Proceedings of the International Conference on Electronics and Renewable Systems (ICEARS 2022)
IEEE Xplore Part Number: CFP22AV8-ART; ISBN: 978-1-6654-8425-1

Attendance Tracking with Face Recognition Through Hidden Markov Models

Dipshikha Biswas1, Sarada Jayan2, Suneel Nadipalli1, Sneha B1, Subramani R2

[1]Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, India

[2]Department of Mathematics, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, India

dipshikha.biswas461@gmail.com, nsuneel89@gmail.com, snehabalaji74@gmail.com , r_subramani@blr.amrita.edu

2022 International Conference on Electronics and Renewable Systems (ICEARS) | 978-1-6654-8425-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICEARS53579.2022.9751888

***Abstract*—Facial recognition is one of the most secure ways to identify a person. Manual attendance in organizations, be it in classrooms or libraries or even attendance for teachers is truly a hassle. Due to the inception of Hidden Markov Model (HMM), they have worked well with image data and it has plethora of facial recognition applications. In this paper, yet another application of face recognition with HMM is explored, where it is integrated with Singular Value Decomposition (SVD) and track the attendance of the students present in a database. HMMs deal with data in the form of states and sequences. Face recognition looked through the lens of HMMs which can be framed in the following manner: a face is split into regions vertically (forehead, chin, etc.) and a particular sequence is always preserved. A rectangular window of fixed size is passed over every test image, and for every vector obtained, the probability of data is calculated. For training, probability computation is done with the help of the Baum Welch algorithm. This whole model is connected to a simple program to keep track of the students leaving and entering the classroom, marking their presence only and updating the same information in the college's database.**

Keywords—Facial Recognition, Attendance Tracking, Hidden Markov Model, Baum Welch algorithm, Singular Value Decomposition.

I. INTRODUCTION

Face Recognition is a prominent area of research as it provides a secure means of identifying an individual. Security Cameras, Location Monitoring, and other areas use facial recognition as their means of identification. It also has a wide variety of applications in various fields such as Robotics, Digital Cameras, Games, and many others. Face recognition research has been conducted, with various benefits and drawbacks. This is due to the complexity of the human face. As a result, the difficulty in developing an ideal computational model for human face recognition is not common.

The security of identification using this method proves to be vital for attendance tracking. In most schools and colleges all over India, attendance tracking is still a manual task and it takes at least fifteen minutes daily to accomplish it. All this time and effort can be saved if the same task was automated. Our model recognizes a face and runs that through a database of students. Once there is a match, the attendance of that student is marked with the appropriate date and time. This is a simple extension of a regular face recognition model, but it proves to be quite effective [1].

Facial recognition can be done using a variety of methods, however, HMM is the most popular one due to its rich mathematical essence. HMMs are also applied in other areas such as Bioinformatics for genome sequencing and gene expression. It has also found its place among speech recognition and gesture recognition models.

In this paper, a seven-state HMM is used, modelling the seven most prominent regions on our face [5]. Existing works had five states HMM but in this paper, a seven state model has been implemented. This model covers more defined regions and partitions of the face. Additionally, this model contains more overlapping segments that would reduce the missing out of facial features unlike in the five-state model that was likely to miss out important facial features[12, 13].

SVD is used to extract the most prominent features from each image. This has a unique property of dimensionality reduction which makes it ideal in our application. It reduces the computing complexity and increases efficiency.

The Olivetti Research Laboratory (ORL) dataset is used to train our model. The dataset used was relatively small as well, having only 400 sample. The dataset file consisted of 40 directories which stored 10 images in each of the directories.

The trained model is then applied on unseen images as well as live ones, taken using a camera (or web camera). This paper is intended to serve as an attendance tracker as well, it matches students' faces as they scan it in real-time to an existing database and to bring about such functionality, SQL was used.

