Face Recognition with Contrastive Convolution

Chunrui Han[1,2][0000−0001−9725−280X], Shiguang Shan[1,3][0000−0002−8348−392X],
Meina Kan[1,3][0000−0001−9483−875X], Shuzhe Wu[1,2][0000−0002−4455−4123], and
Xilin Chen[1][0000−0003−3024−4404]

[1] Key Lab of Intelligent Information Processing of Chinese Academy of Sciences,
Institute of Computing Technology, CAS, Beijing 100190, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] CAS Center for Excellence in Brain Science and Intelligence Technology
{chunrui.han, shuzhe.wu}@vipl.ict.ac.cn, {sgshan, kanmeina,
xlchen}@ict.ac.cn

Abstract. In current face recognition approaches with convolutional neural network (CNN), a pair of faces to compare are independently fed into the CNN for feature extraction. For both faces the same kernels are applied and hence the representation of a face stays fixed regardless of whom it is compared with. As for us humans, however, one generally focuses on varied characteristics of a face when comparing it with distinct persons as shown in Figure 1. Inspired, we propose a novel CNN structure with what we referred to as contrastive convolution, which specifically focuses on the distinct characteristics between the two faces to compare, i.e., those contrastive characteristics. Extensive experiments on the challenging LFW, and IJB-A show that our proposed contrastive convolution significantly improves the vanilla CNN and achieves quite promising performance in face verification task.

Keywords: Face Recognition, Convolutional Neural Networks, Contrastive Convolution, Kernel Generator

1 Introduction

Face recognition is of great practical values as an effective approach for biometric authentication. The task of face recognition includes two categories, face identification which classifies a given face to a specific identity, and face verification which determines whether a pair of faces are of the same identity. The face verification task appears in a wide range of practical scenarios, e.g., phone unlocking with faces, remote bank account opening that uses faces for identity check, electronic payment with face, criminal tracking from surveillance cameras and etc. Though it has been studied for a long time, there still exist a great many challenges for accurate face verification, which is the focus of this work.

The most effective solutions for the face verification at present are employing the powerful CNN models. To verify whether a given pair of faces A and B are of the same identity, most CNN-based methods generally first feed the two faces into a CNN to obtain their feature representations. Then, the similarity of the

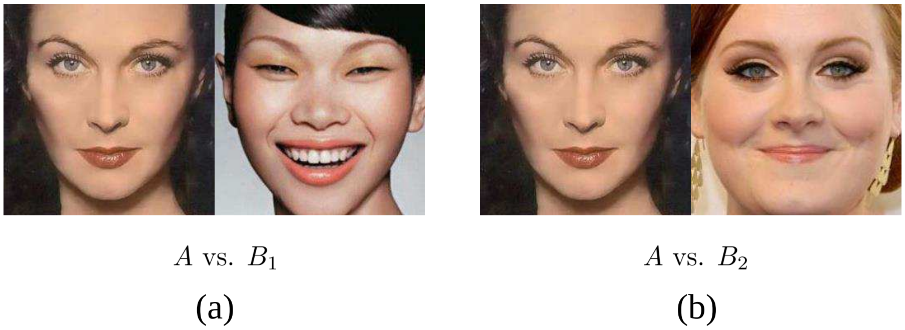

Fig. 1. Illustration of how we humans do face verification by focusing on distinct face characteristics when the same face A is compared with different persons. (a) When comparing A with B_1 who features small eyes, our focus is attracted to regions around the eyes of A; (b) when comparing A with B_2 whose face is round, we pay more attention to the contour of A. This reveals that a face should be described differently by using contrastive charateristics for example, when being compared with different persons.

two features is calculated to determine whether they are the same person. Since the parameters of convolutional kernels are fixed once the training of CNN is completed, all faces are processed with identical kernels and thus mapped into a common discriminative feature space. This means that the representation of A stays unchanged regardless of who it is compared with, and this representation has to be discriminative enough to distinguish A from all other persons, which is quite challenging. By contrast, when we humans compare two faces, the observation of one face is guided by that of the other, i.e., finding the differences and putting more attention on them for better distinguishing of the two faces. Taking Figure 1 for example, the same face A is compared with two different faces B_1 and B_2. When comparing with B_1 who features small eyes relative to A's big eyes, our focus on A will be attracted to regions around the eyes. When comparing with B_2 whose face is round relative to A's oval face, we will tend to pay more attention to the contour of face A during the observation. Naturally, we depict a face differently when comparing it with different persons so as to distinguish them more accurately.

