



Face Recognition by Using 2D Orthogonal Subspace Projections

Semih Ergin¹, Sahin Isik^{2*}, Mehmet Bilginer Gulmezoglu¹

¹ Department of Electrical and Electronics Engineering, Eskisehir Osmangazi University, Meselik Campus, Eskişehir 26480, Turkey

² Department of Computer Engineering, Eskisehir Osmangazi University, Meselik Campus, Eskişehir 26480, Turkey

Corresponding Author Email: sahini@ogu.edu.tr

<https://doi.org/10.18280/ts.380105>

ABSTRACT

Received: 12 November 2020

Accepted: 26 January 2021

Keywords:

face recognition, common matrix approach, support vector machine, convolutional neural networks, 2D feature extraction

In this paper, the implementations and comparison of some classifiers along with 2D subspace projection approaches have been carried out for the face recognition problem. For this purpose, the well-known classifiers such as K-Nearest Neighbor (K-NN), Common Matrix Approach (CMA), Support Vector Machine (SVM) and Convolutional Neural Network (CNN) are conducted on low dimensional face representations that are determined from 2DPCA-, 2DSVD- and 2DFDA approaches. CMA, which is a 2D version of the Common Vector Approach (CVA), finds a common matrix for each face class. From the experimental results, we have observed that the SVM presents a dominant performance in general. When overall results of all datasets are considered, CMA is slightly superior to others in case of 2DPCA- and 2DSVD-based features matrices of the AR dataset. On the other side, CNN is better than other classifiers when it comes to develop a face recognition system based on original face samples and 2DPCA-based feature matrices of the Yale dataset. The experimental results indicate that use of these feature matrices with CMA, SVM, and CNN in classification problems is more advantageous than the use of original pixel matrices in the sense of both processing time and memory requirement.

1. INTRODUCTION

The automatic recognition of individual faces is a challenging area for the pattern classification. Automated face recognition generally involves different stages, namely face detection, to decide the position and size of the facial images, feature extraction to determine the most discriminative properties of facial images and face classification to identify an unknown face according to models of faces in a face dataset. In many applications of face classification algorithms, images taken in a restricted location are controlled by illumination, least occlusion of facial images, regular background [1]. For more robust performance, a face classification system should be flexible in terms of the aspects of pose, lighting and expression.

Subspace methods are usually utilized to attain feature space with low dimension among the existing face recognition techniques. There are both experimental and methodical reasons for utilizing low-dimensional subspaces to model image variations in individual faces under different illumination situations [2]. Subspace methods such as PCA, LDA and CVA are generally used in signal processing and computer vision areas as effective methods for both size decline and orientation of images [2-6]. There are also more efficient linear subspace analysis methods that use null-space face [7].

The motivation under the PCA method is to obtain a subspace in which variation is maximized, meanwhile some undesirable variations (resulting from changes in illumination, facial expressions, presentation points, etc.) can be preserved in order to enhance the recognition rate [8]. Subspace methods

such as Independent Component Analysis (ICA) [8, 9], PCA+Null Space [2] and Kernel PCA (KPCA) [10] are the variations or expansions of PCA to report statistical dependencies with higher order. Besides, Two Dimensional Principal Component Analysis (2D-PCA) is developed in order to easily deal with 2D data and the recognition performance obtained using 2D-PCA is superior to that obtained using 1D-PCA [6]. A new technique called Structural 2D-PCA (S2D-PCA) that identifies the structural information for discrimination of images was proposed [11]. More recently, some authors [12] proposed the directional 2DPCA (D2DPCA) that can obtain features from the matrices in any direction. In order to efficiently utilize all the features obtained from D2DPCA, a D2DPCA bank performed in several directions was combined to produce a multi-directional 2DPCA similarity score level combination method for face recognition. Besides 2D-PCA can be used with the K-Nearest-Neighbor (K-NN) classifier in face recognition studies, it is also one of the effective image representation methods.

Additionally, more effective solutions, specifically the Direct LDA (D-LDA) and Fisher's LDA (FLDA) methods, had also been presented for face classification problems [13-17]. In D-LDA, after the null-space of between-class covariance matrix, S_B , is removed, the projection vectors minimizing the within-class scatter in the transformed space are chosen from the range space of S_B . LDA overcomes the restrictions of PCA by using Fisher's LDA (FLDA) [13]. Park and Sim [18] utilized FLDA for appearance-based face representation in order to implement their Modified Census Transform (MCT) based face recognition approach. While doing well in most cases, LDA-based methods frequently

unsuccessful to perform well when facial images are subjected to complex changes in viewing angles, lighting, or facial expression [15]. FLDA was used as a solution to one sample problem in face classification [19]. Moreover, the efforts had been geared up to deal with the small sample size (sss) problem in LDA using Two Dimensional Fisher Discriminant Analysis (2DFDA) [20].

Furthermore, Singular value decomposition (SVD) is an approach of defining and sorting dimensions in which data points exhibit the greatest variation. The uniform eigen-space SVD method was proposed to recognize faces on enhanced images [21]. Even though many systems have been used to determine faces, they are not effective for different illumination conditions and facial expressions; so it is required to find an alternative solution by combining different face recognition methods and feature extraction techniques.

Nowadays, a great deal of efforts has been invested by developing the Convolutional Neural Network (CNN) based systems for face detection [22] or face recognition [23]. In a study, the local shallow CNNs [24] are performed on predetermined regions that are surrounding the five landmarks (left eye center, right eye center, nose apex, left and right corner of the mouth). The researchers believe that more meaningful and discriminative features would be extracted once the shallow CNNs are operated in different regions. After 50% cross validation strategy, the verification performance was reported as 97.48% accuracy rate. Some researchers focused on the improving of learning capability of CNN features [23]. Two different loss functions, namely Compact Discriminative Loss and Advanced Compact Discriminative Loss, had been evaluated for maximizing the within class similarity and minimizing between class similarity. After using three well-known CNNs (LeNet [25], CNN-M [26] and ResNet-50 [27]), the best scores are achieved with pre-trained ResNet models. In a particular study [28], the authors wanted to see the influence of gender (men and women) and age (youth, middle and old) factors on face recognition. The impact of these factors are independently analyzed with a Multi-Task Cascaded Convolutional Networks (MTCNN) for male and female groups. It was found that there is a high effect of age factor for face recognition scores for men, whereas age's influence is considerably low for women. The main drawback of CNN based face recognition can be explained since the weights of CNN are sensitive to noise in data. Therefore, a pre-trained model may produce unwanted classification scores if some pixels of samples are corrupted with noise.

