Pose-Robust Face Signature for Multi-View Face Recognition

Pengfei Dou, Lingfeng Zhang, Yuhang Wu, Shishir K. Shah, Ioannis A. Kakadiaris
Computational Biomedicine Lab
Department of Computer Science, University of Houston, Houston, TX, USA
{pdou,lzhang34,ywu36,sshah,ioannisk}@uh.edu

Abstract

Despite the great progress achieved in unconstrained face recognition, pose variations still remain a challenging and unsolved practical issue. We propose a novel framework for multi-view face recognition based on extracting and matching pose-robust face signatures from 2D images. Specifically, we propose an efficient method for monocular 3D face reconstruction, which is used to lift the 2D facial appearance to a canonical texture space and estimate the self-occlusion. On the lifted facial texture we then extract various local features, which are further enhanced by the occlusion encodings computed on the self-occlusion mask, resulting in a pose-robust face signature, a novel feature representation of the original 2D facial image. Extensive experiments on two public datasets demonstrate that our method not only simplifies the matching of multi-view 2D facial images by circumventing the requirement for pose-adaptive classifiers, but also achieves superior performance.

1. Introduction

For the past decade, tremendous research effort has focused on face recognition techniques due to their promising potential in biometric authentication. Although great progress has been made, attributed to the expansion in the volume of available training data and use of complex non-linear optimization methods such as deep neural networks, addressing face variations in 2D images still remains a challenging issue. Among those face variations, arbitrary facial pose due to changing viewpoints is probably the most challenging problem. In many real-world applications, the collection of face images is usually dominated by non-frontal faces captured either intentionally, such as selfies for personal albums and/or social networks, or unintentionally, such as surveillance videos. Thus, it is very important for face recognition techniques to have the ability to match non-frontal to frontal facial images, or match two non-frontal facial images.

Existing methods aimed at solving the problem of pose variations in face recognition could be roughly categorized into three groups: robust feature extraction, pose synthesis, and pose normalization. In the first category, massive training data is usually used to learn local features that are discriminative while being consistent over different facial poses. Methods in the second category are usually designed for the face identification task, where a 3D face model is employed to generate virtual face images to synthesize the pose of a query face using the gallery data. In the third category, the problem of pose variations is solved by transforming the original facial image into a new and canonical representation, which is usually a frontal face, converting an unconstrained matching problem into a less challenging constrained matching problem.

In this paper, we propose a novel framework for multi-view face recognition by extracting and matching pose-robust face signatures (**PRFS**) from facial images. The pipeline of our framework is illustrated in Fig. 1. We first use 3D Annotated Face Models (AFM) [13] reconstructed from single 2D images to lift the 2D facial appearance to a canonical representation and estimate self-occlusions. On the canonical face representation, we then extract different local features. Using the estimated self-occlusion mask, the occlusion encodings are computed to further enhance these local features, resulting in a pose-robust representation of the original 2D facial image. The most similar work to our method is by Abiantun *et al*. [1], which also lifts the facial texture to a canonical space and estimates self-occlusions explicitly. However, unlike our method, in Abiantun *et al*. [1] the occluded facial area is recovered via subspace modeling and holistic face features are extracted for face matching.

The major contributions of our method are: (i) We propose an efficient method for 3D AFM reconstruction using single 2D images. The proposed method requires only a small set of 2D landmarks on the image that could be automatically localized. (ii) We propose a novel framework for extracting and matching pose-robust face signatures for face recognition. The framework combines 3D-aided pose

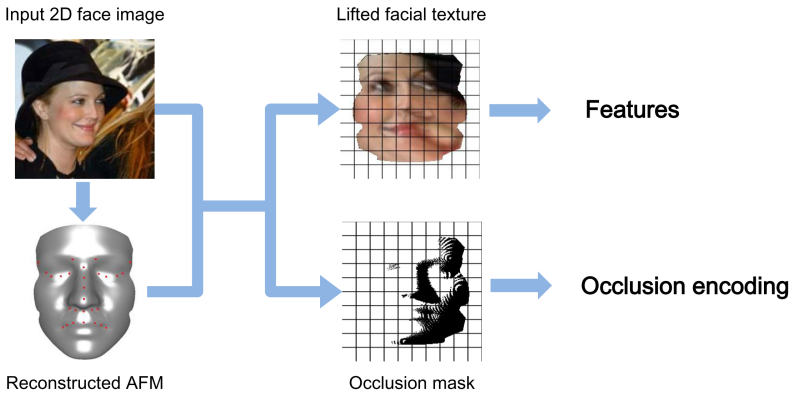

Figure 1. Method overview: Extracting pose-robust face signature from 2D image for face recognition.

normalization and part-based face recognition and employs self-occlusion estimation to further enhance the local features and regularize the feature matching. One of the advantages of our method is that it circumvents the requirement for pose-adaptive classifiers because of the integration of occlusion encodings into PRFS, which makes matching two PRFS's straightforward.

The rest of the paper is organized as follows. Section 2 reviews related work. We present our method in Section 3. Experiments and results are presented in Section 4. Section 5 summarizes our conclusions.

2. Related work

As increasingly high performance is being achieved on popular face recognition benchmarks, such as the Labeled Faces in the Wild (LFW) dataset [12], research emphasis has gradually shifted to solving a practical, yet more challenging problem: pose-robust face recognition. Among the different face variations prevalent in 2D images, facial pose might be the most complex. Facial pose causes not only arbitrary face shape changes, but also self-occlusions and cast shadows, which combined will cause significant changes in facial appearance and greatly increase intra-class variation.

