

Face Recognition using Principal Component Analysis and Clustered Self-Organizing Map

Jasem Almotiri

Department of Computer Science
College of Computers and Information Technology
Taif University, Makkah, Saudi Arabia

Abstract—Face recognition is one of the cornerstones of the face processing schemes that composed the contemporary intelligent vision-based interactive systems between computers and humans. Instead of using neurons of the Self-Organized Map (SOM) neural network to cluster the facial data, in this work, we applied an agglomerative hierarchical clustering to cluster the neurons of the SOM network, which in turns, used to cluster the facial dataset. In prior, Principal Component Analysis (PCA) is employed to reduce the dimension of the facial data as well as to establish the initial state of SOM neurons. The design of the clustered-SOM recognition engine involves post-training steps that labeled the clustered SOM neurons resulting in a supervised SOM network. The effectiveness of the proposed model is demonstrated using the well-known ORL database. Using five images per person for SOM training, the proposed recognizer results in a recognition rate of 94.7%, whereas using nine images raise the recognition rate up to 99.33%. The facial recognizer has attained a notable reliability and robustness against the additive white Gaussian noise, where increasing the level of noise variance from 0 to 0.09, the recognition rate decreased only by 8%. Furthermore, time cost is analyzed, where using 200 images for training takes less than 4 seconds to be performed, whereas testing using a new set of 200 images takes less than 0.013 seconds which is competitive to many artificial intelligence and machine learning based schemes.

Keywords—Artificial intelligence; machine learning; clustering; agglomerative hierarchical clustering; face recognition; neural network; self-organizing map; principal component analysis

I. INTRODUCTION

As a basic definition, facial recognition is the process that utilizes techniques and algorithms to match the physical characteristics against the photos of peoples' faces where face identification allows faster and more accurate face identification rather than that is carried out by the naked human eye. Face recognition can take variants spheres of face recognition-related activities and operations to new ambits. For example, in the fields of security, face recognition can do a lot more to enhance security extends from street crime to airport security where these issues have been dominated the headlines in many countries all over the world. The limited information in security cases opens the doors for a wide band of accusations related to bias or discrimination.

Face recognition systems open the counterpart door that entails no antecedent information related to age, race, or gender especially face recognition that is carried out based on classical techniques that adapt prior saved databases composed of the

faces of persons of interest or persons who are suspected to involve in a serious violent crime. In the nutshell, face recognition systems offer up a further intelligence in people identification, especially in situations where it is considered a tedious task to be done by human staff alone as encountered in the large, crowded areas and establishments.

This recent surge of facial recognition usage increases the demands on recognition performance metrics included: (1) recognition accuracy and (2) the speed of response. The majority of facial recognition methods and schemes in literature have been built based on two major cascaded engines: (1) facial data representation (facial features/characteristics extraction) engines (2) Facial classification engines.

Facial representation engines can be either supervised such as Linear Discriminative Analysis (LDA) [1][2] or unsupervised such as Principal Component Analysis (PCA) [1][3], Locality Preserving Projection (LPP) [4], Discrete Wavelet Transform (DWT) [5] and other sparsity-based techniques [6] [7] or it can be a merged of supervised and unsupervised techniques as proposed in [8].

Facial classification engines can also be categorized as unsupervised [9], [10] or supervised such as using group sparsity representation-based classification [11], or using techniques that built based on deep learning [12] such as artificial neural networks [13], [14],[15], [16][17],[18], Support Vector Machine (SVM) [19], [20], or Decision Tree and Random Forest [21]. Furthermore, the classification engines can be built based on a hybrid of supervised and unsupervised techniques [22].

A considerable portion of artificial intelligent and machine learning based schemes is simple to be implemented, however the recognition performance is moderate, or the reliability of the facial recognizer show dramatic changes against changes occur in the number of training images used per person, or against additive white Gaussian noise.

The other portion of techniques, that adapt complex frameworks (such as complex-structured neural networks) show high to moderate recognition performance associated with high computational overhead and time cost that lessen the applicability of these techniques.

In this work, we aim to design an intelligent facial recognizer model that can be deployed with low computational overhead and time cost yet have competitive facial recognition

rate. Furthermore, we aim a high level of reliability in our designed model through a high robustness against sizeable change in the volume of available training data as well as against high levels of additive Gaussian noise.

To achieve these goals, in this work, we use an effective combination of the simple standard PCA algorithm that modified mathematically to suit the high dimensionality of the facial data and to dramatically lower the computational overhead. Then, we designed a special variant of SOM, denoted as clustered SOM to perform the classification step, where instead of using the converged SOM neurons as our classifiers, we applied an agglomerative clustering algorithm, to obtain a set of clusters composed of SOM neurons, afterwards, these clusters are labeled based on the training dataset yields a supervised version of clustered SOM. The training data with labeled exemplars are used to label the different generated clusters by measuring the distance between the facial vectors that composed the training data and the neurons that play the role of "cluster members". Then, the different labeled SOM clusters used to classify the new coming facial data into their corresponding classes.

This hybrid framework has the ability to learn and response fast due to the flexibility of the supervised clustered SOM, where it can learn easily and with low computational overhead as well as it has a high detection performance where the operational neurons undergo two cascaded refinement one during raw SOM neural network training and the other one during the agglomerative clustering applied on these converged neurons. Thus, we can summarize the contribution of this work in five-fold:

- We present the design and implementation of a human facial recognition system based on PCA as a feature extractor (facial representation) and the agglomerative hierarchical clustered SOM as the recognition engine. Moreover, we study the recognition performance of the system using a benchmark ORL dataset.
- The proposed system shows supremacy in terms of time-cost and computational overhead to perform either training or test stages where the processing time per image in the training stage is as low as 0.0182 sec per training image whereas it expends less than $7e-5$ sec per test image.
- The system shows robustness against the noise addition, where the recognition performance can tolerate high levels of Gaussian noise with zero mean and different levels of variations.
- We compare the performance of the PCA-clustered SOM model with other methods that either involved SOM or SOM-variants, where our proposed method show supremacy in terms of recognition rate. However, it shows comparable or underperformance with nuance differences to other schemes that built using other techniques, where these methods show vast inferiority in other aspects of recognition performance such as time cost, complexity overhead, and robustness especially against adding white Gaussian noise.

- Based on the several performance analyses that conducted to measure the reliability and the robustness of the proposed recognition model, it can be implemented in the real time applications where the high robust against noisy signal as well as the fast time response and fast learning of new coming images, make the model mostly appropriate for security systems that utilized the personal biometric features.

The rest of this paper is organized as follows: related works that constructed based on SOM network is discussed in Section 2. Section 3 details the proposed method. Section 4 presents the experimental results and analysis. Section 5 presents the results discussion and Section 6 concludes the paper.

II. RELATED WORK

One of the efficient yet robust unsupervised neural networks is Self-Organizing Map (SOM) as it can play a dual role in the field of face recognition, where it can be used either to represent the facial data (as a feature extractor and dimensionality reduction tool), or it can be used as a classification engine. SOM network can be even integrated with other dimensionality reduction techniques such as PCA as proposed by Kumar et al. [23]. SOM network can be used as a dimensionality reduction as well as facial data representation as proposed by Lawrence et al. [24].

As a pre-processing step applied to the SOM network before the output of the network can be fed to the classification engine, Ruiz and Jaime [25] applied Fourier transform to the output of the SOM network (optimal weight vectors of the SOM neurons) to attain a translation invariance to the feature map generated by applying a two-dimensional Gabor filter to the input raw facial images. Afterwards, backpropagation neural network was used for sake of classification.

In a facial recognition scheme uses K-Nearest Neighbor (KNN) as a classification engine, Yodkhad et al. [26] used the clustering capability of the SOM network to group the training data and extract the representative prototype of each group, which, accelerate the classification duty of KNN algorithm.

Using SOM network as the major classifier engine of the face recognition system is proposed by Neagoe and Stanculescu [22], where PCA, LDA, and ICA techniques of data representation and feature extraction were used for comparison purposes cascaded by a SOM-variant scheme called concurrent SOM (CSOM) which was proposed and developed by the first author of [22]. In this scheme, the training data is partitioned into multiple sub datasets (partitions), each partition represents one class of data. Then, multiple SOM networks are generated and each one is trained by one partition of training data. In the testing stage, the Euclidean distance between the testing vectorial image and each trained SOM network is calculated, and the winner "SOM network" with the minimum distance gives the class to the input testing image.

