Pairwise Relational Networks for Face Recognition

Bong-Nam Kang[1][0000−0002−6818−7532], Yonghyun Kim[2][0000−0003−0038−7850], and Daijin Kim[1,2][0000−0002−8046−8521]

[1] Department of Creative IT Engineering, POSTECH, Korea
[2] Department of Computer Science and Engineering, POSTECH, Korea
{bnkang,gkyh0805,dkim}@postech.ac.kr

Abstract. Existing face recognition using deep neural networks is difficult to know what kind of features are used to discriminate the identities of face images clearly. To investigate the effective features for face recognition, we propose a novel face recognition method, called a pairwise relational network (PRN), that obtains local appearance patches around landmark points on the feature map, and captures the pairwise relation between a pair of local appearance patches. The PRN is trained to capture unique and discriminative pairwise relations among different identities. Because the existence and meaning of pairwise relations should be identity dependent, we add a face identity state feature, which obtains from the long short-term memory (LSTM) units network with the sequential local appearance patches on the feature maps, to the PRN. To further improve accuracy of face recognition, we combined the global appearance representation with the pairwise relational feature. Experimental results on the LFW show that the PRN using only pairwise relations achieved 99.65% accuracy and the PRN using both pairwise relations and face identity state feature achieved 99.76% accuracy. On the YTF, both the PRN using only pairwise relations and the PRN using pairwise relations and the face identity state feature achieved the *state-of-the-art* (95.7% and 96.3%). The PRN also achieved comparable results to the *state-of-the-art* for both face verification and face identification tasks on the IJB-A, and the *state-of-the-art* on the IJB-B.

Keywords: Pairwise Relational Network · Relations · Face Recognition

1 Introduction

Convolutional neural networks (CNNs) have achieved great success on computer vision community by improving the *state-of-the-art* in almost all of applications, especially in classification problems including object [12–14, 20, 22, 29, 33] scene [43, 44], and so on. The key to success of CNNs is the availability of large scale of training data and the end-to-end learning framework. The most commonly used CNNs perform feature learning and prediction of label information by mapping the input raw data to deep embedded features which are commonly the output of the last fully connected (FC) layer, and then predict labels using

these deep embedded features. These approaches use the deep embedded features holistically for their applications, without knowing what part of the features is used and what it is meaning.

Face recognition in unconstrained environments is a very challenging problem in computer vision. Faces of the same identity can look very different when presented in different illuminations, facial poses, facial expressions, and occlusions. Such variations within the same identity could overwhelm the variations due to identity differences and make face recognition challenging. To solve these problems, many deep learning-based approaches have been proposed and achieved high accuracies of face recognition such as DeepFace [34], DeepID series [30–32, 41], FaceNet [28], PIMNet [17], SphereFace [23], and ArcFace [10].

In face recognition tasks in unconstrained environments, the deeply learned and embedded features need to be not only separable but also discriminative. However, these features are learned implicitly for separable and distinct representations to classify among different identities without what part of the features is used, what part of the feature is meaningful, and what part of the features is separable and discriminative. Therefore, it is difficult to know what kind of features are used to discriminate the identities of face images clearly.

To overcome this limitation, we propose a novel face recognition method, called a pairwise relational network (PRN) to capture unique relations within same identity and discriminative relations among different identities. To capture relations, the PRN takes local appearance patches as input by ROI projection around landmark points on the feature map in a backbone CNN network. With these local appearance patches, the PRN is trained to capture unique pairwise relations between pairs of local appearance patches to determine facial part-relational structures and properties in face images. Because the existence and meaning of pairwise relations should be identity dependent, the PRN could condition its processing on a facial identity state feature. The facial identity state feature is learned from the long short-term memory (LSTM) units network with the sequential local appearance patches on the feature maps. To further improve accuracy of face recognition, we combined the global appearance representation with the local appearance representation (the relation features) (Fig. 1). More details of the proposed face recognition method are given in Section 2.

The main contributions of this paper can be summarized as follows:

- We propose a novel face recognition method using the pairwise relational network (PRN) which captures the unique and discriminative pairwise relations of local appearance patches on the feature maps to classify face images among different identities.
- We show that the proposed PRN is very useful to enhance the accuracy of both face verification and face identification.
- We present extensive experiments on the public available datasets such as Labeled Faces in the Wild (LFW), YouTube Faces (YTF), IARPA Janus Benchmark-A (IJB-A), and IARPA Janus Benchmark-B (IJB-B).

The rest of this paper is as follows: in Section 2 we describe the proposed face recognition method including the base CNN architecture, face alignment,

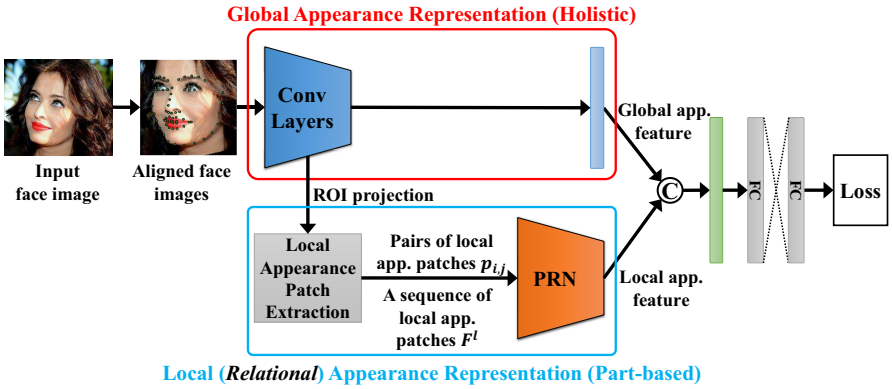

Fig. 1. Overview of the proposed face recognition method

pairwise relational network, facial identity state feature, loss function used for training the proposed method, respectively; in Sections 3 we present experimental results of the proposed method in comparison with the *state-of-the-art* on the public benchmark dataset and discussion; in Section 4 we draw a conclusion.

2 Proposed Methods

In this section, we describe our methods in detail including the base CNN model as backbone network for the global appearance representation, the face alignment method, the pairwise relational network, the pairwise relational network with face identity states, and the loss functions.

