# ARM BASED SECURITY SYSTEM USING LINEAR DISCRIMINANT ANALYSIS

**F. Maria Hadria and S. Jayanthy**

*Department of Electronics and Communication Engineering, Sri Ramakrishna Engineering College, India*

*Abstract*

*This paper presents an ARM Based Security system using facial recognition Techniques. Also a comparison analysis between Principal Component Analysis (PCA), Independent Component Analysis (ICA) and Linear Discriminant Analysis (LDA) algorithms for facial recognition implemented in ARM Cortex M4 processor is done. A training database is created with images of all the authenticated users. The image of the user, whose identity is to be authorized, is captured using a webcam and the ARM microcontroller processes the algorithms to convert the images into vectors and components which are then compared with the images existing in the training database. Results are displayed in a LCD. To compare the performance of the algorithms ORL image database is used. The performance parameters used for comparison are Recognition rate and Cumulative Match score (CMS). The experimental results indicate that LDA algorithm outperforms in terms of recognition accuracy and CMS and gives best results under different illumination conditions, various expressions and poses. It can also be observed that the execution time decreases drastically when executed in ARM Cortex M4 microcontroller compared to execution in MATLAB.*

*Keywords:*

*Facial Recognition, Linear Discriminant Analysis, ARM Cortex M4 Microcontroller, Execution Time, Recognition Accuracy*

## 1. INTRODUCTION

Over the last few years, a new area of engineering allows the identification of a person on the basis of his/her biological characteristics such as voice, features of face and iris patterns. Facial recognition systems are applications that are capable of detecting and identifying an individual from images or a video sequence.

Number of commercial applications requires face recognition technology. Face recognition requires comparing an image with database of stored images. There are different types of face recognition algorithms. For face recognition and comparison in this paper three appearance-based methods namely Principal Component Analysis, Independent Component Analysis and Linear Discriminant Analysis), which have advanced face recognition techniques are used. These methods operate directly on an image-based representation. The features of the training images are extracted into a subspace. In PCA, subspace consists of vectors that correspond to the maximum variance direction in the original image space. The subspace created represents only the face object. Unlike PCA, ICA reduces second order and higher-order dependencies in the input data and converts it into basic vectors that are statistically independent. LDA projects vectors in the subspace that distinguish faces of different persons.

Since face recognition is a real time application it will consist of loop carried dependencies which will include lot of computation complexity. Hence execution time of algorithm increases thus increasing the delay in real time application. Time complexity can be reduced by implementing the algorithm in high speed processors. In the proposed work ARM Cortex M4 portable embedded platform that supports floating-point operation will be used to improve the real time performance of the system.

In these algorithms, the entire face is considered and hence large variation in pose or illumination will affect the recognition rate. Hence the performances of the algorithms are compared for variation in illumination, pose and expressions.

Section 2 deals with Literature survey, section 3 describes the image processing algorithms namely PCA, ICA and LDA, section 4 gives the hardware description of the proposed work and section 5 deals with the experimental results, Conclusion is given in section 6.

## 2. LITERATURE REVIEW

Face recognition technology has developed rapidly and has been applied to feature search system, authentication system, and access control system. An efficient embedded system based on face recognition Technology is a major research topic and still a challenge.

Most technologies are based on passwords, radio frequency identification (RFID) sensors, fingerprints. In [1] a system to enhance security using facial recognition and RFID is proposed. This system was ineffective because the RFID tags can be easily generated if the PIN is known and therefore this system can be hacked even without the image of the person and the camera takes few seconds to become active after the movement is sensed.

Amongst the techniques biometric authentication system based on image processing is efficient for security issues. Face recognition technology using image processing is an option instead of finger print as biometric system [6],[7]. Most of the works on Face recognition system using embedded platform are implemented using PCA.

In [2] authors have proposed a security system that consists of camera, and face-detection and face-recognition algorithms. They have used PCA algorithm for face detection. The disadvantage in their system is their system identified the least possible number of principal components and if the number of principal components were fewer than or the same as the number of original variables, the chances of leading to deviations in recognitions increased and failure rate was more.