This problem has been addressed primarily through methods that focus on robust feature extraction, pose synthesis, or pose normalization. The methods in the first category rely on deriving a new face representation based on features that are discriminative while being consistent over varying facial poses. Inspired by the success of artificial neural networks in high-level feature learning, Zhang *et al*. [32] proposed a framework for pose-invariant feature learning based on single-hidden-layer neural network (S-NN). Kan *et al*. [14] stacked multiple hidden layers to map non-frontal face to frontal face in a progressive manner, where each layer only slightly corrects the pose variation. Zhu *et al*. [35] proposed a deep neural network with six layers, comprising both deterministic hidden neutrons and random hidden neutrons, for extracting pose invariant features.

Most methods in the second category employ 3D facial data to synthesize novel views of the 2D facial image. They either use a generic 3D model or reconstruct the 3D face using a frontal 2D facial image. Niinuma *et al*. [20] used a reconstructed 3D face model to generate a set of synthetic 2D facial images online that ensemble the pose of the query image. Mostafa *et al*. [18] proposed synthesizing a set of virtual images covering a large pose space offline. During testing, the pose of the probe 2D facial image is first estimated and one of the virtual images that best approximates the probe facial pose is then selected as the gallery to match with the probe. Moeini *et al*. [17] proposed a framework that combines offline synthesis and subspace learning. In summary, they generate a large set of virtual images covering different poses offline. For each specific pose a sparse dictionary matrix (SDM) is created via subspace analysis. During face matching, the identity of the probe face is determined based on the minimum reconstruction error criteria.

Pose synthesis based methods are usually limited to the face identification task. For the face verification tasks, pose normalization is more promising. Abiantun *et al*. [1] used a 3D facial model reconstructed for each query image to transform a non-frontal facial image into a frontal facial image. After self-occlusion estimation, the generated frontal 2D facial image is further projected into a pre-learned subspace and the sparse coefficients are used as features. Similarly, [4, 31, 8] proposed reconstructing the 3D facial model for each probe image, and generated a frontal 2D facial image for matching with the frontal faces in the gallery. However, they used local features for face representation. Chu *et al*. [7] proposed to reconstruct an expressive 3D facial model using 3DMM; thus facial expression exhibited in probe images could be neutralized. Hassner *et al*. [10] proposed using a generic 3D facial model and dedicated post-processing to obtain frontal facial images captured in the wild, achieving promising results on the LFW dataset.

Apart from these categories, additional methods include

pose adaptive filter [28], graph model matching [3], and maximum likelihood correspondence [16]. Due to the limited space, we kindly refer readers to the original papers for more details.

3. Method

3.1. Efficient 3D AFM Reconstruction

Despite the recent progress achieved in 3D scanning techniques, integration of 3D sensors in face recognition systems is still challenging in large deployments due to limited effective sensing range of 3D sensors when compared with 2D cameras. We propose to reconstruct a 3D AFM directly from 2D facial images using only a sparse set of 2D facial landmarks. As presented by Dou *et al*. [9], the 3D AFM reconstruction process can be divided into two steps. In the first step, a sparse 3D model with only 28 vertices is reconstructed, based on which a 3D AFM is reconstructed in the second step. We follow the main idea except that we use a different method to reconstruct the sparse 3D model in this work. The algorithm is explained in the following.

Training: From BU3D-FE [29] and FRGC v2 [21], we select $N = 250$ 3D facial scans of different subjects. By fitting a generic 3D Annotated Face Model (AFM) to each facial scan [13], we obtain N registered 3D AFMs denoted by $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_N] \in \mathbb{R}^{3l_1 \times N}$. Each 3D AFM has $l_1 = 7,597$ vertices. From manual annotation of facial landmarks on each AFM, we obtain N sparse 3D facial models $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N] \in \mathbb{R}^{3l_2 \times N}$. Each sparse 3D model has $l_2 = 28$ vertices. To reduce the dimensionality of $\mathbf{X}$, we apply adaptive local linear embedding (ALLE) [30] and obtain a low-dimensional subspace representation denoted by $\mathbf{C} = [\mathbf{c}_1, \cdots, \mathbf{c}_N] \in \mathbb{R}^{e \times N}$, where e is the dimension of the subspace. For clarity, we use $\mathbb{Y}$, $\mathbb{X}$, and $\mathbb{C}$ to denote the set of training samples from $\mathbf{Y}$, $\mathbf{X}$, and $\mathbf{C}$. The sets $\mathbb{X}$ and $\mathbb{C}$ are coupled because, for each sample in $\mathbb{X}$, we can find its corresponding low-dimensional representation in $\mathbb{C}$, and *vice versa*. Following Dou *et al*. [9], using $\mathbf{Y}$ and $\mathbf{X}$ we learn two dictionaries, $\mathbf{\Lambda}_y \in \mathbb{R}^{3l_1 \times M}$ and $\mathbf{\Lambda}_x \in \mathbb{R}^{3l_2 \times M}$, respectively, where M is the number of atoms. As proposed by Dou *et al*. [9], the K-SVD [2] algorithm is employed to solve Eq. 1. The training process is described in Algorithm 1.

Reconstructing a 3D AFM: Given an input 2D facial image $\mathbf{I}_i$, automatic landmark detection [5] is first applied to extract the locations of 28 facial landmarks $\mathbf{a}_i$. Using Levenberg–Marquardt (LM) least squares minimization the facial pose $\mathbf{P}_i$ is estimated via solving $\mathbf{a}_i = \mathbf{P}_i \cdot \bar{\mathbf{x}}$, where $\bar{\mathbf{x}}$ is the average of the training sparse 3D models. After computing $\mathbf{P}_i$, we project $\mathbf{X}$ into a 2D space, forming N 2D training samples $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_N] \in \mathbb{R}^{2l_2 \times N}$, where $\mathbf{b}_j = \mathbf{P}_i \cdot \mathbf{x}_j$ and $\mathbf{x}_j \in \mathbb{X}$. Similarly, we use $\mathbb{B}$ to denote the set of training samples from $\mathbf{B}$. Following the steps listed in Algorithm 2, a sparse 3D model $\hat{\mathbf{x}}_i$ and a 3D AFM $\hat{\mathbf{y}}_i$ are reconstructed. Compared with Zhang *et al*. [30], where also ALLE was used for monocular 3D face reconstruction, our method uses sparse 3D models as training data and applies explicit pose estimation, thus is capable of obtaining reconstructions from non-frontal facial images.

