International Journal of Innovative Computing, Information and Control
Volume **9**, Number **5**, May **2013**

ICIC International ©2013 ISSN 1349-4198
pp. 1819–1834

A NOVEL FRAMEWORK FOR PARTITION-BASED FACE RECOGNITION

Semih Ergin and M. Bilginer Gulmezoglu

Electrical and Electronics Engineering Department
Eskisehir Osmangazi University
Meselik Campus, Eskisehir 26480, Turkey
{ sergin; bgulmez }@ogu.edu.tr

Received February 2012; revised July 2012

Abstract. *The main purpose of this paper is to propose a novel framework based on the separate classification of each partition of a face using the Common Vector Approach (CVA) and the combination of the classification results with the Majority Voting technique to classify test images. First of all, the thirteen different partitions (forehead, two eyebrows, two eyes, two cheeks, nose, mouth, chap and three spaces between some partitions) that effectively represent a face are extracted. When the partition vectors are applied to our proposed framework, higher recognition rates are obtained when compared with those obtained by the PCA, D-LDA and CVA methods for the image vectors obtained by stacking the partition vectors. Although the SVM classifier is superior to all other classifiers in terms of recognition performance, our proposed framework is an appropriate choice when the recognition results, processing time and memory requirement issues are all taken into account.*
Keywords: Face recognition, Face partitioning, Geometric descriptors, Common vector approach, Linear discriminant analysis, Support vector machine

1. **Introduction.** Over the last two decades, face recognition technology has become one of the most important technologies due to its applications, such as use in personal identification, security access control, surveillance, telecommunications, digital libraries, human-computer interaction and in the military. Face recognition can be described as the identification process of a given input facial image from a stored face database.

One of the major difficulties in analyzing and recognizing human faces is the large sample size problem of face images. Extraction of the most representative features from facial images significantly improves the performance of any face recognition system. Therefore, feature extraction is a crucial step for face recognition. In geometrical feature extraction, geometric descriptors of biometric facial components, such as the eye, eyebrow, eyebrow thickness, forehead, mouth, nose, chin, cheek and the zygomatic region, are used [1]. Hess and Martinez [2] extracted all the corners and borders of facial features using the Smallest Univalue Segment Assimilating Nucleus (SUSAN) algorithm for a model-based videophone coding system. Samra et al. [3] attempted to locate the major facial components or feature points in an image. The Local Binary Pattern (LBP) that learns the most discriminative LBP-like features for each facial region in a supervised manner has recently been introduced [4]. In contrast to pixel-based studies, the gradient image has been partitioned on a grid in polar coordinates for probabilistic learning to recognize faces [5]. In their study, the partitions include seven radial sectors and a central region whose radius is chosen to make the areas of all partitions equal. A fast facial feature extraction using a deformable shape model with Haar-wavelet based local texture attributes has also been proposed [6]. De Vel and Aeberhard [1] have used a set of random rectilinear line segments of 2D facial images as the underlying image representation. The line segments are less sensitive to variations in the face and background because they combine the local properties with spatial information of a facial image. The geometrical method of feature extraction from human ear images to perform human identification is presented in [7,8]. The positions of the speaker's face and mouth are reliably localized in videophone sequences by an effective approach based on amplitude projections in the regions of the mouth [9]. Shih and Chuang [10] presented a novel approach called the double-threshold method for the extractions of the human head, face and facial features. In this method, the high-thresholded image is used to trace head boundary, and the low-thresholded image is used to scan face boundary. The concentric circular Fourier-Zernike descriptors (CCFZD) were recently utilized as a new method for face image retrieval [11]. In their paper, the local invariant Zernike moments are computed for each concentric circular subimage to obtain the Fourier-Zernike descriptors. A variety of other approaches are also proposed to extract facial features from images [12-14].

In addition to geometrical approaches, several other analytical methods are also used for the purpose of feature extraction from images. An efficient and stable feature extractor based on the Maximum Margin Criterion (MMC) has been established [15]. Wang et al. [16] newly introduced an Adaptively-weighted Sub-pattern Locality Preserving Projection (Aw-SpLPP) algorithm for face recognition. Bayesian Network [17], localized PCA [18], real-time video object segmentation [19], and LDA [20] are also used for feature extraction. The feature extraction problem has been presented as a multidimensional optimization problem and is overcome by deformable graphs and dynamic programming in [21]. Two-stage 2DPCA and two-stage 2DLDA are also proposed as image feature extraction methods to overcome the disadvantages of 2DPCA and 2DLDA [22]. The Protrusive Graph method is combined with both iterative clustering and smoothing refinement to interactively partition 3D models according to any user-specified partitioning requirements [23].

Subspace methods are widely used among the existing face recognition techniques. In these methods, the variations between images under changeable illumination conditions and/or different facial expressions can be successfully modeled by low-dimensional linear spaces, even when there are many light sources and shadows. Subspace analysis methods, such as PCA, LDA and DCV, have been extensively preferred in signal processing and computer vision problems as efficient methods for both reducing dimension and determining the direction of the projection with certain properties [24-28]. Other subspace methods, such as Independent Component Analysis (ICA), nonlinear PCA (NLPCA) and Kernel PCA (KPCA), are all modifications or extensions of PCA used to address higher-order statistical dependencies [27]. In addition to these methods, LDA-based approaches, namely the Direct LDA (D-LDA), Fisher's LDA (FLDA) and GLU-LDA, have been presented [29-32].

The following also describes some of the popular approaches used in face recognition. Support Vector Machine (SVM) is combined with wavelet statistics [33] and component-based features [34,35] to recognize faces. A feature-based biometric face recognition system together with some initial experiments is also implemented [36]. In this system, informative features in facial images are located by Gabor filters, which provide an automatic system that does not accurately detect facial features. Martinez [37] proposed an approach that uses Hidden Markov Models (HMMs) to identify and index facial images from a database of static and dynamic (video) images. As an interesting application, a cost-effective bi-directional people counter for the pedestrian flow passing through a gate or door is designed based on area estimation and color analysis [38]. Although many algorithms have been developed to classify facial images, they still cannot effectively identify facial images under different illumination conditions and/or the facial expressions. Thus, new combinations of face recognition methods and feature extraction techniques are necessary.