Besides the eigenfaces that can be generated by the PCA method and the fisher face that can be generated by LDA (or Fishers' Linear Discriminant), SOM network can be used to generate what is called SOM-face as proposed in [27] where an

enhanced version of SOM network, called as kernel-based SOM network is used to extract the representative features of the facial data, where authors called it as SOM-face [27].

Based on SOM-face, Zhi and Meng [28] proposed a face recognition method based on multiple training images, where, in addition to the topological shape-feature vector generated by SOM network, a wavelet-feature vector is generated by a wavelet-network, then both feature vector representations fed to the classification engine.

Instead of training the SOM network via the individual vector facial images as a holistic, Tan *et al.* [29], partitioned each facial image into equal-sized nonoverlapping subblocks. Then, the resulted subblocks used to train the SOM map or multiple SOM maps in a step called by the authors as the localizing step. The major goal of this step is to generate the local vector representation of facial data to be fed to a soft KNN-ensemble classifier for sake of facial recognition.

Using the SOM network as a classifier engine, Monteiro *et al.* [30] proposed four schemes of SOM/SOM-variant-based classifiers. Since SOM is used as a classifier, all four classifiers' engines that implemented by the authors use either SOM grid neurons pre-training labeling or post-training labeling step. The first proposed classifier used the labeled

training data for SOM neurons labeling which, in turn, used as the classification engine of the recognition system. The second classifier uses the centroids of the labels available for each class of facial data to label the SOM neurons after the training process, where the centroids of labels of each class are pre-computed. The third classification engine is built by turning SOM into a supervised classifier by augmenting each input vector with its corresponding class label, where these vectors are used to adjust the corresponding augmented weight-vectors of SOM neurons. The fourth SOM-based classifier was built by using an entire SOM network to represent one single class of the available facial data. Afterwards, these networks are trained separately using the corresponding input facial vectors for each class. During the testing phase, however, the best-matched neuron (winner) is chosen via all the available trained SOM networks and the winner trained SOM network assigns its class to the incoming test vector.

III. METHOD

The high-level block diagram of the proposed method is shown in Fig. 1. Seeking an optimal representative subspace of the input facial data, classical PCA is applied as a first step. Then, facial data is projected on the resulting PCA-subspace and split into projected training and projected testing datasets.

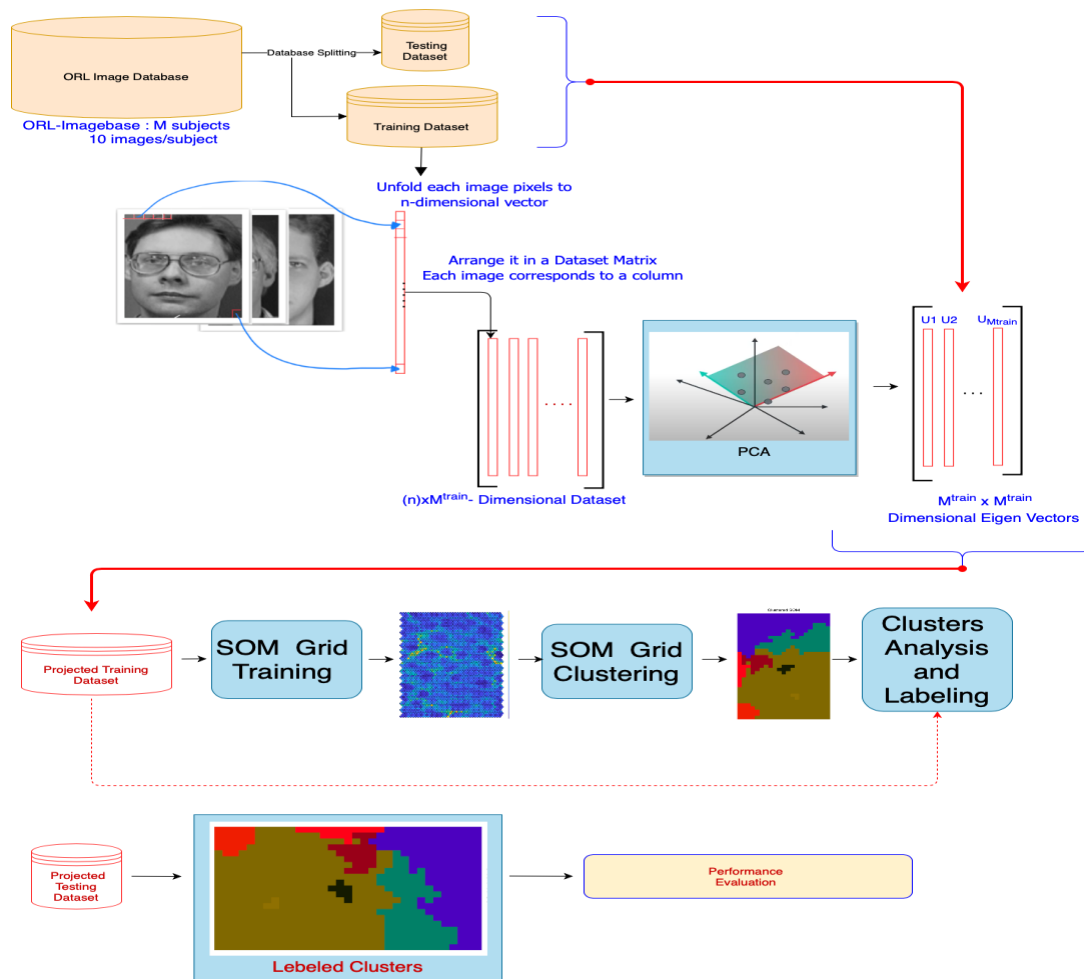


Fig. 1. High-level Block Diagram of the Proposed Face Recognition System Work Flow.

The projected training dataset is used to establish the topological space of the SOM neural grid. Then, the best-matched units of the SOM network undergo a hierarchical clustering for sake of optimal robust and compact representation of facial data. As a post-training step, the projected labeled training data used in analyzing and labeling SOM clusters based on the majority occurrence voting of a specific subject (face image belongs to a specific person in the training dataset). Then, new coming facial data is classified by the proposed facial recognizer into their corresponding classes.

In subsequent sections, each step of the recognition model is explained in detail.

A. Principal Component Analysis (PCA)

Formally, the ORL database is composed of J gray-scale images, that can be represented by the matrix $X_{c,r}^{j,p}$, where $c \times r$ corresponds to the vertical and horizontal dimensions of the j^{th} image (in our case, $r = 92, c = 112$) of the p^{th} person (subject/class). As a first step, the pixels of $X_{c,r}^{j,p}$ are vectorized into an n -dimensional vector: $I_j^p \in \mathbb{R}^n$ where $n = c \times r$ by reading the pixel values in the image $X_{c,r}^{j,p}$ in a raster-scan manner, in our case, $n = 10,304$.

Thus, the set of images in the ORL database can be represented as a rectangular matrix Y of columns $Y = [I_1, I_2, \dots, I_J]$ where J index represents the total number of images in the raw dataset.

The dimensionality of these images is too large to be fed and efficiently analyzed by the recognition engine of the proposed model. For sake of obtaining a more compendious representation of data, the regular form of the principal component analysis technique is used.

Given as input, a rectangular matrix Y whose columns are seen as variables, the main objective of the principal component analysis is to create a new set of variables (called principal components) that have a linear combination of the input variables in such a way that the variance between the principal components (resulted basis vectors) and each of the original variables is maximized.

As a first step, the vectorized dataset Y is split into two sub-datasets: the first one is the training dataset $Y^{train} = [I_1, I_2, \dots, I_M]$ which used to produce the principal components basis vector used to establish and train the SOM network, and to label the clusters of the clustered SOM, M represents the total number of involved images in the training stage. The other sub-dataset is the testing dataset $Y^{test} = [I_1, I_2, \dots, I_m]$ which is used to test the recognition performance of the proposed model, m represents the testing images involved in the testing stage, where $m = J - M$. Fig. 2 shows the pipeline of the facial data representation stage using the PCA method.

