Pose-Robust Face Recognition via Deep Residual Equivariant Mapping

Kaidi Cao[2*] Yu Rong[1,2*] Cheng Li[2] Xiaoou Tang[1] Chen Change Loy[1]

[1]CUHK - SenseTime Joint Lab, The Chinese University of Hong Kong
[2]SenseTime Research
{ry017, ccloy, xtang}@ie.cuhk.edu.hk {caokaidi, chengli}@sensetime.com

Abstract

Face recognition achieves exceptional success thanks to the emergence of deep learning. However, many contemporary face recognition models still perform relatively poor in processing profile faces compared to frontal faces. A key reason is that the number of frontal and profile training faces are highly imbalanced - there are extensively more frontal training samples compared to profile ones. In addition, it is intrinsically hard to learn a deep representation that is geometrically invariant to large pose variations. In this study, we hypothesize that there is an inherent mapping between frontal and profile faces, and consequently, their discrepancy in the deep representation space can be bridged by an equivariant mapping. To exploit this mapping, we formulate a novel Deep Residual EquivAriant Mapping (DREAM) block, which is capable of adaptively adding residuals to the input deep representation to transform a profile face representation to a canonical pose that simplifies recognition. The DREAM block consistently enhances the performance of profile face recognition for many strong deep networks, including ResNet models, without deliberately augmenting training data of profile faces. The block is easy to use, light-weight, and can be implemented with a negligible computational overhead [1].

1. Introduction

The emergence of deep learning greatly advances the frontier of face recognition [29, 30]. The main focus tends to center around near-frontal faces while there can be no assurance of view consistency when face recognition is conducted in unconstrained environments. Although human performance only drops slightly from frontal-frontal to frontal-profile face verification, many existing algorithms can suffer a drop of over 10% [26]. Thus, large pose variation remains to be a significant challenge that confronts real-world face recognition.

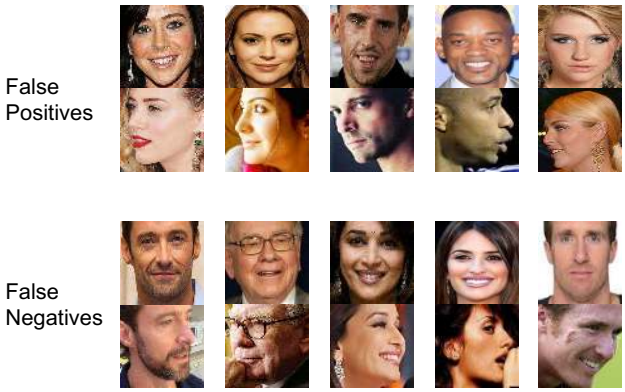


Figure 1. A state-of-the-art face recognition model [34] tested on a challenging frontal-profile faces dataset [26]. It is observed that profile faces of different persons are easily to be mismatched (false positives), and profile and frontal faces of the same identity may not trigger a match leading to false negatives.

We provide an example in Figure 1 to show the failure modes of a state-of-the-art face verification model. We trained the same ResNet-18 model as reported in [34]. This model achieves a high accuracy of 99.3% on the LFW benchmark [12]. Despite the strong model, it tends to falsely match profile faces of different identities yielding a number of false positives. In addition, the model is also likely to miss frontal and profile faces of the same identity leading to false negatives.

Why does face recognition work poorly on profile faces? Modern deep learning is heavily data-driven [11, 7]. The generalization power of deep models is usually proportional to the training data size. Given an uneven distribution of profile and frontal faces in the dataset, deeply learned features tend to bias on distinguishing frontal faces rather than profile faces. When it is infeasible to collect a massive dataset that covers all possible poses with even distribution,

*indicates shared first authorship
[1]Codes and models are available at http://mmlab.ie.cuhk.edu.hk/projects/DREAM/

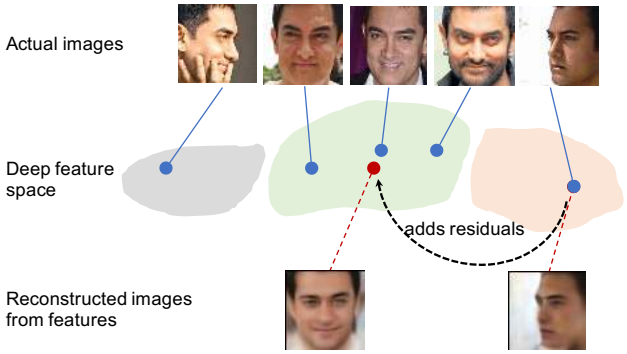


Figure 2. At the top of this figure, we illustrate the deep feature embedding of a subject in different poses. The proposed DREAM block is capable of adding residual to the feature of a profile face and map it to the frontal space. At the bottom of the figure, we show the actual reconstructed image of a profile face and its mapped frontal face.