Inspired by this observation, we propose a novel CNN structure with what we referred to as contrastive convolution, whose kernels are carefully designed and mainly focus on those distinct characteristics, i.e., contrastive features, between the two faces for better verification of them. Specifically, a kernel generator module is designed to generate personalized kernels of a face first. As personalized kernels of a specific person often have high correlation with its own features, the difference of the personalized kernels of the two faces are exploited as the contrastive convolutional kernels, which are expected to focus on the difference between the two faces. This contrastive convolution can be embedded into any kind of convolutional neural networks, and in this work it is embedded into the popular CNN, forming an novel face verification model as shown in Figure 2, which is referred to as *Contrastive CNN*. To demonstrate the effectiveness of the proposed contrastive convolution, extensive experiments are performed on the challenging LFW and IJB-A dataset, and our contrastive CNN achieves quite promising performance.

The rest of this work is organized as follows: Section 2 reviews works related to face recognition in the wild and adaptative convolution. Section 3 describes the proposed deep convolutional neural network with contrastive convolution. Section 4 presents experimental results, and finally Section 5 concludes this work.

2 Related Work

Face Recognition Face recognition is an important and classical topic in computer vision, in which face feature learning plays a significant role. An expressive feature descriptor can substantially improve the accuracy of face recognition. Some early works mainly focus on hand-crafted features, such as the well-known Local Binary Pattern (LBP) [2] and Gabor [36][34] which achieved favorable results in controlled environment. The discrimination ability of hand-crafted features heavily depends on the design principal which may be not beneficial for classification. Go a step further, a few learning-based approaches are proposed to learn more informative but mostly linear feature representation, including the famous Eigenfaces [28] and Fisherfaces [3][6]. Recently, deep convolutional neural networks (CNNs) arise with great performance improvement [16][13] benefitted from its excellent non-linear modeling capability. In [27], a CNN is proposed to extract deep features of the faces that are aligned to frontal through a general 3D shape model and performs better than many traditional face recognition methods. Afterwards, the performance of face recognition is further improved in quick succession by Deep ID2[7], Deep ID2+[26], which even surpass the human's performance for face verification on the Labeled Face in the Wild (LFW). Several recent works mainly focus on exploring better loss functions to imporve the performance of face recognition. In [9], a method named FaceNet is proposed to employ triplet loss for training on large-scale face images without alignment, and it achieves state-of-the-art on multiple challenging benchmarks including LFW [9] and YouTubeFaces [30]. In [18], Large-Margin softmax (L-Softmax) loss is proposed to explicitly reduce the intra-personal variations while enlarging the inter-personal differences. In SphereFace [17], a new loss of angular softmax (A-Softmax) is proposed and achieves excellent results on MageFace challenge [21]. Although the performance of face recognition on LFW and YTF datasets has reached human level [27, 25, 26, 22], there still is a gap between human performance and automatic face recognition with extreme pose, illumination, expression, age, resolution variation in unconstrained environment [23] such as the challenging IJB-A [14], mainly due to the different perception mechanism. Therefore in this work, inspired by the human perception mechanism, we propose a new contrastive convolution for better face recognition.

Adaptive Convolution There are some works exploring adaptive convolution to further improve the performance of CNN. In [4], kernels corresponding to an objective style can transform a given image from the original style to the objective style when convolving with the given image. In [38], scale-adaptive convolution is proposed to acquire flexible-size receptive fields during scene parsing for tackling the issue of inconsistent predictions of large objects and invisibility of small objects in conventional CNN. Most related to our work is those of dynamic input conditioned kernel generation, which includes, as far as we konw, dynamic convolution [15], dynamic filter network [11], and adaptive convolution [12]. Our work is fundamentally different from those works in two folds. First, the purpose of creating conditioned kernels is different. [15, 11] focus on image prediction task, and the dynamically-generated filters are mainly used to predict the movement of pixels between frames. [12] focuses on the supervised learning, and the dynamically-generated kernels aim at incorporating the given side information (e.g., camera tilt angle and camera height) into image features. Differently, our dynamically-generated kernels attempt to highlight the difference between two images for better face verification. Second, the mechanism of kernel generation is different. In [15, 11, 12], kernels are generated only according to one input, which thus characterize the specific feature of input relative to common or general feature, while our contrastive kernels are created according to a pair of images, which characterize the specific feature of an image relative to another.

3 Contrastive CNN

As mentioned above, the conventional CNN-based methods use the same feature of a face image no matter who it is compared with, while our proposed CNNs extract contrastive features of a face image based on who it is compared with. Contrastive features mainly describe those distinct characteristics between two faces which are extracted by the contrastive convolution proposed in this work. An overview of our method can be seen in Figure 2.

Specifically, the whole verification model, referred to as *Contrastive CNN*, consists of a trunk CNN and a kernel generator, forming a successive architecture. The truck CNN C is designed for base feature representation, which is shared between the two images for efficiency although it can be generally different. Based on these base feature representation, the kernel generator G produces personalized kernels for a face image, attempting to highlight those salient features of a face relative to the mean face. And the contrastive kernels are designed as the difference of personalized kernels of two faces, attempting to focus on those contrastive characteristics between them. By performing convolution with those contrastive kernels, contrastive features of two faces are extracted respectively for the similarity calculation.