We propose a procedure having the following steps. First, the facial portions in the images of individuals are cropped and resized because regions other than the facial portions have unnecessary information. Second, the thirteen different partitions that represent a face more effectively than the whole face are automatically extracted; these partitions are the forehead, left eyebrow, right eyebrow, left eye, right eye, nose, left cheek, mouth, right cheek, chap, the area between the eyes and eyebrows, the area between the left cheek and nose, and the area between the right cheek and nose. The feature vectors of images of each individual are then constructed by stacking partition vectors. Third, these feature vectors and the vectors obtained from the whole face images are separately applied to the well-known methods, the Common Vector Approach (CVA), PCA, D-LDA and SVM, for face recognition. Finally, in the proposed framework, each partition is separately assigned to an individual by CVA, and the classification results are combined with the Majority Voting technique to recognize the test image. During the training phase of CVA, the difference subspace and the common vector for each partition are obtained using the corresponding partitions of poses in the training set of each individual. During the test phase of CVA, first, the remaining vector for each partition of a test image is calculated. Then, the distance between the remaining vector and common vector of that partition is used for the classification of the test partition. According to the Majority Voting technique, test image is assigned to an individual class, which is mostly addressed by the classification results of the partitions. All of the recognition rates are obtained by carrying out the "leave-one-out" procedure. In the experimental studies, the AR Face Database is used [39].

The remainder of this paper is organized as follows. In the second section, the feature extraction process is broadly explained. Following the explanations of the PCA, D-LDA, SVM and CVA methods in the third section, the experimental study, discussions and conclusions are given in the fourth, fifth and sixth sections, respectively.

2. Feature Extraction.

2.1. Face localization.

The images in the AR Face Database are bust shots of individuals. The original sizes of the images are 576×768, and the images are represented by 24-bit in depth. These images include not only the facial portions of individuals but also some blank spaces allocated in the left, right, and top parts of the images. One of the original images in the database is shown in Figure 1.

Several pre-processing operations were performed on all of the images. Initially, all of the face images were converted into gray-scale. Then, the gray-scaled images were then

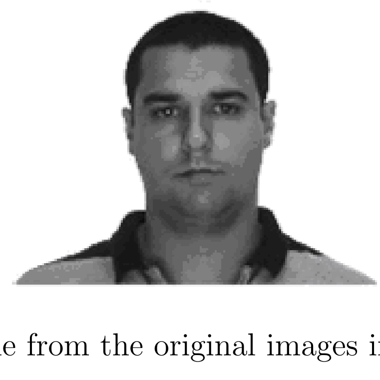

Figure 1. An example from the original images in the AR face database

Figure 2. The cropped facial portion of the original image

resampled to size 143×191 and normalized to a "0(black)-1(white)" range by dividing all pixel values with 255. Next, only the facial portions were cropped by applying a MATLAB program (facefind.m) [40] to the original images in the database. Because the sizes of the cropped facial portions can differ between images, all facial portions are resized to the size of 65×65. The cropped facial portion of the original image shown in Figure 1 is shown in Figure 2. Finally, all of the image matrices were converted into matrices with zero-mean by subtracting the average pixel value from each pixel and with unit-variance by dividing the image matrices with their standard deviations.

2.2. **Face partitioning.** One of the most important steps in the partitioning process is to determine the center of the face. Once the center is correctly determined, the extraction of facial partitions from that image becomes easier. In this study, the tip of the nose is taken as the center of an individual's face and is easily found because its pixel value is closest to 1 among all of the pixels. After a pixel hatching process is applied on the

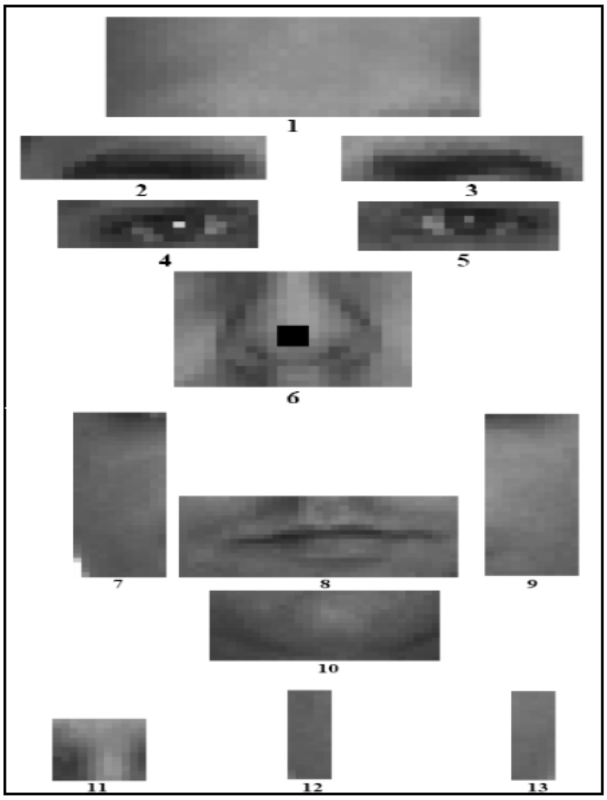

Figure 3. The thirteen partitions extracted from the cropped facial portion of the original image. (Note: The sizes of all partitions are intentionally magnified to clarify the appearance of the partitions.)