In this paper, the implementations and comparison of classification methods, K-Nearest Neighbor (K-NN) [6, 20, 29], Common Matrix Approach (CMA) [30, 31], Support Vector Machine (SVM) and Convolutional Neural Network (CNN), are proposed for the classification of face images represented with the feature matrices extracted from the 2DPCA, 2DSVD and 2DFDA methods. CMA can be considered as a 2D version of the Common Vector Approach (CVA) which is widely used in speech [32-34] and image processing [35-37] and also in motor fault diagnosis [38]. CMA finds a unique common matrix including the common or invariant features of each face class. Therefore, it is also flexible for the aspects of lighting and expression. SVM determines the optimal subplane that maximizes the distance between the hyperplane and the closest sample from hyperplane [39-43]. In addition, CNN finds optimal set of

weights adjusted to the smallest training error rate after feed-forward propagation and feed-backward propagation stages. Experimental findings show that the recognition rates obtained from the original images and feature matrices slightly differ from each other for AR-face dataset, the recognition rates obtained from the feature matrices are greater than those obtained from the original images for ORL and Yale databases. Eventually, it is found that the use of 2DPCA-, 2DFDA- and especially 2DSVD-based feature matrices remarkably decrease the processing time and memory necessity when compared with those obtained for original face matrices.

This paper is organized as follows: The feature extraction methods 2DPCA, 2DSVD and 2DFDA are given in Section 2 whereas Section 3 explains the theories of K-NN, CMA, SVM and CNN classifiers. The experimental studies are introduced in Section 4 and the conclusions are presented in Section 5.

2. 2D ORTHOGONAL PROJECTIONS FOR FEATURE EXTRACTION

In this study, three different methods referred as 2D orthogonal subspace methods were preferred for feature extraction from original face images.

2.1 Feature extraction using 2DPCA

The purpose of 2DPCA [6] is to project an image matrix onto a lower dimensional matrix. Assume that totally M images in a training process, the j^{th} image in this process is indicated with $m \times n$ matrix \mathbf{A}_j , and the mean of all training images is indicated with $\bar{\mathbf{A}}$. The column-column covariance matrix Φ_C can be computed by

$$\Phi_C = \frac{1}{M} \sum_{j=1}^M (\mathbf{A}_j - \bar{\mathbf{A}})^T (\mathbf{A}_j - \bar{\mathbf{A}}) \quad (1)$$

The total scatter of images is characterized by the trace of Φ_C . From this viewpoint, the following criterion are developed as

$$J(\mathbf{X}) = \mathbf{X}^T (\Phi_C) \mathbf{X} \quad (2)$$

where, \mathbf{X} is a unit column vector. The optimal projection axis, \mathbf{X}_{opt} , is the unit vector that maximizes $J(\mathbf{X})$; i.e., \mathbf{X}_{opt} is an eigenvector of Φ_C associated with the maximum eigenvalue. The optimal projection axes, $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_d$, are the eigenvectors of Φ_C , associated with the maximum d eigenvalues (The selection process for d will be explicitly explained in the fourth section.). The projected feature vectors $\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_d$, namely the primary components of \mathbf{A}_j , can be obtained as

$$\mathbf{Y}_k = \mathbf{A}_j \mathbf{X}_k \quad k = 1, 2, \dots, d \quad (3)$$

The obtained principal component vectors are laid together for constructing an $m \times d$ matrix $\mathbf{B}_j = [\mathbf{Y}_1 \mathbf{Y}_2 \dots \mathbf{Y}_d]$, that is referred to as the feature matrix of \mathbf{A}_j .

2.2 Feature extraction using 2DSVD

In the 2DSVD method [29], both the column-column covariance matrix and the row-row covariance matrix of all training images are calculated. In other words, this method considers the spatial relationships among the pixels in two directions (vertical and horizontal) although 2DPCA takes the spatial relationships among the pixels in only horizontal direction into account [6].

The image row-row covariance matrix Φ_R can be evaluated by

$$\Phi_R = \frac{1}{M} \sum_{j=1}^M (\mathbf{A}_j - \bar{\mathbf{A}})(\mathbf{A}_j - \bar{\mathbf{A}})^T \quad (4)$$

The total scatter of images is represented by the trace of Φ_R . By considering this viewpoint, the following criterion can be written:

$$J(\mathbf{P}) = \mathbf{P} (\Phi_R) \mathbf{P}^T \quad (5)$$

where, \mathbf{P} is a unit row vector. A set of projection axes, $\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_e$, are the eigenvectors of Φ_R corresponding to the first e maximum eigenvalues (The selection process of e will be explicitly explained in the fourth section.). The orthonormal eigenvectors, $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_d$, of Φ_C (given in the previous subsection) corresponding to the maximum d eigenvalues are also used for 2DSVD. The feature matrix, \mathbf{B}_j , of \mathbf{A}_j can be obtained as

$$\begin{aligned} \mathbf{B}_j &= \begin{bmatrix} \leftarrow \mathbf{P}_1 \rightarrow \\ \leftarrow \mathbf{P}_2 \rightarrow \\ \vdots \\ \leftarrow \mathbf{P}_e \rightarrow \end{bmatrix}_{e \times m} \times \mathbf{A}_j \times \begin{bmatrix} \uparrow & \uparrow & \dots & \uparrow \\ \mathbf{X}_1 & \mathbf{X}_2 & \dots & \mathbf{X}_d \\ \downarrow & \downarrow & \dots & \downarrow \end{bmatrix}_{n \times d} \\ &= \begin{bmatrix} \uparrow & \uparrow & \dots & \uparrow \\ \mathbf{Y}_1 & \mathbf{Y}_2 & \dots & \mathbf{Y}_d \\ \downarrow & \downarrow & \dots & \downarrow \end{bmatrix}_{e \times d} \end{aligned} \quad (6)$$

It can be clearly seen that a lower-dimensional feature matrix for \mathbf{A}_j is obtained by this method when compared with that obtained by the 2DPCA.

2.3 Feature extraction using 2DFDA

The fundamentals of the 2DFDA method [20] are slightly different from those of both 2DPCA and 2DSVD methods. In order to implement 2DFDA, first of all, the between-class and within-class covariance matrices are calculated for all training images. Assume that totally M images in a training process of S classes, the average for all training images in i^{th} class is $\bar{\mathbf{A}}_i$ and the average of training images in all classes is $\bar{\mathbf{A}}$. The between-class covariance matrix (Φ_B) can be evaluated by

$$\Phi_B = \sum_{i=1}^S N_i (\bar{\mathbf{A}}_i - \bar{\mathbf{A}})^T (\bar{\mathbf{A}}_i - \bar{\mathbf{A}}) \quad (7)$$

where, N_i is the number of images in i^{th} class. The within-class

covariance matrix (Φ_W) is given as:

$$\Phi_W = \sum_{i=1}^S \sum_{j=1}^{N_i} (\mathbf{A}_{j,i} - \bar{\mathbf{A}}_i)^T (\mathbf{A}_{j,i} - \bar{\mathbf{A}}_i) \quad (8)$$

where, $\mathbf{A}_{j,i}$ is the j^{th} training image in i^{th} class. The total scatter of images is represented by the trace of the Fisher's criterion. By considering this viewpoint, the following criterion can be written:

$$J(\mathbf{R}) = \frac{\mathbf{R}^T \Phi_B \mathbf{R}}{\mathbf{R}^T \Phi_W \mathbf{R}} \quad (9)$$

where, \mathbf{R} is a unit column vector. A set of projection axes, $\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_f$, are the eigenvectors corresponding to the maximum f eigenvalues of $(\Phi_W)^{-1} \Phi_B$ (The selection process for f will be explicitly explained in the fourth section.). The projected feature vectors, $\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_f$, of the sample image \mathbf{A}_j , can be obtained as

$$\mathbf{Y}_k = \mathbf{A}_j \mathbf{R}_k \quad k=1, 2, \dots, f \quad (10)$$

The obtained vectors are used to construct an $m \times f$ matrix, $\mathbf{B}_j = [\mathbf{Y}_1 \ \mathbf{Y}_2 \ \dots \ \mathbf{Y}_f]$, which is the feature matrix of \mathbf{A}_j .