II. RELATED WORKS

As discussed in the Introduction section, HMMs have been applied to many research areas including face recognition. Face recognition on its own has been a prominent field of research as it holds the key to many useful applications, namely, protection against theft, information security, and personal identification. And combining the best of both worlds, Face Recognition using HMMs is an active field of research as well. There have been many undertakings on the same.

One method involves modeling the human face as an HMM with 5 states, preprocessing, block extraction and feature extraction. The data is used to train the recognition system and construct the HMM [1]. Another method involves applying Karhunen Loeve Transform coefficients to obtain the feature vectors used to represent the HMM modeled after a given face [3]. Sachin et al [5], take on a different approach to the same problem by using coefficients obtained from SVD and the HMM as a classifier.

Each of the above papers performs testing by comparing a feature vector that represents a given face to those in the dataset. Rajput et al [8] have implemented a two-layer architecture that has been trained to classify regions as containing faces or not. The final stage has an SVM detect the face or non-face class if any of the sub image was left undecided. While traditional methods restrict themselves to 1D data, Bobulski [10] takes on a new challenge as their paper deals with 2D HMM face recognition.

III. THEORETICAL OVERVIEW

A. HMM

HMM has been successfully used for speech recognition and more recently in action and facial detection. This has proved more accurate after the consideration of all the facial features stored in a 2D matrix. This helps the collection of all the facial data without losing any features. Statistical inference also includes that 1D and 2D continuous HMM worked well in facial recognition especially.

Markov chains with HMM are a type of statistical model generally used to describe the statistical properties of a signal [3]. Since the data here is considered with respect to its application in speech recognition, it is naturally one-dimensional along the give time axis. The corresponding fully connected two-dimensional HMM, on the other hand, would provide a significant computational challenge.

The following parameters help define the specifications of an HMM:

- $N = |S|$, number of states in the model
- $M = |V|$, number of various observation symbols
- $A = \{a_{ij}\}$ state transition probability matrix
- $B = \{b_j(k)\}$, observation symbol probability matrix
- π, initial state distribution

B. SVD

SVD is a matrix decomposition method that is primarily used to reduce the dimensionality of matrices. SVD factors a given matrix into three new matrices U, V, and Σ. SVD is applied to each of the blocks that are extracted using

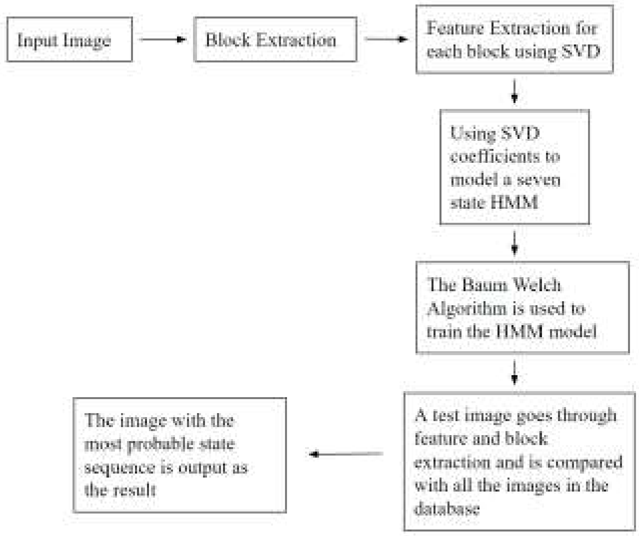
$X = U \times S \times V^T$, where U and V are orthogonal matrices representing eigenvectors and Σ is used for the diagonal matrix of singular numbers representing eigenvalues. Eigenvectors represent the directions along which a linear transformation can act and eigenvalues represent the scale of transformation [2]. One of the properties of SVD is that if the m x n matrix X is real, then the values of the matrices U and V are also real.

There is a plethora of applications of SVD, including facial recognition, watermarking, image compression and detection, signal processing, object detection, scientific computing and many more. SVD is unique in that it decreases the dimensionality of matrices, making calculation easier, and it can be used for any type of matrix [7].