In [3], a security system is implemented using face recognition. In this authors have used PCA algorithm executed in MATLAB. When the face is recognized an SMS alert will be given to authorized person using GSM. The accuracy of the system was just 85%. The execution time is also higher because the program was executed in MATLAB.

In [4], a facial recognition based security system using sensors and 8 bit embedded controller is developed. This algorithm is very

simple and will not be able to work under poor lighting conditions. The algorithm is orientation dependent. Hence end point accuracy is poor. Since their design transmits images in the form of video, the bandwidth had to be compressed. Therefore, performance had to be compromised.

In [5], a system is proposed which uses ARM 11 as the controller for electronic parts. They have used Haar feature based Cascade classifier as face recognition module. The software is implemented using MATLAB thus increasing the computational time and delay in real time applications.

In this paper face recognition algorithm is implemented in ARM Cortex M4 process thus reducing the computation time and algorithm used is LDA which gives very high recognition accuracy.

# 3. IMAGE PROCESSING ALGORITHMS

Three image processing algorithms namely Principal Component Analysis, Independent Component Analysis and Linear Discriminant Analysis are explained in the following sections.

## 3.1 PRINCIPAL COMPONENT ANALYSIS

Principal components analysis is a procedure for identifying a smaller number of uncorrelated variables, called principal components, from a large set of data. The goal of principal components analysis is to explain the maximum amount of variance with the fewest number of principal components. The purpose of PCA is to reduce the large dimensionality of the data space to the smaller dimensionality of feature space. It is used to determine the most representative features among the data [6]. In PCA algorithm, training set is created in the database, consisting of M images.

$$I = \{i_1, i_2, i_3, \ldots, i_M\}$$

These images are then converted into matrices with the dimension $N \times N$.

$$Dimensions = height \times width$$

The next step is to find the image vectors ($\lambda$) also called the eigen vectors for the images under consideration using Eq.(1),

$$A - \lambda I = 0 \tag{1}$$

Normalize the face vector by calculating the average face vector $Y$ and by subtracting the average face vector from each image vector. Next, the eigenvector is computed by calculating the covariance vector $C$, which results in $N^2$ eigenvectors as shown in Eq.(2).

$$Cov(A) = \Sigma[(\lambda_1 - Y)(\lambda_2 - Y),\ldots,(\lambda_i - Y)]/N \tag{2}$$

Then, $K$ eigenvectors are calculated using Eq.(3) which can represent the entire training set and the weight of each eigenvector is calculated which is also known as Principal Components.

$$P = U(Cov(A) \times Y - T) \tag{3}$$

where, $T$ is the transformed data matrix which can be calculated using Eq.(4).

$$T = XP \tag{4}$$

The calculated Principal Components of the image are compared with the components of the images in the database until a match is found. If found, then the user is authenticated.

## 3.2 INDEPENDENT COMPONENT ANALYSIS

Independent component analysis (ICA) is a method for finding underlying factors or components from multi-dimensional data. These components are called independent components [7] looks for components that are both statistically independent, and non-Gaussian. Independent Component Analysis is based on the assumption that source signals are not only uncorrelated, but are also independent. Essentially, if two variables are independent then the value of one variable provides absolutely no information about the value of the other variable. Training set ($I$) of $M$ images is created and the dimensions are calculated by converting the images into 2D matrices.

$$I = \{i_1, i_2, i_3, \ldots, i_M\}$$

Face vectors for each image are calculated using the Eq.(5),

$$A - \lambda I = 0 \tag{5}$$

The average eigen vector ($\alpha$) is calculated using the Eq.(6),to find the common features existing in all the images.

$$\alpha = i/M [\lambda_1 + \lambda_2 + \ldots + \lambda_m] \tag{6}$$

The average vector is subtracted from the images to find out the linear combinations of the $M$ training set face images to form the Eigen faces using the Eq.(7).