TABLE 1. The dimensions and names of the facial partitions

Partition No.	*Location of the Partition*	*Dimension*	*Dimension*
1	*Forehead*	15×37	15×37
2	*Left eyebrow*	7×27	7×23
3	*Right eyebrow*	7×27	7×23
4	*Left eye*	7×25	7×19
5	*Right eye*	7×25	7×19
6	*Nose*	17×23	17×23
7	*Left cheek*	33×11	33×11
8	*Mouth*	17×33	17×33
9	*Right cheek*	33×11	33×11
10	*Chap*	15×27	15×27
11	*The area between the two eyes and the two eyebrows*	*None*	13×11
12	*The area between the left cheek and the nose*	19×5	19×5
13	*The area between the right cheek and the nose*	19×5	19×5

cropped facial portions, the tips of the noses on all of the face images are determined. Initially, the distances between the nose tip and the centers of the thirteen partitions are measured for all of the images in the training set, and the average of these distances for each partition is calculated. Then, the centers of the thirteen partitions for each image are determined with respect to these average distances. The sizes of these partitions are also measured for all of the images in the training set and the average size for each partition is calculated. Furthermore, these thirteen partitions are automatically extracted from the cropped facial portions using the centers and the average sizes of partitions. These extracted partitions are explicitly shown in Figure 3. The thirteen partitions are: the forehead, two eyebrows (right and left), two eyes (right and left), two cheeks (right and left), the nose, the mouth, the chap, and three spaces between some of the partitions. The name and dimension of each partition are listed in Table 1. Whenever the partition numbered as 11 in Figure 3 is incorporated into both the eyes and eyebrows, the number of partitions can be reduced to 12. In this case, the dimensions of the eyes and eyebrows change, but those of other partitions remain the same (see Table 1).

3. PCA, D-LDA, SVM and CVA Methods.

In this paper, four methods (PCA, D-LDA, SVM and CVA) are used for the face recognition. Because these methods are well known and frequently used, they are briefly mentioned in this section to review the basic concepts.

3.1. Principal component analysis (PCA).

The goal of the PCA is to map the vectors $\mathbf{a_i}$ $(i = 1, 2, \ldots, M)$ in a p-dimensional space into another set of vectors in a k-dimensional subspace, where $k < p$ [41]. The covariance matrix of the vectors belonging to all classes is defined as

$$\mathbf{\Phi} = \sum_{i=1}^{M} \left[(\mathbf{a_i} - \mathbf{a_{ave}}) (\mathbf{a_i} - \mathbf{a_{ave}})^T \right] \tag{1}$$

where $\mathbf{a_{ave}}$ is the average of the vectors in all classes. Since the sizes of the facial images are so large, the size of this covariance matrix is large, and an eigenvalue-eigenvector decomposition is difficult to perform. In order to overcome this problem, Turk and Pentland [42] have proposed an eigenface-trick in their study. They first subtracted $\mathbf{a_{ave}}$ from each vector $\mathbf{a_i}$ $(i = 1, 2, \ldots, M)$:

$$\mathbf{d_i} = \mathbf{a_i} - \mathbf{a_{ave}} \quad \text{for} \quad i = 1, 2, \ldots, M. \tag{2}$$

Then, a matrix $\mathbf{A}$ is constructed using the vectors $\mathbf{d_i}$ $(i = 1, 2, \ldots, M)$ as

$$\boldsymbol{A} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ \mathbf{d_1} & \mathbf{d_2} & \cdots & \mathbf{d_M} \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}_{p \times M} . \tag{3}$$

A new covariance matrix is evaluated by utilizing the matrix $\mathbf{A}$:

$$\mathbf{\Phi_{new}} = \mathbf{A}^T \mathbf{A}. \tag{4}$$

This new covariance matrix has a size of $(M \times M)$. An eigenvalue-eigenvector decomposition is performed on this new covariance matrix ($\mathbf{\Phi_{new}}$). Since the information in the directions of the eigenvectors corresponding to k largest eigenvalues has more representative power, these eigenvectors are selected. Thus, p-dimensional vectors can be converted into k-dimensional feature vectors, and these vectors are then used in the classification.

3.2. **Direct-linear discriminant analysis (D-LDA).** In the D-LDA, the main goal is to determine a matrix that simultaneously diagonalizes both the total within-class scatter matrix ($\mathbf{S_w}$) and the between-class scatter matrix ($\mathbf{S_b}$) [32]. The total within-class scatter matrix is defined as:

$$\mathbf{S_w} = \sum_{C=1}^{S} \left\{ \sum_{i=1}^{m} \left[\left(\mathbf{a_i^C} - \mathbf{a_{ave}^C} \right)^T \left(\mathbf{a_i^C} - \mathbf{a_{ave}^C} \right) \right] \right\} \tag{5}$$

where S is the number of classes, m is the number of images in C^{th} class, $\mathbf{a_i^C}$ is the i^{th} vector in the training set of C^{th} class and $\mathbf{a_{ave}^C}$ is the average vector of C^{th} class. The between-class scatter matrix is defined as:

$$\mathbf{S_b} = \sum_{C=1}^{S} \left[m \left(\mathbf{a_{ave}^C} - \mathbf{a_{ave}} \right)^T \left(\mathbf{a_{ave}^C} - \mathbf{a_{ave}} \right) \right] \tag{6}$$

where $\mathbf{a_{ave}}$ is the average of the vectors in all classes. First, an eigenvalue-eigenvector decomposition is performed on $\mathbf{S_b}$. Some of the eigenvalues of $\mathbf{S_b}$ are zero (or close to zero) and are discarded. Then, the following matrix ($\mathbf{Z}$) can be written as:

$$\mathbf{Z} = \mathbf{Y} \left(\mathbf{D_b} \right)^{1/2} \tag{7}$$

where $\mathbf{D_b}$ is the diagonal matrix whose diagonal elements are the nonzero eigenvalues of $\mathbf{S_b}$, and $\mathbf{Y}$ is a matrix whose columns are the eigenvectors corresponding to the nonzero eigenvalues of $\mathbf{S_b}$. Another eigenvalue-eigenvector decomposition is performed on ($\mathbf{Z^T S_w Z}$), and the largest eigenvalues are discarded. It is important to keep the eigenvectors corresponding to the smallest eigenvalues, especially those that are zero. The final transformation matrix is:

$$\mathbf{T} = \left(\mathbf{D_w} \right)^{(-1/2)} \mathbf{U^T Z^T} \tag{8}$$

where $\mathbf{D_w}$ is the diagonal matrix whose diagonal elements are the smallest eigenvalues of $\mathbf{Z^T S_w Z}$, and $\mathbf{U}$ is a matrix whose columns are the eigenvectors corresponding to the smallest eigenvalues of $\mathbf{Z^T S_w Z}$. Using the transformation matrix in Equation (8), the $\mathbf{a_i^C}$ vectors are mapped into the lower dimensional feature vectors and then used in the classification.