researchers have turned to alternative approaches to better handle the recognition of profile faces. A large body of methods normalize images to a single frontal pose before recognition, either through elaborated dense 3D facial landmark detection and warping [30], or another deep model (or generative adversarial network) specialized in face frontalization [31]. Such methods would add processing burden to the whole system. In addition, face frontalization in the wild, especially with extreme profile faces, is still considered challenging. Often, synthesized 'frontal' faces would contain artifacts caused by occlusions and non-rigid expressions. Another potential solution is divide-and-conquer, *i.e.*, training separate models for learning pose-specific identity features [19]. This strategy tends to increase computational cost due to the use of multiple models.

In this study, we hypothesize that the profile face domain possesses a gradual connection with the frontal face domain in the deep feature space. Figure 2 illustrates a deep representation embedding of faces belong to the same subject but in different poses. Given an input image of arbitrary pose, we can actually map its feature to the frontal space through a mapping function that adds residual. This observation is closely connected to the notion of *feature equivariance* [15], which finds the representation of many deep layers depends upon transformations of the input image. Interestingly, such transformations can be learned by a mapping function from data and the function can be subsequently applied to manipulate the representation of an input image to achieve the desired transformation.

Motivated by this observation, we formulate a novel module called *Deep Residual EquivAriant Mapping* (DREAM) block, which can model the transformation between frontal-profile faces in the high-level deep feature space. The block adaptively adds residuals to an input representation to transform a profile face to a canonical pose to simplify recognition. The residuals are generated conditioned on the preceding feature representation via a few additional weight layers. To accommodate input faces of arbitrary pose, a soft gate is introduced to adaptively control the amount of residuals such that more residuals are added to extreme profile faces while keeping the representation unchanged if the input is already in a frontal pose.

Our work is conceptually related to the face frontalization [31] in that our approach also performs 'frontalization' but not in the image space. We observe from our experiments that transforming profile face features to frontal features could yield better performance than image-level frontalization, which is susceptible to the negative influence of artifacts as a result of image synthesis. To our best knowledge, this study is the first attempt to perform *profile-to-frontal face transformation in the deep feature space*.

The DREAM block is appealing in several aspects:

1. It is simple to implement. Specifically, the DREAM block is implemented as a simple yet effective gated residual branch. It can be integrated into existing convolutional neural network (CNN) architectures through stitching the block to the base network. It does not alter the original dimensionality of the face representation and can be trained end-to-end with standard back-propagation.
2. It is light-weight. It adds only a tiny amount of parameters and computation to the base model. For instance, it only adds 0.3% parameters on ResNet-18 and increases its forward time by 1.6%.
3. The proposed approach helps a base network that already does well in near-frontal face recognition to gain better performance in recognizing faces with extreme pose variation. This is done without elaborated data augmentation and face normalization that is practiced by most existing face recognition studies. For instance, the DREAM block reduces the error of ResNet-18 and ResNet-50 by 16.3% and 23.7% on the CFP benchmark [26]. In addition, it also gains 8.5% improvement on the verification task (TAR@FAR=0.001) and reduces the error rate by 17.5% on the identification task (Rank-1 accuracy) on the IJB-A dataset [4] for ResNet-18. For ResNet-50, the numbers are 7.0% and 12.6%, respectively.

2. Related Work

Deep Learning for Face Recognition. Deep learning is the prominent technique for face recognition. Most existing studies deploy CNNs, but with different loss functions, such as contrastive loss [29], triplet loss [25], and center loss [33]. Center loss represents the current state-of-the-art approach that learns a center for deep features of each class and simultaneously minimize the distances between the deep features and their corresponding class centers, thus intra-class features variations are reduced and discrimina-

tive power of learned features are enhanced. Prior to center loss, Joint Bayesian [3] is widely used to derive a similarity metric for robust face verification. The aforementioned metric learning methods facilitate more robust verification given faces with arbitrary poses. Nevertheless, as shown in our experiments, we found that the DREAM block performs better, especially on extreme profile faces (such as those in the Celebrities in Frontal-Profile (CFP) dataset [26]).