$$\Omega_M = \lambda_m - \alpha \tag{7}$$

Covariance matrix ($C$) is calculated using the Eq.(8),

$$C = 1/M \{\Omega_{(1,2\ldots M)} \Omega_M{}^T = AA^T \tag{8}$$

where, $A = \{\Omega_1, \Omega_2, \ldots, \Omega_m\}$

The $K$ vectors that are uncorrelated and independent are calculated using the separating $matrix(S)$ which is computed using the covariance matrix as in Eq.(9).

$$S = C^T \Omega_{(1,2,\ldots,M)} \tag{9}$$

The Independent Components ($I$) that are determined using Eq.(10) are compared with the components of images in the database to verify a user.

$$I = E (SS^T) \tag{10}$$

For a test image, the difference between the weight vector of the input image and the weight vector of training set is calculated to recognize a user [8].

## 3.3 LINEAR DISCRIMINANT ANALYSIS

LDA is a technique used in Machine Learning, Statistics and pattern recognition to find a linear combination of features that discriminates classes of objects. [9]. LDA tries to model the difference between the classes of data.

The main aim of the LDA is to maximize the between-class differences and, minimize the within-class ones. LDA acquires the different variables of a face and calculates which group the face most probably matches. The LDA builds a set of feature vectors in which variations of different faces are highlighted, while different instances of faces due to illumination conditions, facial expression and orientations are not emphasized

LDA makes use of projections of training images into a subspace defined by the fisher faces known as fisher space. Recognition is performed by projecting a new face onto the fisher space.

LDA approach includes two phases: training and classification. In the training phase, a face space is established from the training samples and the training faces are projected onto the same subspace [10]. The transformation can be computed by applying the Eigen decomposition on the scatter matrices. In the classification phase, an input face is projected into the fisher space and classified using the Euclidean distance.

- Three steps in LDA algorithm gives rise to higher execution time.
- Calculation of the conditional covariance matrices.
- Calculation of the pooled covariance matrix.
- Inversion of the pooled covariance matrix.

The second step does not require to access the data. Similarly, at the 3rd step the inversion of $S$ is not a complex calculation. The 1st step is thus the main bottleneck. The deprivation of the processing time is hardly noticeable in this case. Hence if the algorithm is implemented in highly efficient ARM Cortex processor running at high frequency the computational time will be greatly reduced.

A training database is created with I images of authenticated users and the dimensions are found by converting images into matrices.

$$I = \{i_1, i_2, i_3, \ldots, i_M\}$$

For the training phase, dimensional mean vectors are calculated by finding the Eigen vectors ($\lambda_1, \lambda_2, \ldots \lambda_n$). Each vector is referred to as space image. From the main vectors, static vectors ($m_i$) are calculated using Eq.(11),

$$m_i = \Sigma(\lambda_1, \lambda_2, \ldots \lambda_n)N \qquad (11)$$

In the next step, between class scatter matrix ($S_B$) and within class scatter matrix ($S_w$) are calculated using Eq.(12) and Eq.(13).

$$S_w = \Sigma_N (m_1 + m_2 \ldots m_n) \qquad (12)$$

$$S_B = \sum_N (m_n - m)(m_n - m) \qquad (13)$$

The within class scatter $S_w$ represents how face images are distributed closely within classes and between class scatter matrix $S_B$ represents how classes are separated from each other. The goal of LDA is to maximize $S_B$ while minimizing $S_w$. The images in the training set are divided into the corresponding classes.

To calculate the linear discriminant, the covariance matrix ($C$) is found to evaluate the optimizing factor ($O$) using the Eq.(14) and Eq.(15).

$$cov_j = (S_w - N)(S_B - N) \qquad (14)$$

$$O = inv (cov_j) S_B \qquad (15)$$

The optimizing factor from the Eq.(15) is used for comparing the user's image with the images in the database. Given a test image, the mean image of the training set is subtracted and the result is compared with the training database.

# 4. HARDWARE DESCRIPTION

The proposed system consists of an ARM Cortex M4 microcontroller to process the Principal Component Analysis (PCA), Independent Component Analysis (ICA) and Linear Discriminant Analysis (LDA) algorithms. The webcam in the laptop is used to capture the image of the user [11]. The images

are processed in ARM Cortex M4 microcontroller and the results are displayed using LCD.

