Pixel-Level Alignment of Facial Images for High Accuracy Recognition Using Ensemble of Patches

Hoda Mohammadzade, Amirhossein Sayyafan, Benyamin Ghojogh

***Abstract*—The variation of pose, illumination and expression makes face recognition still a challenging problem. As a preprocessing in holistic approaches, faces are usually aligned by eyes. The proposed method tries to perform a pixel alignment rather than eye-alignment by mapping the geometry of faces to a reference face while keeping their own textures. The proposed geometry alignment not only creates a meaningful correspondence among every pixel of all faces, but also removes expression and pose variations. The geometry alignment is performed pixel-wise, i.e., every pixel of the face is corresponded to a pixel of the reference face. The proposed fine alignment method results in a high accuracy recognition using ensemble of random facial patched. In order to obtain the discriminative subspace of each patch, the simple and yet powerful Fisherface method is used. However, any other holistic-based face classification methods can be used instead of Fisherfaces. Experimental results show a great improvement using the proposed method in comparison to eye-aligned recognition. For instance, at the false acceptance rate of 0.001, the true positive rates of recognition are respectively improved by 24% and 33% in Yale and AT&T datasets. In LFW dataset, which is a challenging big dataset, improvement is 20% in false acceptance rate of 0.1. In Cohn-Kanade dataset, however, rates have not changed in false acceptance rate of 0.001.**

***Index Terms*—face recognition, pixel alignment, geometrical transformation, pose and expression variation, ensemble of patches.**

I. Introduction

FACE Recognition is one of the most attractive and practical fields of research in pattern analysis and image processing, receiving much attention from different knowledge backgrounds including pattern recognition, computer vision, image processing, statistical learning, neural networks, and computer graphics [1].

According to [1], face recognition methods can be categorized into two main categories; feature-based and holistic (whole-pixels) methods. Feature-based methods try to create a feature vector out of the face for the learning process. The holistic recognition uses all pixels of face region as raw data for recognition and learning.

Feature-based methods utilize the geometrical and structural features of face [1]. For instance, in [2], features of head width, distances between eyes and eyes to mouth are compared. In [3], angles and distances between eye corners, mouth hole, chin top and the nostrils are used. In [4], face features such as mouth, nose, eyebrows, and face outline are detected using horizontal and vertical gradients. In this method, template matching using correlation is also proposed. In [5], [6] Hidden Markov Model (HMM) is used on pixel strips of different parts of face. Also, recently, a patch-based representation is used in [7] in which each patch tries to learn a transformation dictionary in order to transform the features onto a discriminative subspace.

Some feature-based methods use both features and whole pixels together in order to enhance the performance of recognition [1]. Eigenmodules [8] can be mentioned in which eigenfaces are combined with eigenmodules of face such as eigeneyes, eigenmouth and eigennose. In [9] Principle Component Analysis (PCA) is used in combination with Local Feature Analysis (LFA). Some of the methods in this category which seem more promising are based on "shape-free" face concept. In [10], [11], Active Appearance Model (AAM) has been proposed as a method of warping textures of image patches to a specific geometry in an iterative manner. In this method [12], [13], [14], [15], the patch of the face is labeled by several landmarks (model points), the texture of face is projected onto the texture model frame by applying scale and offset to the intensities, and the residual (error) between the projected and previous image patches is iteratively reduced. In [12], [15], the shape of the face is also modeled using Active Shape Model (ASM) [16]. The authors have shown that different weights of eigenvalues can vary the different aspects and parts of face shape models. In [17], several shape-free (neutral) faces create an ensemble and all the faces are approximated by a linear combination of the eigenfaces of the ensemble.

Despite significant advances of feature-based methods, holistic methods are still being received lots of attention as they use the information of all pixels in the face region. Holistic methods detect and crop the face out of the image and use it as a raw input for classification. Eigenfaces [18], [19], Fisherfaces [20], and Kernel faces [21], [22] are several well-known examples of this category which respectively create a feature space using Principle Component Analysis (PCA), Fisher Linear Discriminant Analysis (LDA) [23] and Kernel Direct Discriminant Analysis (KDDA) for face classification and recognition. Face recognition using support vector machine (SVM) [24] is another method from this category, which formulates face recognition as a two-class problem, one class as dissimilarities between faces of the same person and

Hoda Mohammadzade's e-mail: hoda@sharif.edu
Amirhossein Sayyafan's e-mail: sayyafan@ee.sharif.edu
Benyamin Ghojogh's e-mail: ghojogh_benyamin@ee.sharif.edu
All authors are with Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran.

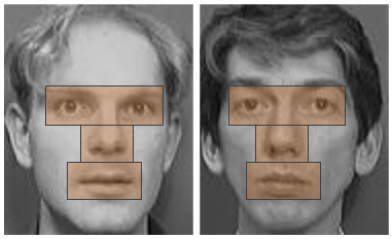

Fig. 1: Important organs of face for face alignment.

the other class as dissimilarities between faces of different individuals. Bayesian classifier [25] can also be mentioned in this category, which has a probabilistic approach toward the similarity of faces. Some other holistic methods of face recognition have used artificial neural networks [1], [26]. As instance, Probabilistic Decision-Based Neural Network (PDBNN) [27] and Convolutional Neural Networks (CNN) [28], [29], [30], [31] can be mentioned. Recently, Sparse Representation based Classification (SRC) [32] is used in order to create a recognition system with robustness to illumination and occlusion.

The remaining of this paper is organized as follows. Section II introduces the proposed method and its motivation; and Section III details the geometrical alignment as the first part of proposed method. Thereafter, geometrical information is more discussed in Section IV. Creating feature vectors using ensemble of patches, using Fisher Linear Discriminant Analysis (LDA), and decision fusion are explained in Section V afterwards. Section VI sums up the proposed method by illustrating the overall structure. The utilized datasets and experimental results are also reported in Section VII. Finally, in Section VIII, article is summarized and concluded. Moreover, in this section, several discussions are performed on alignment of features, warping, and ensemble of patches, the differences and similarities of the proposed warping and AAM method are debated, and future work is introduced.

II. Motivation

It has been shown that holistic methods perform better if faces are aligned as much as possible, i.e., corresponding features of the face are located at same coordinates in the image. Alignment methods are mostly based on aligning eyes. The reason for selecting eyes for alignment is that eyes are the most discriminative features for recognition. The other organs of the face such as the nose and lips do not carry as much information as the eyes since lips may vary much a lot in different faces of even one person and the noses are not discriminative enough among different persons. As is obvious in the two faces in Fig. 1, the eyes are better identifiers of the faces than the lips and nose, encouraging methods to choose them for alignment.

Although eye alignment tries its best to align faces as much as possible, complete alignment using a linear transformation is not possible because of variety of face geometries. The proposed method goals a complete geometry alignment using a non-linear transformation which is detailed in following sections. Faces are transformed and warped onto a reference geometry while their textures are kept unchanged. Having the texture not changed helps to identify people because the texture carries a large part of the identification information of the faces. After having the geometry of faces aligned, every pixel through all faces correspond to the same feature of the face. For instance, the middle pixel might correspond to the nose tip in all warped faces, and so on. Having this property, the vector of pixel intensities of the whole faces (scanned either row-wise or column-wise) are ready to be fed to a classifier. However, what is missing in these feature vectors is the geometry information of each face, which is also important for identification. The proposed method solves this problem by including the geometry information to the feature vectors which is done by fusing the warped intensities and original coordinates of each pixel before warping.