Profile Face Recognition. It is not new for researchers to take pose variations into consideration [26, 19, 38] when dealing with the face recognition problem. Existing methods address profile face recognition through elaborated dense 3D facial landmark detection and warping [30], face frontalization [31], or training separate models for learning pose-specific identity features [19]. There are alternative approaches. For instance, Masi *et al*. [20] enhances the performance of CNN through augmenting training and test data with face images differ in 3D shape, expression and pose. Yin *et al*. [35] propose a multi-task CNN that exploits side tasks, *e.g*., pose, to serve as regularizations for learning pose-specific identity features. In contrast to the aforementioned studies that require elaborated data augmentation or multi-task training, our approach is light-weight and easy to implement.

3. Deep Residual Equivariant Mapping

We first present a background on feature equivariance [15], and subsequently use it to motivate the proposed DREAM block. We will then present the block's design and different ways we could employ it to achieve pose-robust face recognition.

3.1. Feature Equivariance

The notion of feature equivariance is presented in [15]. It looks at how a representation changes upon transformations of the input image. An important finding of this paper is that most of the layers in deep neural networks change in an easily predictable manner with the input. And such transformations can be learned empirically from data.

Formally, a convolutional neural network (CNN) can be regarded as a function ϕ that maps an image $\mathbf{x} \in \mathcal{X}$ to a vector $\phi(\mathbf{x}) \in \mathbb{R}^d$. The representation ϕ is said equivariant with a transformation g of the input image if the transformation can be transferred to the representation output [15]. That is, equivariance with g is obtained when there exists a map $M_g : \mathbb{R}^d \rightarrow \mathbb{R}^d$ such that

$$\forall \mathbf{x} \in \mathcal{X} : \quad \phi(g\mathbf{x}) \approx M_g \phi(\mathbf{x}). \tag{1}$$

Interestingly, by requiring the same mapping M_g to work for any input image, the function would capture intrinsic geometric properties of the representations. In [15], the authors focus on geometric transformation such as affine warping and flips of images. In our problem context, the transformation g is more challenging as it involves 3D geometric changes from profile to frontal faces.

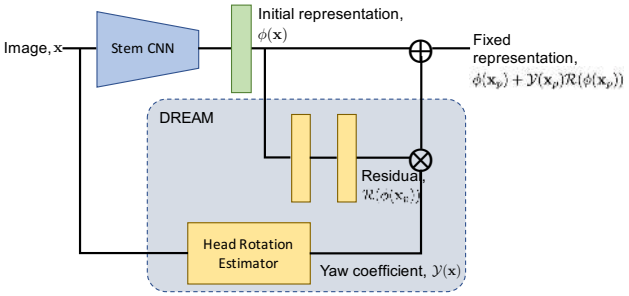


Figure 3. The DREAM block is designed with simplicity in mind and it can be easily added to existing CNNs. The block dynamically adds residuals to an input representation to transform a profile face to a canonical pose to simplify recognition.

3.2. Problem Formulation and the DREAM Block

Face recognition relies on robust representation extracted from deep models. For instance, DeepID series [29] extract features from the penultimate layer (the fully-connected layer before the output layer) as a feature vector and feed it into a classifier. Our goal in this study is to design a light-weight solution to make the representation robust to pose variation.

We begin by introducing the problem formulation. We denote a CNN as a function ϕ and the image representation it maps from image $\mathbf{x}$ as $\phi(\mathbf{x})$. We call the network a stem CNN or base network. Let's assume that we are given two types of face images, namely frontal face image, represented as $\mathbf{x}_f$ and profile face image, denoted by $\mathbf{x}_p$. Note that we assume this setting to facilitate our discussion; the proposed method can work with faces of an arbitrary pose.

Motivated by Eq. (1), we wish to obtain a transformed representation of a profile image $\mathbf{x}_p$ through a mapping function M_g, so that $M_g\phi(\mathbf{x}_p) \approx \phi(\mathbf{x}_f)$. To facilitate the incorporation of $M_g\phi(\mathbf{x}_p)$ to a stem CNN, we formulate it as a sum of the original profile feature $\phi(\mathbf{x}_p)$ with residuals given by a residual function $\mathcal{R}(\phi(\mathbf{x}_p))$ weighted by a yaw coefficient $\mathcal{Y}(\mathbf{x}_p)$. That is

$$\begin{aligned} \phi(g\mathbf{x}_p) &= M_g\phi(\mathbf{x}_p) \\ &= \phi(\mathbf{x}_p) + \mathcal{Y}(\mathbf{x}_p)\mathcal{R}(\phi(\mathbf{x}_p)) \\ &\approx \phi(\mathbf{x}_f). \end{aligned} \tag{2}$$

By performing this transformation, we wish that the fixed representation $\phi(\mathbf{x}_p) + \mathcal{Y}(\mathbf{x}_p)\mathcal{R}(\phi(\mathbf{x}_p))$ will be mapped to the frontal face space as illustrated in Figure 2.