3. CLASSIFIERS

3.1 K-Nearest Neighbor (K-NN) classifier

First of all, let us define the distance between two arbitrary feature matrices, \mathbf{B}_i and \mathbf{B}_j , as the following:

$$distance(\mathbf{B}_i, \mathbf{B}_j) = \sum_{k=1}^g \|\mathbf{Y}_k^i - \mathbf{Y}_k^j\|^2 \quad (i, j = 1, 2, \dots, M) \quad (11)$$

where, \mathbf{Y}_k^i and \mathbf{Y}_k^j are the k^{th} column vectors of \mathbf{B}_i and \mathbf{B}_j , respectively. Suppose that the image feature matrices in the training set are indicated with $\mathbf{B}_1, \mathbf{B}_2, \dots, \mathbf{B}_M$. A test feature matrix, \mathbf{B}_{test} , if

$$L = \arg \min_{1 \leq j \leq M} \{ distance(\mathbf{B}_{test}, \mathbf{B}_j) \} \quad \text{and} \quad (12)$$

$$\mathbf{B}_L \in C_i \quad (i=1, 2, \dots, S) \quad (13)$$

then the test feature matrix \mathbf{B}_{test} belongs to the class C_i . In Eq. (13), \mathbf{B}_L indicates the feature matrix whose distance with all feature matrices in the training set is minimum.

3.2 Common Matrix Approach (CMA)

CMA is a 2D version of the Common Vector Approach (CVA) that is usually preferred in speech recognition [32, 34, 38], speaker recognition [44], image processing [33, 37], and motor fault diagnosis [38]. In CMA, a common matrix, representing the unvarying features for each face class is calculated from the images in the training set by discarding the alterations in images of the same class.

Training Stage of CMA

- Let us assume that the number of classes and images in the training set of class C are S and N , respectively. The matrix of the j^{th} training image in class C is indicated with \mathbf{A}_j^C . Each image has the same size of $(m \times n)$.
- \mathbf{A}_j^C is represented by the sum of the common matrix, \mathbf{A}_{com}^C , and the matrix which includes differences resulting from the illumination and facial expressions, $\mathbf{A}_{j,dif}^C$,

$$\mathbf{A}_j^C = \mathbf{A}_{com}^C + \mathbf{A}_{j,dif}^C \quad (14)$$

- If the first image matrix in class C is selected as the reference matrix, all the difference matrices, \mathbf{D}_k^C , pertaining to class C can be expressed:

$$\mathbf{D}_k^C = \mathbf{A}_{k+1}^C - \mathbf{A}_1^C \text{ for } k = 1, 2, \dots, N-1 \quad (15)$$

Same process is repeated until the difference matrices of all classes are determined.

- The difference matrices are orthogonalized by applying the Gram-Schmidt orthogonalization process [34] as in Eq. (16) so that \mathbf{W}_k^C (the orthogonal matrices) and \mathbf{Z}_k^C (orthonormal matrices) ($k = 1, 2, \dots, N-1$) are obtained.

$$\mathbf{W}_k^C = \mathbf{D}_k^C - \sum_{r=1}^{k-1} \left\{ \frac{\langle \mathbf{D}_k^C, \mathbf{W}_r^C \rangle \mathbf{W}_r^C}{\|\mathbf{W}_r^C\|_F^2} \right\} \quad (16)$$

$$\mathbf{Z}_k^C = \frac{\mathbf{W}_k^C}{\|\mathbf{W}_k^C\|_F} \text{ for } k = 1, 2, \dots, (N-1)$$

In the Eq. 16, $\langle \mathbf{D}, \mathbf{W} \rangle$ denotes the scalar product of \mathbf{D} and \mathbf{W} , and it is calculated as the *trace*($\mathbf{D}^T \mathbf{W}$). The matrices, $\mathbf{Z}_1^C, \mathbf{Z}_2^C, \dots, \mathbf{Z}_{N-1}^C$, span the difference subspace for class C .

- The projection of any image matrix \mathbf{A}_j^C in the training set of class C onto the difference subspace of class C provides the difference matrix $\mathbf{A}_{j,dif}^C$ as

$$\mathbf{A}_{j,dif}^C = \langle \mathbf{A}_j^C, \mathbf{Z}_1^C \rangle \mathbf{Z}_1^C - \langle \mathbf{A}_j^C, \mathbf{Z}_2^C \rangle \mathbf{Z}_2^C - \dots - \langle \mathbf{A}_j^C, \mathbf{Z}_{N-1}^C \rangle \mathbf{Z}_{N-1}^C \quad (17)$$

- The common matrix, \mathbf{A}_{com}^C , can be found by subtracting the matrix $\mathbf{A}_{j,dif}^C$ from the image matrix \mathbf{A}_j^C .

$$\mathbf{A}_{com}^C = \mathbf{A}_j^C - \mathbf{A}_{j,dif}^C \quad (18)$$

Common matrices ($\mathbf{A}_{com}^1, \mathbf{A}_{com}^2, \dots, \mathbf{A}_{com}^S$) for all classes are determined using the above procedure.

Test Stage of CMA

- A test image matrix, \mathbf{A}_{test} , is projected onto the difference subspace of class C . Then, the remaining matrix, \mathbf{R}_{test}^C , can be obtained by subtracting this projection from \mathbf{A}_{test} :

$$\mathbf{R}_{test}^C = \mathbf{A}_{test} - \langle \mathbf{A}_{test}, \mathbf{Z}_1^C \rangle \mathbf{Z}_1^C - \langle \mathbf{A}_{test}, \mathbf{Z}_2^C \rangle \mathbf{Z}_2^C - \dots - \langle \mathbf{A}_{test}, \mathbf{Z}_{N-1}^C \rangle \mathbf{Z}_{N-1}^C \quad (19)$$

- The decision is conducted using the following criterion for the classification:

$$K = \arg \min_{1 \leq C \leq S} \left\{ \|\mathbf{R}_{test}^C - \mathbf{A}_{com}^C\|_F^2 \right\} \quad (20)$$

where, K is the index of the class for which the distance $\|\mathbf{R}_{test}^C - \mathbf{A}_{com}^C\|_F^2$ provides the least value for \mathbf{A}_{test} . Thus, \mathbf{A}_{test} , is classified to that class.