Before explaining the parts of proposed methodology, it should be mentioned that as pre-processing, all the faces of utilized dataset are eye-aligned manually and then histogram equalized.

III. Geometrical Alignment

Geometrical alignment can be defined as aligning the geometries of faces to a unique geometry while saving their own textures. In the proposed geometrical alignment method, a reference geometry is defined and the geometry of all faces of train and test procedures are transformed to this geometry. Here, the geometry of a face is defined as the location of the contours of the facial landmarks. Therefore, geometrical alignment is performed by warping a face such that its facial contours coincide with those of the reference contours.

In the proposed method, in order to detect facial landmarks, every landmark detection method can be used such as Active Shape Model (ASM) [16] or Constrained Local Neural Fields (CLNF) [33]. In this work, CLNF is utilized for this purpose.[1] The landmarks in this work are as follows. There are 17 landmarks around the face region, 14 landmarks for lips, three landmarks for each upper and lower teeth, six landmarks for each eye, nine landmarks for the whole nose and five landmarks for each eyebrow, resulting in 68 total landmarks.

In the following sections, different steps of the proposed method are explained in details.

A. Fitting Face contours

The CLNF method [33], which is an enhanced Constrained Local Model (CLM) [34], is used for detecting landmarks of each train and test face. This method is briefly described in the following. Interested readers are encouraged to refer to [33] for more details.

The CLNF method consists of two main parts: (I) probabilistic patch expert (landmark detector), and (II) non-uniform regularized landmark mean-shift optimization technique.

At first, face or faces are detected with a tree-based method. CLNF method introduces patch experts which are small partitions of pixels around the interest points such as face edge,

[1]The code of CLNF method can be found in https://github.com/TadasBaltrusaitis/OpenFace.

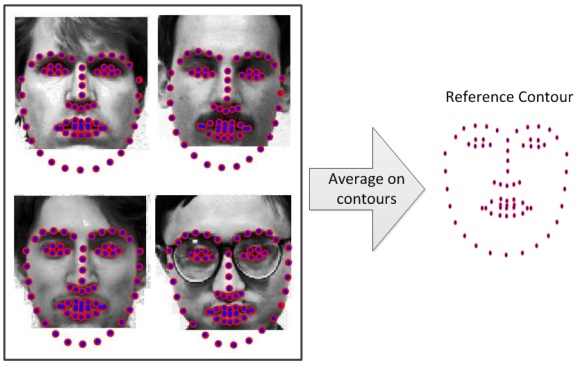

Fig. 2: Obtaining reference contour by averaging landmark contours of several neutral faces.

eyes, eyebrows, nose and lips. The initial patch experts are put on the image. The pixels which fall in the m^{th} patch are named as $X_m = \{x_1, x_2, ..., x_n\}$ where x_i is a two-dimensional vector representing the coordinate of pixels. This method uses a one-layer neural network with X_m as inputs, and outputs $Y_m = \{y_1, y_2, ..., y_n\}$ where y_i is a scalar [33]. A potential function is defined as Ψ which is a function of vertex features f_k and edge features g_k and l_k. The features are defined as [33],

$$f_k(y_i, x_i, \theta_k) = -(y_i - h(\theta_k, x_i))^2, \tag{1}$$

$$g_k(y_i, y_j) = -0.5 S_{ij}^{(g_k)} (y_i - y_j)^2, \tag{2}$$

$$l_k(y_i, y_j) = -0.5 S_{ij}^{(l_k)} (y_i + y_j)^2, \tag{3}$$

where θ_k's are the weights of k^{th} neuron and $h(.,.)$ is the sigmoid activation function of the neural network. $S^{(g_k)}$ and $S^{(l_k)}$ control the smoothness (similarity) and sparsity, respectively. This method attempts to maximize the probability

$$P(y|X) = \frac{e^{\Psi}}{\int_{-\infty}^{\infty} e^{\Psi} dy}. \tag{4}$$

Non-uniform regularized landmark mean-shift optimization technique considers that variance of different patches are not similar and therefore sets several weights W's for them. The contour p of patches is updated as [33],

$$p^{new} = p^{old} + \Delta p, \tag{5}$$

in which the step Δp is defined as,

$$\Delta p = -(J^T W J + r\Lambda^{-1})(r\Lambda^{-1} p - J^T W v), \tag{6}$$

where J is the Jacobian of the landmark locations, r is the regularization factor, Λ^{-1} is the matrix describing the prior on the parameter p, and v is the mean-shift vector over the patch responses [33].

B. Reference Contour

Reference contours are obtained by averaging the contours of landmarks of several neutral faces from the training set. Figure 2 shows an example of reference contours.

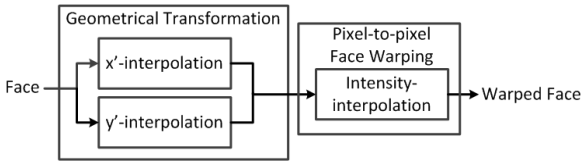

Fig. 3: Procedure of geometrical transformation and pixel-to-pixel face warping.

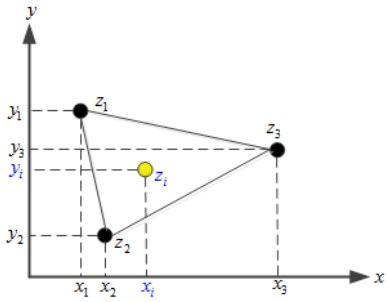

Fig. 4: Affine interpolation.

C. Transformation and Pixel-to-pixel Warping

After fitting the contour of landmarks to the input face, the face is geometrically transformed and warped to reshape to the geometry of reference face. This step is detailed in this section.

For the geometrical transformation and pixel-to-pixel warping, three interpolations are performed as depicted in Fig. 3 which are detailed next. As a result of these interpolations, the intensity of each pixel is transformed to its corresponding location on the warped face. This transformation is guided by the transformation between the location of landmarks on the input face and the location of those on the warped face.

It is important to note that the proposed face warping method differs from the conventional one as the target coordinates for every single pixel of the input face is calculated using the described interpolation procedures.

1) Affine Interpolation: In this work, affine transformation is used in order to perform the coordinate interpolations, i.e., x'- and y'-interpolation. Affine transformation uses three surrounding points to calculate the interpolated value at a new point. Assuming that the points have two dimensions. In affine interpolation method [35], the value at each point is approximated as,

$$z_i = f(x_i, y_i) \simeq a_0 + a_1 x_i + a_2 y_i, \tag{7}$$

where the coefficients a_0 to a_2 are calculated by solving the following linear system, according to Fig. 4,

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}. \tag{8}$$

If this matrix equation is denoted as $XA = Z$, then by solving it using least square method, the coefficients are found as,

$$A = (X^T X)^{-1} X^T Z. \tag{9}$$

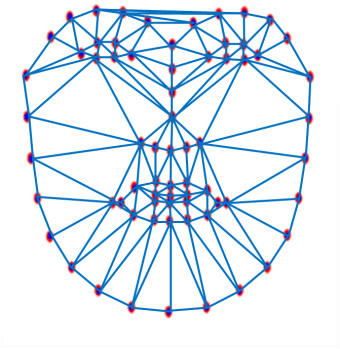

Fig. 5: Delaunay triangulation of face landmarks.