IV. GENERAL WORKFLOW

Our model follows the steps listed below to perform training (on the ORL Dataset) and testing (on an unseen image)

- Block Extraction
- Feature Extraction
- Modelling the HMM
- Training using the Baum Welch Algorithm



A. Block Extraction

HMMs need an observation sequence to model states. Thus, a given 2D image must be transformed into a 1D observation sequence. To accomplish this task, the concept of Block Extraction is used.

These images are segmented into a set of overlapping blocks with the width W, the original width of the image and height L. The overlap between two blocks is measured by the value P. The values of L and P are manually chosen by the programmer. This is a pivotal process as it can greatly increase the accuracy of a model.

By empirical analysis, the value of L is approximately taken to be one-tenth of H ($H/10$), and the value of P should be less than or equal to L - 1 ($P \le L - 1$). Putting all these values together, the length of the observation line T is obtained, which is the total number of blocks for a single image, as follows,

$$T = \frac{(H-L)}{(L-P)} + 1 \tag{1}$$

B. Feature Extraction

Once the block for each image is extracted, feature extraction is applied to each one of these blocks. This is done to extract the most prominent features from each image and use that for recognition and identification. SVD is used to extract features from these blocks. This decomposes each block matrix into a product of three matrices as modelled by the following formula,

$$X_{m\times n} = U_{m\times n} \times S_{m\times n} \times (V_{n\times n})^T \tag{2}$$

Here, S is a diagonal matrix, whereas U and V are orthogonal matrices. After multiple rounds of experimentation and testing, it was found that the coefficients $U(1,1), S(1,1), S(2,2)$ perfectly describe the features of that particular block. Thus, a coefficient matrix of a large size for a single image can be decomposed to just three values per block.

$$C = (coeff_1, coeff_2, \ldots, coeff_n) \tag{3}$$

Here, $coeff$ is the vector containing the three most prominent coefficients for each block in an image[6].

These values cannot be directly used to model an HMM as they are still continuous and discrete values are required for this process. Thus, the procedure of quantization is applied, where every value in the coefficient matrix is made fit to a particular level. This takes place in the following steps,

$$\Upsilon_i = (coeff_{max} - coeff_{min})/D_i \quad (4)$$

Here, $coeff_{max}$ and $coeff_{min}$ are the maximum and minimum coefficients from each coefficient vector in the coefficient matrix. D_i is the level of quantization. Empirical Analysis proves that the values of D_i as 18, 10, 7 for U(1,1), S(1,1), and S(2,2) respectively, give the highest accuracy.

Now each value in the coefficient matrix should be conformed to these distinct levels. The quantized values are obtained from the following,

$$qt_i = (coeff - coeff_{min})/\Upsilon_i \quad (5)$$

Finally, these quantized values are converted into a single discrete value called a label using the following formula,

$$label = qt_1 \times 10 \times 7 + qt_2 \times 7 + qt_3 + 1 \quad (6)$$

These observation vectors are input into the seven-state HMM model.

C. Modelling the Seven-State HMM

For facial recognition, it was proposed that a one-dimensional continuous HMM be used. Assuming that each face is upright and frontal, the different characteristics appear in the known sequence, such as the forehead, nose and then mouth. The image of the face could thus be segmented or divided into seven distinct regions from top to bottom, namely hair, forehead, eyebrows, eyes, nose, mouth, and chin. They are represented as H, F, EB, E, N, M, and C respectively.

The hidden states present in the Markov model would be these seven separate areas.

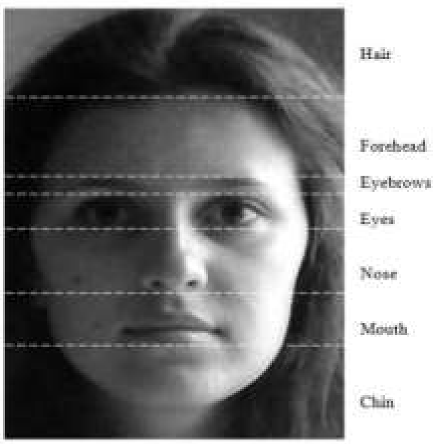


Fig 1. Facial Regions from top to bottom

This arrangement indicates the employment of a top-bottom paradigm, in which only top-to-bottom transitions between neighboring states are permitted [4].