3.3. **Support vector machine (SVM).** SVM is originally proposed as a binary classifier and determines the optimal hyperplane that maximizes the distance between the optimal hyperplane and the nearest sample to this hyperplane [43]. For this reason, SVM is also known as the maximum margin classifier. Support vectors correspond to the data samples that are closest to the optimal hyperplane [44,45]. If a two-class problem is investigated, the training set is denoted as $\mathrm{TS} = \{(\mathbf{x_1}, L_1), (\mathbf{x_2}, L_2), \ldots, (\mathbf{x_M}, L_M)\}$. In this set, $\mathbf{x_i}$ $(i = 1, 2, \ldots, M)$ is the data sample, and L_i $(L_i \in \{-1, 1\})$ is the class label (either negative or positive class). Any vectorial test data $(\mathbf{x_{test}})$ can be classified using the following decision function:

$$f(\mathbf{x_{test}}) = \sum_{i=1}^{M} \left\{ \alpha_i L_i \left(\mathbf{x}_i^{\mathbf{T}} \mathbf{x_{test}} \right) + b \right\} \tag{9}$$

where α_i $(i = 1, 2, \ldots, M)$ are the nonzero coefficients that are the solution of the quadratic programming problem, $(|b|/\|\mathbf{w}\|)$ is the perpendicular distance from the optimal hyperplane to the origin, and $\mathbf{w}$ is the normal vector of the hyperplane. The sign of this decision function determines the label of the class to which the test data $(\mathbf{x_{test}})$ is assigned, and this decision function assumes the linearly separable case. The SVM classifier explained above is a two-class classifier. For dealing with multi-class problems (with S classes), it is possible to construct "$S\,(S-1)/2$" classifiers [46]. In this paper, the linear SVM classifier is preferred and is implemented using the Pattern Recognition Toolbox (PRTools) [47].

3.4. **Common vector approach (CVA).** CVA is a subspace-based pattern recognition method that has been used for speech recognition [48,49], speaker recognition [50], image recognition [24] and motor fault diagnosis [51] problems. In the implementation of CVA, a common vector for each class is calculated using feature vectors in the training set of that class. This common vector represents the common properties or invariant features of that class [48,49]. CVA can also be successfully applied to other types of data patterns that are represented in vector notation.

Let the vectors $\mathbf{a_1^C}, \mathbf{a_2^C}, \ldots, \mathbf{a}_m^{\mathbf{C}} \in \mathbf{R^p}$ be the feature vectors for an image-class C $(C = 1, 2, \ldots, S)$ in the training set, where $m \leq p$. Then, each of these feature vectors, which are assumed to be linearly independent, can be written as [48]:

$$\mathbf{a_i^C} = \mathbf{a_{i,dif}^C} + \mathbf{a_{com}^C} \quad \text{for} \quad i = 1, 2, \ldots, m \tag{10}$$

where the vector $\mathbf{a_{i,dif}^C}$ indicates the differences of class C resulting from facial expressions or illumination conditions, and the vector $\mathbf{a_{com}^C}$ is the common vector of image-class C. In order to find a unique solution for Equation (10), let us first define the difference vectors $\mathbf{b_j^C}$:

$$\mathbf{b_j^C} = \mathbf{a_{j+1}^C} - \mathbf{a_1^C} \quad \text{for} \quad j = 1, 2, \ldots, m-1 \tag{11}$$

where $\mathbf{a_1^C}$ is the subtrahend vector. The orthonormal basis vector set $\left\{\mathbf{z_1^C}, \ldots, \mathbf{z_{m-1}^C}\right\}$ of the difference subspace $\mathbf{B^C}$ is obtained from the difference vectors $(\mathbf{b_j^C})$ using the Gram-Schmidt orthogonalization method [48]. The projection of any vector $\mathbf{a_i^C}$ onto the difference subspace yields $\mathbf{a_{i,dif}^C}$:

$$\mathbf{a_{i,dif}^C} = \left\langle \mathbf{a_i^C}, \mathbf{z_1^C} \right\rangle \mathbf{z_1^C} + \left\langle \mathbf{a_i^C}, \mathbf{z_2^C} \right\rangle \mathbf{z_2^C} + \cdots + \left\langle \mathbf{a_i^C}, \mathbf{z_{m-1}^C} \right\rangle \mathbf{z_{m-1}^C} \tag{12}$$

where $\langle \mathbf{a}, \mathbf{z} \rangle$ indicates the dot product of $\mathbf{a}$ and $\mathbf{z}$. A unique solution for $\mathbf{a_{com}^C}$ can be obtained as [48]:

$$\mathbf{a_{com}^C} = \mathbf{a_i^C} - \mathbf{a_{i,dif}^C} \quad \text{for} \quad i = 1, 2, \ldots, m \tag{13}$$

This vector is unique, i.e., independent from the subtrahend vector and the index "i" for each class.

4. **Experimental Study.**

4.1. **Database.** The AR Face Database was used in the experimental study. This database includes the RGB (Red-Green-Blue)-colored face poses of 126 different individuals: 70 of whom are men, and the remaining 56 are women. The face images have different facial expressions, illumination conditions, or occasional occlusions. All of the poses were captured in two different sessions with a two-week time period between each session. Thirteen different poses in the first session and 13 different poses in the second session were saved for each individual. The resulting 26 different poses were recorded as pixel matrices with dimensions of 576×768 in the database. Fourteen poses with different facial expressions and illumination conditions were selected for the 30 male and 20 female subjects randomly chosen for the experiments. The total number of faces used in the experiments is equal to 700 (50 individuals $\times$ 14 poses per individual). After all of the preprocessing operations, explicitly explained in subsection 2.1, were applied on all of the poses, the facial portions of the individuals were resized to 65×65. In this study, five different types of vectors are used. Among them, the vectors (4225×1) obtained by vectorizing whole faces (65×65), and the vectors (361×1) obtained by vectorizing the resized whole faces (19×19) are two different types of vectors. The feature vectors constructed by the stacking of vectors corresponding to the twelve and thirteen partitions of the face constitute the third and the fourth types of vectors, which have dimensions of 3556×1 and 3581×1, respectively. The last feature vector type includes the partition vectors corresponding to the partitions of the images whose names and dimensions are comprehensively listed in Table 1. All of the experiments were conducted in MATLAB.