As PCA is a variance optimization process, if some variables (vectors of matrix Y) show a large variance compared to other variables, then, during variance maximization, PCA will load on the ones of large variances. Therefore, as a prior step to PCA, is to normalize the data in two succeeded steps. First, the average of vectorized training images $I_j \in Y^{train}$ is obtained as in (1):

$$I_{mean} = \frac{1}{M} \sum_{j=1}^M I_j \quad (1)$$

Each vectorized image vector $I_j \in Y^{train}$ is subtracted from the average image vector I_{mean} to obtain the normalized training dataset as elaborated in (2) and (3):

$$\xi_j^{train} = I_j^{train} - I_{mean} \quad (2)$$

$$\xi_j^{test} = I_j^{test} - I_{mean} \quad (3)$$

Let the set of all standardized image vectors $\xi_j^{train}; j = 1:M$ compose the standardized training dataset matrix $\Xi^{train} = [\xi_1, \xi_2, \dots, \xi_M]$, whereas $\Xi^{test} = [\xi_1, \xi_2, \dots, \xi_m]$ represents the standardized testing dataset matrix composed of all standardized image vectors $\xi_j^{test}; j = 1:m$. $\xi_j, I_j, I_{mean} \in \mathbb{R}^n$.

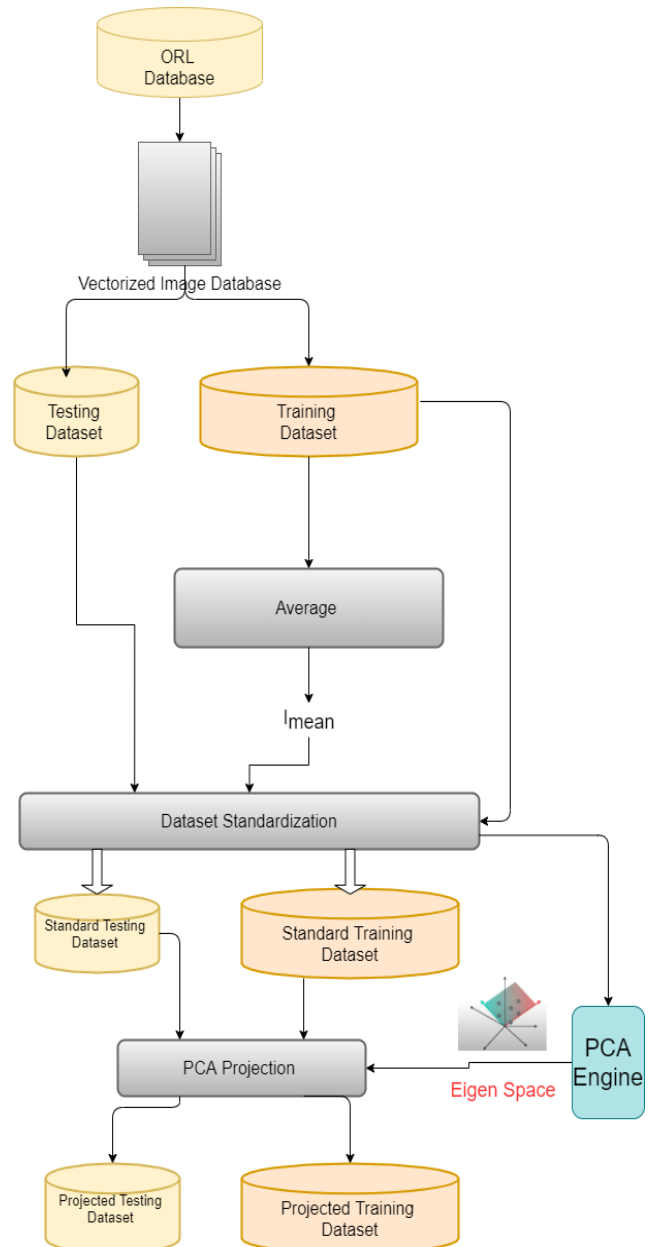


Fig. 2. Flowchart of the PCA-based Feature Reduction Scheme.

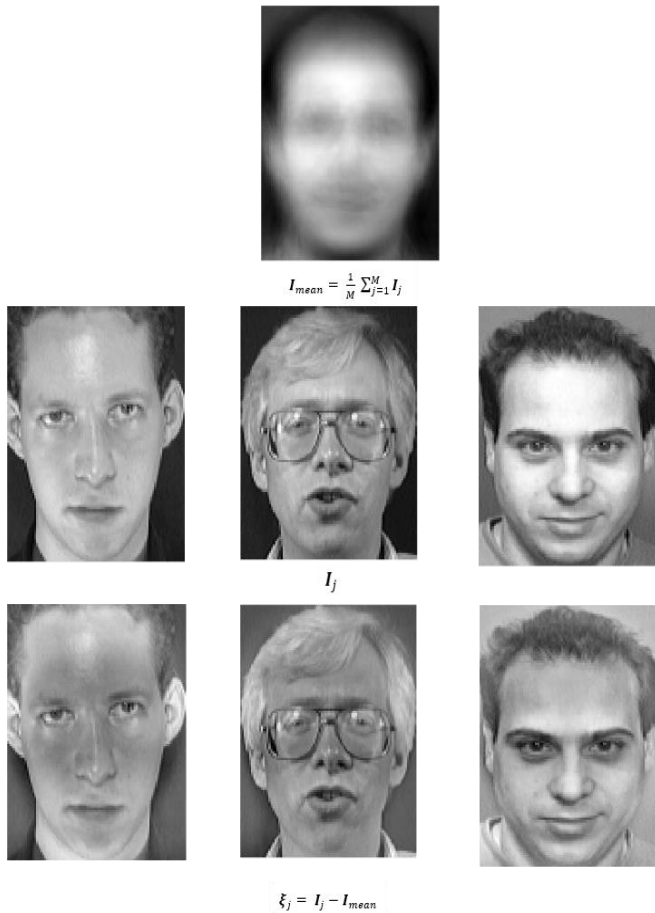


Fig. 3. Average Image I_{mean} and the Deviation of a Randomly Selected Set of Training Images I_j^{train} from the Average Image ξ_j .

As a visual illustration of the outputs of (2) and (3). Fig. 3 shows the average image and the deviation of a randomly selected set of training images from the average image I_{mean} .

Let U be defined as a linear transformation that maps the standard version of n -dimensional Ξ^{train} matrix onto a feature subspace of f -dimensional feature vectors, where $f \ll n$. By projecting Ξ^{train} on U space, the new formed feature vectors $\mathfrak{F}_j^{train} \in \mathbb{R}^f$ are defined as in (4):

$$\mathfrak{F}^{train} = U^T \Xi^{train}, j = 1, 2, \dots, M. \quad (4)$$

Based on (4) we can define the matrix $\mathfrak{F}^{train} = [\mathfrak{F}_1^{train}, \mathfrak{F}_2^{train}, \dots, \mathfrak{F}_M^{train}]$ as that represents the projected training dataset that will be used to train the neurons of the SOM network in the recognition stage of the proposed model.

Same wise is applied on the standardized testing dataset as defined in (5):

$$\mathfrak{F}^{test} = U^T \Xi^{test}, j = 1, 2, \dots, m. \quad (5)$$

Based on (5), we define, $\mathfrak{F}^{test} = [\mathfrak{F}_1^{test}, \mathfrak{F}_2^{test}, \dots, \mathfrak{F}_m^{test}]$, as the result of projecting the standardized testing data Ξ^{test} on

the eigenspace defined by U^T . The columns of U matrix are the eigenvectors e_i that represent the eigenstructure decomposition of the covariance matrix C that can be defined as in (6) and (7):

$$\lambda_i e_i = C e_i, i = 1, 2, \dots, M \quad (6)$$

$$C = \Xi^{train} \Xi^{train^T} \quad (7)$$

where, the scalars λ_i are the eigenvalues of the covariance matrix C .