Eq. (2) is capable of coping with input images of arbitrary pose, thanks to the yaw coefficient $\mathcal{Y}(\mathbf{x}) \in [0, 1]$, which acts as a *soft gate* of the residuals. The role of this soft gate is to provide a higher magnitude of residuals (thus

a heavier fix) to a face that deviates more from the frontal pose. Intuitively, $\mathcal{Y}(\mathbf{x}) = 0$ if the face image is frontal, and its value gradually changes from 0 to 1 when the face pose shifts from frontal to a complete profile. The residual's magnitude is thus the largest at the complete profile pose. The soft gate is essential. Without it residuals $\mathcal{R}(\phi(\mathbf{x}))$ will be added blindly to input images of any poses, affecting the final face recognition performance.

The soft gated residual block presented in this study is related to the residual structure [10] that is used to achieve identity mapping in very deep networks. While [10] utilizes the residual block to increase the effective depth of networks. Our attempt of combining a soft control gate can be viewed as a correction mechanism that adopts top-down information (the yaw in our case) to influence the feed-forward process. The role of the proposed control gate is to determine the amount of residuals to be passed to the next level. Besides, it is worth noticing that roll and pitch angles are not considered. The effect of roll will be eliminated by face alignment while face images with large pitch angles are rare and it is possible to address pitch angles by adding another branch in our DREAM block.

Architecture and Training. The residual formulation in Eq. (2) allows us to design a succinct network structure shown in Figure 3. Specifically, we use a stem CNN, *e.g.*, ResNet-18 or ResNet-50, to extract features from the input face image. To adapt ResNet [10] for our recognition task, we add a 256-dimensional fully connected layer between the average pooling layer and the original fully connected layer. We call this newly added layer as feature layer. The stem CNN can also be of any of the existing face recognition models [29, 25, 33]. A fully connected layer is then used to extract the initial representation, $\phi(\mathbf{x})$, which is subsequently 'fixed' by the DREAM block. The DREAM block, in our current implementation, consists of two branches:

Residual Branch. The first branch generates the residuals $\mathcal{R}(\phi(\mathbf{x}))$. It has two fully-connected layers with Parametric Rectified Linear Unit (PReLU) [9] as the activation function. This branch is learnable separately from the stem CNN. Specifically, we train it by minimizing the Euclidean distance between the mapped profile feature and its corresponding frontal feature using stochastic gradient descent.

$$\min_{\Theta_R} \mathrm{E}\|\phi(\mathbf{x}) + \mathcal{Y}(\mathbf{x})\mathcal{R}(\phi(\mathbf{x}); \Theta_{\mathcal{R}}) - \phi(\mathbf{x}_f)\|_2^2, \quad (3)$$

where $\Theta_{\mathcal{R}}$ denotes the parameters of $\mathcal{R}(\cdot)$. We keep the parameters fixed for the $\mathcal{Y}(\cdot)$ branch. During the training process we use dropout strategy [27] to the last fully connected layer in the branch. In this work, we train this branch on frontal-profile pairs sampled from MS-Celeb-1M dataset.

Soft Gate with Head Rotation Estimator. The second branch produces the soft yaw coefficient $\mathcal{Y}(\mathbf{x})$. This branch assumes an input of 21 facial landmarks following the standard AFLW's protocol [14]. Note that this requirement does not add additional burden to the stem CNN since the face alignment process is a standard preprocessing step of many face recognition pipelines[2].

Given facial landmarks, the head rotation estimator in the second branch (see Figure 3) estimates the head rotation by using the algorithm presented in [37]. Specifically, we adopt the same definition of the face model and head coordinate system as [28]. Slightly different from [37], we extend their 6-point 3D face model into a 21-point model to achieve a better performance. We then fit the model by estimating the initial solution using the EPnP algorithm [16], and further refining the pose via non-linear optimization.

The yaw angle obtained from previous steps is then non-linearly mapped to a positive value within the range of $[0, 1]$. We obtain the yaw coefficient by $\sigma(\lambda(\frac{4}{\pi}|y| - 1))$, where y is the yaw angle of a face image in radian units, λ is the coefficient(λ is set to be 10 in our experiments) and σ is a sigmoid function. Through this mapping the coefficient quickly reaches the value 1 once a face turns more than 45°, exerting more residuals for extreme profile faces. Empirically, we found that by adding a monotonous nonlinear mapping to the coefficient improves the effectiveness of residuals to the representation.