3.3 Support Vector Machine (SVM)

In a briefly explained manner, SVM [39-43] is presented as a two-class classifier and it finds the best subplane which maximizes the distance between the best hyperplane and the closest sample to this hyperplane. If the training set is defined as $TS = \{(\mathbf{x}_1, L_1), (\mathbf{x}_2, L_2), \dots, (\mathbf{x}_M, L_M)\}$ for a two-class problem, a test vector (\mathbf{x}_{test}) can be categorized by considering this decision function:

$$f(\mathbf{x}_{test}) = \sum_{i=1}^M \left\{ \alpha_i L_i (\mathbf{x}_i^T \mathbf{x}_{test}) + b \right\} \quad (21)$$

In this Eq., \mathbf{x}_i ($i = 1, 2, \dots, M$) is a training vector and L_i ($L_i \in \{-1, 1\}$) is the class label. Besides, α_i are the nonzero constants which are solution of the quadratic programming problem, and b is a bias term. The sign of this decision function specifies the class to which \mathbf{x}_{test} is assigned. If one needs to use the SVM classifier in a multi-class problem (with S classes), " $S(S-1)/2$ " classifiers are to be constructed [45]. In this study, the linear SVM classifier is performed [46].

3.4 Convolutional Neural Network (CNN)

The algorithms based on CNN were widely utilized to overcome the problems in image processing and pattern recognition tasks. The key motivation behind a CNN based learning methodology is to map the discriminative information to huge number and size of filters. Typically, a CNN algorithm [47] includes four common layers: convolution layers, pooling/subsampling layers, non-linear layers (activation) and fully-connected layers. The CNNs are able to produce optimal hierarchical features through backpropagation procedures. Since CNNs are successful in terms of performance, efforts have been geared up towards the replacement of traditional algorithms with CNN-based ones.

In order to analyze the capability of CNN for face recognition, some experiments on low dimensional representations of images that are determined with 2DPCA, 2DFDA and 2DSVD methods are conducted. Moreover, we have compared the performance of our shallow CNN with well-known methods after training and testing of both original images and their projections onto a subspace. Typically, the utilized CNN structure is 64-64-64-64-1 and the filter size is 3x3 with stride 1. The output size after convolution remains same by padding zeros prior to convolution procedure at each layer. The tuned parameters for CNN are given 1×10^{-5} for learning rate, 16 for batch size and 100 for epoch. The different

combinations of batch sizes and epoch values are used to increase the accuracy of recognition. Also, the Rectified Linear Unit (ReLU) is chosen as an activation function in hidden layers. The optimization function is Adam.

The Figure 1 shows our utilized shallow CNN model, which is performed on original and projected face data. The dimension of input data is preserved after convolution operations. The FC refers to the Fully Connected layer that holds the meaningful features to be activated for generating n (number of classes) distinct probability scores. At the last head of CNN, the 4096 features are transformed with Sofmax activation in order to get probability scores when assigning the input data to n distinct classes. The loss function is a categorical cross-entropy. For all methods, we have determined CNN structure as 64-64-64-64-1 in order to settle the memory overflow issue. The performance of CNN is measured with leave-one-out cross-validation strategy. The training procedure for particular cross-validations is repeated many times based on different combinations of batch-size and epoch value in order to get the highest accuracy score. All of experimental processes with CNN is performed in Python environment.

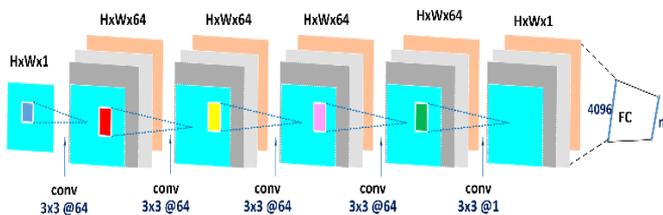


Figure 1. Utilized CNN architecture

4. PROPOSED FACE RECOGNITION SYSTEM

The utilized 2D projection approaches based face recognition is summarized with Figure 2. One can observe that the dimension of a face sample is reduced after projecting onto the orthonormal subspaces obtained from 2DPCA, 2DSVD and 2DFDA methods. Utilizing these low dimensional face representations improve the classification performance, reducing the memory consumption. Besides, the 2D projection based dimension reduction strategy substantially reduce the training time of complex classifiers like SVM and CNN. Additionally, one can emphasize that the 2D projection approaches are boosting the generalization and interpretation of a trained model as well as removing the redundant information from face data. Also, we compare the performance of reduced face samples with original ones.

4.1 Datasets

AR Face Dataset: The AR face dataset was compiled by two scientists [48] and comprises of the RGB face poses of 126 different persons: 70 of whom are men and the remaining 56 are women. Some cropped face portions (100x85) from the original dataset are presented in Figure 3. The poses have different facial expressions and some of them were taken in distinct lighting conditions. Besides this, some of the faces are exposed to irregular occlusions.

All facial images were taken in two completely distinct sessions. The elapsed time between two sessions was two weeks. 13 different poses were recorded for each person in the

first and second sessions, respectively. Therefore, totally 26 different poses with a dimension of 576x768 were recorded and represented in 24-bit in depth.

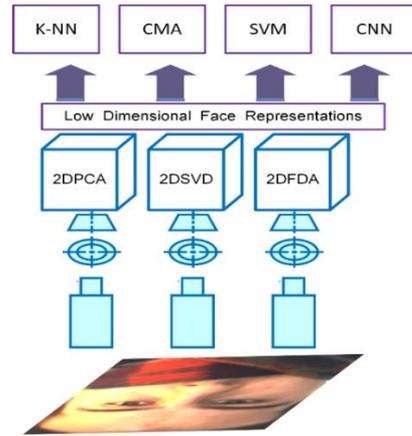


Figure 2. Proposed 2D projection based dimension reduction technique for face recognition



Figure 3. Face portions cropped from the original poses in the AR face dataset

ORL Face Dataset: The ORL dataset [49, 50] comprises of the images of 40 different persons, each of which has 10 images. The 10 images include variations in facial expression, facial position and some other details. All face images were mostly taken in the front position with a dark background. The images are represented in 8-bit gray levels with the size of 92x112. The cropped versions of face samples have the pixel matrices with the size of 60x60. Some face samples from ORL dataset [24] are given in Figure 4.

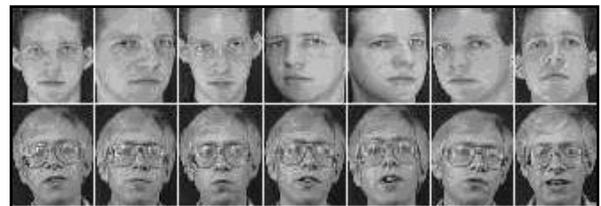


Figure 4. Some face samples from the ORL dataset

Yale Face Dataset: The third dataset utilized in this study is Yale Face dataset and it contains 165 grayscale images of 15 persons (14 of whom are men and the rest one is woman). 11 poses per individual were recorded and each pose was shot in one of these conditions: center-light, wearing glasses, happy, left-light, wearing no glasses, normal, right-light, sad, sleepy, surprised, and wink. The cropped facial images have the pixel matrices with the size of 152x126 in the dataset as given in Figure 5.