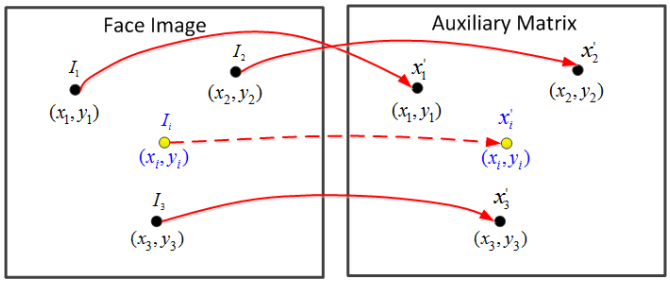

Fig. 6: x'-interpolation.

2) Delaunay Triangulation of Landmarks: According to [36], a triangulation of a finite point set $P \subset \mathbb{R}^2$ is called a Delaunay triangulation, if the circumcircle of every triangle is empty, that is, there is no point from P inside the circumcircle of any triangle.

Each face is triangulated using Delaunay method, as depicted in Fig. 5. By performing triangulation, the triangles needed for affine interpolations are obtained which are used in geometrical transformation as described next.

3) Geometrical Transformation: Let (x, y) and (x', y') denote the coordinates of pixels on the input and warped face, respectively, and $I(x, y)$ and $I'(x', y')$ denote their corresponding intensities.

For x'-interpolation, an auxiliary matrix is created with the same size as the input face. In this matrix, the x' of landmarks are put on the same entry as they were in the input face matrix. The other entries of this matrix are found using affine interpolation resulting in the x' coordinate of other pixels. This procedure is depicted in Fig. 6. The y'-interpolation is performed similarly as shown in Fig. 7. Thereafter, the (x', y') target coordinate of all input pixels are found and each input pixel is known where to be transferred.

4) Pixel-to-pixel Warping: After x'- and y'-interpolations, each (x', y') coordinate gets the intensity of its corresponding (x, y) from the input face, i.e.,

$$I'(x', y') = I(x, y). \tag{10}$$

I' values are then resampled on a uniform grid, e.g., 140×120 pixels, to create the warped face (see Fig. 8).

For the sake of demonstration, an example of geometrical transformation and pixel-to-pixel warping on a sample face with a few number of landmarks is depicted in Fig. 9. In this figure, the yellow diamond points and red square points are respectively input and reference landmarks. The face is warped so that the input landmarks are precisely located at the position of reference landmarks, as it was the goal of geometrical transformation. The other pixels are interpolated as explained previously.

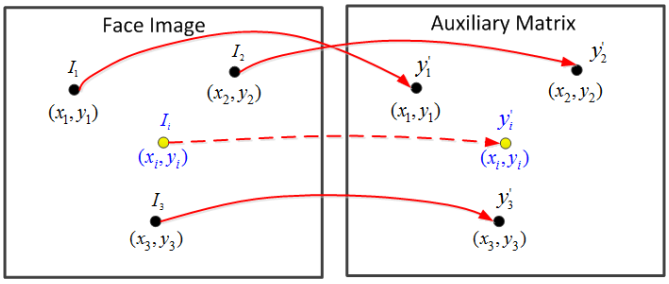

Fig. 7: y'-interpolation.

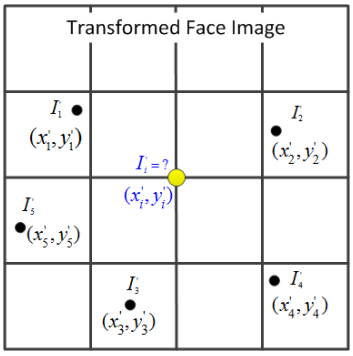

Fig. 8: intensity-interpolation.

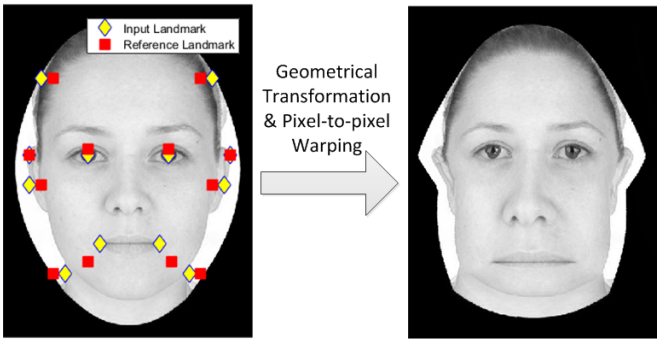

Fig. 9: An example of geometrical transformation and pixel-to-pixel warping.

IV. Geometrical Information

Geometrical information seems to be useful in addition to intensity information of the warped face. Obviously, the geometry information of each face exists in its unwarped (input) face. By finding the original coordinate (i.e., coordinate in the unwarped face image) of each pixel of the warped face, geometry information can be gathered. However, as x' and y' coordinates have been once resampled, their original coordinates cannot be found directly. These coordinates can be obtained by performing two other resamplings on the same grid as before; one for original x values and one for original

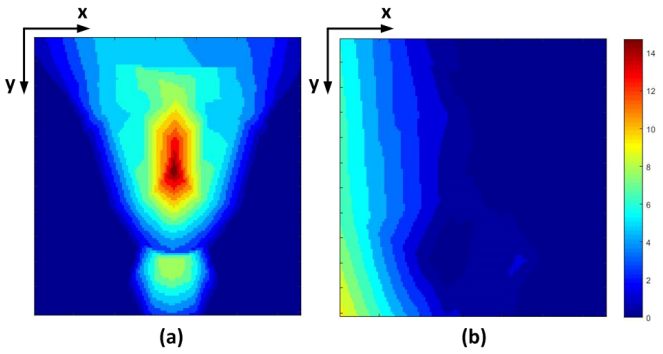

Fig. 10: Illustration of Δx and Δy information for a sample warped face. (a) Δx information, (b) Δy information.

y values. To better explain it, two other interpolations are performed in which the x and y source coordinate of each pixel in the warped face is found using interpolation. These two interpolations are exactly the same as previous intensity-interpolation (Fig. 8) but by replacing I' with x and y.

For the sake of better visualization, the difference of original coordinates x and y of every pixel from its previous pixel is calculated. The differences in original coordinates are denoted as Δx and Δy here, respectively for differences in x and y information. Figure 10 illustrates the information of Δx and Δy for a sample face in Yale dataset [41]. The amount of vertical and horizontal transitions of each pixel after warping can be seen in this figure. This figure shows that for this specific face, warping has changed face more in horizontal direction rather than vertical.

V. Classification Using Ensemble of Patches

A. Ensemble of Patches and Feature Vectors

Instead of using the whole face, a patch-based approach is used in this work. To do this, an ensemble of patches are created in the limit of face frame. The location of patches are selected randomly once, and for all faces of dataset, the same patches are used in both training and testing phases. The optimum number and size of patches were found through trial and error to be 80 and 30×30 pixels, respectively, over various different datasets.