Algorithm 1 : Training

Input: $\mathbf{X}$, $\mathbf{Y}$

Output: $\mathbf{\Lambda}_x$, $\mathbf{\Lambda}_y$, $\mathbf{C}$

1: Train coupled dictionaries $\mathbf{\Lambda}_x$ and $\mathbf{\Lambda}_y$ by solving:

$$\underset{\mathbf{\Psi}, \mathbf{\Lambda}_x, \mathbf{\Lambda}_y}{\arg\min} \left\| \begin{bmatrix} \beta_0 \mathbf{Y} \\ \mathbf{X} \end{bmatrix} - \begin{bmatrix} \beta_0 \mathbf{\Lambda}_y \\ \mathbf{\Lambda}_x \end{bmatrix} \mathbf{\Psi} \right\|_2^2 \; s.t. \; \|\mathbf{\Psi}\|_1 \leq \beta_1 \;, \tag{1}$$

where $\mathbf{\Psi}$ denotes the coding coefficients, β_0 balances two types of training data, and β_1 controls the sparsity of the solution.

2: Compute a low-dimensional representation of $\mathbf{X}$ via ALLE [22, 30]: $\mathbf{C} = ALLE(\mathbf{X}, e)$, where e is the dimension of the subspace and $\mathbb{C}$ is the set of low-dimensional representations corresponding to matrix $\mathbf{C}$.

3.2. Texture Lifting and Self-Occlusion Estimation

By using the reconstructed 3D AFM, we aim to derive a new representation of the 2D facial image, where pose variations in the original image can be normalized. Though several recent works have shown that pose variations in face recognition could be partially solved by using synthetic facial images, we do not follow this approach for two reasons. First, the reconstructed 3D model is only a coarse approximation of the real face. Second, the rendering process is time consuming, introducing additional overhead. Instead, in our method, the reconstructed 3D AFM is used to lift the 2D facial appearance [26] to a canonical 2D space (geometry image space [13]) following the procedures described below.

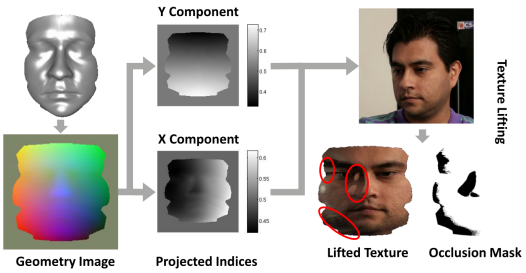

Figure 2. The process of texture lifting.

Using $\hat{\mathbf{x}}_i$ obtained in Sec. 3.1, we first re-estimate the facial pose $\mathbf{P}_i$ of $\mathbf{I}_i$. Then, we compute the geometry image

Algorithm 2 : Reconstruct A Sparse 3D Model and A 3D AFM

Input: $\mathbf{a}_i$, $\mathbb{B}$, $\mathbb{C}$, $\mathbb{X}$, $\mathbf{\Lambda}_x$, $\mathbf{\Lambda}_y$
Output: $\hat{\mathbf{x}}_i$, $\hat{\mathbf{y}}_i$

1: Select the K nearest neighbors of $\mathbf{a}_i$ in $\mathbb{B}$ (K is determined using the method described by Zhang *et al*. [30]) and their indices are: $\mathbb{U}^i = \{u_1^i, \cdots, u_K^i\}$.
2: Compute $\mathbb{W}^i = \{w_j^i | j \in \mathbb{U}^i\}$ by solving:

$$\arg\min_{\mathbb{W}^i} \|\mathbf{a}_i - \sum_{j \in \mathbb{U}^i} w_j^i \cdot \mathbf{b}_j\|_2^2 \quad s.t. \sum_{j \in \mathbb{U}^i} w_j^i = 1 \ .$$

3: From $\mathbb{C}$ select the samples with the same index as the elements of $\mathbb{U}^i$ and form: $\mathbb{\Theta}^i = \{\mathbf{c}_j | j \in \mathbb{U}^i, \mathbf{c}_j \in \mathbb{C}\}$.
4: Compute the low-dimensional representation $\hat{\mathbf{c}}_i$ corresponding to $\mathbf{a}_i$: $\hat{\mathbf{c}}_i = \sum_{j \in \mathbb{U}^i} w_j^i \cdot \mathbf{c}_j$.
5: Select the K nearest neighbors of $\hat{\mathbf{c}}_i$ in $\mathbb{C}$ (K is determined using the method described by Zhang *et al*. [30]) and their indices are: $\mathbb{V}^i = \{v_1^i, \cdots, v_K^i\}$.
6: Compute $\mathbb{D}^i = \{d_j^i | j \in \mathbb{V}^i\}$ by solving:

$$\arg\min_{\mathbb{D}^i} \|\hat{\mathbf{c}}_i - \sum_{j \in \mathbb{V}^i} d_j^i \cdot \mathbf{c}_j\|_2^2 \quad s.t. \sum_{j \in \mathbb{V}^i} d_j^i = 1 \ .$$