The covariance matrix C is of high dimensionality to be computationally processed by the PCA engine for sake of finding the eigenvectors e_i in an efficient manner. To solve this problem, many previous works had handled this difficulty via different tactics. In this work, we adapted the solution proposed by [31].

First, let e_i^C be the eigenvectors correspond to the covariance matrix C that defined in (7) and let e_i^K be the eigenvectors corresponds to the new-defined matrix K that constructed by switching the order of the transpose in (7) as illustrated in (8):

$$K = \Xi^{train^T} \Xi^{train} \quad (8)$$

The eigenvectors e_i^K and the eigenvalues λ_i^K corresponds to the matrix K are given as in (9):

$$K e_i^K = \lambda_i^K e_i^K, i = 1, 2, \dots, M \quad (9)$$

Pre-multiplying both sides of (9) by matrix Ξ^{train} , we have (10):

$$\Xi^{train} K e_i^K = \Xi^{train} \lambda_i^K e_i^K \quad (10)$$

Substitute the value of K of (8) and rearrange the terms, yields (11):

$$\Xi^{train} \Xi^{train^T} \Xi^{train} e_i^K = \lambda_i^K \Xi^{train} e_i^K \quad (11)$$

based on the basic definition of the covariance matrix C in (7) we have (12):

$$C \Xi^{train} e_i^K = \lambda_i^K \Xi^{train} e_i^K \quad (12)$$

It can be noted from (12), that the terms “ $\Xi^{train} e_i^K$ ” and λ_i^K represent the eigenvectors and the eigenvalues of the covariance matrix C respectively. Thus, based on (12), to find the eigenvalues and the eigenvectors of the matrix C , we first construct the matrix K and then find the corresponding eigenvalues λ_i^K and eigenvectors e_i^K . Then, the eigenvalues of C are set to λ_i^K whereas the eigenvectors of C are obtained by multiplying the standardized training matrix Ξ^{train} by the eigenvectors of the matrix K as summarized in (13) and (14):

$$e_i^C = \Xi^{train} e_i^K \quad (13)$$

$$\lambda_i^C = \lambda_i^K \quad (14)$$

Fig. 4 shows the first nine eigenfaces U_j corresponding to the first nine eigenvalues λ_j^C for $i = 1, 2, 3, \dots, 9$.

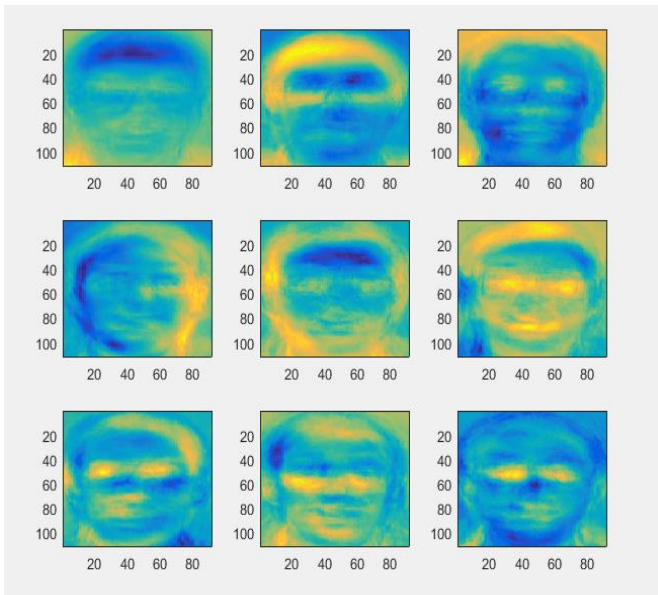


Fig. 4. First Nine Eigenfaces U_j Corresponding to the First Nine 2D - Eigenvalues λ_j^f for $i = 1, 2, 3, \dots, 9$.

B. Hierarchical Clustered SOM Network

The second stage in our proposed model is the recognition engine. The hierarchical agglomerative clustering was applied on the well-trained neurons of the self-organizing map (converged map) yields clusters that were labeled by training data and used as the classification engine of the system.

As illustrated in Algorithm I, the first step of the classification engine is to establish the SOM grid which is used as a second layer of dimensional reduction and facial data representation. SOM networks belong to the family of topographic maps, represents a type of competitive unsupervised learning systems where the input space, which in our case is \mathfrak{F}^{train} is “mapped” in a less-dimensional output space with the following principle: the similar feature vector $\mathfrak{S}_i^{train} \in \mathfrak{F}^{train}$ will be projected into the same neuron or, at least, in the neighborhood of it, in the output space of the SOM grid.

As can be shown in Fig. 1, in our proposed model, we have two cascaded projections: first projection is when the standardized training data Ξ^{train} and the standardized testing data Ξ^{test} projected onto the eigenvectors of the U space producing \mathfrak{F}^{train} and \mathfrak{F}^{test} . The second projection, \mathfrak{F}^{train} is projected onto the SOM grid to construct the output space of the SOM neural network.

Typically, the incremental-learning algorithm of SOM networks proceeds as follows [32]: Let the codebooks of the SOM neurons modeled by the vectors $W_k \in \mathbb{R}^f$ whereas $\mathfrak{F}^{train} \in \mathbb{R}^f$ represents the observation vectors (input space). Then, we can define the regression of a set of weight vectors codebooks $w_k \in W$ into the input space \mathfrak{F}^{train} by the following mathematical relation (15):

$$W_k^{t+1} = W_k^t + h_{win}(\mathfrak{F}_j^{train})(\mathfrak{F}_j^{train} - W_k^t) \quad (15)$$

where:

t : is the sample index.

where, $h_{win}(t)$ is called the neighborhood function, which is often, chosen to be as Gaussian defined by (16):

$$h_{win}(t) = \alpha^t \exp\left(-\frac{\|l_j - l_w\|^2}{2\sigma_t^2}\right) \quad (16)$$

Algorithm I: SOM Training Algorithm

Input : Specify the size (number of grid units) and the dimensions (width, Height) of SOM map.
Initialize wights vectors, Initialize $\sigma(t)$.
Normalize training data $\mathfrak{F}^{train} \in \mathbb{R}^f$
define value for the maximum number of epochs as N^{epoch}
 K : total number of SOM neurons on the grid
 M : total number of training input vector presentations.

$h_{num}^{accum} = 0$ // the accumulated value of the numerator of equation (18)
 $h_{den}^{accum} = 0$ // the accumulated value of the denominator of equation (18)

Output : Converged 2D SOM grid.

- 1 : FOR ($epoch = 1, \dots, N^{epoch}$)
- 2 : Specify a new value for $\sigma(t)$
- 3 : //for each data item \mathfrak{F}^{train}
- 4 : FOR ($t = 1, \dots, M$)
- 5 : $t = t + 1$
- 6 : //for each neuron k on SOM grid.
FOR ($k = 1, \dots, K$)
- 7 : Compute the distances $\mathfrak{D}_k(t)$ using:
$$\mathfrak{D}_k(t) = \|\mathfrak{F}_j^{train} - w_k(t)\|^2$$
- 8 : END FOR
- 9 : Compute winning (BMU) neuron using:
$$\mathfrak{D}_{k,win}(t) \triangleq \min(\mathfrak{D}_k(t))$$
- 10 : //for each neuron k on SOM grid.
FOR ($k = 1, \dots, K$)
- 11 : compute the neighborhood function value for j presentation of input as:
$$h_{k,win}(t) = \exp(-\|r_k - r_{win}\|/\sigma(t)^2)$$
- 12 : accumulate h_{num}^{accum} as:
$$h_{num}^{accum} = h_{num}^{accum} + h_{k,win}(t)\mathfrak{F}_j^{train}$$

accumulate h_{den}^{accum} as:
$$h_{den}^{accum} = h_{den}^{accum} + h_{k,win}(t)$$
- 13 : END FOR
- 14 : END FOR
- 15 : // Weight Update at the end of each Epoch
- 16 : FOR ($k = 1, \dots, K$)
- 17 : Update weight vectors w_k using:
$$w_k = \frac{h_{num}^{accum}}{h_{den}^{accum}}$$
- 18 : END FOR
- 19 : END FOR

where, “win” refers to the “winner” neurons on the SOM grid, where the weight vectors W_{win}^t of winner neurons are defined by the condition (17):

$$\|\mathfrak{F}_t^{train} - W_{win}^t\| \leq \|\mathfrak{F}_t^{train} - W_k^t\| \forall k \in K \quad (17)$$

where, K represents the total number of neurons of SOM grid, $k = 1, 2, \dots, K$.