2.1 Base Convolutional Neural Network

We first describe the base CNN model. It is the backbone neural network to represent the global appearance representation and extract the local appearance patches to capture the relations (Fig. 1). The base CNN model consists of several 3-layer residual bottleneck blocks similar to the ResNet-101 [13]. The ResNet-101 has one convolution layer, one max pooling layer, 30 3-layer residual bottleneck blocks, one global average pooling (GAP) layer, one FC layer, and *softmax* loss layer. The ResNet-101 accepts a image with 224×224 resolution as input, and has 7×7 convolution filters with a stride of 2 in the first layer. In contrast, our base CNN model accepts a face image with 140×140 resolution as input, and has 5×5 convolution filters with a stride of 1 in the first layer (*conv1* in Table 1). Because of different input resolution, size of kernel filters, and stride, the output size in each intermediate layer is also different from the original ResNet-101. In the last layer, we use the GAP with 9×9 filter in each channel and the FC layer. The outputs of FC layer are fed into the *softmax* loss layer. More details of the base CNN architecture are given in Table 1.

Table 1. Base convolutional neural network. The base CNN is similar to ResNet-101, but the dimensionality of input, the size of convolution filters, and the size of each output feature map are different from the original ResNet-101

Layer name	**Output size**	**101-layer**
conv1	140×140	5×5, 64
conv2_x	70×70	3×3 max pool, stride 2
		$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	35×35	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
conv4_x	18×18	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$
conv5_x	9×9	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	global average pool, 8630-d fc, *softmax*

To represent the global appearance representation $\boldsymbol{f}^g$, we use the $1 \times 1 \times 2048$ feature which is the output of the GAP in the base CNN (Table 1).

To represent the local appearance representation, we extract the local appearance patches $\boldsymbol{f}^l$ on the $9 \times 9 \times 2048$ feature maps (*conv5_3*) in the base CNN (Table 1) by ROI projection with facial landmark points. These $\boldsymbol{f}^l$ are used to capture and model pairwise relations between them. More details of the local appearance patches and relations are described in Section 2.3.

2.2 Face Alignment

In the base CNN model, the input layer accepts the RGB values of the face image pixels. We employ a face alignment method to align a face image into the canonical face image, then we adopt this aligned face image as input of our base CNN model.

The alignment procedures are as follows: 1) Use the DAN implementation of Kowalsky *et al.* by using multi-stage neural network [19] to detect 68 facial landmark points (Fig. 2b); 2) rotate the face in the image plane to make it upright based on the eye positions (Fig. 2c); 3) find a central point on the face by taking the mid-point between the leftmost and rightmost landmark points (the red point in Fig. 2d); 4) the center points of the eye and mouth (blue points in Fig. 2d) are found by averaging all the landmark points in the eye and mouth regions, respectively; 5) center the faces in the x-axis, based on the center point (red point); 6) fix the position along the y-axis by placing the eye center point at 30% from the top of the image and the mouth center point at 35% from the bottom of the image, respectively; 7) resize the image to a resolution of

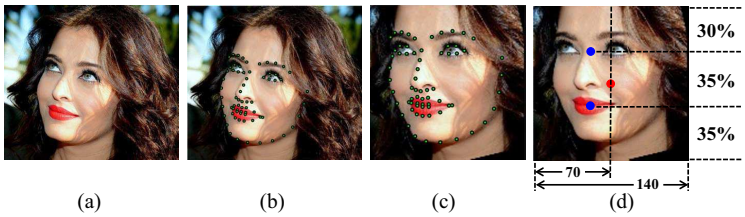

Fig. 2. A face alignment. The original image is shown in (a); (b) shows the detected 68 landmark points; (c) shows the aligned 68 landmark points in the aligned image plane; and (d) is the final aligned face image, where the red circle was used to center the face image along x-axis, and the blue circles denote the two points used for face cropping

140×140. Each pixel which value is in a range of $[0, 255]$ in the RGB color space is normalized by dividing 255 to be in a range of $[0, 1]$.

2.3 Pairwise Relational Network

The pairwise relational network (PRN) is a neural network and takes a set of local appearance patches on the feature maps as input and output a single feature vector as its relational feature for the face recognition task. The PRN captures unique pairwise relations between pairs of local appearance patches within the same identity and discriminative pairwise relations among different identities. In other words, the PRN captures the core common properties of faces within the same identity, while captures the discriminative properties of faces among different identities. Therefore, the PRN aims to determine pairwise-relational structures from pairs of local appearance patches in face images. The relation feature $\boldsymbol{r}_{i,j}$ represents a latent relation of a pair of two local appearance patches, and can be written as follows:

$$\boldsymbol{r}_{i,j} = \mathcal{G}_\theta \left(\boldsymbol{p}_{i,j} \right), \tag{1}$$

where $\mathcal{G}_\theta$ is a multi-layer perceptron (MLP) and its parameters θ are learnable weights. $\boldsymbol{p}_{i,j} = \{\boldsymbol{f}_i^l, \boldsymbol{f}_j^l\}$ is a pair of two local appearance patches ($\boldsymbol{f}_i^l$ and $\boldsymbol{f}_j^l$) which are i-th and j-th local appearance patches corresponding to each facial landmark point, respectively. Each local appearance patches $\boldsymbol{f}_i^l$ is extracted by the ROI projection which projects a $m \times m$ region around i-th landmark point in the input image space to a $m^{'} \times m^{'}$ region on the feature map space. The same MLP operates on all possible parings of local appearance patches.

The permutation order of local appearance patches is a critical for the PRN, since without this invariance, the PRN would have to learn to operate on all possible permuted pairs of local appearance patches without explicit knowledge of the permutation invariance structure in the data. To incorporate this permutation invariance, we constrain the PRN with an aggregation function (Fig. 3):

$$\boldsymbol{f}_{agg} = \mathcal{A}(\boldsymbol{r}_{i,j}) = \sum_{\forall \boldsymbol{r}_{i,j}} (\boldsymbol{r}_{i,j}), \tag{2}$$