For every face, the ensemble of patches are applied on intensity matrix of its warped face, its Δx information, and its Δy matrix. An example of applying ensemble of patches on these three matrices is depicted in Fig. 11. Note that the information of Δx and Δy is the same as x and y. In order to have the feature vectors of each patch, the matrix coefficients fell in the patch are reshaped as a vector. In other words, for the p^{th} patch, if the size of patch is $m\times m$, the feature vectors are obtained as,

$$f_p^I = [I'(1,1), I'(1,2), \dots, I'(m,m)]^T, \tag{11}$$

$$f_p^{\Delta x} = [\Delta x(1,1), \Delta x(1,2), \dots, \Delta x(m,m)]^T, \tag{12}$$

$$f_p^{\Delta y} = [\Delta y(1,1), \Delta y(1,2), \dots, \Delta y(m,m)]^T, \tag{13}$$

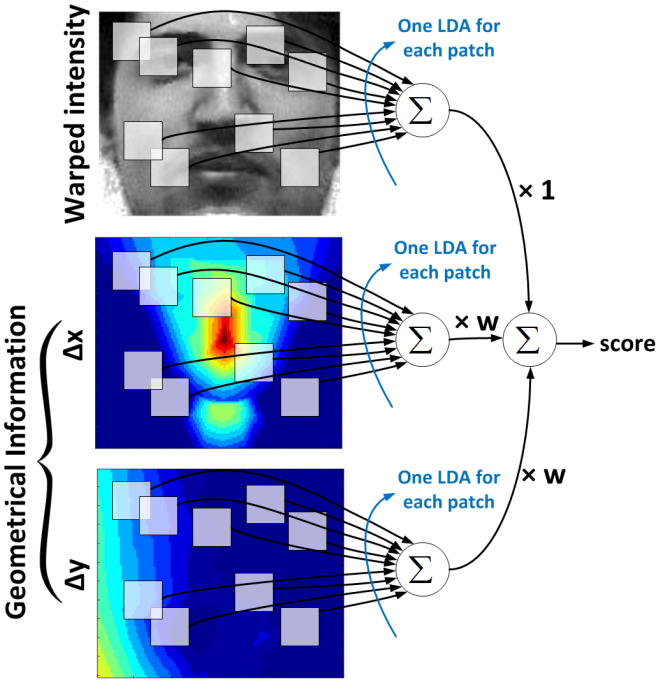

Fig. 11: Classification using ensemble of patches.

where f_p^I, $f_p^{\Delta x}$, and $f_p^{\Delta y}$ are respectively the feature vectors of p^{th} patch with respect to intensity, Δx, and Δy matrices. Moreover, $I'(k,l)$, $\Delta x(k,l)$, and $\Delta y(k,l)$ denote the coefficient of intensity, Δx, and Δy matrices which fall in pixel (k,l) of the patch.

B. Fisher Linear Discriminant Analysis

After constructing the feature vecotrs of ensemble of patches, three separate Fisher Linear Discriminant Analysis (LDA) subspaces are trained for every patch. To better explain, for p^{th} patch in all training set of faces, one Fisher LDA subspace is trained using the feature vectors f_p^I, one for feature vectors $f_p^{\Delta x}$, and one for feature vectors $f_p^{\Delta y}$. In this work, Fisherface method [20] is used for classification of each patch; however, other more complicated learning methods can be used in future works.

The goal of Fisher LDA is maximizing the ratio of,

$$W_{opt} = \arg\max_W \frac{|W^T S_b W|}{|W^T S_w W|}, \tag{14}$$

where S_b and S_w are the between- and within-class scattering matrices, respectively [37], [38], formulated as,

$$S_w = \sum_{i=1}^{C} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T, \tag{15}$$

$$S_b = \sum_{i=1}^{C} N_i (\mu_i - \mu)(\mu_i - \mu)^T, \tag{16}$$

where μ_i is the mean of i^{th} class and μ is the mean of means of classes. N_i is the number of samples of i^{th} class. And x_k is the k^{th} sample of i^{th} class (X_i).

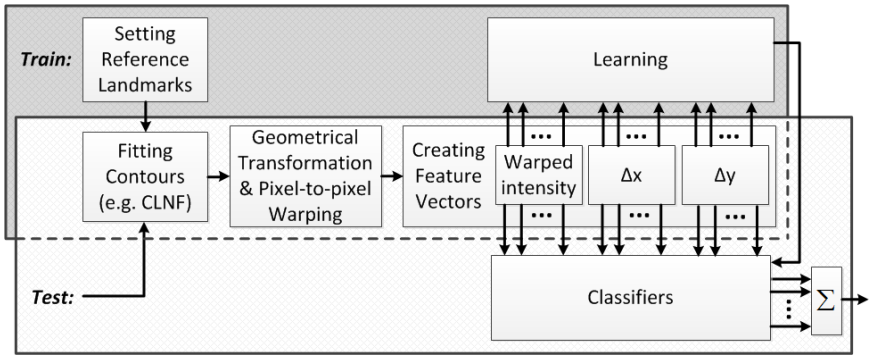

Fig. 12: The overall structure of the proposed method.

After finding scattering matrices, a discriminative subspace is created using the eigenvectors of $S_w^{-1}S_b$ matrix. To extract the discriminative features from each feature vector, it should be projected onto this subspace. If C denotes the number of classes, this projection also reduces the dimension of data to $C-1$ [37], [38].

C. Decision Making

Clearly, there are a lot of different features available rather than one, i.e., intensity, Δx, and Δy features for all patches. Hence, in order to obtain the final similarity/distance score between two face images, a fusion is required to be performed. The fusion can be performed either before, during, or after classification, which are respectively known as data-, feature-, and decision-level fusion. In the fusion of data and feature, respectively, the two feature vectors are concatenated before and after projecting to the discriminative subspace; and in the fusion of decision, the resulting scores are fused. The fusion of decision is found to perform better in this work.

For p^{th} patch in every face image, each of the feature vectors, f_p^I, $f_p^{\Delta x}$, and $f_p^{\Delta y}$, is projected onto their corresponding discriminative LDA subspace, obtained as described in Section V-B. The projections result in projected feature vectors $\widehat{f}_p^I$, $\widehat{f}_p^{\Delta x}$, and $\widehat{f}_p^{\Delta y}$. In the context of face recognition, it has been shown that the cosine of the angle between two discriminative feature vectors, which is obviously a similarity score, results a better recognition rather than distance measures such as Euclidean distance [39], [40]. Hence, cosine is used in this work for matching purposes. Then, the similarity score between two face images i and j is calculated as follows. First, the similarity scores in the discriminative subspaces related to p^{th} patch are obtained as,

$$\text{sim}_p^I(i,j) = \cos(\widehat{f}_{p,i}^I, \widehat{f}_{p,j}^I) = \frac{\widehat{f}_{p,i}^I \widehat{f}_{p,j}^I}{|\widehat{f}_{p,i}^I||\widehat{f}_{p,j}^I|}, \tag{17}$$

$$\text{sim}_p^{\Delta x}(i,j) = \cos(\widehat{f}_{p,i}^{\Delta x}, \widehat{f}_{p,j}^{\Delta x}) = \frac{\widehat{f}_{p,i}^{\Delta x} \widehat{f}_{p,j}^{\Delta x}}{|\widehat{f}_{p,i}^{\Delta x}||\widehat{f}_{p,j}^{\Delta x}|}, \tag{18}$$

$$\text{sim}_p^{\Delta y}(i,j) = \cos(\widehat{f}_{p,i}^{\Delta y}, \widehat{f}_{p,j}^{\Delta y}) = \frac{\widehat{f}_{p,i}^{\Delta y} \widehat{f}_{p,j}^{\Delta y}}{|\widehat{f}_{p,i}^{\Delta y}||\widehat{f}_{p,j}^{\Delta y}|}, \tag{19}$$

where $\widehat{f}_{p,i}^I$, $\widehat{f}_{p,i}^{\Delta x}$, and $\widehat{f}_{p,i}^{\Delta y}$ are respectively projected feature vectors $\widehat{f}_p^I$, $\widehat{f}_p^{\Delta x}$, and $\widehat{f}_p^{\Delta y}$ in i^{th} face image.