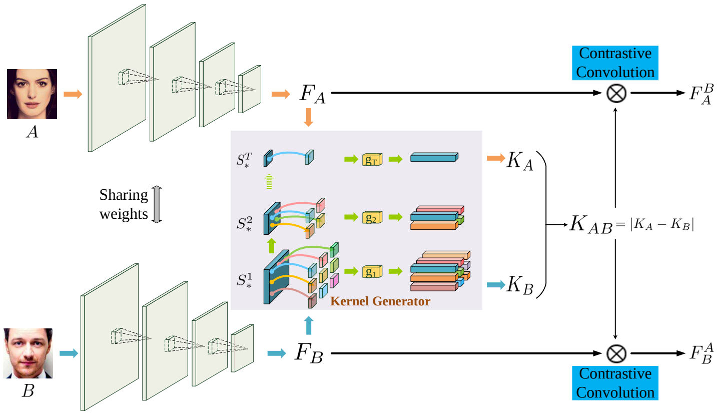

Fig. 2. The pipeline of our contrastive CNN. Given a pair of face images, A and B, a common feature extractor C consisting of several cascaded convolution layers is firstly used to obtain expressive feature representations F_A and F_B of them. Then, the kernel generator G consisting of several sub-generators generates personalized kernels for A and B respectively, based on which the contrastive kernels are achieved as $|K_A - K_B|$. Finally, with those contrastive kernels, the contrastive features of A and B are extracted via convolution operations respectively for the final similarity calculation. Note the subscript $*$ of S in kernel generator can be A or B.

3.1 Kernel generator

Denote a pair of face images as (A, B, L_{AB}), where A and B are face images, and L_{AB} is the label for them, with $L_{AB} = 1$ meaning that A and B are the same person, and $L_{AB} = 0$ meaning that A and B are different persons. The feature maps of A and B extracted from the feature extractor C are denoted as F_A and F_B respectively, i.e.

$$F_A = C(A), F_B = C(B) \in \mathbb{R}^{h_F \times w_F \times c_F} \tag{1}$$

where h_F, w_F and c_F are the height, width, and number of channels respectively.

The kernel generator aims at producing kernels specific to A or B, which is referred to as personalized kernels. Taking A as an example, the kernel generator G takes the feature maps F_A as input, and outputs a set of personalized kernels K_A, generally formulated as follows:

$$K_A = G(F_A) \tag{2}$$

Given only one face image A, i.e. with no reference face, it is impossible to obtain kernels depicting contrastive characteristics. So here, the generated kernels K_A is expected to highlight those intrinsic and salient features of A, which is the foundation of constructing contrastive convolutional kernels.

What's more, the kernel generator is designed with a hierarchical structure, allowing the personalized kernels to capture face characteristics at various scales, which can further effect on contrastive kernels K_{AB}. As shown in Figure 2, there are multiple layers in kernel generator network, one sub-generator for each layer, obtaining kernels with different receptive field as different layers are usually with feature map in different scale. Generally, the generator G consists of multiple layers, e.g. T layers, and on each layer a sub-generator is designed, forming T sub-generators in total:

$$G = \{g_1, g_2, \cdots, g_T\}. \tag{3}$$

Specifically, the feature maps from i^{th} layer are denoted as $S_A^i|_{i=1}^T$, which are usually obtained by using the convolving or fully connected operations on the feature maps of $(i-1)^{th}$ layer, i.e., S_A^{i-1} with $S^0 = F_A$.

On each layer, the sub-generator g_i is constructed to generate a group of kernels in the same scale as below:

$$K_A^i = \{k_A^{i1}, k_A^{i2}, ..., k_A^{iN_i}\}, \tag{4}$$

where N_i is the number of kernels generated from g_i. Each kernel k_A^{ij} is expected to portray the characteristics of a local component of face image A, achieved by using a local patch as input:

$$k_A^{ij} = g_i(p_A^{ij}), \tag{5}$$

$$p_A^{ij} = R(S_A^i, c_{ij}, h_K, w_K), \tag{6}$$

where p_A^{ij} is a local patch cropped from S_A^i with the center at c_{ij}, height of h_K, and width of w_K. Here, R denotes the image crop operation. Generally, these patches can be taken at regular grid for easy implementation. The sub-generator g_i can be any kind of deep network structure, such as convolution layer or full connection layer, and a small one is preferable. In all our experiments, the g_i consists of only one fully connected layer.