7: From $\mathbb{X}$ select the samples with the same index as the elements of $\mathbb{V}^i$ and form: $\mathbb{\Phi}^i = \{\mathbf{x}_j | j \in \mathbb{V}^i, \mathbf{x}_j \in \mathbb{X}\}$.
8: Reconstruct the sparse 3D model corresponding to $\mathbf{a}_i$: $\hat{\mathbf{x}}_i = \sum_{j \in \mathbb{V}^i} d_j^i \cdot \mathbf{x}_j$.
9: Compute the coefficients $\hat{\boldsymbol{\psi}}_i$ of the reconstructed sparse 3D model $\hat{\mathbf{x}}_i$ with dictionary $\mathbf{\Lambda}_x$:

$$\arg\min_{\hat{\boldsymbol{\psi}}_i} \left\| \hat{\mathbf{x}}_i - \mathbf{\Lambda}_x \hat{\boldsymbol{\psi}}_i \right\|_2^2 + \beta_2 \left\| \hat{\boldsymbol{\psi}}_i \right\|_1 + \beta_3 \left\| \hat{\boldsymbol{\psi}}_i \right\|_2 , \tag{2}$$

where β_2 and β_1 are regularization parameters. The lasso algorithm [24] is used to solve Eq. 2.
10: Reconstruct a 3D AFM $\hat{\mathbf{y}}_i$ using dictionary $\mathbf{\Lambda}_y$ and the same coefficients $\hat{\boldsymbol{\psi}}_i$: $\hat{\mathbf{y}}_i = \frac{\mathbf{\Lambda}_y \hat{\boldsymbol{\psi}}_i}{\beta_0}$.

$\mathbf{G}_i$ of the 3D AFM $\hat{\mathbf{y}}_i$ [23]. In this geometry image, as illustrated in Fig. 2, each pixel captures the information of an existing or interpolated vertex on the 3D AFM surface. Based on $\mathbf{G}_i$, a set of 2D coordinates pointing to the pixels on the original 2D facial image can be computed by $\mathbf{Q}_i = \mathbf{P}_i \cdot \mathbf{G}_i$. Facial appearance is then lifted into the UV space, resulting in a new representation of the facial texture $\mathbf{T}_i$. One advantage of using such a representation is that dense semantic correspondence can be naturally established via pose normalization.

As shown in Fig. 2, marked by red ellipses, arbitrary facial pose results in self-occlusion of certain facial parts and creates artifacts on the lifted facial texture. To estimate self-occlusions and locate these artifacts, we use the 3D AFM and employ Z-buffer technique to estimate for each pixel on the lifted texture its occlusion status, indicating whether that pixel is occluded or not [13]. This process generates an occlusion mask $\mathbf{Z}_i$ of the same size as $\mathbf{T}_i$, where each pixel has a binary value, with 0 indicating occlusion and 1 indicating non-occlusion. The lifted texture and the occlusion mask form a new representation of the original 2D image, $\mathbb{I}_i = \{\mathbf{T}_i, \mathbf{Z}_i\}$.

3.3. Pose-Robust Feature Extraction and Matching

Inspired by the success and superior performance of part-based methods in face recognition over its holistic counterparts, we propose to employ local features and extract part-based face representations from $\mathbb{I}_i$. Specifically, we first divide the facial texture $\mathbf{T}_i$ into m non-overlapping patches. For each patch, we extract dense local features $\mathbf{f}_i^k$, such as LBP, Gabor responses, or SIFT. From the occlusion mask $\mathbf{Z}_i$, an occlusion encoding o_i^k is extracted for each local patch by Eq. 3:

$$o_i^k = \begin{cases} 1 & \text{if } \frac{\#(\mathbf{Z}_i(,)=0)}{\#(\mathbf{Z}_i(,))} < \delta \\ 0 & \text{otherwise} \end{cases} , \tag{3}$$

where $\#(\mathbf{Z}_i(,) = 0)$ and $\#(\mathbf{Z}_i(,))$ denote the number of occluded pixels and the total number of pixels in the k^{th} patch and δ is a threshold defined empirically.

After feature extraction, each facial image is then represented as an ensemble of local features enhanced by occlusion encodings $\mathbb{H}_i = \{\boldsymbol{\mathcal{F}}_i, \boldsymbol{\mathcal{O}}_i\}$, where $\boldsymbol{\mathcal{F}}_i = [\mathbf{f}_i^1, \cdots, \mathbf{f}_i^m]$ and $\boldsymbol{\mathcal{O}}_i = [o_i^1, \cdots, o_i^m]$. We call this part-based representation Pose Robust Face Signature (**PRFS**) because: (i) it is a generic face representation without regarding for pose variations; (ii) PRFS could be extracted offline and stored independently of the original 2D facial image; (iii) it explicitly encodes the pose and the associated self-occlusion information in the feature space, thus is pose-aware; and (iv) matching two faces represented in PRFS is straightforward and pose-robust as two signatures are intrinsically aligned.

For a pair of facial images represented by their PRFS's $\{\mathbb{H}_i, \mathbb{H}_j\}$, the similarity score is computed as the summation of similarities over all local patches. Specially, to make this feature matching process robust to pose variations, occlusion encodings of both $\mathbb{H}_i$ and $\mathbb{H}_j$ are integrated into similarity score computation, as shown in Eq. 4:

$$s_{ij} = s(\mathbb{H}_i, \mathbb{H}_j) = \frac{1}{\sum_{l=1}^m o_i^l \cdot o_j^l} \cdot \sum_{l=1}^m o_i^l \cdot o_j^l \cdot k(\mathbf{f}_i^l, \mathbf{f}_j^l) \ , \tag{4}$$

where $k : \mathbb{R}^d \rightarrow \mathbb{R}$ is the metric function. The two faces are classified as the same identity if their similarity (or dissimilarity) s_{ij} is larger (or smaller) than a threshold τ.

4. Experiments and results

In this section, we evaluate our method on two face datasets: the UHDB11 [25] and the Labeled Faces in the Wild (LFW) [12], in both face identification and face verification settings. UHDB11 is used to evaluate the performance of our method in a controlled environment. LFW is further used to evaluate our method in an unconstrained environment.

4.1. UHDB11

UHDB11 [25] comprises of data from 23 subjects with both their 3D facial scans and 2D images captured in a controlled environment. The 2D images are acquired under six illumination conditions and different head poses, specifically around $[-30°, +30°]$ in yaw rotation and $[-50°, +50°]$ in roll rotation. The protocol of the UHDB11 dataset defines a face identification task, where only one image with frontal face for each subject is collected in the gallery set.