Then, the final similarity score is simply obtained by a weighted summation of all the scores of patches (decision fusion),

$$\text{sim}(i,j) = \sum_{p=1}^{80} (\text{sim}_p^I(i,j) + w\ \text{sim}_p^{\Delta x}(i,j) + w\ \text{sim}_p^{\Delta y}(i,j)), \tag{20}$$

where w is the weight associated to the geometrical information, and the weight of intensity information is considered to be one for simplicity. The classification using ensemble of patches is summarized in Fig. 11.

VI. Overall Structure of the Proposed Face Recognition Framework

The proposed method can be summarized as is depicted in Fig. 12. In this method, a set of reference contours is constructed, landmarks of each train/test face are detected using CLNF method, the faces are transformed geometrically to the reference, warping is performed, and feature vectors are created for classification. In preparing feature vectors, the ensemble of patches are considered for matrices of warped intensity, Δx, and Δy. A separate Fisher LDA is trained for every patch in each of these matrices. Finally, in the test phase, the feature vectors are projected onto the corresponding LDA subspaces and the similarity scores are summed up together in order to have the total score.

VII. Experimental Results

A. Datasets

Four different datasets are used for evaluating the recognition performance using the proposed alignment method, which are Yale [41], AT&T [42], Cohn-Kanade [43], [44], and LFW datasets [45], [46] detailed in the following. In this work, as a pre-processing, the datasets are eye-aligned and then cropped using CLNF method [33]. To more explain, the location of eyes are found using CLNF and by a translation and rotation, the faces become eye-aligned.

1) Yale face dataset: The Yale face dataset [41] was created by the Center of Computational Vision and Control at Yale University, New Haven. It consists of 165 grayscale face images of 15 different persons. There exist 11 images per person depicting different facial expressions.

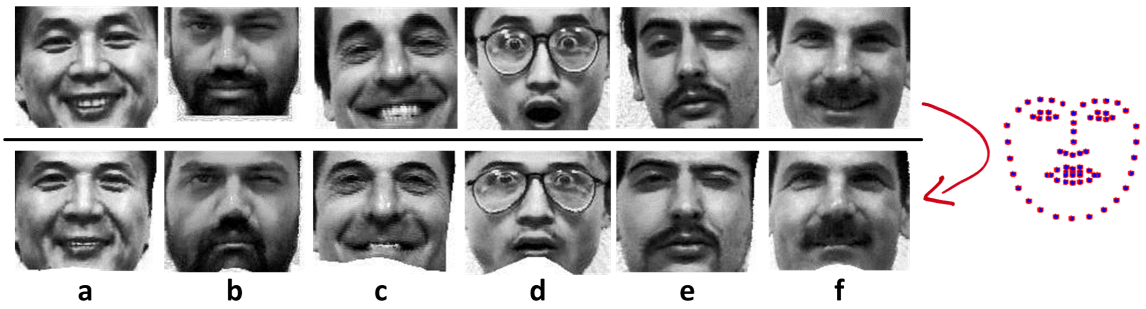

Fig. 13: Several samples of pixel alignment in Yale dataset

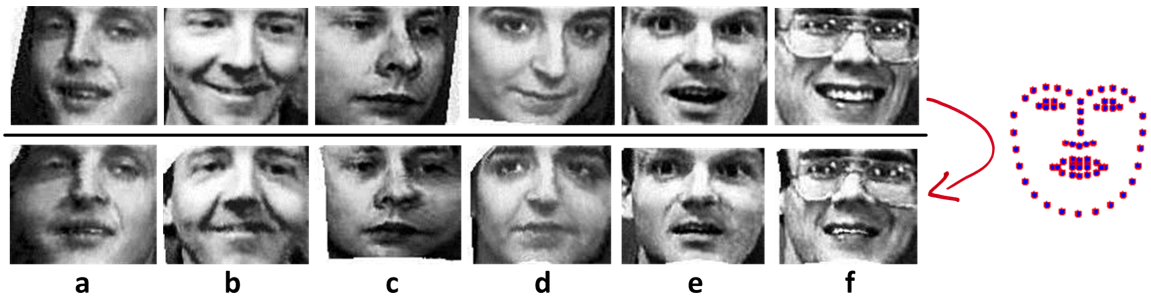

Fig. 14: Several samples of pixel alignment in AT&T dataset

2) AT&T face dataset: The AT&T face dataset [42] was created by the AT&T Laboratories Cambridge in 2002. There are pictures of 40 different persons with 10 different facial expressions.

3) Cohn-Kanade face dataset: Cohn-Kanade dataset [43], [44] includes 486 face sequences from 97 persons. Every sequence starts with neutral face and ends with extreme versions of expressions. Different expressions exist in this dataset, such as laughing, surprising, and etc. The first version of this dataset is used here. For every person in this dataset, merely one neutral face, all middle expressions, and all extreme expressions are utilized in this work to perform experiments.

4) LFW face dataset: Labeled Faces in the Wild (LFW) dataset [45], [46] is a very big and challenging dataset including 13,000 images of faces collected from the web. The faces have various poses, expressions, and locations in images. The distances of camera from persons are not necessarily the same in images. There are different number of images for every subject, from one to sometimes 10. The not-cropped and not-processed version of this dataset is used in this work for experiments.

B. Warped Faces

In this section, for the sake of visualization, several warped faces which are pixel aligned are shown and analyzed. Several samples of warped faces from Yale and AT&T datasets are illustrated in figures 13 and 14, respectively. In these figures, the first and second row are faces before and after warping, respectively. At the right-hand side of these figures, the reference contours are shown.

As seen in Fig. 13, faces (a), (c) and (f) are smiling originally but the mouths in the corresponding warped faces are closed and the teeth are roughly removed. Face (d), however, is wondering originally while the mouth is totally closed after warping. Similarly, in Fig. 14, faces (b), (d), (e) and (f) have different expressions while their corresponding warped faces have neutral expression with closed mouths. As shown in these figures, removing the facial expression is obviously one of the results of the proposed pixel-by-pixel alignment method, which of course can greatly improve the recognition task. Moreover, faces (b), (c) and (e) in Fig. 13 show that this method can also change the pose of faces to the pose of the reference contours. Similarly, in Fig. 14, faces (a), (b), (c) and (d) have frontal pose after warping. Clearly, in all the warped faces in figures 13 and 14, not only are different organs of the face aligned, but also other features of the face are almost aligned. However, due to the drawback of the landmark detection method in converging to exact landmark points, some features may not become well-aligned. For instance, in Fig. 13, the eyes are not completely open in the warped faces (a), (b), (c), (e) and (f).

C. Experiments

In all the experiments mentioned in this section, the dataset is firstly shuffled randomly and then 5-fold cross validation is performed. In the following, experimenting the impact of patch size are reported and analyzed. Thereafter, classification using ensemble of patches is compared to classification using the whole face. Finally, the proposed method is examined and compared to eye-aligned classification.