4.2. **Studies using PCA classifier.** In this part, the classes are recognized with PCA for the first four different types of feature vectors. Because each individual does not have a sufficient number of poses to perform a hold-out scheme (train/test separation), the "leave-one-out" procedure is followed in the experimental studies of PCA. Thirteen of the 14 images belonging to each individual are used for the training phase, and the remaining image in each "leave-one-out" step is used for the testing phase. Since 50 individuals are used in the experimental study, a total of 650 poses are used in the training phase, and the remaining 50 poses are used in the test phase for each step of the "leave-one-out" procedure. This process is repeated 14 times, leaving all of the images out once for each individual. In each step of the "leave-one-out" procedure, a covariance matrix with a size of 650×650 is obtained. Then, an eigenvalue-eigenvector decomposition is performed on the covariance matrix, and 650 eigenvalues (λ_i, $i = 1, 2, \ldots, 650$) and 650 eigenvectors are obtained. After these 650 eigenvalues are decreasingly sorted, the k largest eigenvalues are selected. We can choose k such that the sum of the largest eigenvalues is less than some fixed percentage L of the entire set [52]. Thus, we set k to fulfill

$$\frac{\left(\sum\limits_{i=1}^{k} \lambda_i\right)}{\left(\sum\limits_{i=1}^{650} \lambda_i\right)} < L. \tag{14}$$

If $L = 90$ %, good performance is obtained while retaining a large proportion of the variance presented in the original space [53]. $L = 90$ % yields the different number of eigenvalues for each step of the leave-one-out procedure. The average number of these eigenvalues is equal to 18, 10, 13 and 12 for the four different types of input vectors (whole

TABLE 2. The average recognition rates (%) obtained using PCA

Leave-one-out procedure step no.	*Whole faces with the size of* (4225 × 1)	*Whole faces with the size of* (361 × 1)	*Faces with twelve partitions* (3556 × 1)	*Faces with thirteen partitions* (3581 × 1)
1	62	62	82	84
2	56	52	84	78
3	78	74	84	88
4	60	60	70	74
5	30	26	50	54
6	42	42	72	66
7	40	38	70	68
8	68	68	92	92
9	76	66	90	88
10	74	66	90	94
11	56	50	66	70
12	28	24	48	48
13	40	38	60	62
14	34	34	72	66
Average	**53.1**	**50.0**	**73.6**	**73.7**

faces with the size of 4225×1, whole faces with the size of 361×1, twelve partitioned faces with the size of 3556×1, and thirteen partitioned faces with the size of 3581×1), respectively. The value of k is also determined from the point where the eigenvalues of the training data start to slowly vary. This point approximately corresponds to the number of eigenvalues determined by the fixed percentage of L in Equation (14). The feature vectors are mapped into lower dimensional spaces using the eigenvectors corresponding to the k largest eigenvalues and then classified. The classification process is performed using the Nearest-Neighbor Classifier (NNC), and the average recognition rates of all of the individual classes for the four different types of feature vectors are listed in Table 2.

4.3. **Studies using D-LDA classifier.** In this section, the classes are recognized with the D-LDA classifier for the first four different types of feature vectors. During the training phase of D-LDA, first, the between-class scatter matrix is calculated. After an eigenvalue-eigenvector decomposition is performed on the between-class scatter matrix, 4225, 361, 3556, and 3581 eigenvalues and the corresponding eigenvectors are separately obtained for the four different types of feature vectors. Forty-nine eigenvalues are nonzero, and the remaining eigenvalues are zero for each type of feature vector. The eigenvectors corresponding to nonzero eigenvalues are then selected, and thus, the null space of the between class scatter matrix is discarded. The "leave-one-out" procedure is also followed in the experimental studies of D-LDA. The recognition process is performed using the Nearest-Neighbor Classifier, and the average recognition rates of all individual classes for the four different types of feature vectors are listed in Table 3.

4.4. **Studies using SVM classifier.** The first four different types of feature vectors are also classified with the linear SVM classifier, which is implemented using the Pattern Recognition Toolbox in MATLAB. The average recognition rates of all individual classes are listed in Table 4. While obtaining the recognition rates, the "leave-one-out" procedure is also followed.

TABLE 3. The average recognition rates (%) obtained using D-LDA

Leave-one-out procedure step no.	*Whole faces with the size of* (4225 × 1)	*Whole faces with the size of* (361 × 1)	*Faces with twelve partitions* (3556 × 1)	*Faces with thirteen partitions* (3581 × 1)
1	76	74	96	98
2	58	70	80	84
3	66	68	88	88
4	48	66	74	76
5	78	82	92	92
6	70	78	96	86
7	70	70	90	88
8	70	82	98	98
9	74	74	90	86
10	64	68	88	90
11	38	48	70	76
12	72	78	92	94
13	82	78	96	92
14	80	84	94	94
Average	**67.6**	**72.9**	**88.9**	**88.7**

TABLE 4. The average recognition rates (%) obtained using SVM

Leave-one-out procedure step no.	*Whole faces with the size of* (4225 × 1)	*Whole faces with the size of* (361 × 1)	*Faces with twelve partitions* (3556 × 1)	*Faces with thirteen partitions* (3581 × 1)
1	92	86	98	100
2	78	80	96	94
3	84	84	98	94
4	78	70	94	90
5	94	90	98	96
6	80	78	96	94
7	84	86	94	92
8	92	90	100	100
9	96	92	94	100
10	94	90	96	100
11	72	64	88	92
12	92	82	100	96
13	88	86	100	100
14	90	86	98	98
Average	**86.7**	**83.1**	**96.4**	**96.1**