α^t : represents the learning rate which decreases in a monotonic manner according to learning steps (iterations) and has the value of: $0 < \alpha^t < 1$.

σ_t : represents the width of the neighborhood function which decreases monotonically with learning steps.

l_j, l_w : represent the 2D vectorial locations in the display SOM grid, where $l_j \in \mathbb{R}^2$ and $l_w \in \mathbb{R}^2$.

In the incremental (online) learning described by (15) The unsupervised learning is accomplished recursively for each presentation of the t^{th} training feature vector \mathfrak{F}_t^{train} . However, in our work, we use batch SOM [33] instead of an online SOM variant. Thus, weight updating takes place at the end of each epoch. Mathematically, let's define t_o and t_f as the start and finish of each epoch, then, weight updating is given by (18):

$$w_k(t_f) = \frac{\sum_{t_o}^{t_f} h_{win,k}(t) \mathfrak{F}_t^{train}}{\sum_{t_o}^{t_f} h_{win,k}(t)} \quad (18)$$

A self-organizing map in its raw version described in equations (15) to (18), serves as a dimensional reduction and facial representation analysis scheme and the next step of the recognition stage is to predict the SOM neuron's membership of a new (testing) facial feature vector matrix \mathfrak{F}^{test} that is presented to the output layer of the SOM network.

This process yields a set of contiguous neurons in correspondence to a particular facial pattern in the testing dataset. As a consequence, a set of facial patterns might belong to different persons mapped to the same set of contiguous neurons in the output layer of the SOM network. However, in our model, the SOM network is required to serve as a facial recognition engine that can recognize among 40 classes represents the subjects (person labels) of the original dataset. This entails SOM-nodes to be highly subject-oriented (high dependable on the person class). To enhance the uniqueness of SOM response, we apply a hierarchical agglomerative clustering on the SOM neurons themselves successively.

There are two main schemes of hierarchical cluster analysis: Agglomerative Hierarchical Clustering (HAC) and Divisive Hierarchical Clustering (DAC). In our proposed model, we used the agglomerative hierarchical clustering scheme as illustrated in Algorithm II. This type is of a bottom-up approach where each SOM codebook neuron is treated as a singleton cluster at the outset, then it agglomerates each pair of clusters successively. This process continues until clusters are merged into a pre-specified number of clusters that are specified at the beginning of the process. The prespecified number of clusters, in turn, represents the different target classes of the recognition process.

Algorithm II: Agglomerative Hierarchical Clustering (AHC) Algorithm

Input	:	Converged SOM network (weight vectors) with K neurons. \mathcal{M} : pre-specified number of clusters resulted from AHC.
Output	:	Clustered SOM
1	:	Let each neuron on SOM grid form one cluster NO clusters _initial = K // each cluster contain a single SOM Neuron. $\mathcal{M} = K, \text{ where } \mathcal{M} \ll K$
2	:	$I = K$ // initialize the index of while loop.
3	:	WHILE ($I \geq \mathcal{M}$)
4	:	Find a pair of clusters with the smallest cluster-to-cluster distance (linkage) that given by : $\mathcal{D}_{average}(cluster^A, cluster^B)$ $= \frac{1}{n_A n_B} \left(\sum_{i=1}^{n_A} \sum_{j=1}^{n_B} dist(x_{Ai}, x_{Bj}) \right)$ where, n_A, n_B : are the number of elements in $cluster^A$ and $cluster^B$ respectively. x_A and x_B represents the members of $cluster^A$ and $cluster^B$ respectively.
5	:	Based on a distance computed. Merge the two clusters of least distance into one cluster. Update while loop index: $I = K - 1$
6	:	ENDWHILE

As a post-training step, the projected labeled training data used in analyzing and labeling the resulted SOM clusters based on the majority occurrence voting of a specific subject (face image belongs to a specific person in the training dataset). Then, new coming facial data is classified by the proposed facial recognizer into their corresponding classes.

IV. EXPERIMENTAL RESULTS

In this section, the facial recognition performance of our proposed system as well as a comparison to other facial recognition systems are represented. Moreover, a series of experiments are carried out to evaluate the efficiency and the robustness of the proposed facial recognition model.

A. Experimental Setup

All experiments are performed using the ORL (Olivetti Research Lab) [34] dataset, which is a classical dataset composed of 400 sample images, each of 92×112 grayscale pixel resolution with 256 intensity levels. The dataset contains images for 40 persons (subjects), 10 images for each person (subject). As shown in Fig. 5, The images were taken at different lighting conditions, even for some subjects they are taken at different sessions, which adds kind of facial distortions such as different facial expressions (smiling, nonsmiling, open eyes, and closed eyes) and different facial details (wearing glasses or no glasses).



Fig. 5. Examples of Raw Facial Images of ORL Database show the different Lighting Conditions, different Facial Expressions and different Facial Details.

Although all images are taken in an upright position in the frontal view, it exhibits a slight left-right rotation in the pose angle and alignment, which, in turn, can be exploited to examine the robustness of the proposed system against imprecise facial alignment.

We measured the performance based on the recognition rate criterion basically, which required setting up an experimental protocol same as used in several previous works in this field, where the images per person are randomly permuted. Then, five images of each person are used for the training stage whereas the other five images are used for testing purposes. Moreover, for sake of examining the efficiency and the reliability of the proposed system, we experimented with different sizes of training dataset per person. The results for 30 runs of the experiment were recorded and the average is taken and analyzed in the following experiments.

B. Experiment 1

The performance of our proposed model as well as the recognition performance of other recognition schemes are shown in Table I and Table II.

TABLE I. COMPARATIVE RESULTS OF THE PROPOSED TECHNIQUE WITH OTHER EXISTING SOM-BASED FACIAL RECOGNITION SCHEMES BASED ON THE ORL DATASET.

Author	Year	Method	train to test ratio	Recognition Rate
Lawrence <i>et al.</i> [24]	1997	SOM + Convolutional ANN	5:5	94.25%
		PCA	5:5	89.5%
Zhi and Ming [35]	2005	SOM + Wavelet Network	7:3	92.5
		Regular SOM	7:3	84
		RBF-Kernel SOM	7:3	85
Kumar <i>et al.</i> [36]	2005	PCA + SOM	5:5	62.64
Neagoe and Anton [38]	2010	PCA+ Concurrent SOM	5:5	93
		Comon ICA + Concurrent SOM	5:5	88
		INFOMAX PCA + concurrent SOM	5:5	94
Proposed Model	2022	PCA + Hierarchical Agglomerative Clustered SOM	5:5	94.7

TABLE II. COMPARATIVE RESULTS OF THE PROPOSED TECHNIQUE WITH OTHER EXISTING FACIAL RECOGNITION SCHEMES BASED ON THE ORL DATASET

Author	Year	Method	train to test ratio	Recognition Rate
Tan <i>et al.</i> [37]	2009	Partial Similarity	5:5	97.3
Feng <i>et al.</i> [6]	2016	Fast Superimposed Parameter Classifier	5:5	94.5
Hamdan and Mukhtar [19]	2016	Moments-based Angular Radial Transform	5:5	87.7
Abuzneid <i>et al.</i> [13]	2018	LBPH + multi-KNN + Backpropagation ANN	5:5	98
Kong <i>et al.</i> [39]	2018	CSGF(2D) 2 PCA Net + Linear SVM	5:5	97.5
Sun <i>et al.</i> [17]	2020	Gradient Number Pattern +Fuzzy Convex-Concave Partition + CNN	4:6	95.69
			6:4	98.28
Qin <i>et al.</i> [40]	2020	Collaborative Representation (CR)+ Enhanced Nearest Neighbour	5:5	92.5
Gupta <i>et al.</i> [21]	2020	Combination of SIFT (64-components) and SURF (64-components) + Random Forest	8:2	94.7
			8:2	86.2
Proposed Model	2022	PCA + Hierarchical Agglomerative Clustered SOM	5:5	94.7

In Table I, we compared favorably with other facial recognizers which were built using either SOM or SOM-variants. However, based on a recently published survey, the SOM network has limited usage [41] in face recognition systems either using SOM as a feature extractor, data representative or using it as a face recognition engine. Therefore, to better discuss and interpret the results obtained from our proposed method, a comparison with other existing schemes that used different machine learning and artificial neural networks other than SOM network are shown in Table

II. It is worth to mention that all schemes listed in Table I and Table II are built based on ORL dataset.

C. Experiment 2: Impact of Training Dataset Volume

In this experiment, the impact of increasing the ratio of the size of facial data involved in the training stage to that used for testing purposes is demonstrated. Fig. 6 and Fig. 7 show the relationship between increasing the ratio of training to testing data versus the achieved recognition performance. The experiment is set up by varying the number of involved training images from 5 to 9 images. The characteristic curve between the recognition rate and the number of training images involved is computed for every case.