## 4.1 BLOCK DIAGRAM

The block diagram of the proposed security system using facial recognition is shown in the Fig.1. The webcam in the laptop is used to capture the image of the user whose identity is to be authorized. The three algorithms namely PCA, ICA and LDA are executed in MATLAB. The MATLAB program is given as input to the embedded coder that converts the M file into C code for execution in ARM Cortex M4 microcontroller. The generated C code is compiled and linked using Keil µvision IDE. The code is ported into the microcontroller and executed. An LCD display is interfaced with the ARM Cortex M4 to display the results. The LCD would display whether the person is authorized or unauthorized.

A data base is initially created for a set of users. The image is captured in the webcam and stored in the database along with the user name. This is the training phase. In the face recognition part, the algorithms compare the image captured by the webcam with set of images in the database.

If the captured image matches with image in the database the LCD displays it as authenticated user along with his name as stored in the database. If match is not found, it displays that the user is not an authenticated user.
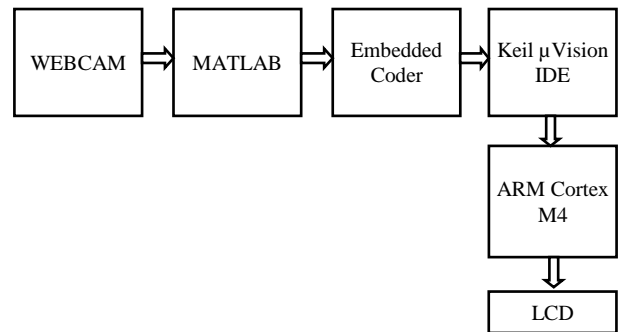


Fig.1. Block Diagram of the Proposed System

## 4.2 SCHEMATIC DIAGRAM

The Security system is designed with the LPC4088 Embedded board. The board consists of ARM Cortex M4 core with 512KB on-chip flash memory for storing the code. The ARM Cortex-M4 was chosen as it supports digital signal processing and SIMD instructions. The core has an integrated hardware floating-point processor. All the algorithms are implemented in The ARM processor. The webcam present in the PC is used for capturing images. The algorithms are coded in MATLAB software installed in the same PC. Embedded Coder is used to convert the MATLAB code into C code. Embedded Coder significantly extends the real-time execution framework provided by MATLAB Coder. By default, the code can be executed with or without a real-time operating system (RTOS) and in single-tasking, multitasking, or asynchronous mode. It is also possible to verify the code execution results using software-in-the-loop (SIL) and processor-in-the-loop (PIL) testing.

The application code is tested, verified, and optimized using Keil µvision IDE which provides source code editing, and program debugging in a single powerful environment. The C code

is downloaded into the flash memory of the ARM Cortex M4 in the LPC4088 board by using a Keil ULINK Debugger. The Fig.2 shows the schematic diagram for the proposed system. LCD controller HD44780 is interfaced to the ARM Cortex M4 microcontroller through a potentiometer to vary the contrast of the display. Data pins 4,5,6,7 are connected to the digital output pins of ARM Cortex M4.
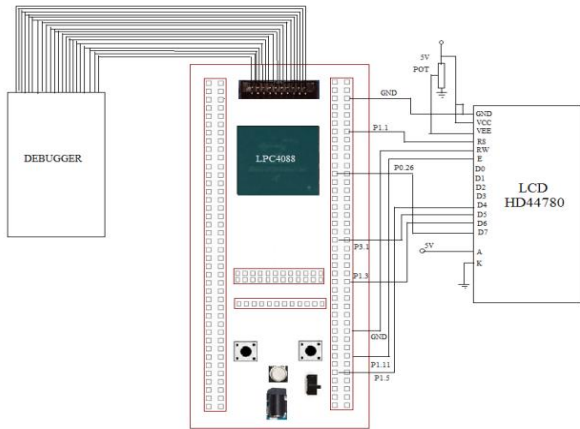


Fig.2. Schematic diagram of ARM based Security System

## 5. EXPERIMENTAL RESULTS

The hardware implementation of the proposed system is shown in Fig.3 and Fig.4. The image of the user is captured using the webcam in the PC. The webcam in the PC is accessed by the MATLAB software using the necessary video packages and defining the correct adaptor name.