The kernels from one sub-generator share similar receptive field but focus on different components. Kernels from different sub-generators have different receptive fields paying attention to characteristics in different scales. Altogether, a set of personalized kernels can be obtained as the union of kernels from all the sub-generators as below:

$$K_A = \{k_A^{11}, ..., k_A^{1N_1}, ..., k_A^{ij}, ..., k_A^{T1}, ..., k_A^{TN_T}\}. \tag{7}$$

The personalized kernels generated from the generator G are expected to capture the intrinsic characteristics of an image, regardless of pose, illuminations, expression and etc, leading to a loss in Eq. (15). The personalized kernels K_B of B can be generated similarly.

Finally, the contrastive kernels are achieved as the difference of personalized kernels of two face images, attempting to only focus on those distinct characteristics between two faces and subtract the commonality, formulated as follows:

$$K_{AB} = |K_A - K_B|. \tag{8}$$

The contrastive kernels are dynamically generated by considering the two faces to compare in testing stage, which is flexible and adaptive to the testing faces, resulting in more accurate feature representation. As shown in Figure 4, the contrastive kernels created by Eq. (8) have high response to those different features, while low response to those common features between of the two faces as expected.

3.2 Contrastive Convolution

The contrastive convolution is very similar to conventional convolution, except that kernels used in contrastive convolution are dynamically generated according to different pairs being compared in the process of testing, while kernels used in conventional convolution are learned by large scale data and are fixed after training.

When comparing a pair of face images A and B, the contrastive features between A and B are extracted by convolving F_A and F_B with the contrastive kernels K_{AB} as follows:

$$F_A^B = K_{AB} \bigotimes F_A = [k_{AB}^{11} \otimes F_A; \cdots ; k_{AB}^{ij} \otimes F_A; \cdots , k_{AB}^{TN_T} \otimes F_A] \tag{9}$$

$$F_B^A = K_{AB} \bigotimes F_B = [k_{AB}^{11} \otimes F_B; \cdots ; k_{AB}^{ij} \otimes F_B; \cdots , k_{AB}^{TN_T} \otimes F_B] \tag{10}$$

where $\bigotimes$ means element-wise convolution. $K_{AB} \bigotimes F_A$ means each contrastive kernel in set K_{AB} is convolved with F_A.

With the contrastive feature representation of A and B, a simple linear regression followed by sigmoid activation is used to calculate the similarity S_A^B and S_B^A between A and B as follows:

$$S_A^B = \sigma(F_A^B \cdot W) \tag{11}$$

$$S_B^A = \sigma(F_B^A \cdot W) \tag{12}$$

Here, σ is sigmoid function with $\sigma(x) = \frac{\mathrm{e}^x}{1+\mathrm{e}^x}$, and $\cdot$ means dot product.

The final similarity S_{AB} between A and B is calculated as the average of the two similarities, i.e.

$$S_{AB} = \frac{1}{2}(S_A^B + S_B^A). \tag{13}$$

3.3 Overall Objective

With the contrastive convolution, the similarity between a pair of images from the same person is expect to be 1, i.e. $s_{AB} = 1$ and that from different persons is expect to be 0, i.e. $s_{AB} = 0$. The cross entropy loss is used to maximize the similarity of same face pairs, while minimize the similarity of different face pairs as follows:

$$\min_{C,G,W} L_1 = -\frac{1}{N}\sum_{A,B}[L_{AB}\log(S_{AB}) + (1 - L_{AB})\log(1 - S_{AB})] \tag{14}$$

Here, N means the number of face pairs, L_{AB} is the label of the face pair of A and B, in which $L_{AB} = 1$ means the positive face pair, and $L_{AB} = 0$ means the negative face pair.

Moreover, the personalized kernels is expected to capture the intrinsic characteristics of a face, which means that the personalized kernels of face images of the same person should have high similarity even if with various pose, illuminations or expressions, forming another cross entropy loss in the following:

$$L_2 = -\frac{1}{2N}\left[\sum_{A} l_A \log(H(K_A)) + \sum_{B} l_B \log(H(K_B))\right] \tag{15}$$

where $l_A \in \{0, 1\}^M$ and $l_B \in \{0, 1\}^M$ are the identity coding of A and B respectively in the form of one-hot coding with the number of persons as M. Here, $H(K) \in R^{M\times 1}$ is a small network used to regress the kernels to a one-hot code for classification.

Overall, the objective function of our CNN with contrastive convolution can be formulated as follows:

$$\min_{C,G,W,H} L_1 + \alpha L_2 \tag{16}$$

The α is a balance parameter, and is set as 1 in our experiments in addition to special instructions. This objective can be easily optimized by using the gradient decent same as most CNN based methods.

4 Experiments

In this section, we will evaluated our proposed CNN with contrastive convolution w.r.t. different architectures and compare with the state-of-art methods for face verification task on two wild challenging datasets: Labeled Faces in the Wild (LFW) [10], and IARPA Janus Benchmark A (IJB-A) [14].

4.1 Experimental settings

Datasets Three datasets are used for evaluation. The CASIA-WebFace [35] dataset is used for training, the LFW [10], and IJB-A [14] datasets are used for testing. The details of each dataset are as follows.