Obviously, as the size of training data increases, the principal components analysis algorithm generates more representative eigenvectors which, in turn, reflects in more accurate dataset projection and further enhanced dataset presentation enabling clustered SOM network to recognize better.

D. Experiment 3: Impact of Adding Noise

To discuss the robustness of our model against additive noise, we have conducted noise sensitivity experiments on the ORL dataset where the noisy testing images were generated by adding Gaussian additive noise of zero mean and different values of variation to each test image as illustrated in Fig. 8.

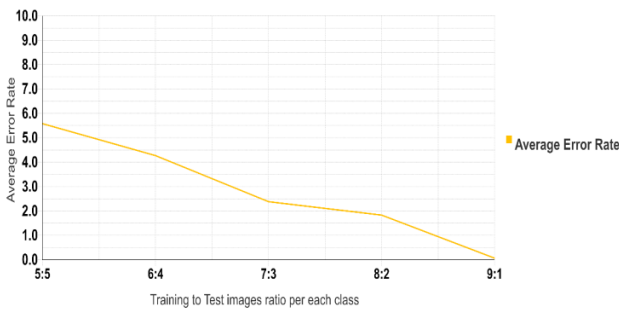


Fig. 6. Impact of Increasing the Size of Training Dataset on the Achieved Recognition Performance of our Proposed System in Terms of Average Recognition Error.

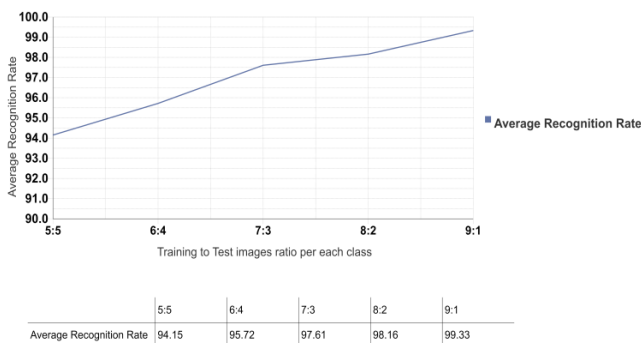


Fig. 7. Impact of Increasing the Size of Training Dataset on the Achieved Recognition Performance of our Proposed System in Terms of Average Recognition Rate.

TABLE III. AVERAGE RATE RECOGNITION RATE (OVER 30 ITERATIONS WITH 5:5 TRAINING TO TEST DATA VOLUME RATIO) OF OUR PROPOSED PCA-CLUSTERED SOM MODEL AGAINST ORL TEST IMAGES DISTORTED BY ADDING GAUSSIAN NOISE OF ZERO MEANS AND FIVE DIFFERENT VALUES OF VARIATIONS σ

Variations (σ)	Average Recognition Rate (%)	Average Recognition Error (%)
0.00	94.15	5.58
0.01	94.15	5.85
0.02	93.88	6.11
0.05	91.95	8.05
0.09	86.88	13.11

We run the experiment at five levels of deviation σ and the corresponding average recognition rate over 30 iterations using 200 images of the ORL dataset as training images (training to test ratio is 5:5) was reported as shown in Table III. In addition to using the raw images of the ORL dataset without any type of pre-processing.

TABLE IV. TIME COST OF PROPOSED PCA-CLUSTERED SOM MODEL AVERAGED OVER 30 RUNS WITH DIFFERENT TRAINING TO TESTING DATASET VOLUMES USING ORL DATASET

Size of Training Dataset	$v^{train} : v^{test} *$ Ratio	Average Training Time (sec)	Average Test Time (sec)
200	5:5	3.65	0.0129
240	6:4	4.389	0.0120
280	7:3	5.128	0.0114
320	8:2	5.98	0.0102
360	9:1	7.00	0.0085

* $v^{train} : v^{test}$: ratio of training dataset volume to testing dataset volume.

E. Experiment 4: Time Cost

The objective of this experiment is to verify the proposed model in terms of overhead complexity where the average training and average testing time corresponding to different sizes of training to test ratios were recorded in Table IV. Simulations are done on MATLAB 2021a, executed on an Intel Core (TM) i7-4500U CPU, 8 GB RAM in Windows 10 platform using customized code developed for this model and using SOM toolbox [42].

V. DISCUSSION

It can be noted from Table I and Table II, our proposed model shows supremacy against most of the different methods that used SOM/SOM-variants network either as a classifier or as a feature extractor as in using the regular SOM in [23] or as RBF-kernel based SOM presented in [42] or concurrent SOM-based technique proposed in [22].

The system proposed by Tan *et al.*[11] has about 3.15% average improvement over our proposed method, however, this improvement comes with a computational cost as a normal result to portioning each image into sub-block for sake of extracting local-features of facial data, although Tan *et al.*[11] didn't refer to the time expended, extensive computations always reflect as time-consuming and power-hungry face recognition style. The same scenario repeated for [24] where SOM is used as a feature extractor that fed a CNN network with optimal facial data representation.

Zhi and Ming [35] had achieved comparable performance to our method, however, the training data size used is large than that used in our method. Moreover, for the same training data size used by [35], our proposed system can achieve an average performance reaches up to 97.61% as elaborated in Fig. 7. Although superior results are reported in Table I for methods that used techniques other than PCA and SOM, these methods have underperformance in contrary to our proposed system in other aspects of recognition performance.

Abuzneid et al. [13] combined different types of machine learning methods that led to a computationally intensive solution including increasing the system latency resulted from by recognizing facial images in the test phase of the system.

On the other hand, authors [13] had conducted image-preprocessing including cropping, resizing, and histogram equalization and it took 25 hours to perform the backpropagation network training, where nothing was recorded about the time required for the testing stage. The proposed method proposed by Abuzneid et al. [13] has several cascaded computational blocks as LBPH, BBNN representation, and multi-KNN which is considered a huge computational overhead.

As another example, in the proposed system by Sun et al. [43], two types of descriptors, Local Gradient Number Pattern (LGNP) and Fuzzy Convex-Concave Partition (FCCP) were used to represent facial data. Moreover, a deep neural network was used as a classification engine which represents a recognition system with high complexity.

Lawrence et al. [24] reported in their experimental results that without preprocessing step, the resultant error yielded two times greater error rate which means that the average recognition rate will decrease down to 88.5%.

The authors reported that the training time required to train the CNN network was approximately 4 hours. Although Gupta et al. [21] achieved a comparable recognition performance using SIFT-64 and SURF-64 facial data representation that cascaded by random forest as a classifier, authors [21], as can be noted from Table II used 80% of the ORL dataset to train their model. However, as shown in Table III, our model can achieve a higher recognition performance reached up to 98.16% for using this percentage of training data.

As a typical example of image pre-processing that can enhance the overall performance but at the same time can blur many of the facial recognition performance drawbacks is that proposed by Qin et al [44] where down sampling algorithm is used to resize the facial image down into 46×56 pixel matrix proceeded by a non-linear transformation stretching gray image enhancement as a pre-processing step. Collaborative Representation (CR) was used as a feature extractor whereas an enhanced KNN was used as a classification engine. In our case, we have used raw images for training and testing to test the robustness of our system against different facial effects in one hand and to keep the computational cost down to the minimum in the other hand.