1) Experiment on Size of Patches: In this experiment, the effect of patch sizes are mentioned and reported. Different experiments on AT&T dataset were performed with different sizes of patches, which are 10×10, 20×20, 30×30, 40×40, 50×50, and random-sized patches each with one of the mentioned sizes. In these experiments, 80 random

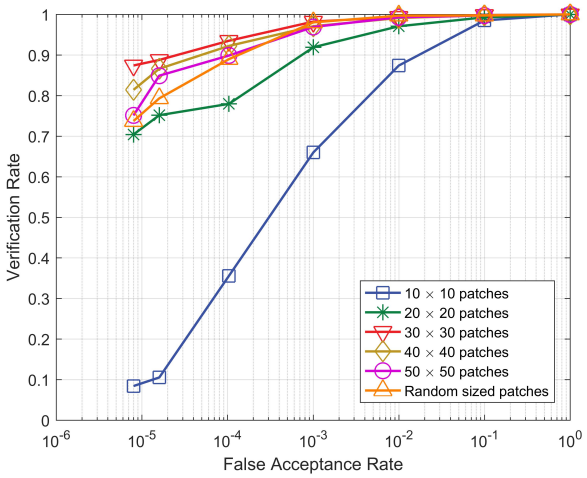

Fig. 15: Effect of size of patches in classification using ensemble of patches

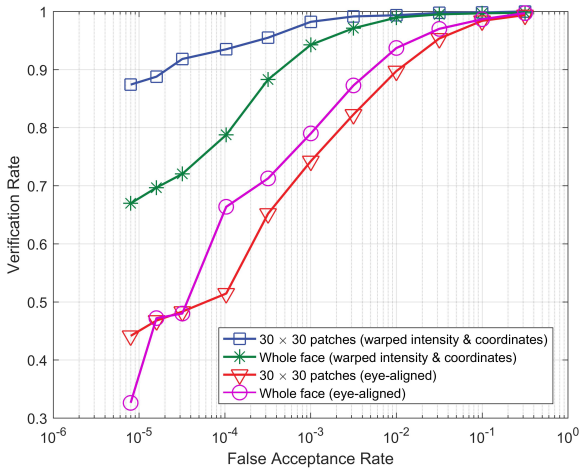

Fig. 16: Comparison of classification using the whole face or ensemble of patches

patches were utilized, and the optimum weight of geometrical information was found to be 0.2 through trial and error.

In each iteration of the experiments, the similarity score between every pair of gallery and probe images is calculated and the Receiver Operating Characteristic (ROC) curve using all the scores are plotted. The ROC curves of experiments are depicted in Fig. 15. As is obvious in this figure, the size of patches has important impact on the recognition rate. According to the curves, 30×30 patches perform better; therefore, merely 30×30 patches are used in the next experiments.

2) Patch-based Recognition Using Eye-aligned and Pixel-aligned Faces: Several other experiments were performed evaluating the effect of using ensemble of patches for both eye-aligned and pixel-aligned face images. In these experiments, 80 random patches were utilized with size 30×30. First, classification using ensemble of patches and not using patches were tested on eye-aligned images. Note that in classification using ensemble of patches for eye-aligned faces, the ensemble was solely applied on intensity matrix of eye-aligned images because warping does not exist anymore and thus there is no geometrical information. Figure 16 shows ROC curves of the two experiments performed on AT&T dataset. As can be seen in this figure, using ensemble of patches results in overall worse performance than not using patches when eye-aligned method is utilized.

On the other hand, the same two experiments were performed using pixel-aligned faces rather than eye-aligned ones. The ROC curves of these experiments on AT&T dataset are also depicted in Fig. 16. As obvious in this figure, when pixel-aligned faces are used, patch-based recognition produces superior results compared to not using patches. For instance, in FAR of 0.001, verification rates are roughly 99% and 94% in recognition using ensemble of patches and not using it, respectively. This result verifies the effectiveness of using ensemble of patches alongside having faces pixel-to-pixel warped.

3) Eye-aligned Versus Proposed Method: Eye-aligned face recognition is compared with the proposed pixel-aligned classification method in Fig. 17. This comparison is performed for four datasets, which are Yale [41], AT&T [42], Cohn-Kanade [43], [44], and LFW datasets [45], [46]. LFW dataset is a very challenging and big dataset and includes images which might have more than one face, but still there is only one subject label associated with each image. For this dataset, using CLNF method [33], the faces in image were detected. If there were several detected faces in the image, the face with the biggest area (multiplication of height and width of face) and minimum distance from the center of image was extracted as the main face. Thereafter, the main face was cropped out of the image.

As can be seen in the ROC curves of Fig. 17, the proposed method significantly outperforms eye-aligned face recognition with a wonderful enhancement in Yale, AT&T, and LFW dataset. Notice that LFW is a very challenging and big dataset, and Yale and AT&T datasets are two medium well-known datasets. The proposed method results very good in both big and small datasets, showing its power and effectiveness in different types of datasets.

In Cohn-Kanade dataset, however, eye-aligned method performs slightly better than the proposed method; although the ROC curves show that the rates of proposed method is almost near the rate of eye-aligned face recognition. The reason of this failure is that the CLNF method [33], which was used for warping, did not work precisely in detecting very open mouths in extremely surprising faces. Thus, warping could not be performed successfully because of imprecise detected landmarks. Therefore, this failure is not because of the weakness of the proposed method, but because of not having correct and accurate landmarks as input.

VIII. Conclusion, Discussion, and Future Work

A. Summary and Conclusion

In this article, a pixel-level facial alignment method, i.e., a method to align the whole pixels of faces is proposed. This alignment is achieved by mapping the face geometry onto a reference geometry, where the mapping is guided