The outputs of these two branches are multiplied and added to the initial representation $\phi(\mathbf{x})$. The resulting feature $\phi(\mathbf{x}) + \mathcal{Y}(\mathbf{x})\mathcal{R}(\phi(\mathbf{x}))$ is the final feature output. It is worth noting that pose estimation accuracy has little effect on the final performance. We use the state-of-the-art face alignment method proposed in [39] to obtain the 21 facial landmarks. The method is specially designed to handle large-pose alignment. We have tried adding 20% noise to the estimated yaws in both training and test sets and found that EER in face recognition increases by no more than 2% under multiple settings. This result is still a lot more better than baselines.

3.3. The Usages of DREAM

We describe three ways of utilizing DREAM. Comparative results on all strategies will be provided in the experiment section.

Stitching. The most convenient way of deploying the DREAM block is by 'stitching' the block directly to a trained stem CNN. In particular, given a base network, we can just stitch the DREAM block onto the final feature layer of the base network without changing any learned parameters of the original model.

End-to-end. The proposed light-weight block can also be trained together with the stem CNN in an end-to-end manner. Given a plain base network, we insert the DREAM

[2]In many face recognition pipelines, face alignment is used to center the face, rotate the face such that the eyes lie along a horizontal line, and scale the faces such that they are approximately identical in size.

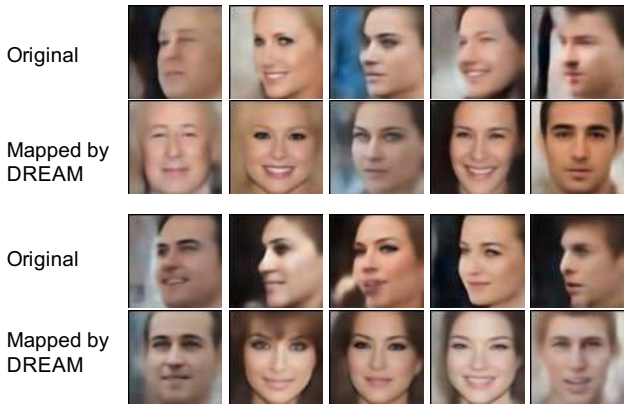


Figure 4. Visualization of deep features. The first and third rows show the reconstructed original features of profile faces. The second and fourth row depict the reconstructed features after the mapping by DREAM block.

block and directly train the new network with random initialization on all of its parameters. If the stem CNN is not plain but trained previously, we can first fine-tune the stem CNN while training the DREAM block end-to-end using an existing face recognition loss (*e.g.*, verification loss, identification loss, or both). We name the strategy as 'end2end'. With this strategy, the performance on profile faces is not guaranteed since the DREAM block may not be able to distinguish frontal and profile cases, since no specific frontal-profile face pairs are used for training the block.

End-to-end+retrain. We train the stem CNN and DREAM block together then train the DREAM block separately with frontal-profile face pairs.

As a way to demonstrate the effectiveness of DREAM block, we use the GAN model in [21] that could map deep features back to reconstructed images. This GAN model is used to visualize the original and mapped features generated by our model. Some representative results are shown in Figure. 4. Note that the reconstruction only serves for visualization purposes. The usefulness of DREAM can only be fully validated through examining its performance on the face recognition task. We show strong performance of DREAM in the following section.

4. Experiments

We divide our experiments into a few sections. In the first section, we show the effectiveness of our method on the task of frontal to complete profile face verification. In the second section, we compare with state-of-the-art methods on a dataset with full-pose variation. Lastly, we provide a further analysis of the influence of face yaw and present an ablation study.

We provide a description of the training set we use before moving to the next section. To train our own stem CNNs (ResNet-18 and ResNet-50) and DREAM block, we employ a subset of MS-Celeb-1M [7]. The original data provides images of 100,000 top celebrities in the form of URL, which, however, are collected from Google without manual cleaning. To facilitate the learning of our model, we clean the data and select a subset to form our training and test partitions. The training partition consists of 696,446 images from 13,385 identities. The testing partition contains 70,730 images from 3,084 celebrities. The training and test partitions have exclusive identities. The test partition is only used in our further analysis and ablation study since it contains more profile faces. All face images in training and testing sets are prepocessed to a size of 224×224. We use the method presented in [36] and [39] for face detection and face alignment, respectively. To train our stem CNN, we use an identification loss [29]. As in [22], face verification is performed by measuring the cosine distance between feature representation of queries.