The **CASIA-WebFace** [35] dataset is a large scale face dataset containing about 10,000 subjects and 500,000 images collected from the internet. This

Table 1. Architectures of the CNN used in our method with 4, 10, 16 layers respectively. Conv1.x, Conv2.x, Conv3.x and Conv4.x mean convolution layers that contain multiple convolution units. For example, conv[256, 3, 1] denotes convolution with 256 filters of size 3 × 3, and stride 1. The max[3, 2] denotes the max pooling within a region of size 3 × 3, and stride 2. In CNNs with 10 and 16 layers, the residual network structure is used for better performance and the residual units are shown in the double-column brackets. In the last contrastive convolutional layer, the convolution is the same as conventional convolution except that its kernels are dynamically generated during testing.

Layer	4-layer CNN	10-layer CNN	16-layer CNN
input	112 × 112 × 3		
Conv1.x	conv[64, 3, 1]	conv[64, 3, 1]	conv[64, 3, 1]
Pool1	max[3, 2]		
Conv2.x	conv[128, 3, 1]	conv[128, 3, 1] $\begin{bmatrix}\text{conv}[128, 3, 1] \\ \text{conv}[128, 3, 1]\end{bmatrix} \times 1$	conv[128, 3, 1] $\begin{bmatrix}\text{conv}[128, 3, 1] \\ \text{conv}[128, 3, 1]\end{bmatrix} \times 2$
Pool2	max[3, 2]		
Conv3.x	conv[256, 3, 1]	conv[256, 3, 1] $\begin{bmatrix}\text{conv}[256, 3, 1] \\ \text{conv}[256, 3, 1]\end{bmatrix} \times 2$	conv[256, 3, 1] $\begin{bmatrix}\text{conv}[256, 3, 1] \\ \text{conv}[256, 3, 1]\end{bmatrix} \times 3$
Pool3	max[3, 2]		
Conv4.x	conv[512, 3, 1]	conv[512, 3, 1]	conv[512, 3, 1] $\begin{bmatrix}\text{conv}[512, 3, 1] \\ \text{conv}[512, 3, 1]\end{bmatrix} \times 1$
Pool4	max[3, 2]		
Contrastive Conv	conv[14, 3, 1]	conv[14, 3, 1]	conv[14, 3, 1]
features	686 dimensions		

dataset is often used to develop a deep network for face recognition in the wild, such as in [35, 8, 18, 17].

The **LFW** dataset [10] includes 13,233 face images from 5,749 different identities with large variations in pose, expression and illuminations. On this dataset, we follow the standard unrestricted protocol of with labeled outside data, i.e. training on the outside labeled CASIA-WebFace, and testing on 6,000 face pairs from LFW. Please refer to [10] for more details.

The **IJB-A** dataset [14] contains 5,712 images and 2,085 videos from 500 subjects captured from the wild environment. Because of the extreme variation in head pose, illumination, expression and resolution, so far IJB-A is regarded as the most challenging dataset for both verification and identification. A few example images of a subject from IJB-A can be seen in Figure 3. The standard protocol on this dataset performs evaluations by using template-based manner,

Fig. 3. Examplar images of a person from IJB-A dataset. Note the extreme variations of head poses, expression and image resolutions.

instead of image-based or video-based. A template may include images and/or videos of a subject.

Preprocessing For all three datasets, [31] is firstly used to detect the faces, then each detected face is aligned to a canonical one according to the five landmarks (2 eyes centers, 1 nose tip, and 2 mouth corners) obtained from CFAN [37], and finally all aligned images are resized into 128 × 128 for training or testing.

Settings of CNNs Tensorflow is used to implement all our experiments. For extensive investigation of our method, the proposed contrastive CNNs with base layers of 4, 10, and 16 are evaluated respectively. The detailed settings of the three CNNs are given in Table 1. Note that kernels in the last convolutional layer of our contrastive CNNs are dynamically generated in the testing stage with the kernel generator learnt in the training stage. We also compare our contrastive CNN with the conventional CNN, which is constructed by adding additional layer to the base CNN (referred to as L-Vanilla CNN) so that it has the same network structure as ours for fair comparison. The batch size is 128 for both methods, i.e. 128 images for baseline models and 64 pairs for our models. The face image pairs used in our method are randomly chosen from CASIA-WebFace with the same possibility between positive pairs and negative pairs when training. The length of personalized kernels are normalized to be 1 before they are used to calculate contrastive kernel. All models are trained with iterations as 200K, with learning rate as 0.1, 0.01 and 0.001 at the beginning, 100K iterations, and 160K iterations. Our contrastive CNN is designed with 3 sub-generators which generate 9, 4, and 1 contrastive kernels respectively, i.e. $T = 3, N_1 = 9, N_2 = 4, N_3 = 1$.