The states of the model correspond to significant facial features like H, F, EB, E, N, M, and C. The observation sequence O is created by overlapping $X \times M$ pixels in an $X \times Y$ image with an $X \times L$ sampling window. A block of L lines makes up each observation vector. Between subsequent observations, there are an M number of lines that overlap successive observations.

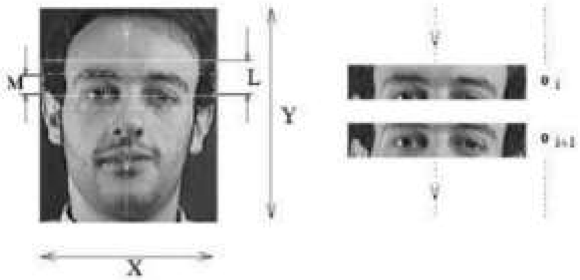


Fig 2. Sampling Technique

The initial state distribution is,

	H	F	EB	E	N	M	C
π=[	1	0	0	0	0	0	0]

The transition matrix A is modelled using

$$A_{ij} = P[q_t = S_j | q_{t-1} = S_i] \quad , \quad 1 \le (i,j) \le N \quad (7)$$

The emission matrix B is modelled using the given formula

$$B = (1/M) \times Ones\,(N, M) \quad (8)$$

where N denotes the number of states and M denotes the number of observables.

The training stage's purpose is to optimize the parameters $\lambda = (A, B, \pi)$ to best characterize the observations O. The test image is matched to each of the learnt models to achieve recognition. Model likelihoods for each lambda are produced once the picture is converted to an observation sequence. The unknown face's identity is revealed by the model with the highest likelihood.

D. Training Process

HMMs aim to solve three problems:

- Evaluation - How likely is it that something observable will happen?
- Decoding - What is the reason for the observation that happened?
- Learning - What I can learn from the observation data I have?

Each of the three problems stated above can be solved by different algorithms. Given our problem statement, it can be seen that our paper aims to tackle the Learning problem, i.e, learning certain patterns from the given set of faces (observations) to help identify these faces later on.

Baum Welch was the algorithm chosen to help solve the Learning problem for this paper. Baum Welch is based on Dynamic Programming, a widely popular programming paradigm used to solve complex problems by breaking them into smaller chunks and using those solutions at higher levels.

An iterative technique is used to obtain the greatest likelihood for the three parameters in the HMM, the emission matrix B, the state transition matrix A, and the initial state distribution [8-9]. After the algorithm has run its course, these parameters will have been set to values that are ideal for the observed data, thus allowing the model to execute with more precision.

The algorithm consists of three steps:

- Initial Phase → Initialises the value of the parameters randomly
- Forward Phase → Calculation of the alpha function in a recursive manner. (9)
- Backward Phase → Calculation of the alpha function in a recursive manner. (10)

The initial phase involves setting the initial, as the name suggests, values of the 3 parameters A, B, and π_0 . This is usually done randomly or by setting all of them to 0.

The forward phase of the Baum Welch algorithm entails utilising the recursive function alpha to calculate the joint probability of the observed data up to a specified time (in this instance k) and the state at that time. It is governed by the formula below:

$$\alpha(X_k) = P[Y_{0:k}, X_k]$$
$$= \sum_{Xk-1} \alpha(X_{k-1}) P(X_k \mid X_{k-1}) P(Y_k \mid X_k) \quad (9)$$

The function becomes recursive because the first term in the summation refers to the function's previous value. The transition probabilities from matrix A are the second term, while the emission probabilities from matrix B are the last term. At time *k-1*, the whole summation includes all potential states.