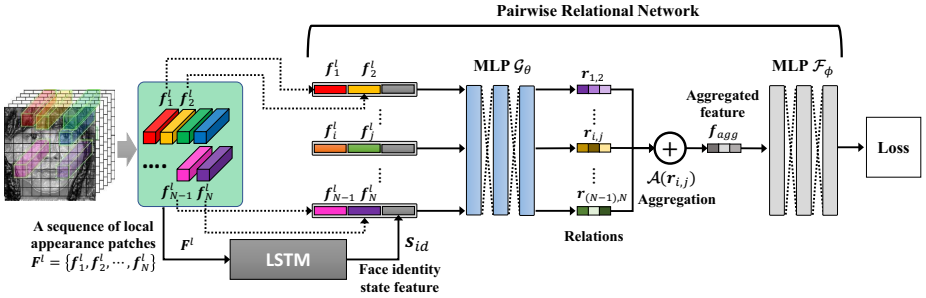

Fig. 3. Pairwise Relational Network (PRN). The PRN is a neural network module and takes a set of local appearance patches on the feature maps as input and outputs a single feature vector as its relational feature for the recognition task. The PRN captures unique pairwise relations between pairs of local appearance patches within the same identity and discriminative pairwise relations among different identities

where $\boldsymbol{f}_{agg}$ is the aggregated relational feature, and $\mathcal{A}(\cdot)$ is the aggregation function which is summation of all pairwise relations among all possible pairing of the local appearance patches. Finally, a prediction $\widetilde{\boldsymbol{r}}$ of the PRN can be performed with:

$$\widetilde{\boldsymbol{r}} = \mathcal{F}_\phi\left(\boldsymbol{f}_{agg}\right), \tag{3}$$

where $\mathcal{F}_\phi$ is a function with parameters ϕ, and are implemented by the MLP. Therefore, the final form of the PRN is a composite function as follows:

$$PRN(\boldsymbol{P}) = \mathcal{F}_\phi\left(\mathcal{A}\left(\mathcal{G}_\theta\left(\boldsymbol{p}_{i,j}\right)\right)\right), \tag{4}$$

where $\boldsymbol{P} = \{\boldsymbol{p}_{1,2}, \ldots, \boldsymbol{p}_{i,j}, \ldots, \boldsymbol{p}_{(N-1),N}\}$ is a set of all possible pairs of local appearance patches where N denotes the number of local patches on the feature maps.

To capture unique pairwise relations within same identity and discriminative pairwise relations among different identities, a pairwise relation should be identity dependent. So, we modify the PRN such that $\mathcal{G}_\theta$ could condition its processing on the identity information. To condition the identity information, we embed a face identity state feature $\boldsymbol{s}_{id}$ as the identity information in the PRN as follows:

$$PRN^+(\boldsymbol{P}, \boldsymbol{s}_{id}) = \mathcal{F}_\phi\left(\mathcal{A}\left(\mathcal{G}_\theta\left(\boldsymbol{p}_{i,j}, \boldsymbol{s}_{id}\right)\right)\right). \tag{5}$$

To get this $\boldsymbol{s}_{id}$, we use the final state of a recurrent neural network composed of LSTM layers and two FC layers that process a sequence of total local appearance patches (Fig. 1, 4).

Face Identity State Feature Pairwise relations should be identity dependent to capture unique and discriminative pairwise relations. Based on the feature maps which are the output of the *conv5_3* layer in the base CNN model, the face is divided into 68 local regions by ROI projection around 68 landmark

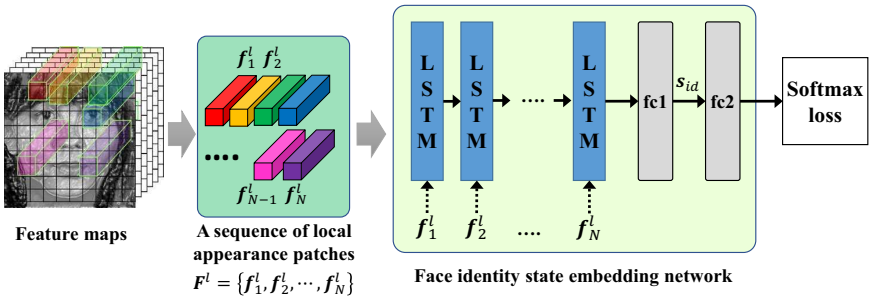

Fig. 4. Face identity state feature. A face on the feature maps is divided into 68 regions by ROI projection around 68 landmark points. A sequence of local appearance patches in these regions are used to encode the face identity state feature from LSTM networks

points. In these local regions, we extract the local appearance patches to model the facial identity state feature $\boldsymbol{s}_{id}$. Let $\boldsymbol{f}_i^l$ denote the local appearance patches of $m^{'} \times m^{'}$ i-th local region. To encode the facial identity state feature $\boldsymbol{s}_{id}$, an LSTM-based network has been devised on top of a set of local appearance patches $\boldsymbol{F}^l = \{\boldsymbol{f}_1^l, \ldots, \boldsymbol{f}_i^l, \ldots, \boldsymbol{f}_N^l\}$ as followings:

$$\boldsymbol{s}_{id} = \mathcal{E}_\psi(\boldsymbol{F}^l), \tag{6}$$

where $\mathcal{E}_\psi(\cdot)$ is a neural network module which composed of the LSTM layers and two FC layers with learnable parameters ψ. We train $\mathcal{E}_\psi$ with *softmax* loss function (Fig. 4). The detailed configuration of $\mathcal{E}_\psi$ used in our proposed method will be presented in Section 3.1.

2.4 Loss Function

To learn the proposed PRN, we jointly use the triplet ratio loss L_t, pairwise loss L_p, and identity preserving loss L_{id} (*softmax*) to minimize distances between faces that have the same identity and to maximize distances between faces that are of different identities:

$$L = \lambda_1 L_t + \lambda_2 L_p + \lambda_3 L_{id}. \tag{7}$$

During training the PRN, we empirically set $\lambda_1 = 1$, $\lambda_2 = 0.5$, and $\lambda_3 = 1$.

Triplet Ratio Loss Triplet ratio loss L_t is defined to maximize the ratio of distances between the positive pairs and the negative pairs in the triplets of faces. To maximize L_t, the Euclidean distances of positive pairs should be minimized and those of negative pairs should be maximized. Let $F(I) \in \mathbb{R}^d$, where I is the input facial image, denote the output of a network (the output of $\mathcal{F}_\phi$ in the PRN), the L_t is defined as follows:

$$L_t = \sum_{\forall T} \max\left(0, 1 - \frac{\|F(I_a) - F(I_n)\|_2}{\|F(I_a) - F(I_p)\|_2 + m}\right), \tag{8}$$

where $F(I_a)$ is the output of the network for an anchor face I_a, $F(I_p)$ is the output of the network for a positive face image I_p, and $F(I_n)$ is the output of the network for a negative face I_n in the triplets of faces T, respectively. m is a margin that defines a minimum ratio in Euclidean space. From recent work by B-.N. Kang *et al.* [17], they reported that an unbalanced range of distance measured between the pairs of data using only L_t during training; this result means that although the ratio of the distances is bounded in a certain range of values, the range of the absolute distances is not. To overcome this problem, they constrained L_t by adding the pairwise loss L_p.