4.2 Ablation study of contrastive convolution

Effectiveness of contrastive convolution To show the improvement of our contrastive convolution, we compare our contrastive CNN with what we referred to

Table 2. Comparion between the vanilla CNN and our contrastive CNN. They share the same architecture for fair comparison.

Method	Loss	mAcc on LFW (%)	TAR(%)@FAR on IJB-A		
			0.1	0.01	0.001
L-VanillaCNN	Pairwise Loss	91.80	64.13	22.43	5.88
Contrastive CNN		95.20	78.73	52.51	31.37
L-VanillaCNN	Pairwise Loss +Softmax Loss	97.50	88.43	71.51	52.72
Contrastive CNN		98.20	90.24	74.55	58.04

Table 3. Performance of our Contrastive CNN with different number of sub-generators on LFW in terms of mean accuracy (mAcc) and IJB-A in terms of TAR (%) at FAR = 0.1, 0.01, and 0.001.

# sub-generator	mAcc on LFW	TAR(%)@FAR on IJB-A		
		0.1	0.01	0.001
1	97.83	87.06	64.95	37.32
2	98.17	89.92	75.08	57.08
3	98.20	90.24	74.55	58.04

as L-Vanilla CNN, which is constructed by adding additional layers that have similar structure with our kernel generator to the base CNN so that it has the same network structure as ours. In our contrastive CNN, a kernel classification loss in Eq. (15) is used to make the personalized kernels of a specific face image capture the intrinsic characteristics regardless of pose, illumination or expression. Therefore, the comparison is conducted on two cases that one is with pairwise loss + softmax loss, and the other is with only pairwise loss. The results are shown in Table 2. As can be seen, for both vanilla CNN and our contrastive CNN, the results with softmax loss+pairwise loss are better than that only with pairwise loss, demonstrating the superiority of the softmax loss as that in [33]. More importantly, for both cases with softmax loss+pairwise loss and pairwise loss only, our proposed contrastive convolution performs much better than the conventional convolution, with an improvement up to 30% at FAR = 0.01. These comparison clearly and convincingly show that our contrastive CNN can significantly improve the conventional CNN.

Contrastive convolution w.r.t. number of sub-generator Our kernel generator is organized with a hierarchical structure consisting of several sub-generators, where kernels created from different sub-generators are equipped with different scales. Here, we investigate the influence of the number of sub-generator, i.e. three contrastive CNNs of which the number of sub-generator is 1, 2, 3, re-

Table 4. Results of our Contrastive CNN with different base CNNs on LFW in terms of mean accuracy (mAcc) and IJB-A in terms of TAR at FAR = 0.1, 0.01, 0.001. Three base CNN structures with layers 4, 10, 16 are evaluated respectively, with architecture detailed in Table 1.

# Layers of base CNN	mAcc on LFW(%)	TAR(%)@FAR on IJB-A		
		0.1	0.01	0.001
4	98.20	90.24	74.55	58.04
10	98.93	93.17	80.35	61.83
16	99.12	95.31	84.01	63.91

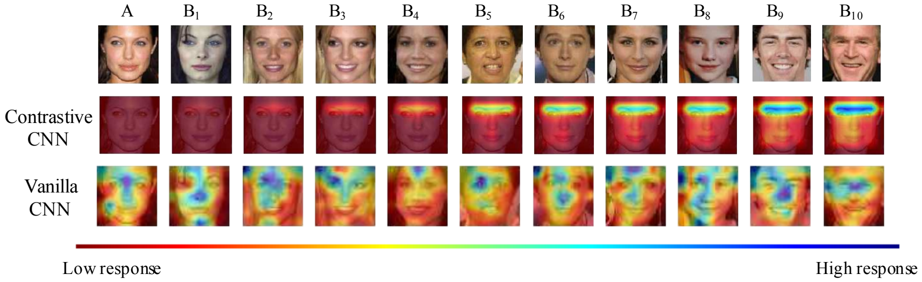

Fig. 4. Feature maps from our Contrastive CNN and Vanilla CNN for a given image A when comparing to images $B_1 \sim B_{10}$. These feature maps for contrastive CNN mainly focus on the region of eyes and eyebrows.

spectively and accordingly there are 9, 13, 14 contrastive kernels orderly in the 4-layer CNN shown in Table 1. The performance of Contrastive CNN with different number of sub-generator can be found in Table 3, where the performance is constantly improved with the increasing of number of sub-generator.

Contrastive convolution w.r.t. different architectures To further investigation, we demonstrate our contrastive convolution with different base CNNs. Three types of architecture with 4, 10, and 16 layers are used for evaluation, and the results are shown in Table 4. As can be seen, performance of our contrastive CNNs is constantly improved with the increasing of the depth of base CNN.