4.5. **Studies using CVA classifier.** In this section, those classes having any of the four different types of feature vectors are classified with the CVA classifier. Similar to the other classifiers in this paper, the CVA is also performed with the "leave-one-out" procedure. During the training phase of CVA, the difference subspace and the common vector are obtained for each class. During the test phase of CVA, a test image vector, $\mathbf{a_{test}}$, is first projected onto the difference subspace of class C. Then, the resulting vector

TABLE 5. The average recognition rates (%) obtained using CVA

Leave-one-out procedure step no.	*Whole faces with the size of* (4225 × 1)	*Whole faces with the size of* (361 × 1)	*Faces with twelve partitions* (3556 × 1)	*Faces with thirteen partitions* (3581 × 1)
1	78	78	94	100
2	72	68	94	96
3	86	86	94	94
4	66	66	88	92
5	74	70	92	90
6	62	62	94	96
7	66	62	90	90
8	84	82	96	96
9	84	84	96	98
10	84	74	96	96
11	62	60	86	86
12	70	72	92	88
13	74	70	96	96
14	80	76	98	98
Average	**74.4**	**72.1**	**93.3**	**94.0**

is subtracted from the test vector $\mathbf{a_{test}}$, and the remaining vector $\mathbf{a^C_{rem}}$ is obtained:

$$\mathbf{a^C_{rem}} = \mathbf{a_{test}} - \left[\left\langle \mathbf{a_{test}}, \mathbf{z^C_1} \right\rangle \mathbf{z^C_1} + \left\langle \mathbf{a_{test}}, \mathbf{z^C_2} \right\rangle \mathbf{z^C_2} + \cdots + \left\langle \mathbf{a_{test}}, \mathbf{z^C_{m-1}} \right\rangle \mathbf{z^C_{m-1}}\right]. \tag{15}$$

The decision criterion used for the classification purpose is given as

$$K = \underset{1 \le C \le S}{\arg\min} \left\{ \left\| \mathbf{a^C_{rem}} - \mathbf{a^C_{com}} \right\|^2 \right\} \tag{16}$$

where K is the index of the class for which the distance $\left\| \mathbf{a^C_{rem}} - \mathbf{a^C_{com}} \right\|^2$ yields the minimum value for $\mathbf{a_{test}}$. Therefore, the test image vector, $\mathbf{a_{test}}$, is assigned to that class. The average recognition rates of all individual classes are listed in Table 5.

As a new approach for CVA classification, each partition of a test image is separately assigned to an individual. During the training phase of this partition-based study, the difference subspace and the common vector are determined for each partition using same partitions in the 13 poses of each individual. During the testing process, the remaining vector for each partition of a test image is calculated and compared with the common vector of that partition. After each partition is assigned to an individual by CVA, the classification results of the partitions are combined with the Majority Voting technique. The results of this technique are accepted as the final recognition, and the recognition rates are listed in Table 6, and again, the "leave-one-out" procedure is followed. This study was realized for two differently partitioned (twelve or thirteen) faces. Therefore, a new approach is implemented for the face recognition by separately classifying each partition of a face.

5. **Discussion.** One of the greatest difficulties in automatic face recognition systems is the large sample size of the facial images. In order to construct lower-sized feature vectors that can effectively represent the facial images, unnecessary information in the facial images must be eliminated. For this purpose, thirteen different partitions, including discriminative features of a face, were cropped from the facial images. Following cropping, the raw data matrices belonging to these partitions were converted into column vectors

Table 6. The average recognition rates (%) obtained using CVA with the majority voting technique

Leave-one-out procedure step no.	*Faces with twelve partitions*	*Faces with thirteen partitions*
1	96	96
2	98	98
3	94	92
4	92	94
5	92	94
6	94	92
7	92	94
8	100	100
9	96	96
10	94	94
11	90	90
12	92	92
13	98	98
14	98	98
Average	**94.7**	**94.9**

and stacked to construct the feature vectors (3581×1). This process corresponds precisely to a 15.2% decrease in the dimension of the vectors belonging to a whole facial image. Meanwhile, these thirteen partitions are reduced to twelve partitions by adding the area between the two eyes and the two eyebrows (the 11^{th} partition) into both of the partitions for the eyes and the eyebrows; the other partitions remained the same. A 15.8% decrease in the dimension of the vectors belonging to the whole facial images was achieved, and the feature vectors with the size of 3556×1 were obtained. These feature vectors and the original facial images were applied to four well-known classifiers, PCA, D-LDA, SVM, and CVA. When the recognition rates obtained for the four classifiers are considered, they are ordered as SVM, CVA, D-LDA and PCA, with SVM as the best and PCA as the worst. Our results are superior when compared with the results of two closest studies in the literature [34,35].

In order to further increase the recognition rates determined using CVA, a new partition-based study is proposed. For this purpose, each partition is assigned to an individual separately, and the classification results are combined with the Majority Voting technique to classify the test image. As clearly observed in Table 6, the recognition performance of this partition-based study is slightly higher than those of the studies in which CVA was directly applied to the feature vectors obtained by the stacking of the partition vectors of the images.

The training and testing time per image for the eighteen different situations are comparatively provided in Table 7 and Table 8, respectively. These time values were calculated on a personal computer whose processor and memory are Pentium 3.0 GHz and 1 GB, respectively. When the testing time is the criterion for evaluation, the classifiers can be sorted as PCA, D-LDA, CVA, and SVM from the shortest to the longest time. Furthermore, a ranking on the classifiers' memory requirements is provided in Table 9. This ranking is also helpful to identify which case is useful to implement in terms of memory requirements. In Table 9, while the '1' indicates the case that requires the least memory, the '18' implies the case that requires the most memory. The classifiers can also be ordered

TABLE 7. Training time per image in eighteen cases (sec)

Classifiers	*Whole faces with the size of* 4225×1	*Whole faces with the size of* 361×1	*Faces with twelve partitions*	*Faces with thirteen partitions*
PCA	0.011	0.005	0.010	0.010
D-LDA	1.350	0.355	1.276	1.287
SVM	177.415	109.342	160.013	164.528
CVA	1.180	0.421	1.150	1.160
CVA with Majority Vote	*Not Applicable*	*Not Applicable*	3.540	3.731

TABLE 8. Testing time per image in eighteen cases (sec)