Table 1. Rank-1 Accuracy (%) on the UHDB11 Dataset.

Method	**Acc. (%)**
LIRIS [34]	80.2
UR2D [34]	85.2
UR2D+unlighting [33]	86.7
Progressive Pose Estimation [31]	88.8
PRFS	**94.1**
PRFS without occlusion encoding	91.1

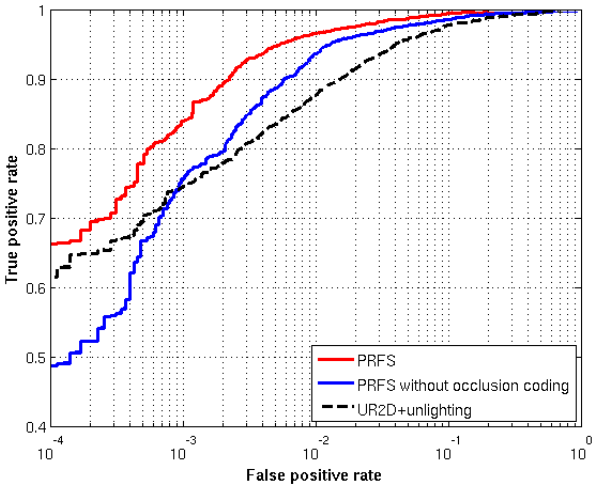

Figure 3. ROC curves for the UHDB11 dataset.

Parameters and settings: In our experiment, we first apply automatic landmark detection [5] and extract 28 facial landmarks on each facial image $\mathbf{I}_j$ in the gallery set. Then, a 3D AFM is reconstructed accordingly using our method detailed in Sec. 3.1. The parameters are empirically selected as $e = 20$, $\beta_0 = 0.15$, $\beta_1 = 20$, $\beta_2 = 0.3$, $\beta_3 = 0.7$, $M = 249$, and $\delta = 0.35$, which are used throughout our experiments. Using the 3D AFM facial texture $\mathbf{T}_j$ of each gallery 2D facial image is lifted, $\mathbf{Z}_j$ is estimated, and PRFS $\mathbb{H}_j$ is extracted, where $j \in \{1, 2, \cdots, 23\}$. Following the same approach, PRFS of probe facial image $\mathbf{I}_i$ is also extracted using the 3D AFM of each gallery subject. Similar to [34, 33], we compute the similarity score of each probe-gallery pair $\{\mathbb{H}_j, \mathbb{H}_i\}$ using Eq. 4, where cosine similarity is used as the metric function. From all the similarity scores $\{s_{ij}\}$, where $j \in \{1, 2, \cdots, 23\}$, we choose the maximum value and assign the identity of the corresponding gallery subject to the probe.

Experiment 1: The objective of this experiment is to compare the performance of proposed method with other approaches on UHDB11. The facial texture is set to be 200×200 and divided into 8×8 non-overlapping blocks. On each block, we extract a discriminative face descriptor (DFD) proposed by Lei *et al*. [15], which is trained on 907 images of 109 subjects selected from the FRGC v2 dataset [21]. Experimental results listed in Table 1 show that our method achieves the best Rank-1 accuracy of $\mathbf{94.1}\%$, which is superior to previous methods by a large margin. To evaluate the importance of occlusion encoding, we run another experiment without using this information. As expected, the performance degrades significantly, with Rank-1 accuracy of only 91.1%. The ROCs are plotted in Fig. 3, indicating the superior performance of our proposed method on the UHDB11 dataset.

Table 2. Rank-1 Accuracy (%) on the UHDB11 Dataset with MSLBP feature.

Method	**Acc. (%)**
PRFS	88.2
PRFS + WPCA	**92.8**

Experiment 2: The objective of this experiment is to compare the performance of proposed method on UHDB11 using different local features. Our method is flexible to integrate different local features. Besides learning based features such as DFD, we choose to use the multi-scale LBP (MSLBP) feature, a variant of the widely used LBP features in face recognition. Specifically, we first divide the facial texture into 5×5 non-overlapping blocks and compute the corresponding occlusion encodings according to Eq. 3. Within each block, we further divide it into 2×2 and 4×4 sub-blocks, which form a block-pyramid. On the bottom-

level block, we extract uniform LBP with $s_1 = 8$ neighbors and radius $r_1 = 1$ and compute the histogram. Similarly, we set $r_2 = 2$, $s_2 = 16$ on the middle-level block and $r_3 = 3$, $s_3 = 24$ on the top-level block. By concatenating the histograms of all 21 blocks in the same pyramid, we obtain the local feature vector. Occlusion encoding is computed only on the top-level block. As shown in Table 2, using MSLBP feature our method achieves 88.2% Rank-1 identification accuracy, which is higher than previous results of [34, 33]. Although Zhang *et al*. [31] reported a Rank-1 accuracy of 88.8%, they used only a subset of the dataset with neutral lighting, which is less challenging than our experiment. As the feature vector extracted with our method is very high-dimensional, we further applied whitened PCA (WPCA) to reduce the dimensionality. We use the same images from FRGC v2 as used in Experiment 1 to learn the PCA for each of the 5×5 blocks and keep 95% of the energy. This improves the Rank-1 accuracy to 92.8%, which is slightly lower than the performance obtained using the DFD feature.

4.2. LFW

The LFW dataset comprises a large set of facial images collected from the Internet. In total there are 13,233 images of 5,749 different subjects covering many real-world variations including illumination, facial pose, facial expression, occlusion, and resolution. The dataset is split into two views, and View 2 that contains 3,000 matched and 3,000 mismatched pairs is used to test face verification performance. In our experiment, we evaluate the performance of our method following the unsupervised protocol where no labeled training data is used during the evaluation.