Visualization Comparison of Contrastive Features and Vanilla Features To further verify that those contrastive kernels can capture the differences between the two faces being compared. We visualize those feature maps from our contrastive CNN and vanilla CNN in Fig. 4. Specifically, an image A is compared to 10 images from B_1 to B_{10}. As can be seen, the high response of our contrastive CNN only lies in the area where A differs from the compared image, while the

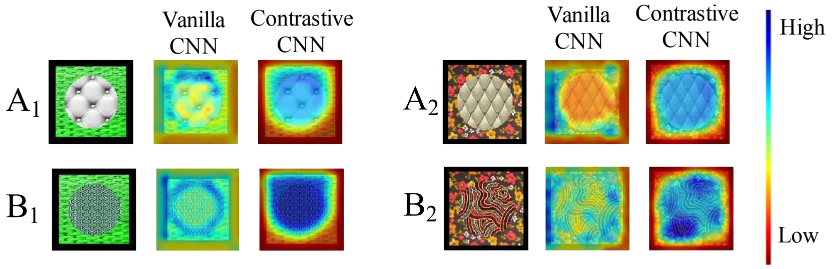

Fig. 5. Illustration of feature maps from contrastive CNN and vanilla CNN on the toy data for A_1 comparing with B_1, and A_2 comparing with B_2.

Table 5. Comparison on LFW in terms of mean accuracy (mAcc). * denotes the outside data is private (not publicly available).

Methods	# Models	Depth	Data	mAcc on LFW
DeepFace [27]	3	7	4M*	97.35
DeepID2+ [26]	1	5	300K*	98.70
Deep FR [22]	1	15	2.6M	98.95
FaceNet [25]	1	14	200M*	99.65
Yi et al. [35]	1	10	WebFace	97.73
Ding et al. [8]	1	14	WebFace	98.43
LargeMargin [18]	1	17	WebFace	98.71
SphereFace [17]	1	64	WebFace	99.42
Contrastive CNN(ours)	1	16	WebFace	99.12

high response of conventional CNN scatters over the whole image. Moreover, a toy experiment with images filled in simple geometry patterns is designed for more obvious illustration of feature maps from our contrastive CNN and conventional CNN, and the visualization is shown in Fig. 5. Both experiments clearly demonstrate that our contrastive CNN can focuses on the distinct characteristics between the two faces to compare as claimed.

4.3 Comparison with existing methods

Furthermore, our proposed method is compared with a few state-of-the-art methods. In this experiment, contrastive CNN with 16 layers is used for fair comparison as most existing methods are equipped with large architectures. All methods are tested on the LFW and IJB-A datasets as shown in Table 5 and Table 6. In Table 5, the proposed contrastive convolution outperforms all methods that are

Table 6. Comparison on IJB-A in terms of TAR (%) at FAR = 0.1, 0.01, and 0.001. Results of GOTS and OPENBR are from [14]. It is worth noting that our Contrastive CNN is not finetuned on the training splits of IJB-A, while some of those methods, such as [5][24] are finetuned on the training splits of IJB-A for better performance.

Methods	TAR(%)@FAR on IJB-A		
	0.1	0.01	0.001
OPENBR	43.3	23.6	10.4
GOTS	62.7	40.6	19.8
ReST [32]	-	63.0	54.8
FastSearch [29]	89.3	72.9	51.0
PAM [20]	-	73.3	55.2
DR-GAN [19]	-	75.5	51.8
Deep Multi-pose [1]	91.1	78.7	-
Triplet Similarity [24]	94.5	79.0	59.0
Joint Bayesian [5]	96.1	81.8	-
Contrastive CNN(ours)	95.31	84.01	63.91

trained on WebFace with reasonable number of layer. In Table 6, our method achieves the best results of TAR = 63.91% for FAR = 0.001 on IJB-A, which demonstrates the effectiveness of our contrastive CNN.

5 Conclusion

In this work, we propose a novel CNN architecture with what we referred to as contrastive convolution for face verification. Instead of extracting the same features of a face no matter who it is compared in conventional CNN, our method extracts contrastive features of a given face according to who it is compared with. The contrastive convolution is beneficial owing to its dynamitic generation of contrastive kernels based on the pair of faces being compared. The proposed contrastive convolution can be incorporated into any kind of CNN architecture. As evaluated on two wild benchmarks of LFW and IJB-A, the contrastive CNN achieves promising performance with significant improvement, demonstrating its effectiveness.

6 Acknowledgement

This work was partially supported by National Key Research and Development Program of China Grant 2017YFA0700804, Natural Science Foundation of China under contracts Nos. 61390511, 61650202, and 61772496.