Figure 5. Face portions cropped from the original poses in the Yale dataset

4.2 Preprocessing stage

Some operations were conducted on all images in the AR and Yale face datasets. Initially, all images were converted into gray-scale images. Then, the pixel values of images were normalized to a (0-1) range corresponding to (black-white) values. The portions including only faces of individuals are automatically cropped. These cropped face portions are resized to 100x85 for AR face dataset and 152x126 for the Yale face dataset. These pre-processing operations were not implemented for the ORL dataset because the images are already cropped in the downloaded ORL dataset. The only operation made for the ORL dataset was that all images were resized to 60x60. Finally, zero-mean and unit-variance matrices were obtained by subtracting the mean pixel value from each pixel and dividing by the standard deviation in the cropped image matrices, respectively. 14 images with different facial expressions and lighting situations were chosen per 30 male and 20 female subjects, randomly selected for AR face dataset. Totally 700 (50 persons x 14 poses per person) faces were used in AR face experiments. Meanwhile, the total number of faces used in the ORL experiments is equal to 400 (40 persons x 10 poses per person) whereas the total number of faces used in the Yale dataset is 165 (15 persons x 11 poses per person).

4.3 Obtaining 2DPCA-based feature matrices

First of all, the image column-column covariance matrix with a size of $n \times n$ is computed (n is 85, 60, and 126 for the AR, ORL and Yale face datasets, respectively), an eigenvalue-eigenvector decomposition was applied. Thus, n eigenvalues and n eigenvectors were found. After these n eigenvalues were decreasingly ordered, the eigenvectors associated with the maximum d eigenvalues were selected. We can select d so that the sum of the largest eigenvalues is less than a constant percentage L of the whole eigenvalue set [51]. Thus, we perform the rule that is asserted by Eq. (22) for evaluating d value.

$$\left(\frac{\sum_{i=1}^d \lambda_i}{\sum_{i=1}^n \lambda_i} \right) < L \quad (22)$$

If $L = 90\%$, a good performance is achieved while preserving a large amount of variance presented in the original space [52]. $L = 90\%$ gives the different number of eigenvalues for each leave-one-out step. The average number of these eigenvalues was equal to 12, 12, and 9 for the AR, ORL and Yale face datasets, respectively. A feature matrix, with a size of 100x12, 60x12, and 126x9 for the AR, ORL and Yale face datasets, respectively, was obtained by using Eq. (3). These size values (100x12, 60x12, and 126x9) obviously state an appreciable reduction compared to original face images. Therefore, a noticeable reduction in processing power and computational load were achieved since the sizes of the input matrices carry big importance for training and classification

parts. These feature matrices were then used in different classifiers.

4.4 Obtaining 2DSVD-based feature matrices

The 2DSVD method was applied to extract the feature matrices. At first, an eigenvalue-eigenvector decomposition was applied on the image row-row covariance matrix with a size of $m \times m$ (m is 100, 60, and 152 for the AR, ORL and Yale face datasets, respectively). Thus, m eigenvalues and m eigenvectors were obtained from each dataset. After these m eigenvalues were decreasingly ordered, the largest e eigenvalues (e is equal to 12, 14, and 12 for the AR, ORL and Yale face datasets, respectively) whose summation is less than 90% of total energy were chosen. An image column-column covariance matrix with a size of $n \times n$ was already found in the previous subsection. Again, the eigenvectors associated with the largest d eigenvalues were used in this method. Since the number of eigenvectors taken from the row-row and column-column covariance matrices were e and d , respectively; the feature matrices of the face images have the size of $e \times d$ (12x12, 14x12, and 12x9 for the AR, ORL and Yale face datasets, respectively). This explicitly shows a remarkable decrease compared with the sizes of original face poses. This circumstance results in a drastically high reduction from the points of processing time. These feature matrices were then applied to four different classifiers.

4.5 Obtaining 2DSVD-based feature matrices

First of all, an eigenvalue-eigenvector decomposition on $(\Phi_W)^{-1} \Phi_B$ was implemented so that n eigenvalues and n eigenvectors were computed. If these eigenvalues were decreasingly ordered, the maximum f eigenvalues (f is equal to 5, 45, and 80 for the AR, ORL and Yale face datasets, respectively) whose summation is less than 90% of total energy were chosen (refers to Eq. (22)). A feature matrix (with a size of 100x5, 60x45, and 152x80 for the AR, ORL and Yale face datasets, respectively) was attained for each image using the eigenvectors corresponding to the maximum f eigenvalues. These size values obviously imply an appreciable decrease compared with the sizes of original facial image matrices for each dataset. Therefore, a clearly advantageous case in processing power and computational load were realized with a purpose of boosting classification performance. These feature matrices were then treated for utilized classifiers.

5. PERFORMANCE EVALUATION

The original face images, 2DPCA-, 2DSVD- and 2DFDA-based feature matrices obtained from the AR, ORL and Yale datasets were separately classified using the K-NN, SVM, CMA and CNN classifiers. The number of nearest-neighbor is 1 for K-NN classifier. Additionally, we have analyzed the performance of Large-Softmax concept for boosting the face recognition accuracy. The Additive Marging Softmax (AMS) [53] is integrated in place of Softmax layer in CNN. Instead of Categorical Cross Entropy, the loss function is chosen as Additive Margin Softmax Loss. Furthermore, the Random Forest (RF) classifier is applied to compare the performances of integrated learning versus single model learning. The motivation is that the decision made by multiple members may further boost the classification accuracy of system. For all

classifiers, the “leave-one-out” procedure was used since the number of images for each class is inadequate. Thus, this procedure was repeated 14, 10, and 11 times for the AR, ORL and Yale datasets, respectively. The average scores of “leave-one-out” steps are comparatively presented in Table 1 for original face samples, and the other three types of feature matrices.

In order to comprehensively compare the performances of classifiers together with the feature extraction methods, three datasets were considered as a benchmark evaluation. Experimental studies point out that the CMA and SVM classifiers are superior to the K-NN classifier for each type of feature matrices in all datasets. CMA gives remarkably high recognition rates only for 2DPCA- and 2DSVD-based feature matrices in the AR face dataset when compared with the SVM and CNN classifiers. However, SVM is more successful than CMA and CNN when considering the 2DFDA-based feature matrices and original images in AR and ORL face datasets and all feature matrices in ORL and Yale face datasets except 2DPCA based feature matrices obtained from Yale face dataset.

Further insights into the face classification could be emphasized by inspecting results that are obtained from the proposed shallow CNN model and AMS. When the average results of all three datasets in Table 1 are examined, for original samples, performance of AMS is superior to CNN. Of course, both Softmax and AMS provide good results in high dimensional data. It was observed that the accuracy of CNN with Softmax gives slightly better accuracy rates than ones obtained from 2D projections. It means that there is a negative tradeoff between reduced face representations and CNN performance. The key reason of this performance degradation can be explained that the CNN is not able to extract meaningful features once the face data is projected onto orthonormal subspace. The key reason of weakness of CNN can be attributed over-fitting or under-fitting cases reasoned by size of data or utilized hyper-parameters. To improve the performance of CNN, a crowded set of studies performed the fine-tuning idea on CNN features, which is widely preferred for boosting the CNN performance by dropping the last head of CNN (Softmax) and replacing with a new classifier and re-training the activated features that are returned from the last FC layer [54-56]. Nevertheless, the CNN presents a steady performance for all original face samples. By zooming the results in the rows of Table 1, one emphasizes that the RF model gives remarkable results in ORL dataset.

