline, Kaushik Roy (2020),"An Investigation of Biometric Authentication in the Healthcare Environment, Array", Volume 8, https://doi.org/10.1016/j.array.2020.100042

[8] Zulfiqar M,Syed F, Khan M. J and Khurshid K (2019), "Deep Face Recognition for Biometric Authentication," (ICECCE),(pp.1-6).

[9] Schiller Dominik, Huber Tobias, Dietz Michael, and André Elisabeth (2020). "Relevance-based data masking: a model-agnostic transfer learning approach for facial expression recognition," Frontiers in Computer Science, Vol.2, Article 6.

[10] Almabdy, S.; Elrefaei, L (2019)," Deep Convolutional Neural Network-Based Approaches for Face Recognition." Appl. Sci. https://doi.org/10.3390/app9204397

[11] Zhao W, Chellappa R, Rosenfeld A, Phillips P. J(2003)," Face Recognition: A Literature Survey," ACM Computing Surveys, pp. 399-458, 2003

[12] Harguess, J., Aggarwal, J.K. (2009)," A case for the average-half-face in 2D and 3D for face recognition," IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, (pp. 7- 12).