The C code is downloaded into the flash memory by using a ULINK2 Keil Debug and an LCD is interfaced to the board to display the results. A potentiometer is used to control the contrast of the LCD display. Power supply to the LPC4088 Embedded board is through the USB cable from PC. The board can also be powered from a DC adaptor.

For checking whether a person is authorized a training database was created. AT&T database is used as the training database for evaluation of parameters. The three algorithms are compared for illumination variation, pose variation and expression variation using AT&T database. The database is having sufficient sample images under variations illumination, pose and facial expressions.

### 5.1 CREATING A DATABASE WITH IMAGES OF AUTHENTICATED USERS

Initially a training database is created with images of all the authenticated users. Webcam in PC is used to capture the image of the user and the face is detected from the image as shown in Fig.5. After the face is detected, the image is cropped and stored in the database with the name given by the user.

The algorithms are developed in MATLAB. Once the training data base is created the algorithm is processed. When a user image is detected the linear discriminants of the image are found by comparing the scatter matrices and then the training database is compared.

In Fig.6, the user is authorized because a similar image of the user was found in the training database. Therefore, it displays that the user is authenticated along with their name as stored in the database.
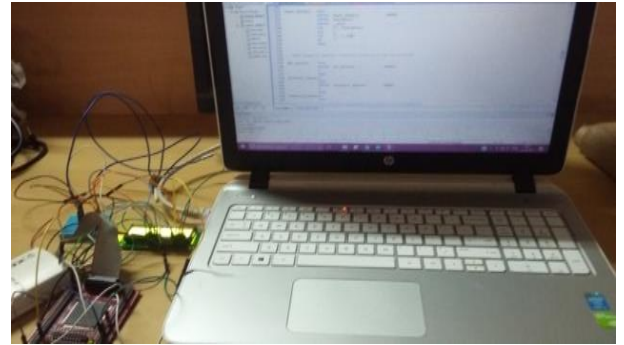


Fig.3. Experimental setup for Security System



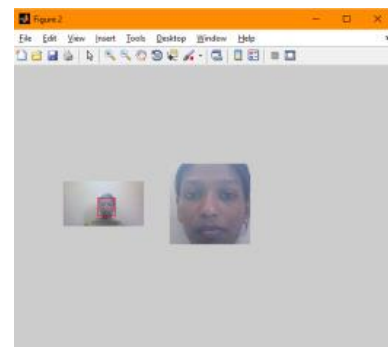Fig.4. Expanded view of the LPC4088 development Board and associated components



Fig.5. Creating a training database

It can be seen from Fig.7 that the algorithm does not authenticate a user if an image of the user is not found in the training database.

### 5.2 GENERATING CODE FOR ARM CORTEX M4

The .m files developed in MATLAB are converted into .c files using embedded coder. The Fig.8 shows the Code generator report displaying the C code that has been converted into .c file using Embedded Coder.

The generated C code from embedded coder is compiled and linked in Keil μVision IDE .The code is tested and debugged and ported in ARM Cortex M4 microcontroller. The porting of code to LPC4088 Embedded board is shown in Fig.9.

During execution in ARM Cortex M4 microcontroller the face recognition algorithm compares the captured image with image stored in training database to check for authentication.