Each alpha, in accordance with HMM principles, contains information on the absolved data till time k, and just the current alpha value is needed to compute the next alpha [10].

The backward phase of the Baum Welch method includes utilising beta, a recursive function, to calculate the conditional probability of the observed data from a certain time (in this instance k+1) and the state at that given time. It is backed by the formula:

$$\beta(X_k) = P[Y_{k+1:T}, X_k]$$
$$= \sum_{Xk+1} \beta(X_{k+1}) P(X_{k+1} \mid X_k) P(Y_{k+1} \mid X_{k+1}) \quad (10)$$

The function becomes recursive because the first term in the summation refers to the function's future value. The transition probabilities from matrix A are the second term, while the emission probabilities from matrix B are the last term. At time k+1, the whole summation includes all potential states. This phase is known as the backward phase because the next value of beta is employed in the calculation of the present value [11].

The alpha and beta formulae may be used to determine the probability distribution of the state variable at a particular time (in this case, k) given the whole sequence of observed data.

In the final step, the update phase, the alpha and beta features are also useful.

During the update phase, the parameter matrices are fine-tuned to best suit the observed data and predicted hidden states. After that, the four phases are repeated until the parameters converge or the model reaches a particular level of accuracy. These four processes have a unique nomenclature and may be neatly encapsulated under the Expectation-Minimisation method, which is a popular technique. The Expectation step (E-step) is made up of the first three phases, whereas the Minimisation step is made up of the last one (M-step)

E. SQL

This paper is intended to serve as an attendance tracker, meaning it matches students' faces as they scan it in real-time to an existing database provided by the institution, as well as stores the date and time of the scan to make things more organized.

To bring about such functionality, SQL was used. A table, stud_info, storing the student id, class, roll no and student name was created and filled with the appropriate details for a certain number of students. This was done using a local MySQL server. When the program is run and the face has been verified to exist in the database, the date and time of verification are recorded and stored along with student id, student name, and student class in another table, att_info, using the corresponding SQL query.

stud_id	stud_name	stud_class
24	Person24	A
25	Person25	A
26	Person26	A
27	Person27	A
28	Person28	A
29	Person29	A
30	Person30	A
31	Person31	A
32	Person32	A
33	Person33	A
34	Person34	A
35	Person35	A
36	Person36	A
37	Person37	A
38	Person38	A
39	Person39	A
40	Person40	A
41	Person41	A
42	Sneha	A
43	Suneel	A
44	Dipshikha	A
NULL	NULL	NULL

Fig 3. Snippet of SQL 'stud_info' table

att_id	stud_id	att_date	att_status
1	1	2021-11-29 16:44:17	1
2	42	2021-11-29 16:46:04	1
3	42	2021-11-29 16:46:32	1
4	1	2021-11-29 17:58:00	1
5	5	1 21-11-29 22:10:11	1
6	5	2021-11-29 22:10:35	1

Fig 4. Snippet of SQL 'att_info' table

V. RESULT

This attendance tracker system was implemented using Matlab R2021b and tested on an Intel(R) Core(TM) i7-10710U system having 16 GB RAM. The images in the ORL dataset were converted to grayscale and resized to 56 x 46 which yielded a facial recognition rate of 95.122%. This rate was obtained by using the SVD matrix coefficients U (1,1), Σ(1,1), and Σ(2,2) that were both tested and empirically found to give out better accuracy.

A. MATLAB GUI

The culmination of the paper code results in a GUI that the user can interact with to:

- Upload their image/Capture their image
- Verify their image
- View their attendance until that day, once they have been verified

Upload Image

Once the user has clicked the "Upload Image" button, a file selector window pops up, letting them select an image to verify their identity. This image is then converted to greyscale.

Capture Image

Alternatively, the user can also capture their image real-time using the device's webcam. This image is then converted to greyscale.