Classifiers	*Whole faces with the size of* 4225×1	*Whole faces with the size of* 361×1	*Faces with twelve partitions*	*Faces with thirteen partitions*
PCA	0.004	0.002	0.004	0.004
D-LDA	0.007	0.003	0.006	0.006
SVM	0.762	0.257	0.714	0.736
CVA	0.561	0.194	0.509	0.523
CVA with Majority Vote	*Not Applicable*	*Not Applicable*	0.619	0.674

TABLE 9. Ranking of the memory requirements in eighteen cases

Classifiers	*Whole faces with the size of* 4225×1	*Whole faces with the size of* 361×1	*Faces with twelve partitions*	*Faces with thirteen partitions*
PCA	4	1	2	3
D-LDA	15	6	13	14
SVM	18	12	16	17
CVA	9	5	7	8
CVA with Majority Vote	*Not Applicable*	*Not Applicable*	10	11

as PCA, CVA, D-LDA, and SVM from the classifier requiring the least memory to the classifier requiring the most memory. When all of the above comparisons are considered, it can be concluded that CVA is superior to the other classifiers.

6. **Conclusions.**

Among many applications in computer vision, face recognition is still a very important and attractive subject. In order to correctly identify facial images, many feature extraction and face recognition methods have been investigated. Despite these methods, facial images are still unable to be efficiently recognized under different illumination conditions and/or facial expressions. Therefore, new feature extraction and face recognition systems are needed.

In this paper, a novel framework, which includes the combination of partition-based classification results obtained by CVA with the Majority Voting technique, is proposed. To analyze the performance of our proposed framework, the feature vectors obtained by stacking the partition vectors are applied to the well-known methods, PCA, D-LDA, SVM, and CVA. The classification results obtained from the application of the proposed framework are superior to the results obtained from all other studies, except for those with the SVM classifier. However, the proposed framework has advantages in terms of processing power and memory requirement when compared with the only partition-based

SVM classifier. Therefore, the proposed framework provides new insights into partition-based recognition studies realized using CVA.

The effectiveness of partitioning a face on the recognition performance is also examined in this paper. Despite an approximately 15% decrease in the dimension of the vectors obtained from whole faces, the recognition rates obtained for twelve or thirteen partitions are appreciably greater than those obtained with whole faces for all classification methods, as observed from Tables 2-5. Thus, the remaining parts other than the thirteen partitions possess disruptive features, i.e., the remaining parts result in the degradation of face recognition performance. Reduction in the dimension of facial images also results in an evidently advantageous situation in terms of processing power and memory requirements for the training as well as the classification parts of all of the recognition methods.

REFERENCES

[1] O. De Vel and S. Aeberhard, Line-based face recognition under varying pose, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.21, no.10, pp.1081-1088, 1999.

[2] M. Hess and M. Martinez, Facial feature extraction based on the Smallest Univalue Segment Assimilating Nucleus (SUSAN) algorithm, *Proc. of Picture Coding Symposium*, 2004.

[3] A. S. Samra, S. E. Allah, T. G. Allah and R. M. Ibrahim, Face recognition using wavelet transform, fast fourier transform and discrete cosine transform, *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp.272-275, 2004.

[4] D. Maturana, D. Mery and Á. Soto, Face recognition with decision tree-based local binary patterns, *Lecture Notes in Computer Science*, vol.6495, pp.618-629, 2011.

[5] M. S. Sarfraz and O. Hellwich, Probabilistic learning for fully automatic face recognition across pose, *Image and Vision Computing*, vol.28, pp.744-753, 2010.

[6] F. Zuo and P. H. N. De With, Fast facial feature extraction using a deformable shape model with Haar-wavelet based local texture attributes, *Proc. of Int. Conf. on Image Processing*, pp.1425-1428, 2004.

[7] M. Chorás, Ear biometrics based on geometrical feature extraction, *Electronic Letters on Computer Vision and Image Analysis*, vol.5, no.3, pp.84-95, 2005.

[8] D. J. Hurley, M. S. Nixon and J. N. Carter, Force field feature extraction for ear biometrics, *Computer Vision and Image Understanding*, vol.98, pp.491-512, 2005.

[9] C. Kervrann, F. Davoine, P. Pe'rez, H. Li, R. Forchheimer and C. Labit, Generalized likelihood ratio-based face detection and extraction of mouth features, *Lecture Notes in Computer Science*, vol.1206, pp.27-34, 1997.

[10] F. Y. Shih and C.-F. Chuang, Automatic extraction of head and face boundaries and facial features, *Information Sciences*, vol.158, pp.117-130, 2004.

[11] Y. Xie, L. Setia and H. Burkhardt, Face image retrieval based on concentric circular Fourier-Zernike descriptors, *International Journal of Innovative Computing, Information and Control*, vol.4, no.6, pp.1433-1443, 2008.

[12] M.-M. Chuang, R.-F. Chang and Y.-L. Huang, Automatic facial feature extraction in model-based coding, *Journal of Information Science and Engineering*, vol.16, pp.447-458, 2000.

[13] H. Gu, G. Su and C. Du, Feature points extraction from faces, *Proc. of Conf. on Image and Vision Computing*, pp.154-158, 2003.

[14] M. Castrillón, O. Déniz, D. Hernández and J. Lorenzo, A comparison of face and facial feature detectors based on the Viola-Jones general object detection framework, *Machine Vision and Applications*, vol.22, pp.481-494, 2011.

[15] H. Li, T. Jiang and K. Zhang, Efficient and robust feature extraction by maximum margin criterion, *IEEE Trans. on Neural Networks*, vol.17, no.1, pp.157-165, 2006.

[16] J. Wang, B. Zhang, S. Wang, M. Qi and J. Kong, An adaptively weighted sub-pattern locality preserving projection for face recognition, *Journal of Network and Computer Applications*, vol.33, pp.323-332, 2010.

[17] Z. Dol, R. A. Salam and Z. Zainol, Face feature extraction using Bayesian network, *Proc. of the 4th Int. Conf. on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia*, pp.261-264, 2006.