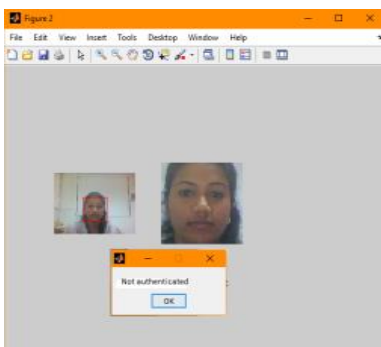Fig.6. Output of LDA algorithm if User is authorized
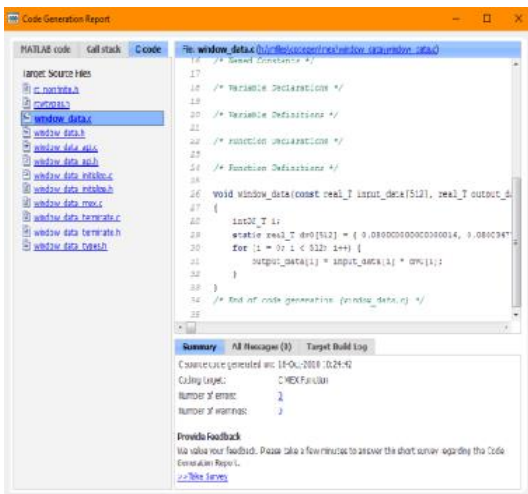


Fig.7. Output of LDA when User is not authorized



Fig.8. Code generator report

The results are displayed in the LCD module connected to the LPC4088 board as shown in Fig.10.

## 5.3 RESULT ANALYSIS

Face expression can have an effect on the appearance of face. The face can be viewed from side view or front view. Environment where images are captured can be dark or bright which can affect the recognition rate. Hence the algorithms are executed for different lighting conditions, facial expressions and poses. For all the experiments, the training database is the same,

consisting images from AT&T database. The testing database is changed according to the variation that has to be evaluated.
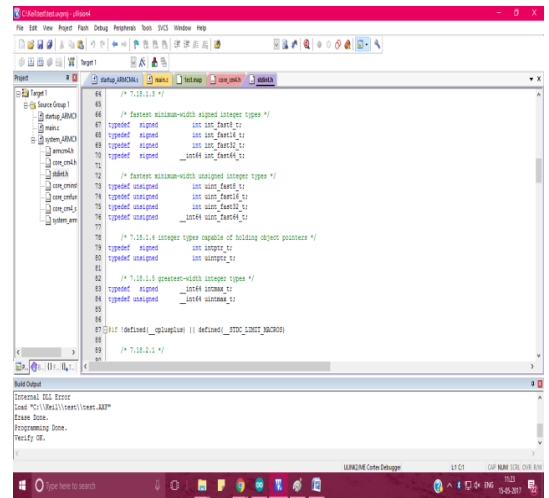


Fig.9. Execution of algorithm in Keil IDE



Fig.10. LCD Output

The Fig.11 shows the training database for evaluation of all three variations.



Fig.11. Training database for evaluation of variations

### 5.3.1 Illumination Variation:

This parameter checks the ability of the algorithm to detect and recognize an individual under different lighting conditions. For this purpose, testing database shown in Fig.12 is created with

images of the same users as in training database but with different lighting conditions.



Fig.12. Testing database for Illumination variation

### 5.3.2  Expression Variation:

Fig.13 shows the testing database for the calculation of expression variation parameter. It determines the ability of the algorithm to identify an individual even when their expression (anger, smile, cry, surprise) changes.

### 5.3.3  Pose Variation:

For calculating the pose variation, a testing database shown in Fig.14 with images of users in different poses is created.

### 5.3.4  Cumulative Match Score Curve:

For analyzing the results of the PCA, ICA and LDA algorithms, the Cumulative Match score, the recognition rates and recognition accuracy of all three algorithms in different cases are evaluated. For calculation of performance and recognition rate, algorithms are executed under various lighting conditions, expressions and poses.

Cumulative Match Score (CMS) curve is the graph showing the differences in the recognition rates of algorithms with respect to the ranks. The performance of the system is directly proportional to the CMS curve. The higher the value of CMS, the better the performance.

The Fig.15 shows the graph of the CMS curve plotted between recognition rate and rank of the algorithm. From the graph it can be shown that LDA CMS.



Fig.13. Testing database for expression variation



Fig.14. Testing database for Pose variation

### 5.3.5  Recognition Rate:

The algorithms were executed for all the three variations using the AT&T database. Each algorithm is executed in a loop for 15 times and the number of times the algorithm correctly identified a user is noted. This value is then converted into percentage to give the recognition rate.