4.1. Evaluation on CFP with Frontal-Profile Setting

Test dataset. We first conduct evaluations on the Celebrities in Frontal-Profile (CFP) dataset [26], a challenging dataset created to examine the problem of frontal to profile face verification 'in the wild'. The dataset contains 500 celebrities, each of which has ten frontal and four profile face images. We follow the standard 10-fold protocol [26] in our evaluation. In particular, the whole dataset is divided into 10 folds each containing 350 same and 350 not-same pairs generated from 50 individuals (7 same and 7 not-same pairs for each individual). The same protocol is applied on both the Frontal-Profile and Frontal-Frontal settings.

Stem CNN. To show the benefits of DREAM on different base networks, we perform our experiments using the following stem CNN architectures. Following [34], we employ a ResNet-18 as our base network. We also try ResNet-50. In addition, we experiment with a state-of-the-art network presented in [33][3], which stacks 11 residual blocks and trains with Center-Loss.

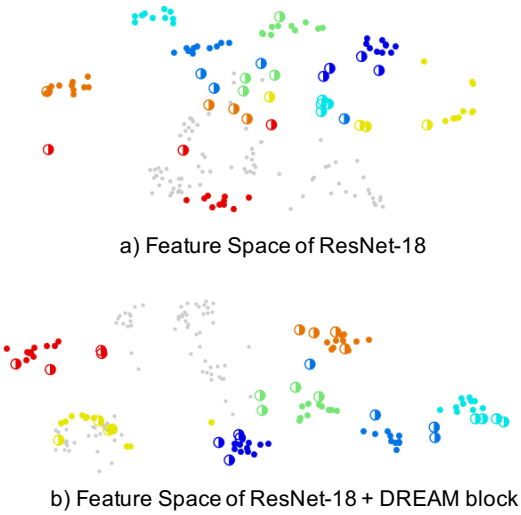
Baselines. We compare our method with three representative baselines that also help to alleviate the gap in frontal-profile face verification:

- CDFE [17] - Common Discriminant Feature Extraction is a representative method that is specially tailored to the inter-modality problem. In the algorithm, two transforms are simultaneously learned to map the samples in two modalities respectively to the common feature space.
- JB [3, 2] - Joint Bayesian is a widely used metric learning approach in face verification, which also supports the transfer from one domain to another. We train a JB model on the CFP dataset. Each time, we use 9 splits for training and the rest for testing. We repeat

[3]We use the codes released by the authors at https://github.com/ydwen/caffe-face.

Table 1. Results on Celebrities in Frontal-Profile (CFP) with Frontal-Profile setting. Equal error rate (EER) is reported. Lower is better. The best result in each row are given in bold.

			Other Strategies			DREAM Variants		
Model	Training Data	Naïve	CDFE [17]	JB [3]	FF [31]	stitching	end2end	end2end+retrain
ResNet-18	MS-Celeb-1M	8.40	8.30	8.37	14.40	7.71	7.63	**7.03**
ResNet-50	MS-Celeb-1M	7.89	7.71	8.49	14.26	7.29	6.43	**6.02**
Center-Loss	MS-Celeb-1M	8.54	8.49	8.29	14.53	7.82	7.81	**7.26**



a) Feature Space of ResNet-18

b) Feature Space of ResNet-18 + DREAM block

Figure 5. Visualization of deep feature space. Here we use solid dots • to represent frontal faces and symbols ◑ to denote profile faces. The features of different subjects are represented in different colors. Figure (a) shows the embedding yielded by a ResNet-18 model. It tends to cluster profile faces around the same point in the space. Figure (b) shows the embedding obtained by ResNet-18 with the DREAM block. Profile faces are separated more clearly and clustered with their own class of frontal faces.

this process for 10 times and report the average error. The DREAM block does not need such fine-tuning.

- FF [31] - Face Frontalization morphs faces from profile to frontal with a generative adversarial network. This baseline converts all profile face images in the CFP test partition to frontal ones.

In addition, we also evaluate the effectiveness of different training strategies of DREAM as discussed in Section 3.3, namely, 'stitching', 'end2end', and 'end2end+retrain'.