Comparing CNN and CMA, one may observe that CMA is dominant for 2DPCA, 2DSVD and 2DFDA feature matrices,

when the average performance scores given in Figure 6 is considered. Results showed that a coupling of CNN model with well-tuned parameters could achieve nice recognition scores that are 97.6% and 94.5% accuracy rates in case of experiments with original samples and 2DPCA based features in Yale dataset, respectively. Moreover, during tests on original samples, we have found that the overall discriminative capability of CNN model was about 90.1% when the results for all datasets and classifiers are taken into consideration in testing stage.

When inspecting the performance results shown in Figure 6, one can clearly say that the CMA and SVM are in a competition. AMS is superior to all classifiers for original samples. Among all classifiers, the K-NN is collapsed at all cases due to its short-sight classification methodology that works only with a distance based similarity estimation. When comparing the efficiency of 2DFDA, 2DPCA and 2DSVD features, one may note that the 2DFDA based features produce more valuable identification scores that resembles to those determined from original samples.

The 2D projection method based comparison of classifiers can also be made according to their respective training and testing processing time. The training and testing time per one image for only the AR face dataset are comparatively given in Table 2.

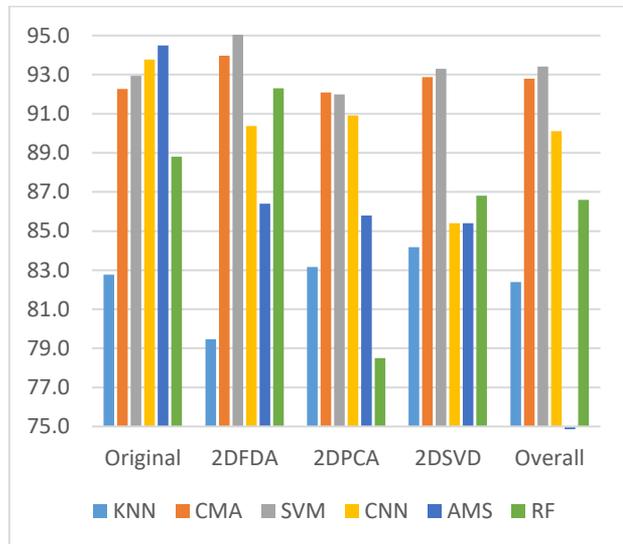


Figure 6. Performance comparison with average accuracy scores of all datasets for each approach

Table 1. The detailed average recognition rates (%) obtained from each approach over datasets

		Size	N	Classifiers					
				KNN	CMA	SVM	CNN	AMS	RF
Original	AR	100x85	50	83.9	97.4	97.6	95.4	95.9	91.3
	ORL	60x60	40	80.8	90.3	90.8	88.3	89.5	87.8
	Yale	152x126	15	83.6	89.1	90.4	97.6	98.2	87.3
2DFDA	AR	100x5	50	94.1	96.7	97.3	92.0	88.1	91.4
	ORL	60x45	40	64.3	91.3	91.9	85.8	78.3	93.4
	Yale	152x80	15	80.0	93.9	97.1	93.3	92.7	92.0
2DPCA	AR	100x12	50	86.6	97.4	90.1	92.0	87.7	65.0
	ORL	60x12	40	80.5	91.5	92.3	86.8	77.5	81.3
	Yale	152x9	15	86.1	85.5	90.1	94.5	91.5	75.5
2DSVD	AR	12x12	50	85.6	97.0	94.4	92.4	66.6	86.0
	ORL	14x12	40	80.8	92.5	93.2	82.0	69.5	93.3
	Yale	12x9	15	86.1	89.1	92.3	81.8	69.1	81.2

Table 2. Execution time per one image for the AR dataset

	Method Size	Time (sec)			
		K-NN	CMA	SVM	CNN
Training	2DPCA (100×12)	0.013	0.022	12.212	0.01
	2DSVD (12×12)	0.0014	0.019	12.221	0.007
	2DFDA (100×5)	0.020	0.026	13.464	0.01
	Original (100×85)	0.057	0.083	21.831	0.14
Testing	2DPCA (100×12)	0.005	0.141	0.090	0.03
	2DSVD (12×12)	0.003	0.093	0.032	0.03
	2DFDA (100×5)	0.004	0.107	0.077	0.03
	Original (100×85)	0.033	0.918	0.597	0.04

All experiments were employed in the MATLAB and Python Environment (for CNN). For all experiments, we have utilized a modest PC (Intel(R) Xeon(R) CPU E5-1620 v3 with 3.50 GHz CPU, 24 GB memory and 4 GB GPU). Due to batch-wise system evaluation, the training processing time of CNN is considerably lower than K-NN, CMA and SVM. The testing processing time of CNN are much less than that of CMA and SVM.

6. CONCLUSIONS

Automatic face recognition is important for many applications in computer vision and is a challenging area in the pattern classification. Even though many systems have been evolved for face recognition applications, face images still are not able to be quickly identified with high accuracy and low memory requirement under various lighting situations, and/or facial expressions so that new face recognition algorithms are required.

The main purpose of this paper is to present a new face recognition system which combines the popular classifiers including K-NN, CMA, CNN, SVM, AMS and RF classifiers with the feature matrices obtained by using 2DPCA, 2DSVD, and 2DFDA methods. 2DPCA pays more interest to the spatial relationship between pixels in the horizontal direction, and projects the image onto the unit column vector to calculate the covariance; 2DSVD takes into account the spatial relationships between pixels in both the horizontal and vertical directions, and projects the image in both the horizontal and vertical directions to calculate the covariance; 2DFDA calculates the inter-class and intra-class covariances of all training images as a support to decrease the intra-class distance and increase the inter-class distance.

In CMA, a common matrix, which is unique and includes the common or invariant features of that class, is calculated and it can be successfully utilized in image classification systems. Since CMA calculates and uses different class subspaces, it may also be applied rapidly. That is, only the common matrix of a class, whose training samples are changed (e.g. new face image/images insertion to that class), should be recalculated. This situation explicitly implies that it is not necessary to create subspaces and evaluate common matrices for other classes. In addition, linear SVM is very efficient in solving classification problems. It can model highly nonlinear

relationships in high dimensional spaces and is more intuitive than other machine learning methods. However, training stage of a traditional SVM algorithm should be repeated if a new face image is inserted. Although the CNN is a parameter-free approach for classification tasks, but it suffers to capture the meaningful information from projected data.

In almost all pattern classification systems, the working with original image pixel matrices is difficult since it significantly consumes memory and time. Therefore, feature matrices having relatively low dimension are more useful for many pattern classifiers. For this purpose, in this study, three types of subspace-based methods were applied to obtain feature matrices. Since the dimensions for these three types of feature matrices are very low compared to the original matrices, these feature matrices are very beneficial from the points of both processing time and memory necessity. Furthermore, these feature matrices can efficiently represent face images because the recognition rates obtained from the feature matrices and the original images slightly differ from each other. If one makes a comparison among the 2DPCA, 2DSVD, and 2DFDA methods, the 2DFDA method is more useful and valuable since slightly higher recognition scores were found with remarkably lower dimensional feature matrices.