Parameters and settings: In this experiment, for each pair of facial images in LFW we reconstruct two 3D AFMs and lift the facial textures and estimate the occlusion masks separately. Due to the low resolution of the face images, we set the size of the facial texture to be 80×80 throughout the experiment. Similarly, we divide the facial texture into 10×10 non-overlapping blocks, extract the PRFS's, and compute the similarity score for the pair. In our experiment, we use several different local features, namely LBP, MSLBP, Gabor wavelets (GW), three patch LBP (TPLBP), and four patch LBP (FPLBP) [27]. For LBP, we set the number of neighbors to be 8 and the radius to be 1. For multi-scale LBP, we compute for each block four LBP histograms with radius increasing from one to four. For TPLBP and FPLBP, we use the parameters reported in [27], except that the radius is set to be 1 for TPLBP and $\{1, 2\}$ for FPLBP. For Gabor wavelets, we divide the facial texture into 4×4 non-overlapping blocks and use 40 Gabor filters covering 5 scales and 8 orientations to extract dense local features at a downsampling rate of 2. To reduce the feature dimension, we also apply WPCA to extract compact features. The PCA subspace model is learned using the unlabeled training data provided in View 1.

Experiment 3: The objective of this experiment is to show how much the face verification performance could be improved by using our method. For comparison we extract the same features on the deep-funneled LFW images [11] with parameters empirically selected to obtain the best results. The best performance is achieved on deep-funneled facial images cropped to 80×150, which coincides with several previous works [19, 6]. We divide the cropped 2D face into 8×15 non-overlapping blocks to extract LBP, MSLBP, TPLBP, and FPLBP features, and 4×6 non-overlapping blocks to extract Gabor wavelets.

Table 4. Mean accuracy and standard error (%) on LFW in an unsupervised setting.

Descriptor	**Deep-Funneled [11]**	**PRFS**
LBP+WPCA	76.0±0.5	78.6±0.4
MSLBP+WPCA	77.0±0.4	79.0±0.4
Gabor+WPCA	72.7±0.6	78.3±0.5
TPLBP+WPCA	73.0±0.5	76.5±0.8
FPLBP+WPCA	73.4±0.4	76.7±0.6
Fusion	77.4±0.4	**80.6±0.4**

For each pair of facial images, we compute their similarity score according to Eq. 4. Using cross validation, we choose the score at Equal Error Rate on the training set as threshold τ and report the face verification accuracy on the testing set. Table 4 summarizes our experimental results, demonstrating that our method can constantly improve face verification performance over its counterparts using the deep-funneled 2D facial images. To further improve the face verification accuracy, we apply majority voting to

Table 3. Comparison with different alignment methods in terms of mean accuracy (ACC) (%) and area under the ROC curve (AUC).

Descriptor	**Deep-Funneled [11]**				**LFW3D [10]**				**PRFS**			
	L_2		Hellinger		L_2		Hellinger		L_2		Hellinger	
	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
LBP	66.1	0.72	69.6	0.76	**74.7**	**0.80**	73.2	0.79	72.3	0.79	72.2	0.79
TPLBP	65.8	0.72	64.3	0.71	**75.0**	**0.81**	67.2	0.72	70.4	0.77	68.6	0.75
FPLBP	66.5	0.73	66.4	0.72	72.7	0.80	**73.5**	**0.81**	71.8	0.79	71.5	0.79

the prediction results achieved with different local features. As shown in Table 4, fusing different prediction results is effective and achieves better accuracy. Though further improvements might be achieved via supervised learning approaches such as metric learning, this will be future work.

Figure 4. Mean facial appearances obtained by averaging multiple images of four subjects from LFW. Top Row: Deep-Funneled [11], Middle Row: LFW3D [10], and Bottom Row: PRFS.

Experiment 4: A recent work that employs a generic 3D facial model to "frontalize" non-frontal facial images is presented by [10] (LFW3D), where they apply 3D-aided pose normalization, self-occlusion estimation, and dedicated post-processing to fill the missing facial parts to derive frontal faces from non-frontal facial images. Figure 4 depicts the mean faces of four subjects obtained by averaging all facial images provided in the deep-funneled images, the frontalized images of [10], and the facial textures generated by our method. We can observe that, compared with the other two methods, fine details around the eyes, nose, and mouth are preserved in the facial textures, indicating the superiority of our method for alignment. Table 3 presents the face verification accuracy and the area under ROC (AUC) of these three methods using two distance metrics [10], namely L_2 distance and Hellinger distance. The face verification accuracies of both [10] and our method outperform that achieved on the deep-funneled face images and [10] performs slightly better than our method.

5. Conclusions

To handle pose variations and the associated self-occlusion problem in face recognition, different strategies have been exploited in the literature. Different from existing works that either try to fill the occluded regions using facial symmetry or subspace modeling or simply discard the occluded facial parts, resulting in inconsistent feature representations, we present in this paper a novel framework for extracting and matching pose robust face signatures, a consistent and pose-aware feature representation of multi-view 2D faces. Matching two faces is straightforward and simplified using pose robust face signatures, circumventing the requirement for pose-adaptive classifiers. As demonstrated in our experiments, better performance is also achieved using our new face representation.

6. Acknowledgment

This research was funded in part by the US Army Research Lab (W911NF-13-1-0127) and the UH Hugh Roy and Lillie Cranz Cullen Endowment Fund. All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies of the sponsors.

References

[1] R. Abiantun, U. Prabhu, and M. Savvides. Sparse feature extraction for pose-tolerant face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(10):2061–2073, 2014.

[2] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4321, 2006.

[3] S. Arashloo and J. Kittler. Efficient processing of MRFs for unconstrained-pose face recognition. In *Proc. IEEE International Conference on Biometrics: Theory, Applications and Systems*, pages 1–8, Arlington, VA, Sep. 29-Oct. 2 2013.

[4] A. Asthana, T. K. Marks, M. J. Jones, K. H. Tieu, and M. Rohith. Fully automatic pose-invariant face recognition via 3D pose normalization. In *Proc. IEEE International Conference on Computer Vision*, pages 937–944, Barcelona, Spain, Nov. 6-13 2011.