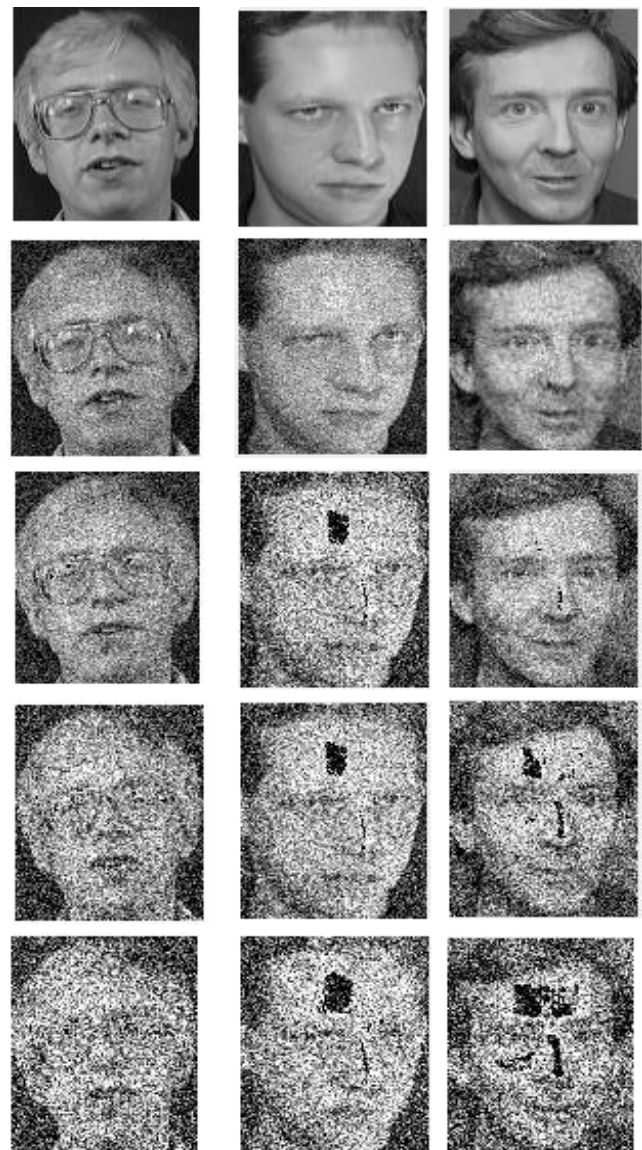


Fig. 8. Subset of Original Test ORL Dataset and the Corresponding Noisy Ones after Adding Gaussian Noise with Zero mean $\mu = 0$ and different Levels of Variations σ . First Row Represents the Original ORL Test Dataset. Second Row Represents Noisy Test Images using $\sigma = 0.01$. Third Row Represents the Noisy Test Image using $\sigma = 0.02$. Forth Row Represents the Noisy Test Images using $\sigma = 0.05$. Fifth Row Represents the Noisy Test Images using $\sigma = 0.09$.

As shown in Fig. 8 and as revealed in Table III, our proposed system can handle adding additional white Gaussian noise at different levels of variations and beyond that can occur in real-time photo capturing.

As with all face recognizers that built on the machine learning techniques, one major limitation to our proposed model is the need to re-train in case of a new persons (subjects) are added to the database. However, as shown in Table IV, the time cost required for training 200 images is less than 4 seconds and that required for 360 images is less than 8 seconds which is substantially low if it is to be used in real-time applications.

VI. CONCLUSION

This paper has presented an agglomerative clustered SOM-based face recognition model where regular PCA was used to extract the eigenfaces of the facial data for purpose of facial data representation whereas supervised clustered SOM was used as the recognition engine of the proposed model. The proposed model is found to be efficient in terms of time cost for both training and testing stages where it takes less than 3.7 seconds to train the model using 200 training images whereas identifying one single image takes less than $7e-5$ seconds. Therefore, the online training version of the system can be used efficiently for real-life applications where the cost of training and testing is as important issue same as the recognition accuracy. The proposed model is rigorously validated using the ORL dataset and based on the comparative analysis conducted in this work, the recognition performance is superior to methods that used SOM/SOM-variants. Moreover, the system shows robustness against adding Gaussian noise at different levels of variations as well as robustness against using the raw facial data without the need for an image pre-processing step. Using the clustered SOM with features processors other than PCA or using an ensemble of SOM networks are a possible extension for our future work.