The results for all the three algorithms under different variations are tabulated as shown in Table.1. Linear Discriminant Analysis algorithm has the highest recognition rate compared to the Principal Component Analysis and Independent Component Analysis algorithms under all three conditions.
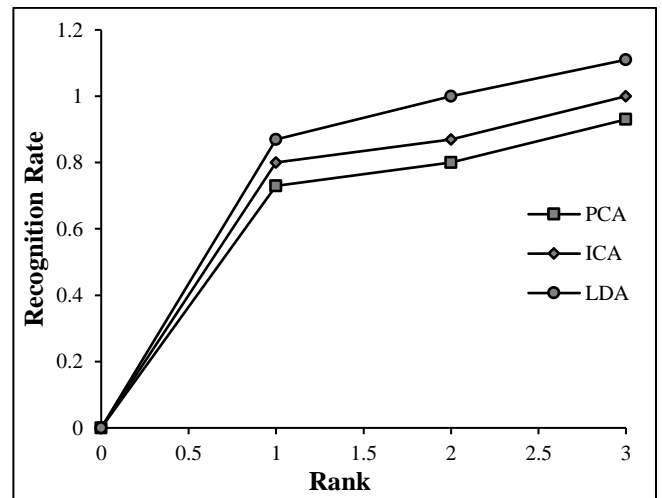


Fig.15. Cumulative Match Score Curve

Table.1. Recognition rate of algorithms under different conditions

| Algorithm | Illumination Variation (%) | Pose Variation (%) | Expression Variation (%) |
|---|---|---|---|
| PCA | 93.34 | 73.34 | 80 |
| ICA | 100 | 80 | 86.67 |
| LDA | 100 | 86.67 | 93.34 |

Recognition Accuracy for all the three algorithms was calculated. It is the number of samples recognized divided by the total number of samples. The results are tabulated in Table.2. LDA gives the highest recognition accuracy of 93% compared to PCA and LDA.

Table. 2. Recognition Accuracy of algorithms

| Algorithm | Recognition Accuracy |
|---|---|
| PCA | 0.8223 |
| ICA | 0.8889 |
| LDA | 0.9334 |

### 5.3.6 Execution Time:

The Table.3 shows the differences between the execution times when the algorithms are executed in MATLAB software and when executed in ARM cortex M4 microcontroller.

Table.3. Execution time in MATLAB and ARM Cortex M4 microcontroller

| Algorithm | Execution Time in MATLAB (ms) | Execution time in ARM Microcontroller (µs) |
|---|---|---|
| PCA | 7.562 | 250 |
| ICA | 7.257 | 370 |
| LDA | 11.154 | 967 |

From the Table.3, it can be observed that the execution of algorithms in ARM Cortex M4 microcontroller effectively reduces the execution time when compared to execution in MATLAB software. Further it is noted that LDA has a slightly larger execution time compared to PCA and ICA. The Table.4 gives the comparison of execution time and recognition accuracy of the proposed work with related works.

Table.4. Comparison with related works

| Parameter | Proposed work | [12] | [13]) | [14] |
|---|---|---|---|---|
| Recognition Accuracy in % | 93.34 | 97.50 | 74.47 | 80 |
| Execution Time | 967µs | 2.14s | - | - |

The database used in the proposed work and [12] is AT&T Face Database. Results are given for LDA face recognition algorithm. In [12], a two dimensional LDA is used and hence the accuracy is slightly higher compared to the proposed work, but execution time is very high compared to proposed work.

In [13], Yale database is used. It gives an accuracy of 74%. In paper [14], SURF algorithm is used for feature matching. In this paper, authors have used LDA for the edge dimensions reduction to live faces from their set of data images. They have got an accuracy of 80% for a data set of 40 images.

Compared to these results method proposed gives better results both in terms of accuracy and execution time.