[5] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Incremental face alignment in the wild. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1859 – 1866, Columbus, OH, June 24-27 2014.

[6] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz. Fast high dimensional vector multiplication face recognition. In *Proc. IEEE International Conference on Computer Vision*, pages 1960–1967, Sydney, Dec. 1-8 2013.

[7] B. Chu, S. Romdhani, and L. Chen. 3D-aided face recognition robust to expression and pose variations. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1907–1914, Columbus, OH, June 23-28 2014.

[8] L. Ding, X. Ding, and C. Fang. Continuous pose normalization for pose-robust face recognition. *IEEE Signal Processing Letters*, 19(11):721–724, Nov. 2012.

[9] P. Dou, Y. Wu, S. K. Shah, and I. A. Kakadiaris. Robust 3D face shape reconstruction from single images via two-fold coupled structure learning. In *Proc. British Machine Vi-*

sion Conference, pages 1–13, Nottingham, United Kingdom, September 1-5 2014.

[10] T. Hassner, S. Harel, E. Paz, and R. Enbar. Effective face frontalization in unconstrained images. *arXiv preprint arXiv:1411.7964*, pages 1–10, 2014.

[11] G. Huang, M. Mattar, H. Lee, and E. G. Learned-miller. Learning to align from scratch. In *Advances in Neural Information Processing Systems 25*, pages 764–772. 2012.

[12] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A database for studying face recognition in unconstrained environments. Technical Report 07–49, University of Massachusetts, Amherst, MA, Oct. 2007.

[13] I. A. Kakadiaris, G. Passalis, G. Toderici, M. N. Murtuza, Y. Lu, N. Karampatziakis, and T. Theoharis. Three-dimensional face recognition in the presence of facial expressions: An annotated deformable model approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):640–649, 2007.

[14] M. Kan, S. Shan, H. Chang, and X. Chen. Stacked progressive auto-encoders (SPAE) for face recognition across poses. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1883 – 1890, Ohio, Columbia, June 23-28 2014.

[15] Z. Lei, M. Pietikainen, and S. Li. Learning discriminant face descriptor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(2):289–302, Feb. 2014.

[16] S. Li, X. Liu, X. Chai, H. Zhang, S. Lao, and S. Shan. Maximal likelihood correspondence estimation for face recognition across pose. *IEEE Transactions on Image Processing*, 23(10):4587–4600, Oct. 2014.

[17] A. Moeini, H. Moeini, and K. Faez. Unrestricted pose-invariant face recognition by sparse dictionary matrix. *Image and Vision Computing*, 36:9 – 22, 2015.

[18] E. Mostafa, A. Ali, N. Alajlan, and A. Farag. Pose invariant approach for face recognition at distance. In *Proc. European Conference on Computer Vision*, pages 15–28, Firenze, Italy, Oct. 7-13 2012.

[19] H. V. Nguyen and L. Bai. Cosine similarity metric learning for face verification. In *Proc. Asian Conference on Computer Vision*, pages 709–720, New Zealand, Nov. 8-12 2011.

[20] K. Niinuma, H. Han, and A. Jain. Automatic multi-view face recognition via 3D model based pose regularization. In *Proc. IEEE International Conference on Biometrics: Theory, Applications and Systems*, pages 1–8, Arlington, VA, Sep. 29-Oct. 3 2013.

[21] P. J. Phillips, W. T. Scruggs, A. J. O'Toole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe. FRVT 2006 and ICE 2006 large-scale experimental results. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(5):831–846, May 2010.

[22] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[23] T. Theoharis, G. Passalis, G. Toderici, and I. A. Kakadiaris. Unified 3D face and ear recognition using wavelets on geometry images. *Pattern Recognition*, 41(3):796–804, 2008.

[24] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[25] G. Toderici, G. Evangelopoulos, T. Fang, T. Theoharis, and I. A. Kakadiaris. UHDB11 database for 3D-2D face recognition. In *Proc. 6^{th} Pacific-Rim Symposium on Image and Video Technology*, pages 73–86, Guanajuato, Mexico, Oct. 28–Nov. 1 2013.

[26] G. Toderici, G. Passalis, S. Zafeiriou, G. Tzimiropoulos, M. Petrou, T. Theoharis, and I. A. Kakadiaris. Bidirectional relighting for 3D-aided 2D face recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 2721–2728, San Francisco, CA, June 13-18 2010.

[27] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *Proc. Real-Life Images workshop at the European Conference on Computer Vision*, pages 1–14, Marseille, France, Oct. 12-18 2008.

[28] D. Yi, Z. Lei, and Z. Li. Towards pose robust face recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 3539–3545, Portland, Oregon, June 25-27 2013.

[29] L. Yin, X. Wei, Y. Sun, J. Wang, and M. Rosato. A 3D facial expression database for facial behavior research. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, pages 211–216, Southampton, UK, Apr. 10-12 2006.

[30] J. Zhang and Y. Zhuang. Sample based 3D face reconstruction from a single frontal image by adaptive locally linear embedding. *Journal of Zhejiang University SCIENCE A*, 8(4):550–558, 2007.

[31] W. Zhang, D. Huang, D. Samaras, J.-M. Morvan, Y. Wang, and L. Chen. 3D assisted face recognition via progressive pose estimation. In *Proc. IEEE International Conference on Image Processing*, pages 728–732, Paris, Oct. 27-30 2014.

[32] Y. Zhang, M. Shao, E. Wong, and Y. Fu. Random faces guided sparse many-to-one encoder for pose-invariant face recognition. In *Proc. IEEE International Conference on Computer Vision*, pages 2416–2423, Sydney, Australia, Dec. 1-8 2013.