References

1. AbdAlmageed, W., Wu, Y., Rawls, S., Harel, S., Hassner, T., Masi, I., Choi, J., Lekust, J., Kim, J., Natarajan, P., et al.: Face recognition using deep multi-pose representations. In: IEEE Winter Conference on Applications of Computer Vision (WACV) (2016)
2. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2006)
3. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis Machine Intelligence (TPAMI) (2002)
4. Chen, D., Yuan, L., Liao, J., Yu, N., Hua, G.: StyleBank: An explicit representation for neural image style transfer. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
5. Chen, J.C., Patel, V.M., Chellappa, R.: Unconstrained face verification using deep CNN features. In: IEEE Winter Conference on Applications of Computer Vision (WACV) (2016)
6. Chen, J.C., Patel, V., Chellappa, R.: Landmark-based fisher vector representation for video-based face verification. In: IEEE International Conference on Image Processing (ICIP) (2015)
7. Chen, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification. In: Advances in Neural Information Processing Systems (NIPS) (2014)
8. Ding, C., Tao, D.: Robust face recognition via multimodal deep face representation. IEEE Transactions on Multimedia (TMM) (2015)
9. Huang, G.B., Learned-Miller, E.: Labeled faces in the wild: Updates and new reporting procedures. In: Dept. Comput. Sci., Univ. Massachusetts Amherst, Amherst, MA, USA, Tech. Rep (2014)
10. Huang, G.B., Mattar, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments (2007)
11. Jia, X., De Brabandere, B., Tuytelaars, T., Gool, L.V.: Dynamic filter networks. In: NIPS (2016)
12. Kang, D., Dhar, D., Chan, A.: Incorporating side information by adaptive convolution. In: NIPS (2017)
13. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
14. Klare, B.F., Klein, B., Taborsky, E., Blanton, A.: Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
15. Klein, B., Wolf, L., Afek, Y.: A dynamic convolutional layer for short range weather prediction. In: CVPR (2015)
16. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (NIPS) (2012)
17. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

18. Liu, W., Wen, Y., Yu, Z., Yang, M.: Large-margin softmax loss for convolutional neural networks. In: International Conference on Machine Learning (ICML) (2016)
19. Luan, T., Yin, X., Liu, X.: Disentangled representation learning GAN for pose-invariant face recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
20. Masi, I., Rawls, S., Medioni, G., Natarajan, P.: Pose-aware face recognition in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
21. Miller, D., Brossard, E., Seitz, S., Kemelmachershlizerman, I.: MegaFace: A million faces for recognition at scale. arXiv preprint arXiv:1505.02108 (2015)
22. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition. In: British Machine Vision Conference (BMVC) (2015)
23. Phillips, P.J., Hill, M.Q., Swindle, J.A., O'Toole, A.J.: Human and algorithm performance on the pasc face recognition challenge. In: Biometrics Theory, Applications and Systems (BTAS) (2015)
24. Sankaranarayanan, S., Alavi, A., Chellappa, R.: Triplet similarity embedding for face verification. arXiv preprint arXiv:1602.03418 (2016)
25. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: A unified embedding for face recognition and clustering. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
26. Sun, Y., Wang, X., Tang, X.: Deeply learned face representations are sparse, selective, and robust. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
27. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: DeepFace: Closing the gap to human-level performance in face verification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
28. Turk, M.A., Pentland, A.P.: Face recognition using eigenfaces. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2002)
29. Wang, D., Otto, C., Jain, A.K.: Face search at scale. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2017)
30. Wolf, L., Hassner, T., Maoz, I.: Face recognition in unconstrained videos with matched background similarity. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2011)
31. Wu, S., Kan, M., He, Z., Shan, S., Chen, X.: Funnel-structured cascade for multi-view face detection with alignment-awareness. Neurocomputing (2017)
32. Wu, W., Kan, M., Liu, X., Yang, Y., Shan, S., Chen, X.: Recursive spatial transformer (ReST) for alignment-free face recognition. In: IEEE International Conference on Computer Vision (ICCV) (2017)
33. Wu, Y., Liu, H., Li, J., Fu, Y.: Deep face recognition with center invariant loss. In: ACM Multimedia ThematicWorkshops (2017)
34. Xie, S., Shan, S., Chen, X., Chen, J.: Fusing local patterns of gabor magnitude and phase for face recognition. IEEE Transactions on Image Processing (TIP) (2010)
35. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. arXiv preprint arXiv:1411.7923 (2014)
36. Zhang, B., Shan, S., Chen, X., Gao, W.: Histogram of gabor phase patterns (HGPP): A novel object representation approach for face recognition. IEEE Transactions on Image Processing (TIP) (2007)
37. Zhang, J., Kan, M., Shan, S., Chen, X.: Leveraging datasets with varying annotations for face alignment via deep regression network. In: IEEE International Conference on Computer Vision (ICCV) (2015)

38. Zhang, R., Tang, S., Zhang, Y., Li, J., Yan, S.: Scale-adaptive convolutions for scene parsing. In: IEEE International Conference on Computer Vision (ICCV) (2017)