[18] I. King and L. Xu, Localized principal component analysis learning for face feature extraction and recognition, *Proc. of Workshop on 3D Computer Vision*, pp.124-128, 1997.

[19] T.-Y. Chen, T.-H. Chen, D.-J. Wang and Y.-C. Chiou, Real-time video object segmentation algorithm based on change detection and background updating, *International Journal of Innovative Computing, Information and Control*, vol.5, no.7, pp.1797-1810, 2009.

[20] Y. Pang, Z. Liu and N. Yu, A new nonlinear feature extraction method for face recognition, *Neurocomputing*, vol.69, pp.949-953, 2006.

[21] J. Ahlberg, Facial feature extraction using eigenspaces and deformable graphs, *Proc. of Int. Workshop on Synthetic/Natural Hybrid Coding and 3-D Imaging*, 1999.

[22] V. D. M. Nhat and S. Lee, New feature extraction approaches for face recognition, *Lecture Notes in Computer Science*, vol.3809, pp.489-497, 2005.

[23] Y.-S. Wang, C.-H. Lin and T.-Y. Lee, Interactive model decomposition using protrusive graph, *International Journal of Innovative Computing, Information and Control*, vol.4, no.8, pp.1887-1896, 2008.

[24] H. Çevikalp, M. Neamtu, M. Wilkes and A. Barkana, Discriminative common vectors for face recognition, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.27, no.1, pp.4-13, 2005.

[25] K.-C. Lee, J. Ho and D. J. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.27, no.5, pp.684-698, 2005.

[26] X. Wang and X. Tang, An improved Bayesian face recognition algorithm in PCA subspace, *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol.3, pp.129-132, 2003.

[27] X. Wang and X. Tang, A unified framework for subspace face recognition, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.26, no.9, pp.1222-1228, 2004.

[28] H. Çevikalp, M. Neamtu and M. Wilkes, Discriminative common vector method with kernels, *IEEE Trans. on Neural Networks*, vol.17, no.6, pp.1550-1565, 2006.

[29] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.19, no.7, pp.711-720, 1997.

[30] J. Lu, K. N. Plataniotis and A. N. Venetsanopoulos, Regularized D-LDA for face recognition, *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol.3, pp.125-128, 2003.

[31] W. Li, C. Zhou and J. Xu, A novel face recognition method with feature combination, *Journal of Zhejiang University – Science A*, vol.6, no.5, pp.454-459, 2005.

[32] H. Yu and J. Yang, A direct LDA algorithm for high-dimensional data with application to face recognition, *Pattern Recognition*, vol.34, pp.2067-2070, 2001.

[33] D. Xi, I. T. Podolak and S.-W. Lee, Facial component extraction and face recognition with support vector machines, *Proc. of the 5th IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pp.83-88, 2002.

[34] B. Heisele, P. Ho, J. Wu and T. Poggio, Face recognition: Component-based versus global approaches, *Computer Vision and Image Understanding*, vol.91, pp.6-21, 2003.

[35] B. Heisele, T. Serre and T. Poggio, A component-based framework for face detection and identification, *International Journal of Computer Vision*, vol.74, no.2, pp.167-181, 2007.

[36] E. Hjelmås, Biometric systems: A face recognition approach, *Proc. of the Norwegian Conf. on Informatics*, pp.189-197, 2000.

[37] A. Martinez, Face image retrieval using HMMs, *Proc. of IEEE Workshop Content-Based Access of Image and Video Libraries*, pp.25-39, 1999.

[38] T.-Y. Chen, T.-H. Chen and D.-J. Wang, A cost-effective people-counter for passing through a gate based on image processing, *International Journal of Innovative Computing, Information and Control*, vol.5, no.3, pp.785-800, 2009.

[39] A. Martinez and R. Benavente, The AR face database, *CVC Technical Report #24*, 1998.

[40] In Web: http://www.mathworks.com/matlabcentral/fileexchange/, 2012.

[41] A. Martinez and A. C. Kak, PCA versus LDA, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.23, no.2, pp.228-233, 2001.

[42] M. Turk and A. Pentland, Face recognition using eigenfaces, *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp.586-591, 1991.

[43] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer Verlag, Berlin, 2000.

[44] H. Bredin, N. Dehak and G. Chollet, GMM-based SVM for face recognition, *Proc. of the 18th Int. Conf. on Pattern Recognition*, pp.1111-1114, 2006.

[45] K. Hotta, Support vector machine with local summation kernel for robust face recognition, *Proc. of the 17th Int. Conf. on Pattern Recognition*, pp.482-485, 2004.

[46] J. Ruiz-Pinales, J. J. Acosta-Reyes, A. Salazar-Garibay and R. Jaime-Rivas, Shift invariant support vector machines face recognition system, *Proc. of World Academy of Science, Engineering and Technology*, vol.16, pp.167-171, 2006.

[47] R. P. W. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. De Ridder and D. M. J. Tax, PRTools4, A Matlab toolbox for pattern recognition, *Delft University of Technology*, 2004.

[48] M. B. Gülmezoğlu, V. Dzhafarov, M. Keskin and A. Barkana, A novel approach to isolated word recognition, *IEEE Trans. on Acoustic Speech and Signal Processing*, vol.7, no.6, pp.620-628, 1999.

[49] M. B. Gülmezoğlu, V. Dzhafarov and A. Barkana, The common vector approach and its relation to principal component analysis, *IEEE Trans. on Speech and Audio Processing*, vol.9, no.6, pp.655-662, 2001.

[50] M. B. Gülmezoğlu and A. Barkana, Text-dependent speaker recognition by using Gram-Schmidt orthogonalization method, *Proc. of Int. Conf. on Signal Processing and Applications*, pp.438-440, 1998.

[51] M. B. Gülmezoğlu and S. Ergin, An approach for bearing fault detection in electrical motors, *European Trans. on Electrical Power*, vol.17, no.6, pp.628-641, 2007.

[52] E. Oja, *Subspace Methods of Pattern Recognition*, John Wiley and Sons, New York, 1983.

[53] D. L. Swets and J. Weng, Using discriminant eigenfeatures for image retrieval, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.18, no.8, pp.831-836, 1996.