[33] X. Zhao, D. Chu, G. Evangelopoulos, S. K. Shah, and I. A. Kakadiaris. UHAE: Minimizing illumination differences for 3D to 2D face recognition using lighting maps. *IEEE Transactions on Cybernetics*, 44(5):725–736, 2013.

[34] X. Zhao, W. Zhang, G. Evangelopoulos, D. Huang, S. K. Shah, Y. Wang, I. A. Kakadiaris, and L. Chen. Benchmarking asymmetric 3D-2D face recognition systems. In *Proc. 10^{th} International Conference on Automatic Face and Gesture Recognition*, pages 1–8, Shanghai, China, April 22–26 2013.

[35] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning multi-view representation for face recognition. *arXiv preprint arXiv:1406.6947*, pages 1–10, 2014.

Table 5. Notations

Notation	Type	Explanations
3D Reconstruction		
N	Scalar	Total number of training 3D faces
l_1	Scalar	Total number of vertices on 3D AFM
l_2	Scalar	Total number of facial landmarks (vertices on the sparse 3D model)
$\mathbf{Y}$	Matrix	N training 3D AFMs
$\mathbb{Y}$	Set	Collection of training 3D AFMs
$\mathbf{X}$	Matrix	N training sparse 3D models
$\mathbb{X}$	Set	Collection of training sparse 3D models
$\mathbf{y}_j$	Vector	The j^{th} training AFM
$\mathbf{x}_j$	Vector	The j^{th} training sparse 3D model
$\bar{\mathbf{x}}$	Vector	The average of training sparse 3D models
$\hat{\mathbf{x}}_i$	Vector	Reconstructed sparse 3D model of the input 2D image $\mathbf{I}_i$
$\mathbf{\Psi}$	Matrix	Sparse coding coefficients of $\mathbf{Y}$ and $\mathbf{X}$ during coupled dictionary learning
$\hat{\boldsymbol{\psi}}_i$	Vector	Sparse coefficient of the reconstructed sparse 3D model
$\mathbf{P}_i$	Matrix	3D-2D projection matrix of 2D facial image $\mathbf{I}_i$
$\mathbf{a}_i$	Vector	2D landmarks on the input 2D image $\mathbf{I}_i$
$\mathbf{b}_j$	Vector	Projected 2D landmarks of the j^{th} training sparse 3D model $\mathbf{x}_j$
$\mathbf{B}$	Matrix	N 2D training shapes projected from $\mathbf{X}$ with $\mathbf{P}_i$
$\mathbb{B}$	Set	Collection of 2D training shapes
$\mathbf{C}$	Matrix	Low-dimensional subspace model learned from $\mathbf{X}$
$\mathbb{C}$	Set	Collection of low-dimensional representations of $\mathbf{X}$
e	Scalar	Dimension of the subspace model $\mathbf{C}$
$\mathbf{\Lambda}_x$	Matrix	Trained dictionary of training sparse 3D models $\mathbf{X}$
$\mathbf{\Lambda}_y$	Matrix	Trained dictionary of training 3D AFMs $\mathbf{Y}$
M	Scalar	Number of atoms in the trained dictionaries
$\hat{\mathbf{c}}_i$	Vector	The low dimensional representation of $\hat{\mathbf{x}}_i$ in $\mathbf{C}$
$\beta_0, \beta_1, \beta_2, \beta_3, M$	Scalars	Parameters used in dictionary learning and sparse coding
K	scalar	Number of nearest neighbors adaptively determined
$\mathbb{W}^i$	Set	Weights of the K nearest neighbors of $\mathbf{a}_i$ in $\mathbb{B}$
$\mathbb{D}^i$	Set	Weights of the K nearest neighbors of $\hat{\mathbf{c}}_i$ in $\mathbb{C}$
$\mathbb{U}^i$	Set	Indices of the K nearest neighbors of $\mathbf{a}_i$ selected from $\mathbb{B}$
u_j^i	Scalar	$u_j^i \in \mathbb{U}^i$
$\mathbb{\Theta}^i$	Set	The K samples selected from $\mathbb{C}$ using the same indices as the elements of $\mathbb{U}^i$
$\mathbb{V}^i$	Set	Indices of the K nearest neighbors of $\hat{\mathbf{c}}_i$ selected from $\mathbb{C}$
v_j^i	scalar	$v_j^i \in \mathbb{V}^i$
$\mathbb{\Phi}^i$	Set	The K samples selected from $\mathbb{X}$ using the same indices as the elements of $\mathbb{V}^i$
Texture Lifting		
$\mathbf{I}_i$	Matrix	Input 2D facial image
$\mathbf{G}_i$	Matrix	Geometry map of reconstructed dense 3D facial model
$\mathbf{Q}_i$	Matrix	Projection of geometry map into 2D image space
$\mathbf{T}_i$	Matrix	Lifted facial texture
$\mathbf{Z}_i$	Matrix	Estimated occlusion mask
$\mathbb{I}_i$	Set	Representation of the i^{th} input 2D facial image $\{\mathbf{T}_i, \mathbf{Z}_i\}$
Local Features		
$\mathbf{f}_i^k$	Vector	Local feature vector extracted at the k^{th} local patch of $\mathbf{T}_i$
δ	Scalar	Threshold used in computing occlusion encoding
o_i^k	Scalar	Occlusion encoding of the k^{th} local patch of $\mathbf{Z}_i$
m	Scalar	Total number of non-overlapping local patches
$\boldsymbol{\mathcal{F}}_i$	Vector	Concatenation of local feature vectors
$\boldsymbol{\mathcal{O}}_i$	Vector	Concatenation of local occlusion encodings
$\mathbb{H}_i$	Set	Part-based face representation $\{\boldsymbol{\mathcal{F}}_i, \boldsymbol{\mathcal{O}}_i\}$
τ	Scalar	Threshold for similarity score
r	Scalar	Radius of neighborhood in computing LBP features
s, s_1, s_2, s_3	Scalars	Total number of heighbors in computing LBP features