In order to comprehensively compare the performances of classifiers together with the feature extraction methods, we have integrated three large data sets for face recognition, and compared the processing and classifier accuracy respectively. While increasing training, we have also analyzed a new influencing factor - pixel size, which revealed the advantages and disadvantages of these classifiers for data of different features.

Experimental studies point out that the CMA and SVM classifiers are superior to the CNN and K-NN classifiers for each type of feature matrices in almost all datasets. However, the CNN gives the highest recognition scores for 2DPCA features and original face samples of Yale dataset. CMA gives remarkably high recognition rates only for 2DPCA- and 2DSVD-based feature matrices in the AR face dataset when compared with the SVM classifier. However, SVM is more successful than CMA when the 2DFDA-based feature matrices and original images in AR face dataset and all feature matrices in ORL and Yale face datasets are used. Especially face classification system which combines the SVM classifier with the 2DFDA-based feature matrices seems to be the most plausible choice. In the future work, we will attempt to apply CMA classifier for face classification in videos by processing video frames as still images.

REFERENCES

- [1] De Vel, O., Aeberhard, S. (1999). Line-based face recognition under varying pose. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 21(10): 1081-1088. <https://doi.org/10.1109/34.799912>
- [2] Cevikalp, H., Neamtu, M., Wilkes, M., Barkana, A. (2005). Discriminative common vectors for face recognition. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 27(1): 4-13. <https://doi.org/10.1109/TPAMI.2005.9>
- [3] Cevikalp, H., Neamtu, M., Wilkes, M. (2006). Discriminative common vector method with kernels. *IEEE Transactions on Neural Networks*, 17(6): 1550-

1565. <https://doi.org/10.1109/TNN.2006.881485>
- [4] Lee, K.C., Ho, J., Kriegman, D.J. (2005). Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 27(5): 684-698. <https://doi.org/10.1109/TPAMI.2005.92>
- [5] Wang, X., Tang, X. (2004). A unified framework for subspace face recognition. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 26(9): 1222-1228. <https://doi.org/10.1109/TPAMI.2004.57>
- [6] Yang, J., Zhang, D.D., Frangi, A.F., Yang, J.Y. (2004). Two-dimensional PCA: A new approach to appearance-based face representation and recognition. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 26(1): 131-137. <https://doi.org/10.1109/TPAMI.2004.1261097>
- [7] Liu, W., Wang, Y., Li, S.Z., Tan, T. (2004). Null space-based kernel fisher discriminant analysis for face recognition. *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, Seoul, Korea (South)*, pp. 369-374. <https://doi.org/10.1109/AFGR.2004.1301558>
- [8] Yang, M.H. (2002). Kernel eigenfaces vs. Kernel fisherfaces: Face recognition using kernel methods. *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition, Washington, DC, USA*, pp. 215-220. <https://doi.org/10.1109/AFGR.2002.4527207>
- [9] Bartlett, M.S., Movellan, J.R., Sejnowski, T.J. (2002). Face recognition by independent component analysis. *IEEE Transactions on Neural Networks*, 13(6): 1450-1464. <https://doi.org/10.1109/TNN.2002.804287>
- [10] Kim, K.I., Franz, M.O., Scholkopf, B. (2005). Iterative kernel principal component analysis for image modeling. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 27(9): 1351-1366. <https://doi.org/10.1109/TPAMI.2005.181>
- [11] Wang, H. (2011). Structural two-dimensional principal component analysis for image recognition. *Machine Vision Applications*, 22: 433-438. <https://doi.org/10.1007/s00138-009-0229-4>
- [12] Zhu, Q., Xu, Y. (2013). Multi-directional two-dimensional PCA with matching score level fusion for face recognition. *Neural Computing Applications*, 23: 169-174. <https://doi.org/10.1007/s00521-012-0851-3>
- [13] Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 19(7): 711-720. <https://doi.org/10.1109/34.598228>
- [14] Chen, S., Liu, J., Zhou, Z.H. (2004). Making FLDA applicable to face recognition with one sample per person. *Pattern Recognition*, 37(7): 1553-1555. <https://doi.org/10.1016/j.patcog.2003.12.010>
- [15] Lu, J., Plataniotis, K.N., Venetsanopoulos, A.N. (2003). Regularized D-LDA for face recognition. *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)*, Hong Kong, China, pp. III-125. <https://doi.org/10.1109/ICASSP.2003.1199123>
- [16] Lu, J., Plataniotis, K.N., Venetsanopoulos, A.N. (2003). Face recognition using LDA-based algorithms. *IEEE Transactions on Neural Networks*, 14(1): 195-200. <https://doi.org/10.1109/TNN.2002.806647>
- [17] Yu, H., Yang, J. (2001). A direct LDA algorithm for high-dimensional data—with application to face recognition. *Pattern Recognition*, 34(10): 2067-2070. [https://doi.org/10.1016/S0031-3203\(00\)00162-X](https://doi.org/10.1016/S0031-3203(00)00162-X)
- [18] Park, S.K., Sim, D.G. (2011). New MCT-based face recognition under varying lighting conditions. *International Journal of Control, Automation Systems*, 9: 542-549. <https://doi.org/10.1007/s12555-011-0314-0>
- [19] Koc, M., Barkana, A. (2011). A new solution to one sample problem in face recognition using FLDA. *Applied Mathematics Computation*, 217(24): 10368-10376. <https://doi.org/10.1016/j.amc.2011.05.048>
- [20] Kong, H., Teoh, E.K., Wang, J.G., Venkateswarlu, R. (2005). Two-dimensional fisher discriminant analysis: forget about small sample size problem [face recognition applications]. *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005., Philadelphia, PA, USA*, pp. ii/761-ii/764. <https://doi.org/10.1109/ICASSP.2005.1415516>
- [21] He, J., Zhang, D. (2011). Face recognition using uniform eigen-space SVD on enhanced image for single training sample. *Journal of Computational Information Systems*, 7(5): 1655-1662.
- [22] Farfate, S.S., Saberian, M.J., Li, L.J. (2015). Multi-view face detection using deep convolutional neural networks. *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, ACM2015*, pp. 643-650. <https://doi.org/10.1145/2671188.2749408>
- [23] Zhang, M.M., Shang, K., Wu, H. (2019). Deep compact discriminative representation for unconstrained face recognition. *Journal Signal Processing: Image Communication*, 75: 118-127. <https://doi.org/10.1016/j.image.2019.03.015>
- [24] Leng, B., Liu, Y., Yu, K., Xu, S., Yuan, Z., Qin, J. (2016). Cascade shallow CNN structure for face verification and identification. *Neurocomputing*, 215: 232-240. <https://doi.org/10.1016/j.neucom.2015.08.134>
- [25] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278-2324. <https://doi.org/10.1109/5.726791>
- [26] Hu, G., Yang, Y., Yi, D., Kittler, J., Christmas, W., Li, S.Z., Hospedales, T. (2015). When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition. *2015 IEEE International Conference on Computer Vision Workshop (ICCVW), Santiago, Chile*, pp. 142-150. <https://doi.org/10.1109/ICCVW.2015.58>
- [27] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA*, pp. 770-778. <https://doi.org/10.1109/CVPR.2016.90>
- [28] Wu, S., Wang, D. (2019). Effect of subject's age and gender on face recognition results. *Journal of Visual Communication Image Representation*, 60: 116-122. <https://doi.org/10.1016/j.jvcir.2019.01.013>
- [29] Ding, C., Ye, J. (2005). 2-dimensional singular value decomposition for 2d maps and images. *Proceedings of the 2005 SIAM International Conference on Data Mining, SIAM2005*, pp. 32-43. <https://doi.org/10.1137/1.9781611972757.4>
- [30] Işık, Ş., Özkan, K. (2019). Common matrix approach-