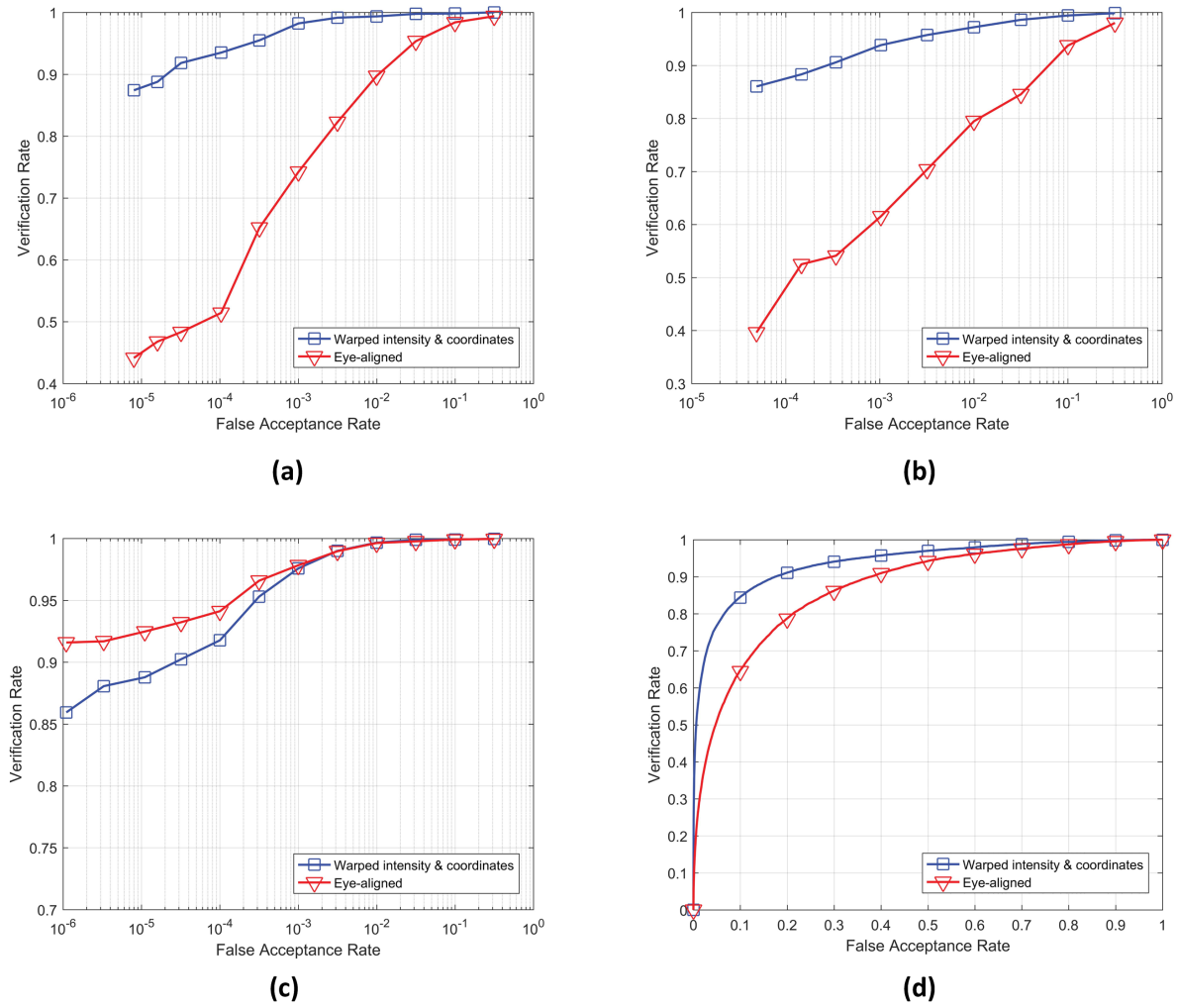

Fig. 17: Comparison of proposed method with eye-aligned face recognition on (a) Yale dataset, (b) AT&T dataset, (c) Cohn-Kanade dataset, (d) LFW dataset.

by contours of facial landmarks which are fitted to each face using landmark detection methods such as CLNF [33]. The resulting aligned intensity and coordinate features create superior recognition results when they are used in a patch-based recognition framework.

The experiments were performed on four well-known datasets and Fisherfaces [20] was used as an instance of a holistic-based face recognition method. Results showed the significantly better performance of patch-based pixel-aligned face recognition in comparison to eye-aligned face recognition in all utilized datasets (except on Cohn-Kanade dataset with slightly rate difference). The reason of not having outperformance in this dataset is that the landmark detection, which is not a contribution of this work, did not work properly in extreme expressions resulting in not qualified warping. The proposed method guarantees better performance in comparison to eye-aligned face recognition when the landmarks are detected properly.

B. Discussions

1) Discussion on Alignment of Features: The most important contribution of this work was to introduce a method for fine alignment of intensity features of the face and at the same time, preserving its geometry information. Moreover, it is important to note that the key finding of this work, i.e., finer alignment results in better classification, is not limited to face recognition problem and is applicable to any other pattern recognition/classification challenge where features are badly corresponded. However, what is required to be solved is to construct an alignment framework, i.e., defining proper correspondences and creating a method for aligning to a reference assortment of features.

2) Discussion on Active Appearance Model and Proposed Warping: One of the contributions of this paper was to introduce a new warping algorithm for faces which can be also used for other applications in future works. Active Appearance Model (AAM) [10], [11] is also another powerful warping method which is widely used for face warping in literature [12], [13], [14], [15]. Algorithm of AAM was briefly explained

in Section I. Here, some differences and similarities of this two methods are mentioned and discussed. Before starting discussion, see Fig. 18 in which the expression is removed using both the proposed warping and AAM method, and a shape-free face is obtained. As can be seen in this figure, the results of the two warping methods are slightly different but they both have removed the expression. In the following, some advantages and disadvantages of AAM method and our method are expressed over each other.

- Both AAM method [11] and the proposed warping guarantee translation of landmarks to exactly the reference (goal) landmarks.
- Different expressions and poses can be generated by both AAM and our warping method. However, AAM method has another ability to create new facial identities by changing facial appearance model parameters [11].
- AAM deals with an optimization problem aiming to minimize the residual error between the projected and previous image patches [11], while our method does not include any optimization and performs warping using coordinate and intensity interpolations.
- AAM method has an iterative manner and converges whenever no improvement is made to residual error [18]. This is while the proposed warping method warps faces in one shot using three interpolations (see Fig. 3). Requiring convergence makes AAM method a little more time consuming than our method, although AAM method is roughly fast enough (see times consumed reported in [11] and [12]). In addition, AAM method also utilizes several Gaussian pyramids for the sake of multi-resolution model. This approach, although increases the precision, makes AAM methd more time consuming as the iterations are performed at each level [18].
- AAM method requires to be trained first using various face images. For instance, it is mentioned in [18] that 10,000 intensity values are sampled for training and generating the appearance model. However, the proposed warping method does not require any train phase. The only thing needed in this method is having the landmarks of input face and also the reference landmarks.
- As mentioned above, our method requires to have landmarks as input; however, this requirement does not exist in AAM method [11] and generation of landmarks and building statistical shape model are performed using Procrustes analysis [16].
- As in AAM method, the texture warping is performed in an eigen-analysis model [11], finding the corresponding pixels in original and warped faces is almost impossible. The proposed warping, however, easily prepares the corresponding pixels because of using interpolations. The corresponding pixels carry important information and having them helps to create valuable feature vectors for the sake of classification, as they are used in this work.

3) Discussion on Classification Using Ensemble of Patches: As it was experimented in Section VII-C2 and shown in Fig. 16, it was observed that for warped faces, classification using ensemble of patches enhances performance in comparison to classification using the whole face. However, this enhancement is not seen for eye-aligned (not warped) faces. This might be because of the fact that in warped faces, every pixel corresponds to a specific region (such as lip corners) in all faces; however, it is not true in eye-aligned images. Therefore, in warped faces, every patch covers similar and corresponding pixels in all faces but may cover not related pixels in eye-aligned ones. That is why using patches has made the result worse in eye-aligned faces, as well as improving result in warped faces.

Fig. 18: Removing expression in original face (figure (a)) using the proposed warping (figure (b)) and AAM warping (figure (c)). Figures (a) and (c) are originally shown in [11].

C. Future Work

In the proposed warping, all faces are warped to a unique neutral face from different expressions and poses. When the geometrical information is obtained using the warped and original faces, three different type of information are included in it, i.e., the face itself, expression and pose. Among these three pieces of information, merely the face itself is important for us because it reflects which pixel has gone where. The other two ones, which are expression and pose, make geometrical information impure because two different expressions or poses of one person result in different geometrical information which is not good. One solution to this problem is not to have only one reference face, but to have one reference face per every expression or pose. This can be performed using regression for every expression or pose with landmarks of non-neutral face as input and landmarks of neutral image as output. We are looking to it as a future work.