Pairwise Loss Pairwise loss L_p is defined to minimize the sum of the squared Euclidean distances between $F(I_a)$ for the anchor face I_a and $F(I_p)$ for the positive face I_p. These pairs I_a and I_p are in the triplets T.

$$L_p = \sum_{(I_a, I_p) \in T} \|F(I_a) - F(I_p)\|_2^2. \tag{9}$$

The joint training with L_t and L_p minimizes the absolute Euclidean distance between face images of a given pair in the triplets of facs T.

3 Experiments

The implementation details are given in Section 3.1. Then, we investigate the effectiveness of the PRN and the PRN with the face identity state feature in Section 3.2. In Section 3.3, 3.4, 3.5, and 3.6, we perform several experiments to verify the effectiveness of the proposed method on the public face benchmark datasets including LFW [15], YTF [38], IJB-A [18], and IJB-B [37].

3.1 Implementation Details

Training Data We used the web-collected face dataset (VGGFace2 [3]). All of the faces in the VGGFace2 dataset and their landmark points are detected by the recently proposed face detector [42] and facial landmark point detector [19]. We used 68 landmark points for the face alignment and extraction of local appearance patches. When the detection of faces or facial landmark points is failed, we simply discard the image. Thus, we discarded $24,160$ face images from $6,561$ subjects. After removing these images without landmark points, it roughly goes to 3.1M images of $8,630$ unique persons. We generated a validation set by selecting randomly about 10% from each subject in refined dataset, and the remains are used as the training set. Therefore, the training set roughly has 2.8M face images and the validation set has $311,773$ face images, respectively.

Detailed settings in the PRN For pairwise relations between facial parts, we first extracted a set of local appearance patches $\boldsymbol{F}^l = \{\boldsymbol{f}_1^l, \ldots, \boldsymbol{f}_i^l, \ldots, \boldsymbol{f}_{68}^l\}$,

$\boldsymbol{f}_i^l \in \mathbb{R}^{1\times1\times2,048}$ from each local region (nearly 1×1 size of regions) around 68 landmark points by ROI projection on the $9 \times 9 \times 2,048$ feature maps (*conv5_3* in Table 1) in the backbone CNN model. Using this $\boldsymbol{F}^l$, we make $2,278$ ($= {}^{68}C_2$) possible pairs of local appearance patches. Then, we used three-layered MLP consisting of $1,000$ units per layer with batch normalization (BN) [16] and rectified linear units (ReLU) [25] non-linear activation functions for $\mathcal{G}_\theta$, and three-layered MLP consisting of $1,000$ units per layer with BN and ReLU non-linear activation functions for $\mathcal{F}_\phi$. To aggregate all of relations from $\mathcal{G}_\theta$, we used summation as an aggregation function. The PRN is jointly optimized by *triplet ratio* loss L_T, *pairwise* loss L_p, and *identity preserving* loss L_{id} (*softmax*) over the ground-truth identity labels using stochastic gradient descent (SGD) optimization method with learning rate 0.10. We used mini-batch size of 128 on four NVIDIA Titan X GPUs. During training the PRN, we froze the backbone CNN model to only update weights of the PRN model.

To capture unique and discriminative pairwise relations dependent on identity, the PRN should condition its processing on the face identity state feature $\boldsymbol{s}_{id}$. For $\boldsymbol{s}_{id}$, we use the LSTM-based recurrent network $\mathcal{E}_\psi$ over a sequence of the local appearance patches which is a set ordered by landmark points order from $\boldsymbol{F}^l$. In other words, there is a sequence of 68 length per face. In $\mathcal{E}_\psi$, it consist of LSTM layers and two-layered MLP. Each of the LSTM layer has $2,048$ memory cells. The MLP consists of 256 and $8,630$ units per layer, respectively. The cross-entropy loss with *softmax* was used for training the $\mathcal{E}_\psi$ (Fig. 4).

Detailed settings in the model We implemented the base CNN and the PRN models using the Keras framework [7] with TensorFlow [1] backend. For fair comparison in terms of the effects of each network module, we train three kinds of models (**model A**, **model B**, and **model C**) under the supervision of cross-entropy loss with *softmax*:

- **model A** is the baseline model which is the base CNN (Table 1).
- **model B** combining two different networks, one of which is the base CNN model (**model A**) and the other is the PRN (Eq. (4)), concatenates the output feature $\boldsymbol{f}^g$ of the GAP layer in **model A** as the global appearance representation and the output of the MLP $\mathcal{F}_\phi$ in the PRN without the face identity state feature $\boldsymbol{s}_{id}$ as the local appearance representation. $\boldsymbol{f}^g$ is the feature of size $1 \times 1 \times 2,048$ from each face image. The output of the MLP $\mathcal{F}_\phi$ in the PRN is the feature of size $1 \times 1 \times 1,000$. These two output features are concatenated into a single feature vector with $3,048$ size, then this concatenated feature vector is fed into the FC layer with $1,024$ units.
- **model C** is the combined model with the output of the base CNN model (**model A**) and the output of the PRN^+ (Eq. (5)) with the face identity state feature $\boldsymbol{s}_{id}$. The output of **model A** in **model C** is the same of the output in **model B**. The size of the output in the PRN^+ is same as compared with the PRN, but output values are different.

All of convolution layers and FC layers use BN and ReLU as nonlinear activation functions except for LSTM layers in $\mathcal{E}_\psi$.

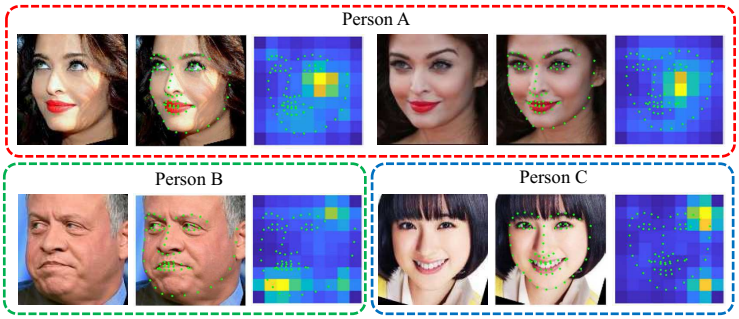

Fig. 5. Visualization of the localized facial parts

3.2 Effects of the PRNs

To investigate the effectiveness of the PRN and the face identity state feature $\boldsymbol{s}_{id}$, we performed experiments in terms of the accuracy of classification on the validation set during training. For these experiments, we trained two different network models, one of which is a network PRN (Eq. (4)) using only the PRN model, and the other is a network PRN^+ (Eq. (5)) using the PRN with the $\boldsymbol{s}_{id}$. We achieved 94.2% and 96.7% accuracies of classification for PRN and PRN^+, respectively. From these evaluations, when using PRN^+, we observed that the face identity state feature $\boldsymbol{s}_{id}$ represents the identity property, and the pairwise relations should be dependent on an identity property of a face image. Therefore, these evaluations validate the effectiveness of using the PRN and the importance of the face identity state feature. We visualize the localized facial parts in Fig. 5, where *Col. 1*, *Col. 2*, and *Col. 3* of each identity are the aligned facial image, detected facial landmark points, and localized facial parts by ROI projection on the feature maps, respectively. We can see that the localized appearance representations are discriminative among different identities.