Verify Image

After the user has either uploaded or captured their image, they can click the "Verify Image" button to verify their face within the existing database. Upon clicking this button, the image is passed to the test function, *upload image*. The function resizes the image to a size of 56x56 and uses the trained HMM to run through each person in the database and see whose face the given face has the most similarity with, i.e., get the most probable match. This function returns the index (id) of the person and the corresponding name is displayed in the GUI. The identified name is taken and run through an SQL query to get the corresponding student id from the stud_info table. Then this id, along with the date and time of verification is run through another SQL query to insert a new record into the att_info table, which is stored for later use.

View Attendance

Once the person has been verified, the "View Attendance" button appears and they can choose to view their attendance from the att_info table. If they go ahead with this, the corresponding callback function runs an SQL query with the previously obtained student id to get the corresponding student attendance details from the att_info table and display them with a MATLAB UI table.

The user can perform the above order of operations as many times as needed before closing the window when required

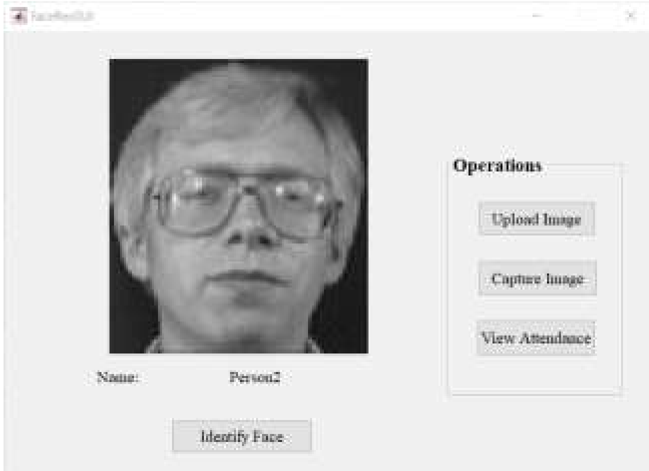


Fig 5.1 Final GUI Window (Window 1)

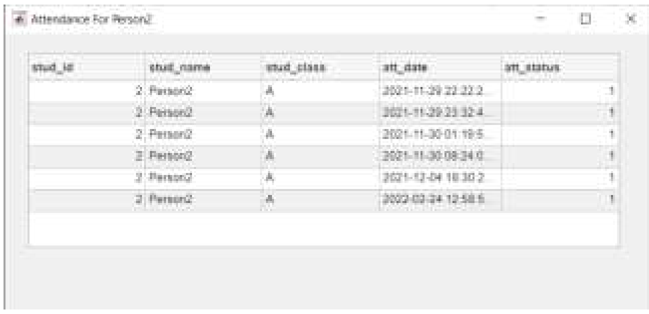


Fig 5.2 Final GUI Window (Window 2)

VI. CONCLUSION AND FUTURE SCOPE

This paper looked at the implementation of an Attendance Tracker using HMMs and face recognition. It was implemented using MATLAB and involved heavy usage of HMM concepts, such as using the Baum Welch algorithm for training and SVD to extract the most important features of a given face.

The dataset used was relatively small as well, having only 400 sample. The dataset file consisted of 40 directories which stored 10 images in each of the directories. Each of the directories corresponded to the images of single person taken in 10 different angles having different facial expressions. The images were taken in such a way that face was visible under a proper amount of light.

The dataset was split into 80% and 20% for testing and training. The training time was bought to minimal, totaling around 20-30 minutes while still maintaining an accuracy of 95.122%. These minimalist conditions leverage this paper as a very efficient implementation of HMMs for face verification and could prove very useful for any sort of institution.

However, there is still scope for improvement, that could further the usefulness of the paper and increase its applications. One such improvement is the use of One-Shot Learning. This method makes it such that only one face would be required for testing and identifying new faces, making it very convenient for institutions to gather required data to fuel the working of the tracker.

Another extension involves the integration of face detection. Instead of having each user come up to the scanner and get their faces scanned; the scanner itself could be placed in a room full of students/employees and scan, recognize and verify all the faces present. This could be done using any one of the many available algorithms for face detection like Haar Cascades and would greatly improve the ease of use of this tracker.