REFERENCES

[1] W. Zhao, R. Chellappa, P. J. Phillips, A. Rosenfeld, "Face Recognition: A Literature Survey," *ACM Computing Surveys*, Vol. 35, No. 4, pp. 399-458, 2003.
[2] M. D. Kelly, "Visual identification of people by computer," *Technical report AI-130, Stanford AI Project, Stanford, CA*, 1970.
[3] T. Kanade, "Computer recognition of human faces," *Birkhauser Verlag, Basel, and Stuttgart*, 1977.
[4] R. Brunelli, T Poggio, "Face Recognition: Features versus Templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1042–1052, 1993.
[5] A. V. Nefian, M. H. Hayes, "Hidden Markov models for face recognition," *In Proceedings, International Conference on Acoustics, Speech and Signal Processing*, pp. 2721-2724, 1998.
[6] F. Samaria, S. Young, "HMM based architecture for face identification," *Image and Vision Computing*, vol. 12, pp. 537-543, 1994.
[7] Changxing Ding, Chang Xu, Dacheng Tao, "Multi-task Pose-Invariant Face Recognition," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 980–993, 2015.

[8] A. Pentland, B. Moghaddam, T. Starner, "View-based and modular eigenspaces for face recognition," *In Proceedings, IEEE Conference on Computer Vision and Pattern Recognition, IEEE*, 1994.

[9] P. Penev, J. Atick, "Local feature analysis: A general statistical theory for object representation," *Network: Computation in Neural Systems*, vol. 7, issue 3, pp. 477-500, 1996.

[10] T. F. Cootes, G. J. Edwards, C. J. Taylor, "Active Appearance Models," *In Proceedings, European Conference on Computer Vision*, vol. 2, pp. 484–498, 1998.

[11] T. F. Cootes, G. J. Edwards, C. J. Taylor, "Active Appearance Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Computer Society*, vol. 23, No. 6, pp. 681–685, 2001.

[12] A. Lanitis, C. J. Taylor, T, F. Cootes, "Automatic face identification system using flexible appearance models," *Image and Vision Computing*, vol. 13, no. 5, pp. 393-401, 1995.

[13] Mikkel B. Stegmann, "Analysis and Segmentation of Face Images using Point Annotations and Linear Subspace Techniques," *IMM Technical Report, IMM-REP-2002-22*, 2002.

[14] G. J. Edwards, T. F. Cootes, C. J. Taylor, "Face Recognition Using Active Appearance Models," *Computer Vision – ECCV'98, Springer*, 2006.

[15] Andreas Lanitis, Christopher J. Taylor, Timothy F. Cootes, "A unified approach to coding and interpreting face images," *Computer Vision, 1995. Proceedings., Fifth International Conference on. IEEE*, 1995.

[16] Cootes T. F., Taylor C. J., Cooper D. H., Graham J., "Active shape models – their training and application," *Computer vision and image understanding*, 61(1), pp. 38–59, 1995.

[17] Ian Craw, Peter Cameron, "Face recognition by computer," *BMVC92, Springer London*, pp. 498–507, 1992.

[18] Mathew A. Turk, Alex P. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, pp. 72-86, 1991.

[19] Mathew A. Turk, Alex P. Pentland, "Face recognition using eigenfaces," *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pp. 586–591, 1991.

[20] Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, No. 7, pp. 711–720, 1997.

[21] Juwei Lu, Konstantinos N. Plataniotis, Anastasios N. Venetsanopoulos, "Face Recognition Using Kernel Direct Discriminant Analysis Algorithms," *IEEE Transactions on Neural Networks*, vol. 14, No. 1, pp. 117–126, 2003.

[22] Juwei Lu, Konstantinos N. Plataniotis, Anastasios N. Venetsanopoulos, "Kernel Discriminant Learning with Application to Face Recognition," *Support Vector Machines: Theory and Applications, Book Chapter, Springer*, vol. 117, pp. 275–296, 2005.

[23] R.A. Fisher, "The Use of Multiple Measures in Taxonomic Problems," *Ann. Eugenics*, vol. 7, pp. 179–188, 1936.

[24] P. J. Phillips, "Support vector machines applied to face recognition," *Advances in Neural Information, Processing Systems 11*, pp. 803-809, 1999.

[25] Baback Moghaddam, Tony Jebara, Alex Pentland, "Bayesian face recognition," *Pattern Recognition, Elsevier*, 2000.

[26] P. J. Phillips, "Face Recognition Using Neural Network: A Review," *International Journal of Security and Its Applications*, Vol. 10, No. 3, pp.81–100, 2016.

[27] S. H. Lin, S. Y. Kung, L. J. Lin, "Face recognition/detection by probabilistic decisionbased neural network," *IEEE Transactions on Neural Network*, pp. 114-132, 1997.

[28] Marc Oliu Simon, et al, "Improved RGB-D-T based Face Recognition," *IET Biometrics*, 2016.

[29] Wael AbdAlmageed, et al, "Face Recognition Using Deep Multi-Pose Representations," *Computer Vision and Pattern Recognition, arXiv*, 2016.

[30] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, "Deep Face Recognition," *University of Oxford*, 2015.

[31] Yaniv Taigman, Ming Yang, Marc' Aurelio Ranzato, Lior Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," *2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE*, 2014.

[32] John Wright, Allen Y. Yang, Arvind Ganesh, S. Shankar Sastry, Yi Ma, "Robust Face Recognition via Sparse Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, No. 2, pp. 210–227, 2009.

[33] Tadas Baltrusaitis, Peter Robinson, Louis-Philippe Morency, "Constrained Local Neural Fields for robust facial landmark detection in the wild," *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2013.

[34] D. Cristinacce, T.F. Cootes, "Feature detection and tracking with constrained local models," *British Machine Vision Conference (BMVC)*, vol. 1, no. 2, 2006.

[35] Simon J.D. Prince, "Computer vision: models, learning and inference," *Cambridge University Press*, 2012.

[36] Bernd Gärtner, Michael Hoffmann, "Computational Geometry," *Lecture Notes HS 2013, ETH Zürich University*, chapter 6, pp. 65–81, 2014.

[37] Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning, Data Mining, Inference, and Prediction," *Springer*, Second Edition, 2008.

[38] Christopher M. Bishop, "Pattern Recognition and Machine Learning," *Springer*, 2006.

[39] Vytautas Perlibakas, "Distance measures for PCA-based face recognition," *Pattern Recognition Letters*, vol. 25, no. 6, pp. 711–724, 2004.

[40] Hoda Mohammadzade, Dimitris Hatzinakos, "Projection into Expression Subspaces for Face Recognition from Single Sample per Person," *IEEE Transactions on Affective Computing (TAFFC)*, vol. 4, no. 1, pp. 69–82, 2013.

[41] Yale Face Dataset, http://vision.ucsd.edu/content/yale-face-database.

[42] AT&T Face Dataset, http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[43] Cohn-Kanade Face Dataset, http://www.pitt.edu/ emotion/ck-spread.htm.

[44] Takeo Kanade, Jeffrey F. Cohn, Yingli Tian, "Comprehensive database for facial expression analysis," *In Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pp. 46–53, 2000.

[45] LFW Face Dataset, http://vis-www.cs.umass.edu/lfw/.

[46] Gary B. Huang, Manu Ramesh, Tamara Berg, Erik Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," *Technical Report, University of Massachusetts, Amherst*, vol. 1, no. 2, pp. 7–49, 2007.