## 6. CONCLUSION

The Security system using Linear Discriminant Analysis (LDA) algorithm proves to be more efficient when compared to Principal Component Analysis (PCA) and Independent Component Analysis (ICA) under different conditions. From the results it can also be noticed that LDA algorithm has better recognition rates in diverse conditions of illumination, different expressions and various poses of users. Therefore, when LDA algorithm is used, it is not required for the users to be in good lighting conditions or have the same expressions and poses. From the results, it can be observed that the execution times of algorithm decreases drastically when executed in ARM Cortex M4 microcontroller, which proves the efficacy of the proposed system for real time application

## REFERENCES

[1] Adnan Affandi, Mohammed Awedh, Mubashshir Husain and Ahmed Alghamdi, "RFID and Face Recognition based Security and Access Control System", *International Journal of Innovative Research in Science, Engineering and Technology,* Vol. 2, No. 11, pp. 27-34, 2013.

[2] Michel Owayjan, Amer Dergham, Gerges Haber, Nidal Fakih, Ahmad Hamoush, Elie Abdo, "Facial Recognition Security System", *Proceedings of International Conference on Electrical, Electronics and Optimization Techniques*, pp. 75-81, 2016

[3] C.A. Athira, Hashlin P Jose, T. Jini John and Aswathi Wilson, "Security Alert using Face Recognition", *International Journal of Advances in Computer Science and Technology*, Vol. 5, No. 12, pp. 176-179, 2016.

[4] J. Shankar Kartik, K. Ram Kumar and V.S. Srimadhavan, "Security System with Face Recognition and Embedded Network Video Monitoring Terminal", *International Journal of Security, Privacy and Trust Management*, Vol. 2, No. 5, pp. 53-59, 2015.

[5] Sameerchand Pudaruth, Faugoo Indiwarsingh and Andrakant Bhugun, "A Unified Intrusion Alert System using Motion Detection and Face Recognition", *Proceedings of 2nd International Conference on Machine Learning and Computer Science*, pp. 17-20, 2013.

[6] Chathunika Gamage and Lasantha Seneviratne, "Development of a Learning Algorithm for Facial Recognition under Varying Illumination", *Proceedings of International Conference on Information and Automation for Sustainability*, pp. 119-126, 2014.

[7] Chengyun Liu, Zhenxue Chen, Faliang Chang and Kaifang Wang, "Face Recognition Algorithm based on Improved Facial Model", *Proceedings of International Conference on Natural Computation*, pp. 927-931, 2014

[8] Maria Rosario D. Rodavia and Orlando Bernaldez, "Web and Mobile based Facial Recognition Security System using Eigenfaces Algorithm", *Proceedings of IEEE International Conference on Teaching, Assessment, and Learning for Engineering*, pp. 86-92, 2016.

[9] Sizhi Zhong, Youguang Chen and Shuchun Liu, "Facial Recognition using Local Feature Selection and Extended Nearest Neighbour Algorithm", *Proceedings of International Conference on Computational Intelligence and Design*, pp. 328-331, 2014.

[10] Chaoqun Jing Tian and Liang Qijun Lei, "Camera Security Network Design and Realization based on PCA Facial Recognition Algorithms", *Proceedings of IEEE International Conference on Electronic Information and Communication* Technology, pp. 68-75, 2016.

[11] Ala Eldin Omer and Adil Khurran, "Facial Recognition using Principal Component Analysis based Dimensionality

Reduction", *Proceedings of International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering*, pp. 73-78, 2015.

[12] J. Ye, R. Janardan and Q. Li, "Two-Dimensional Linear Discriminant Analysis", *Proceedings of International Conference on Advances in Neural Information Processing* Systems, pp. 1569-1576, 2004.

[13] Riddhi A. Vyas and M. Shah, "Comparison of PCA and LDA Techniques for Face Recognition Feature based

Extraction with Accuracy Enhancement", *International Research Journal of Engineering and Technology*, Vol. 4, No. 6, pp. 3332-3336, 2017.

[14] Narpat A. Singh, Manoj B. Kumar and Manju C. Bala, "Face Recognition System based on SURF and LDA Technique", *International Journal Intelligent Systems and Applications*, Vol. 2, pp. 13-19, 2016.