3.3 Experiments on the Labeled Faces in the Wild (LFW)

We evaluated the proposed method on the LFW dataset, which reveals the *state-of-the-art* of face verification in unconstrained environments. The LFW dataset is excellent benchmark dataset for face verification in image and contains $13,233$ web crawling images with large variations in illuminations, occlusions, facial poses, and facial expressions, from $5,749$ different identities. Our models such as **model A**, **model B**, and **model C** are trained on the roughly 2.8M outside training set (VGGFace2), with no people overlapping with subjects in the LFW. Following the test protocol of *unrestricted with labeled outside data* [21], we test on $6,000$ face pairs by using a squared L_2 distance threshold to determine classification of *same* and *different*, and report the results in comparison with the *state-of-the-art* methods (Table 2).

From the experimental results (Table 2), we have the following observations. First, PRN itself provides slightly better accuracy than the baseline **model A**

Table 2. Comparison of the number of images, the number of networks, the dimensionality of feature, and the accuracy of the proposed method with the *state-of-the-art* methods on the LFW

Method	Images	Networks	Dimension	Accuracy (%)
DeepFace [34]	4M	9	$4,096 \times 4$	97.25
DeepID [30]	202,599	120	150×120	97.45
DeepID2+ [32]	300,000	25	150×120	99.47
DeepID3 [41]	300,000	50	300×100	99.52
FaceNet [28]	200M	1	128	99.63
Learning from Scratch [40]	494,414	2	160×2	97.73
CenterFace [36]	0.7M	1	512	99.28
$\text{PIMNet}_{\text{TL-Joint Bayesian}}$ [17]	198,018	4	1,024	98.33
$\text{PIMNet}_{\text{fusion}}$ [17]	198,018	4	6	99.08
SphereFace [23]	494,414	1	1,024	99.42
ArcFace [10]	3.1M	1	512	99.78
model A (baseline, only $\boldsymbol{f}^g$)	2.8M	1	2,048	**99.6**
PRN	2.8M	1	1,000	**99.61**
PRN$^+$	2.8M	1	1,000	**99.69**
model B ($\boldsymbol{f}^g + PRN$)	2.8M	1	1,024	**99.65**
model C ($\boldsymbol{f}^g + PRN^+$)	2.8M	1	1,024	**99.76**

(the base CNN model, just uses $\boldsymbol{f}^g$) and PRN^+ outperforms **model B** which is jointly combined both $\boldsymbol{f}^g$ with PRN. Second, **model C** (jointly combined $\boldsymbol{f}^g$ with PRN^+) beats the baseline model **model A** by a significantly margin, improving the accuracy from 99.6% to 99.76%. This shows that combination of $\boldsymbol{f}^g$ and PRN^+ can notably increase the discriminative power of deeply learned features, and the effectiveness of the pairwise relations between facial local appearance parts (local appearance patches). Third, compared to **model B**, **model C** achieved better accuracy of verification (99.65% *vs.* 99.76%). This shows the importance of the face identity state feature to capture unique and discriminative pairwise relations in the designed PRN model. Last, compared to the *state-of-the-art* methods on the LFW, the proposed method **model C** is among the top-ranked approaches, outperforming most of the existing results (Table 2). This shows the importance and advantage of the proposed method.

3.4 Experiments on the YouTube Face Dataset (YTF)

We evaluated the proposed method on the YTF dataset, which reveals the *state-of-the-art* of face verification in unconstrained environments. The YTF dataset is excellent benchmark dataset for face verification in video and contains 3,425 videos with large variations in illuminations, facial pose, and facial expressions, from 1,595 different identities, with an average of 2.15 videos per person. The length of video clip varies from 48 to 6,070 frames and average of 181.3 frames. We follow the test protocol of *unrestricted with labeled outside data*. We test on 5,000 video pairs and report the test results in comparison with the *state-of-the-art* methods (Table 3).

Table 3. Comparison of the number of images, the number of networks, the dimensionality of feature, and the accuracy of the proposed method with the *state-of-the-art* methods on the YTF

Method	Images	Networks	Dimension	Accuracy (%)
DeepFace [34]	4M	9	$4,096 \times 4$	91.4
DeepID2+ [32]	$300,000$	25	150×120	93.2
FaceNet [28]	200M	1	128	95.1
Learning from Scratch [40]	$494,414$	2	160×2	92.2
CenterFace [36]	0.7M	1	512	94.9
SphereFace [23]	$494,414$	1	$1,024$	95.0
NAN [39]	3M	1	128	95.7
model A (baseline, only $\boldsymbol{f}^g$)	2.8M	1	$2,048$	**95.1**
PRN	2.8M	1	$1,000$	**95.3**
PRN$^+$	2.8M	1	$1,000$	**95.8**
model B ($\boldsymbol{f}^g + PRN$)	2.8M	1	$1,024$	**95.7**
model C ($\boldsymbol{f}^g + PRN^+$)	2.8M	1	$1,024$	**96.3**

From the experimental results (Table 3), we have the following observations. First, PRN itself provides slightly better accuracy than the baseline **model A** (the base CNN model, just uses $\boldsymbol{f}^g$) and PRN^+ outperforms **model B** which is jointly combined both $\boldsymbol{f}^g$ with PRN. Second, **model C** (jointly combined $\boldsymbol{f}^g$ with PRN^+) beats the baseline model **model A** by a significant margin, improving the accuracy from 95.1% to 96.3%. This shows that combination of $\boldsymbol{f}^g$ and PRN^+ can notably increase the discriminative power of deeply learned features, and the effectiveness of the pairwise relations between facial local appearance patches. Third, compared to **model B**, **model C** achieved better accuracy of verification (95.7% *v.s.* 96.3%). This shows the importance of the face identity state feature to capture unique pairwise relations in the designed PRN model. Last, compared to the *state-of-the-art* methods on the YTF, the proposed method **model C** is the *state-of-the-art* (96.3%), outperforming the existing results (Table 3). This shows the importance and advantage of the proposed method.