Results. The results are summarized in Table 1. We report the Equal Error Rate (EER). We have following observations:

1) Through the comparisons between Naïve and DREAM variants, we found that all DREAM strategies are capable of reducing the EER of various strong models, in spite of their different designs and training sources. The strategy 'end2end+retrain' gives the best performance overall. Through visualizing the deep feature space of 'Naïve' ResNet-18 via t-SNE in Figure 5, we could observe that features of frontal faces are clearly separated based on identities, but the profile ones are mostly messed up exhibiting severe feature overlapping in the space. In contrast, ResNet-18 with the proposed DREAM block clearly separates the features of different subjects without affected by the pose factor.

2) Baselines such as CDFE, JB and FF utilize the same supervision as our approach but they are less effective. The fact that FF performs poorer than DREAM in this case suggests that performing 'frontalization' in the deep feature space rather than the image space turns out to be more fruitful. We conjecture that the artifacts of 'frontalized' faces lead to the inferior results of FF to Naïve baseline. Some frontal images synthesized by FF are given in the supplementary material.

3) Our best result, which is obtained by ResNet-18 and ResNet-50 trained with MS-Celeb-1M and 'end2end+retrain' strategy, outperforms many existing methods, including [5], [23], and [26] that achieves 8.00, 8.85, and 14.97, respectively.

It is worth pointing out that the impact introduced by the proposed block on the frontal face recognition is minimal. In particular, the block hardly affects the performance of non-profile face recognition since the soft yaw coefficient prevents the block from altering the frontal representation. In particular, for ResNet-18, the EER of 'Naïve' on Frontal-Frontal setting is 2.66, while that of 'stitching', 'end2end'. and 'end2end+retrain' are 2.69, 2.00 and 2.00, respectively.

4.2. Evaluation on IJB-A with Full Pose Variation

Test dataset. In previous experiments on CFP we focus on frontal-profile face verification. In this experiment, we further evaluate our method on another challenging benchmark called IARPA Janus Benchmark A (IJB-A) [13] that covers full pose variation (yaw angles between $-90°$ to $+90°$). The dataset contains 500 subjects with of 5,712 images and 20,414 frames extracted from videos. The faces in the IJB-A dataset contain extreme poses and illuminations, more challenging than LFW [12]. Following the standard protocol in [13], we evaluate our method on both verification (1:1) and identification tasks (1:N).

Stem CNN. We use ResNet-18 and ResNet-50 trained on MS-Celeb-1M as our stem CNN. The DREAM block is deployed using our 'end2end+retrain' strategy.

Results. Table 2 reports our results on IJB-A. It is noteworthy that we employ a strong baseline: a ResNet-18 trained

Table 2. Comparative performance analysis on IJB-A benchmark. Results reported are the 'average±standard deviation' over the 10 folds specified in the IJB-A protocol. Symbol '-' indicates that the metric is not available for that protocol. Standard deviation is not available for all the methods. f.t. denotes fine tuning a deep network multiple times for each training split.

Methods ↓	Verification		Identification	
Metrics →	TAR @ FAR=0.01	TAR @ FAR=0.001	Rec. Rate @ Rank-1	Rec. Rate @ Rank-5
Our Approach with MS-Celeb-1M subset:				
ResNet-18 (naïve)	0.840±0.026	0.656±0.040	0.897±0.016	0.951±0.011
ResNet-18 (end2end+retrain)	0.872±0.018	0.712±0.035	0.915±0.012	0.962±0.008
ResNet-50 (naïve)	0.881±0.018	0.714±0.034	0.913±0.013	0.957±0.010
ResNet-50 (end2end+retrain)	0.891±0.016	0.764±0.031	0.924±0.016	0.962±0.010
Our Approach with full MS-Celeb-1M:				
ResNet-18 (naïve)	0.934±0.009	0.836±0.016	0.939±0.012	0.960±0.010
ResNet-18 (end2end+retrain)	**0.944±0.009**	0.868±0.015	**0.946±0.011**	**0.968±0.010**
Existing Methods:				
Wang *et al*. [32]	0.729±0.035	0.510±0.061	0.822±0.023	0.931±0.014
Pooling Faces [8]	0.819± ——	0.631± ——	0.846± ——	0.933± ——
Deep Multi-Pose [1]	0.787± ——	–	0.846± ——	0.927± ——
Multi-task CNN [35]	0.775±0.025	0.539±0.042	0.858±0.014	0.938±0.009
PAMs [19]	0.826±0.018	0.652±0.037	0.840±0.012	0.925±0.008
$\text{DCNN}_{\text{fusion}}$ (f.t.) [4]	0.838±0.042	–	0.903±0.012	0.965±0.008
Augmentation+Video Pooling+Rendered Test [20]	0.886± 0.017	0.725± 0.044	0.906± 0.013	0.962± 0.007
$\text{CNN}_{\text{media}}$+TPE (f.t.) [24]	0.900±0.010	0.813±0.020	0.932±0.010	–
Template Adaptation (f.t.) [6]	0.939±0.013	–	0.928±0.010	–
Quality Aware Network (f.t.) [18]	0.942±0.015	**0.893±0.039**	–	–