- based multispectral image fusion and its application to edge detection. *Journal of Applied Remote Sensing*, 13: 016515. <https://doi.org/10.1117/1.JRS.13.016515>
- [31] Turhal, Ü.Ç., Gülmezoglu, M.B., Barkana, A. (2005). Face recognition using common matrix approach. 2005 13th European Signal Processing Conference, IEEE2005, pp. 1-4.
- [32] Gulmezoglu, M.B., Dzhafarov, V., Barkana, A. (2001). The common vector approach and its relation to principal component analysis. *IEEE Transactions on Speech Audio Processing*, 9(6): 655-662. <https://doi.org/10.1109/89.943343>
- [33] Gülmezoglu, M.B., Dzhafarov, V., Edizkan, R., Barkana, A. (2007). The common vector approach and its comparison with other subspace methods in case of sufficient data. *Journal of Computer Speech Language*, 21(2): 266-281. <https://doi.org/10.1016/j.csl.2006.06.002>
- [34] Gulmezoglu, M.B., Dzhafarov, V., Keskin, M., Barkana, A. (1999). A novel approach to isolated word recognition, *IEEE Transactions on Speech Audio Processing*, 7(6): 620-628. <https://doi.org/10.1109/89.799687>
- [35] Işık, Ş., Özkan, K., Gerek, Ö.N. (2019). CVABS: Moving object segmentation with common vector approach for videos. *IET Computer Vision*, 13(8): 719-729. <https://doi.org/10.1049/iet-cvi.2018.5642>
- [36] Özkan, K., Işık, Ş., Topsakal Yavuz, B. (2019). Identification of wheat kernels by fusion of RGB, SWIR, VNIR samples over feature and image domain. *Journal of the Science of Food Agriculture*, 99(11): 4977-4984. <https://doi.org/10.1002/jsfa.9732>
- [37] Özkan, K., Işık, Ş. (2015). A novel multi-scale and multi-expert edge detector based on common vector approach. *AEU-International Journal of Electronics Communications*, 69(9): 1272-1281. <https://doi.org/10.1016/j.aeue.2015.05.011>
- [38] Gülmezoglu, M.B., Ergin, S. (2007). An approach for bearing fault detection in electrical motors. *European Transactions on Electrical Power*, 17(6): 628-641. <https://doi.org/10.1002/etep.161>
- [39] Bredin, H., Dehak, N., Chollet, G. (2006). GMM-based SVM for face recognition. 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, pp. 1111-1114. <https://doi.org/10.1109/ICPR.2006.611>
- [40] Cevikalp, H., Yavuz, H.S., Edizkan, R., Gündüz, H., Kandemir, C.M. (2011). Comparisons of features for automatic eye and mouth localization. 2011 International Symposium on Innovations in Intelligent Systems and Applications, Istanbul, Turkey, pp. 576-580. <https://doi.org/10.1109/INISTA.2011.5946152>
- [41] Guo, G., Li, S.Z., Chan, K. (2000). Face recognition by support vector machines. *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, Grenoble, France, pp. 196-201. <https://doi.org/10.1109/AFGR.2000.840634>
- [42] Hotta, K. (2004). Support vector machine with local summation kernel for robust face recognition. *Proceedings of the 17th International Conference on Pattern Recognition*, 2004. ICPR 2004., Cambridge, UK, pp. 482-485. <https://doi.org/10.1109/ICPR.2004.1334571>
- [43] Vapnik, V. (2013). *The Nature of Statistical Learning Theory*. Springer Science & Business Media, New York. <https://doi.org/10.1007/978-1-4757-3264-1>
- [44] Gulmezoglu, M., Barkana, A. (1998). Text-dependent speaker recognition by using Gram-Schmidt orthogonalization method. *Applied Informatics-Proceedings-1998*, pp. 438-440.
- [45] Ruiz-Pinales, J., Acosta-Reyes, J., Salazar-Garibay, A., Jaime-Rivas, R. (2006). Shift invariant support vector machines face recognition system. *CiteSeerX.psu*:10.1.1.306.4449
- [46] Duin, R.P., Juszczak, P., Paclik, P., Pekalska, E., de Ridder, D., Tax, D.M. (2007). *Prtools4. 1, a matlab toolbox for pattern recognition*. Delft University of Technology.
- [47] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint*.
- [48] Martinez, A., Benavente, R. The AR Face Database, <http://www.cat.uab.cat/Public/Publications/1998/MaB1998/CVCReport24.pdf>, accessed on 2 March 2020.
- [49] ORL Face Dataset, <https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>, accessed on 2 March 2020.
- [50] Samaria, F.S., Harter, A.C. (1994). Parameterisation of a stochastic model for human face identification. *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, USA, pp. 138-142. <https://doi.org/10.1109/ACV.1994.341300>
- [51] OJE, E. (1983). *Subspace methods of pattern recognition*. Pattern recognition and image processing series. Research Studies Press.
- [52] Swets, D.L., Weng, J.J. (1996). Using discriminant eigenfeatures for image retrieval. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 18(8): 831-836. <https://doi.org/10.1109/34.531802>
- [53] Wang, F., Cheng, J., Liu, W., Liu, H. (2018). Additive margin softmax for face verification. *IEEE Signal Processing Letters*, 25(7): 926-930. <https://doi.org/10.1109/LSP.2018.2822810>
- [54] Ciocca, G., Napoletano, P., Schettini, R. (2018). CNN-based features for retrieval and classification of food images. *Journal Computer Vision Image Understanding*, 176-177: 70-77. <https://doi.org/10.1016/j.cviu.2018.09.001>
- [55] Liu, X., Wang, C., Bai, J., Liao, G., (2020). Fine-tuning pre-trained convolutional neural networks for gastric precancerous disease classification on magnification narrow-band imaging images. *Neurocomputing*, 392: 253-267. <https://doi.org/10.1016/j.neucom.2018.10.100>
- [56] Cetinic, E., Lipic, T., Grgic, S. (2018). Fine-tuning Convolutional Neural Networks for fine art classification. *Expert Systems with Applications*, 114: 107-118. <https://doi.org/10.1016/j.eswa.2018.07.026>