REFERENCES

- [1] M. Martinez and A. C. Kak, "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, 2001, doi: 10.1109/34.908974.
- [2] A. Eleyan and H. Demirel, "PCA and LDA Based Face Recognition Using Feedforward Neural Network Classifier," in *Multimedia Content Representation, Classification and Security*, vol. 4105, B. Günsel, A. K. Jain, A. M. Tekalp, and B. Sankur, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 199–206. doi: 10.1007/11848035_28.
- [3] C. L. Chowdhary and D. Acharjya, "Singular Value Decomposition–Principal Component Analysis-Based Object Recognition Approach," in *Bio-Inspired Computing for Image and Video Processing*, Chapman and Hall/CRC, 2018, pp. 323–341.
- [4] J. Pan, X.-S. Wang, and Y.-H. Cheng, "Single-Sample Face Recognition Based on LPP Feature Transfer," *IEEE Access*, vol. 4, pp. 2873–2884, 2016, doi: 10.1109/ACCESS.2016.2574366.
- [5] K. Sett et al., "Face recognition using fusion of spatial and temporal features," in *2018 Emerging Trends in Electronic Devices and Computational Techniques (EDCT)*, Kolkata, India, Mar. 2018, pp. 1–4. doi: 10.1109/EDCT.2018.8405063.
- [6] Q. Feng et al., "Superimposed Sparse Parameter Classifiers for Face Recognition," *IEEE Trans. Cybern.*, vol. 47, no. 2, Art. no. 2, Feb. 2017, doi: 10.1109/TCYB.2016.2516239.
- [7] J. Sun, Y. Lv, C. Tang, H. Sima, and X. Wu, "Face Recognition Based on Local Gradient Number Pattern and Fuzzy Convex-Concave Partition," *IEEE Access*, vol. 8, pp. 35777–35791, 2020, doi: 10.1109/ACCESS.2020.2975312.
- [8] B. Jyostna Devi, N. Veeranjanyulu, and K. V. K. Kishore, "A novel face recognition system based on combining eigenfaces with fisher faces using wavelets," *Procedia Computer Science*, vol. 2, pp. 44–51, 2010, doi: 10.1016/j.procs.2010.11.007.
- [9] Lying Ma, Yegui Xiao, K. Khorasani, and R. K. Ward, "A new facial expression recognition technique using 2-D DCT and K-means algorithm," in *2004 International Conference on Image Processing*, 2004. ICIP '04., Singapore, 2004, vol. 2, pp. 1269–1272. doi: 10.1109/ICIP.2004.1419729.
- [10] C. Cifarelli, G. Manfredi, and L. Nieldu, "Statistical Face Recognition via a k-Means Iterative Algorithm," in *2008 Seventh International Conference on Machine Learning and Applications*, San Diego, CA, USA, 2008, pp. 888–891. doi: 10.1109/ICMLA.2008.146.
- [11] S. Tan, X. Sun, W. Chan, L. Qu, and L. Shao, "Robust Face Recognition With Kernelized Locality-Sensitive Group Sparsity Representation," *IEEE Trans. on Image Process.*, vol. 26, no. 10, pp. 4661–4668, Oct. 2017, doi: 10.1109/TIP.2017.2716180.
- [12] J. Kong, M. Chen, M. Jiang, J. Sun, and J. Hou, "Face Recognition Based on CSGF(2D)2PCANet," vol. 6, p. 13, 2018.
- [13] M. A. Abuzneid and A. Mahmood, "Enhanced Human Face Recognition Using LBPH Descriptor, Multi-KNN, and Back-Propagation Neural Network," *IEEE Access*, vol. 6, pp. 20641–20651, 2018, doi: 10.1109/ACCESS.2018.2825310.
- [14] A. Bouzalmat, N. Belghini, A. Zarghili, and J. Kharroubi, "Face Detection and Recognition Using Back Propagation Neural Network and Fourier Gabor Filters," *SIPIJ*, vol. 2, no. 3, pp. 15–21, Sep. 2011, doi: 10.5121/sipij.2011.2302.
- [15] S. Lawrence, C. L. Giles, and A. D. Back, "Face Recognition: A Convolutional Neural-Network Approach," *IEEE TRANSACTIONS ON NEURAL NETWORKS*, vol. 8, no. 1, p. 16, 1997.
- [16] S. Almadby and L. Elrefaei, "Deep Convolutional Neural Network-Based Approaches for Face Recognition," *Applied Sciences*, vol. 9, no. 20, p. 4397, Oct. 2019, doi: 10.3390/app9204397.
- [17] J. Sun, Y. Lv, C. Tang, H. Sima, and X. Wu, "Face Recognition Based on Local Gradient Number Pattern and Fuzzy Convex-Concave Partition," *IEEE Access*, vol. 8, pp. 35777–35791, 2020, doi: 10.1109/ACCESS.2020.2975312.
- [18] T. Goel and R. Murugan, "Classifier for Face Recognition Based on Deep Convolutional - Optimized Kernel Extreme Learning Machine," *Computers & Electrical Engineering*, vol. 85, p. 106640, Jul. 2020, doi: 10.1016/j.compeleceng.2020.106640.
- [19] B. Hamdan and K. Mokhtar, "Face recognition using Angular Radial Transform," *Journal of King Saud University - Computer and Information Sciences*, vol. 30, no. 2, pp. 141–151, Apr. 2018, doi: 10.1016/j.jksuci.2016.10.006.
- [20] J. Kong, M. Chen, M. Jiang, J. Sun, and J. Hou, "Face Recognition Based on CSGF(2D) 2 PCANet," *IEEE Access*, vol. 6, pp. 45153–45165, 2018, doi: 10.1109/ACCESS.2018.2865425.
- [21] S. Gupta, K. Thakur, and M. Kumar, "2D-human face recognition using SIFT and SURF descriptors of face's feature regions," *Vis Comput*, vol. 37, no. 3, pp. 447–456, Mar. 2021, doi: 10.1007/s00371-020-01814-8.
- [22] V.-E. Neagoe, A.-C. Mugioiu, and I.-A. Stanculescu, "Face Recognition using PCA versus ICA versus LDA cascaded with the neural classifier of Concurrent Self-Organizing Maps," in *2010 8th International Conference on Communications, Bucharest, Romania*, Jun. 2010, pp. 225–228. doi: 10.1109/ICCOMM.2010.5509041.
- [23] D. Kumar, C. S. Rai, and S. Kumar, "Face Recognition using Self-Organizing Map and Principal Component Analysis," in *2005 International Conference on Neural Networks and Brain*, Beijing, China, 2005, vol. 3, pp. 1469–1473. doi: 10.1109/ICNNB.2005.1614908.
- [24] S. Lawrence, C. L. Giles, Ah Chung Tsoi, and A. D. Back, "Face recognition: a convolutional neural-network approach," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 98–113, Jan. 1997, doi: 10.1109/72.554195.
- [25] J. Ruiz and R. Jaime, "Backpropagation and SOM for face feature recognition," in *Proceedings of 3rd IEEE International Conference on Image Processing*, Lausanne, Switzerland, 1996, vol. 3, pp. 487–490. doi: 10.1109/ICIP.1996.560537.
- [26] P. Yodkhad, A. Kawewong, and K. Patanukhom, "Approximate nearest neighbor search using self-organizing map clustering for face recognition system," in *2014 International Computer Science and Engineering Conference (ICSEC)*, Khon Kaen, Thailand, Jul. 2014, pp. 151–156. doi: 10.1109/ICSEC.2014.6978186.
- [27] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang, "Robust face recognition from a single training image per person with kernel-based som-face," 2004, pp. 858–863.
- [28] Yang Zhi and Gu Ming, "A SOM-Wavelet Networks for Face Identification," in *2005 IEEE International Conference on Multimedia and Expo*, Amsterdam, The Netherlands, 2005, pp. 852–855. doi: 10.1109/ICME.2005.1521557.

- [29] Xiaoyang Tan, Songcan Chen, Zhi-Hua Zhou, and Jun Liu, "Face Recognition Under Occlusions and Variant Expressions With Partial Similarity," *IEEE Trans. Inform. Forensic Secur.*, vol. 4, no. 2, pp. 217–230, Jun. 2009, doi: 10.1109/TIFS.2009.2020772.
- [30] I. Q. Monteiro, S. D. Queiroz, A. T. Carneiro, L. G. Souza, and G. A. Barreto, "Face recognition independent of facial expression through SOM-based classifiers," in *2006 International Telecommunications Symposium, Fortaleza, Ceara, Brazil, Sep. 2006*, pp. 263–268. doi: 10.1109/ITS.2006.4433281.
- [31] A. Eleyan and H. Demirel, "PCA and LDA Based Face Recognition Using Feedforward Neural Network Classifier," in *Multimedia Content Representation, Classification and Security*, vol. 4105, B. Günsel, A. K. Jain, A. M. Tekalp, and B. Sankur, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 199–206. doi: 10.1007/11848035_28.
- [32] T. Kohonen, *Self-Organizing Maps*, 2nd ed., vol. 30. Berlin: Springer-Verlag Berlin Heidelberg, 1997.
- [33] R. D. Lawrence, "A Scalable Parallel Algorithm for Self-Organizing Maps with Applications to Sparse Data Mining Problems," . . . , p. 25.
- [34] AT & T Laboratories Cambridge, "ORL: The Database of Faces." <https://cam-orl.co.uk/facedatabase.html> (accessed Jul. 01, 2021).
- [35] Yang Zhi and Gu Ming, "A SOM-Wavelet Networks for Face Identification," in *2005 IEEE International Conference on Multimedia and Expo, Amsterdam, The Netherlands, 2005*, pp. 852–855. doi: 10.1109/ICME.2005.1521557.
- [36] D. Kumar, C. S. Rai, and S. Kumar, "Face Recognition using Self-Organizing Map and Principal Component Analysis," p. 5.
- [37] S. Tan, X. Sun, W. Chan, L. Qu, and L. Shao, "Robust Face Recognition With Kernelized Locality-Sensitive Group Sparsity Representation," *IEEE Trans. on Image Process.*, vol. 26, no. 10, Art. no. 10, Oct. 2017, doi: 10.1109/TIP.2017.2716180.
- [38] V.-E. Neagoe, A.-C. Mugioiu, and I.-A. Stanculescu, "Face Recognition using PCA versus ICA versus LDA cascaded with the Neural Classifier of Concurrent Self-Organizing Maps," p. 4.
- [39] J. Kong, M. Chen, M. Jiang, J. Sun, and J. Hou, "Face Recognition Based on CSGF(2D)2PCANet," vol. 6, p. 13, 2018.
- [40] Y. Qin, L. Sun, and Y. Xu, "Exploring of alternative representations of facial images for face recognition," *Int. J. Mach. Learn. & Cyber.*, vol. 11, no. 10, Art. no. 10, Oct. 2020, doi: 10.1007/s13042-020-01116-4.
- [41] G. Guo and N. Zhang, "A survey on deep learning based face recognition," *Computer Vision and Image Understanding*, vol. 189, p. 102805, Dec. 2019, doi: 10.1016/j.cviu.2019.102805.
- [42] "SOM toolbox programming team," SOM Toolbox. <http://www.cis.hut.fi/projects/somtoolbox/>.
- [43] J. Sun, Y. Lv, C. Tang, H. Sima, and X. Wu, "Face Recognition Based on Local Gradient Number Pattern and Fuzzy Convex-Concave Partition," *IEEE Access*, vol. 8, pp. 35777–35791, 2020, doi: 10.1109/ACCESS.2020.2975312.
- [44] Y. Qin, L. Sun, and Y. Xu, "Exploring of alternative representations of facial images for face recognition," *Int. J. Mach. Learn. & Cyber.*, vol. 11, no. 10, Art. no. 10, Oct. 2020, doi: 10.1007/s13042-020-01116-4.