3.5 Experiments on the IARPA Janus Benchmark A (IJB-A)

We evaluated the proposed method on the IJB-A dataset [18] which contains face images and videos captured from unconstrained environments. It features full pose variation and wide variations in imaging conditions thus is very challenging. It contains 500 subjects with $5,397$ images and $2,042$ videos in total, and 11.4 images and 4.2 videos per subject on average. We detect the faces using face detector [42] and landmark points using DAN landmark point detector [19], and then align the face image with our face alignment method explained in Section 2.2. In this dataset, each training and testing instance is called a 'template', which comprises 1 to 190 mixed still images and video frames. IJB-A dataset provides 10 split evaluations with two protocols (1:1 face verification and 1:N face identification). For face verification, we report the test results by using true

Table 4. Comparison of performances of the proposed PRN method with the *state-of-the-art* on the IJB-A dataset. For verification, TAR *vs.* FAR are reported. For identification, TPIR *vs.* FPIR and the Rank-N accuracies are presented

Method	1:1 Verification TAR			1:N Identification TPIR				
	FAR=0.001	FAR=0.01	FAR=0.1	FPIR=0.01	FPIR=0.1	Rank-1	Rank-5	Rank-10
B-CNN [8]	-	-	-	0.143 ± 0.027	0.341 ± 0.032	0.588 ± 0.020	0.796 ± 0.017	-
LSFS [35]	0.514 ± 0.060	0.733 ± 0.034	0.895 ± 0.013	0.383 ± 0.063	0.613 ± 0.032	0.820 ± 0.024	0.929 ± 0.013	-
$DCNN_{manual}$+metric [6]	-	0.787 ± 0.043	0.947 ± 0.011	-	-	0.852 ± 0.018	0.937 ± 0.010	0.954 ± 0.007
Triplet Similarity [27]	0.590 ± 0.050	0.790 ± 0.030	0.945 ± 0.002	0.556 ± 0.065	0.754 ± 0.014	0.880 ± 0.015	0.95 ± 0.007	0.974 ± 0.005
Pose-Aware Models [24]	0.652 ± 0.037	0.826 ± 0.018	-	-	-	0.840 ± 0.012	0.925 ± 0.008	0.946 ± 0.005
Deep Multi-Pose [2]	-	0.876	0.954	0.52	0.75	0.846	0.927	0.947
$DCNN_{fusion}$ [5]	-	0.838 ± 0.042	0.967 ± 0.009	0.577 ± 0.094	0.790 ± 0.033	0.903 ± 0.012	0.965 ± 0.008	0.977 ± 0.007
Triplet Embedding [27]	0.813 ± 0.02	0.90 ± 0.01	0.964 ± 0.005	0.753 ± 0.03	0.863 ± 0.014	0.932 ± 0.01	-	0.977 ± 0.005
VGG-Face [26]	-	0.805 ± 0.030	-	0.461 ± 0.077	0.670 ± 0.031	0.913 ± 0.011	-	0.981 ± 0.005
Template Adaptation [9]	0.836 ± 0.027	0.939 ± 0.013	0.979 ± 0.004	0.774 ± 0.049	0.882 ± 0.016	0.928 ± 0.010	0.977 ± 0.004	0.986 ± 0.003
NAN [39]	0.881 ± 0.011	0.941 ± 0.008	0.978 ± 0.003	0.817 ± 0.041	0.917 ± 0.009	0.958 ± 0.005	0.980 ± 0.005	0.986 ± 0.003
VGGFace2 [3]	0.921 ± 0.014	0.968 ± 0.006	0.990 ± 0.002	0.883 ± 0.038	0.946 ± 0.004	0.982 ± 0.004	0.993 ± 0.002	0.994 ± 0.001
model A (baseline, only $\boldsymbol{f}^g$)	$\mathbf{0.895 \pm 0.015}$	$\mathbf{0.949 \pm 0.008}$	$\mathbf{0.980 \pm 0.005}$	$\mathbf{0.843 \pm 0.035}$	$\mathbf{0.923 \pm 0.005}$	$\mathbf{0.975 \pm 0.005}$	$\mathbf{0.992 \pm 0.004}$	$\mathbf{0.993 \pm 0.001}$
model B ($\boldsymbol{f}^g$ + PRN)	$\mathbf{0.901 \pm 0.014}$	$\mathbf{0.950 \pm 0.006}$	$\mathbf{0.985 \pm 0.002}$	$\mathbf{0.861 \pm 0.038}$	$\mathbf{0.931 \pm 0.004}$	$\mathbf{0.976 \pm 0.003}$	$\mathbf{0.992 \pm 0.003}$	$\mathbf{0.994 \pm 0.003}$
model C ($\boldsymbol{f}^g$ + PRN^+)	$\mathbf{0.919 \pm 0.013}$	$\mathbf{0.965 \pm 0.004}$	$\mathbf{0.988 \pm 0.002}$	$\mathbf{0.882 \pm 0.038}$	$\mathbf{0.941 \pm 0.004}$	$\mathbf{0.982 \pm 0.004}$	$\mathbf{0.992 \pm 0.002}$	$\mathbf{0.995 \pm 0.001}$

accept rate (TAR) *vs.* false accept rate (FAR) (Table 4). For face identification, we report the results by using the true positive identification (TPIR) *vs.* false positive identification rate (FPIR) and Rank-N (Table 4). All measurements are based on a squared L_2 distance threshold.

From the experimental results (Table 4), we have the following observations. First, compared to **model A** (base CNN model), **model C** (jointly combined $\boldsymbol{f}^g$ with PRN^+) achieved a consistently superior accuracy (TAR and TPIR) on both 1:1 face verification and 1:N face identification. Second, compared to **model B** (jointly combined $\boldsymbol{f}^g$ with PRN), **model C** achieved also a consistently better accuracy (TAR and TPIR) on both 1:1 face verification and 1:N face identification. Last, more importantly, **model C** is trained from scratch and achieves comparable results to the *state-of-the-art* (VGGFace2 [3]) which is first pre-trained on the MS-Celeb-1M dataset [11], which contains roughly 10M face images, and then is fine-tuned on the VGGFace2 dataset. It shows that our proposed method can be further improved by training on the MS-Celeb-1M and our training dataset.