With the right improvements, this paper could prove to be another application of HMMs, amongst the already existing ones. It could also be helpful for educational institutions and potentially even offices, removing the need for manual attendance checks and making tracking of attendance over the entire college easier and more organized.

REFERENCES

[1] Tjahyanto, Aris and Hidayat, Nurul and Shahab, Muhammad Luthfi and Adzkiya, Dieky and Nisa, Alvina Khairun, Individual Recognition System Through Face Identification Using Hidden Markov Model (August 1, 2018). International Conference on Information Technology, Engineering, Science & its Applications, Available at SSRN: https://ssrn.com/abstract=3248108

[2] S. S. Teja Gontumukkala, Y. S. Varun Godavarthi, B. R. Ravi Teja Gonugunta, R. Subramani and K. Murali, "Analysis of Image Classification using SVM," *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2021, pp. 01-06, doi: 10.1109/ICCCNT51525.2021.9579803.

[3] A. V. Nefian and M. H. Hayes, "Face detection and recognition using hidden Markov models," Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269), 1998, pp. 141-145 vol.1, doi: 10.1109/ICIP.1998.723445.

[4] M. Tanuj, A. Virigineni, A. Mani and R. Subramani, "Comparative Study of Gradient Domain Based Image Blending Approaches," *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*, 2021, pp. 1-5, doi: 10.1109/ICSES52305.2021.9633858.

[5] Serign Modou Bah, Fang Ming, "An improved face recognition algorithm and its application in attendance management system", Array, Vol 5, (2020), 100014, https://doi.org/10.1016/j.array.2019.100014 .

[6] Y. L. Prasanna, Y. Tarakaram, Y. Mounika and R. Subramani, "Comparison of Different Lossy Image Compression Techniques," *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*, 2021, pp. 1-7, doi: 10.1109/ICSES52305.2021.9633800.

[7] Huan, X. and Zhou, H.,"Cost-Effective Attendance Management System Using Cloud Computing and Face Recognition", *Journal of Management & Engineering Integration*, (2021) Vol.*14 No.* 1, pp.81-89.

[8] B. Murugadoss, S. N. R. Kama, J. S. Kode and R. Subramani, "Blind Digital Image Watermarking using Henon Chaotic Map and Elliptic Curve Cryptography in Discrete Wavelets with Singular Value Decomposition," *2021 International Symposium of Asian Control Association on Intelligent Robotics and Industrial Automation (IRIA)*, 2021, pp. 203-208, doi: 10.1109/IRIA53009.2021.9588744.

[9] Gawande U., Joshi P., Ghatwai S., Nemade S., Balkothe S., Shrikhande N. (2022) Efficient Attendance Management System Based on Face Recognition. In: Tuba M., Akashe S., Joshi A. (eds) ICT Systems and Sustainability. Lecture Notes in Networks and Systems, vol 321. Springer, Singapore. https://doi.org/10.1007/978-981-16-5987-4_12

[10] Goud, Kathi Mohan; Hussain, Shaik Jakeer, "Implementation of investigating system using face detection and face recognisation techniques: A case study", Journal of Engineering Science & Technology Review. 2021, Vol. 14, No 5, pp. 109-120.

[11] M. Vamsi, Soman K. P., and Guruvayurappan, K., "Automatic Seat Adjustment using Face Recognition", in 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2020.

[12] Ali, N.S., Alhilali, A.H., Rjeib, H.D., Alsharqi, H. and Al-Sadawi, B., "Automated attendance management systems: systematic literature review", *International Journal of Technology Enhanced Learning*, (2022), Vol. *14 No.* 1, pp.37-65.

[13] Kumbhar, A.A., Wanjara, K.S., Trivedi, D.H., Khairatkar, A.U. and Sharma, D., "Automated attendance monitoring system using android platform", *International Journal of Current Engineering and Technology*,(2014), Vol. *4, No.* 2, pp.1096-1099.