3.6 Experiments on the IARPA Janus Benchmark B (IJB-B)

We evaluated the proposed method on the IJB-B dataset [37] which contains face images and videos captured from unconstrained environments. The IJB-B dataset is an extension of the IJB-A, having $1,845$ subjects with 21.8K still images (including $11,754$ face and $10,044$ non-face) and 55K frames from $7,011$ videos, an average of 41 images per subject. Because images in this dataset are labeled with ground truth bounding boxes, we only detect landmark points using DAN [19], and then align face images with our face alignment method. Unlike the IJB-A, it does not contain any training splits. In particular, we use the 1:1 Baseline Verification protocol and 1:N Mixed Media Identification protocol for the IJB-B. For face verification, we report the test results by using TAR *vs.* FAR (Table 5). For face identification, we report the results by using TPIR *vs.* FPIR and Rank-N (Table 5). We compare our proposed methods with VGGFace2

Table 5. Comparison of performances of the proposed PRN method with the *state-of-the-art* on the IJB-B dataset. For verification, TAR *vs.* FAR are reported. For identification, TPIR *vs.* FPIR and the Rank-N accuracies are presented

Method	1:1 Verification TAR				1:N Identification TPIR				
	FAR=0.0001	FAR=0.0001	FAR=0.001	FAR=0.01	FPIR=0.01	FPIR=0.1	Rank-1	Rank-5	Rank-10
VGGFace2 [3]	0.671	0.800	0.0.888	0.949	0.746 ± 0.018	0.842 ± 0.022	0.912 ± 0.017	0.949 ± 0.010	0.962 ± 0.007
VGGFace2_ft [3]	0.705	0.831	0.908	0.956	0.763 ± 0.018	0.865 ± 0.018	0.914 ± 0.029	0.951 ± 0.013	0.961 ± 0.010
FPN [4]	-	0.832	0.916	0.965	-	-	0.911	0.953	0.975
model A (baseline, only $\boldsymbol{f}^g$)	**0.673**	**0.812**	**0.892**	**0.953**	$\mathbf{0.743 \pm 0.019}$	$\mathbf{0.851 \pm 0.017}$	$\mathbf{0.911 \pm 0.017}$	$\mathbf{0.950 \pm 0.013}$	$\mathbf{0.961 \pm 0.010}$
model B ($\boldsymbol{f}^g + PRN$)	**0.692**	**0.829**	**0.910**	**0.956**	$\mathbf{0.773 \pm 0.018}$	$\mathbf{0.865 \pm 0.018}$	$\mathbf{0.913 \pm 0.022}$	$\mathbf{0.954 \pm 0.010}$	$\mathbf{0.965 \pm 0.013}$
model C ($\boldsymbol{f}^g + PRN^+$)	**0.721**	**0.845**	**0.923**	**0.965**	$\mathbf{0.814 \pm 0.017}$	$\mathbf{0.907 \pm 0.013}$	$\mathbf{0.935 \pm 0.015}$	$\mathbf{0.965 \pm 0.017}$	$\mathbf{0.975 \pm 0.007}$

[3] and FacePoseNet (FPN) [4]. All measurements are based on a squared L_2 distance threshold.

From the experimental results, we have the following observations. First, compared to **model A** (base CNN model, just uses $\boldsymbol{f}^g$), **model C** (jointly combined $\boldsymbol{f}^g$ with PRN^+ as the local appearance representation) achieved a consistently superior accuracy (TAR and TPIR) on both 1:1 face verification and 1:N face identification. Second, compared to **model B** (jointly combined $\boldsymbol{f}^g$ with the PRN), **model C** achieved also a consistently better accuracy (TAR and TPIR) on both 1:1 face verification and 1:N face identification. Last, more importantly, **model C** achieved consistent improvement of TAR and TPIR on both 1:1 face verification and 1:N face identification, and achieved the *state-of-the-art* results on the IJB-B.

4 Conclusion

We proposed a novel face recognition method using the pairwise relational network (PRN) which takes local appearance patches around landmark points on the feature maps, and captures unique pairwise relations between a pair of local appearance patches. To capture unique and discriminative relations for face recognition, pairwise relations should be identity dependent. Therefore, the PRN conditioned its processing on the face identity state feature embedded by the LSTM based network using a sequential local appearance patches. To further improve accuracy of face recognition, we combined the global appearance representation with the PRN. Experiments verified the effectiveness and importance of our proposed PRN and the face identity state feature, which achieved 99.76% accuracy on the LFW, the *state-of-the-art* accuracy (96.3%) on the YTF, comparable results to the *state-of-the-art* for both face verification and identification tasks on the IJB-A, and the *state-of-the-art* results on the IJB-B.

Acknowledgment

This research was supported by the MSIT, Korea, under the SW Starlab support program (IITP-2017-0-00897), and "ICT Consilience Creative program" (IITP-2018-2011-1-00783) supervised by the IITP.

References

1. Abadi, M., et al.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015), https://www.tensorflow.org/, software available from tensorflow.org
2. AbdAlmageed, W., Wu, Y., Rawls, S., Harel, S., Hassner, T., Masi, I., Choi, J., Lekust, J., Kim, J., Natarajan, P., Nevatia, R., Medioni, G.: Face recognition using deep multi-pose representations. In: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1–9 (March 2016). https://doi.org/10.1109/WACV.2016.7477555
3. Cao, Q., Shen, L., Xie, W., Parkhi, O.M., Zisserman, A.: Vggface2: A dataset for recognising faces across pose and age. CoRR **abs/1710.08092** (2017), http://arxiv.org/abs/1710.08092
4. Chang, F.J., Tran, A.T., Hassner, T., Masi, I., Nevatia, R., Medioni, G.: Faceposenet: Making a case for landmark-free face alignment. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW). pp. 1599–1608 (Oct 2017). https://doi.org/10.1109/ICCVW.2017.188
5. Chen, J.C., Patel, V.M., Chellappa, R.: Unconstrained face verification using deep cnn features. In: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1–9 (March 2016). https://doi.org/10.1109/WACV.2016.7477557
6. Chen, J.C., Ranjan, R., Kumar, A., Chen, C.H., Patel, V.M., Chellappa, R.: An end-to-end system for unconstrained face verification with deep convolutional neural networks. In: 2015 IEEE International Conference on Computer Vision Workshop (ICCVW). pp. 360–368 (Dec 2015). https://doi.org/10.1109/ICCVW.2015.55
7. Chollet, F., et al.: Keras. https://github.com/fchollet/keras (2015)
8. Chowdhury, A.R., Lin, T.Y., Maji, S., Learned-Miller, E.: One-to-many face recognition with bilinear cnns. In: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1–9 (March 2016). https://doi.org/10.1109/WACV.2016.7477593
9. Crosswhite, N., Byrne, J., Stauffer, C., Parkhi, O., Cao, Q., Zisserman, A.: Template adaptation for face verification and identification. In: 2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017). pp. 1–8 (May 2017). https://doi.org/10.1109/FG.2017.11
10. Deng, J., Guo, J., Zafeiriou, S.: ArcFace: Additive Angular Margin Loss for Deep Face Recognition. ArXiv e-prints (Jan 2018)
11. Guo, Y., Zhang, L., Hu, Y., He, X., Gao, J.: MS-Celeb-1M: A dataset and benchmark for large scale face recognition. In: European Conference on Computer Vision. pp. 87–102. Springer International Publishing (2016)
12. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 1026–1034 (Dec 2015). https://doi.org/10.1109/ICCV.2015.123
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (June 2016). https://doi.org/10.1109/CVPR.2016.90
14. Huang, G., Liu, Z., v. d. Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2261–2269 (July 2017). https://doi.org/10.1109/CVPR.2017.243
15. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Tech. Rep. 07-49, University of Massachusetts, Amherst (October 2007)

16. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. pp. 448–456 (2015), http://jmlr.org/proceedings/papers/v37/ioffe15.html
17. Kang, B.N., Kim, Y., Kim, D.: Deep convolutional neural network using triplets of faces, deep ensemble, and score-level fusion for face recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 611–618 (July 2017). https://doi.org/10.1109/CVPRW.2017.89
18. Klare, B.F., Klein, B., Taborsky, E., Blanton, A., Cheney, J., Allen, K., Grother, P., Mah, A., Burge, M., Jain, A.K.: Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1931–1939 (June 2015). https://doi.org/10.1109/CVPR.2015.7298803
19. Kowalski, M., Naruniec, J., Trzcinski, T.: Deep alignment network: A convolutional neural network for robust face alignment. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 2034–2043 (July 2017). https://doi.org/10.1109/CVPRW.2017.254
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1. pp. 1097–1105. NIPS'12 (2012), http://dl.acm.org/citation.cfm?id=2999134.2999257
21. Learned-Miller, G.B.H.E.: Labeled faces in the wild: Updates and new reporting procedures. Tech. Rep. UM-CS-2014-003, University of Massachusetts, Amherst (May 2014)
22. Liu, S., Deng, W.: Very deep convolutional neural network based image classification using small training sample size. In: 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR). pp. 730–734 (Nov 2015). https://doi.org/10.1109/ACPR.2015.7486599
23. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6738–6746 (July 2017). https://doi.org/10.1109/CVPR.2017.713
24. Masi, I., Rawls, S., Medioni, G., Natarajan, P.: Pose-aware face recognition in the wild. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4838–4846 (June 2016). https://doi.org/10.1109/CVPR.2016.523
25. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on International Conference on Machine Learning. pp. 807–814. ICML'10 (2010), http://dl.acm.org/citation.cfm?id=3104322.3104425
26. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition. In: Proceedings of the British Machine Vision Conference (BMVC). pp. 41.1–41.12 (September 2015). https://doi.org/10.5244/C.29.41
27. Sankaranarayanan, S., Alavi, A., Castillo, C.D., Chellappa, R.: Triplet probabilistic embedding for face verification and clustering. In: 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS). pp. 1–8 (Sept 2016). https://doi.org/10.1109/BTAS.2016.7791205
28. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 815–823 (June 2015). https://doi.org/10.1109/CVPR.2015.7298682

29. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014), http://arxiv.org/abs/1409.1556
30. Sun, Y., Wang, X., Tang, X.: Deep learning face representation from predicting 10,000 classes. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1891–1898 (June 2014). https://doi.org/10.1109/CVPR.2014.244
31. Sun, Y., Wang, X., Tang, X.: Deeply learned face representations are sparse, selective, and robust pp. 2892–2900 (June 2015). https://doi.org/10.1109/CVPR.2015.7298907
32. Sun, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification pp. 1988–1996 (2014), http://papers.nips.cc/paper/5416-deep-learning-face-representation-by-joint-identification-verification.pdf
33. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1–9 (June 2015). https://doi.org/10.1109/CVPR.2015.7298594
34. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: Closing the gap to human-level performance in face verification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1701–1708 (June 2014). https://doi.org/10.1109/CVPR.2014.220
35. Wang, D., Otto, C., Jain, A.K.: Face search at scale. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(6), 1122–1136 (June 2017). https://doi.org/10.1109/TPAMI.2016.2582166
36. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: Computer Vision – ECCV 2016. pp. 499–515. Springer International Publishing (2016). https://doi.org/10.1007/978-3-319-46478-7-31
37. Whitelam, C., Taborsky, E., Blanton, A., Maze, B., Adams, J., Miller, T., Kalka, N., Jain, A.K., Duncan, J.A., Allen, K., Cheney, J., Grother, P.: Iarpa janus benchmark-b face dataset. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 592–600 (2017). https://doi.org/10.1109/CVPRW.2017.87
38. Wolf, L., Hassner, T., Maoz, I.: Face recognition in unconstrained videos with matched background similarity. In: CVPR 2011. pp. 529–534 (June 2011). https://doi.org/10.1109/CVPR.2011.5995566
39. Yang, J., Ren, P., Zhang, D., Chen, D., Wen, F., Li, H., Hua, G.: Neural aggregation network for video face recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5216–5225 (July 2017). https://doi.org/10.1109/CVPR.2017.554
40. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. CoRR **abs/1411.7923** (2014), http://arxiv.org/abs/1411.7923
41. Yi Sun, Ding Liang, X.W., Tang, X.: Deepid3: Face recognition with very deep neural networks. CoRR **abs/1502.00873** (2015), http://arxiv.org/abs/1502.00873
42. Yoon, J., Kim, D.: An accurate and real-time multi-view face detector using orfs and doubly domain-partitioning classifier. Journal of Real-Time Image Processing (Feb 2018). https://doi.org/10.1007/s11554-018-0751-6
43. Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A.: Object detectors emerge in deep scene cnns. CoRR **abs/1412.6856** (2014), http://arxiv.org/abs/1412.6856
44. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Ghahramani, Z.,

Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 487–495. Curran Associates, Inc. (2014), http://papers.nips.cc/paper/5349-learning-deep-features-for-scene-recognition-using-places-database.pdf