on the MS-Celeb-1M dataset, which achieved 65.6% True Acceptance Rate (TAR) at False Acceptance Rate (FAR) of 0.001 on the verification task and a Rank-1 recognition accuracy of 89.7% on the identification task, comparable to a state-of-the-art method [4]. By adding the DREAM block, the performance of this strong baseline is improved by **8.5**% on the verification task (TAR@FAR=0.001) and the error rate is reduced by **17.5**% on the identification task (Rank-1 accuracy). For the even stronger ResNet-50, the DREAM block also offers a compelling error reduction of 7.0% and 12.6%, respectively.

The aforementioned results are generated by using a sampled subset of MS-Celeb-1M dataset for training. Once we use the full MS-Celeb-1M dataset, we achieve the state-of-the-art results on the identification task and comparable verification performance against Quality Aware Network [18] that uses a more elaborated loss (triplet+softmax) for training.

There are other strategies that can be used to further improve the performance but we did not attempt each of them. We included these methods in Table 2 for completeness. Most of the techniques perform excellently when they fine-tune their model multiple times for each training split. Such methods are denoted with 'f.t.' in Table 2. Our model is trained on MS-Celeb-1M and directly tested on IJB-A without fine-tuning. Data augmentation technique can be useful too. Masi *et al*. [20] perform task-specific data augmentation on pose, shape, and expression on CASIA-WebFace. Their method also performs pose synthesis at test time. We believe the same technique can be used to improve the performance of our approach. Sankaranarayanan *et al*. [24] deploy a metric learning approach to fine-tune the network on the training splits of IJB-A. Liu *et al*. [18] train a network with joint triplet and softmax losses. Our method is only trained using the typical identification loss and without fine-tuning.

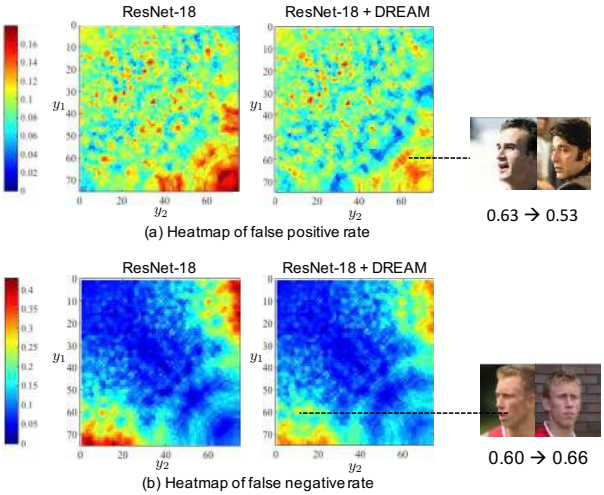


Figure 6. A comparison between the false positive rate and false negative rate between the Naïve ResNet-18 and ResNet-18 + DREAM on the yaw space. (a) and (b) show heatmaps of False positive rate and False negative rate given face pairs of different yaws (y_1, y_2). $0.63 \rightarrow 0.53$ means the cosine similarity drops from 0.63 to 0.53 after the DREAM block was utilized.

4.3. A Further Analysis on Influences of Face Yaw

We conduct a more in-depth analysis on the influence of face yaw to the performance of face verification to better un-

Table 3. Comparative analysis of different architectures of DREAM block. Evaluation is conducted on MS-Celeb-1M with ResNet-18. Results are reported as EER error.

Position of DREAM	Components of DREAM	EER on MS-Celeb-1M
After pooling layer	two fc	9.22
After feature layer	one fc	8.92
After feature layer	one fc, one conv	9.18
After feature layer	two fc	**8.45**

derstand the effectiveness of DREAM block on profile face recognition. In the analysis, we assume a query face and a matching face with yaw angle (y_1, y_2), respectively. We then select a threshold of equal error rate. Given image pairs of different (y_1, y_2) and the threshold, we quantify their false positive rate and false negative rate and plot these values as heatmaps in Figure 6. We compare the performance of two approaches, namely the Naïve ResNet-18 as a baseline, and a ResNet-18 equipped with the proposed DREAM block. It can be observed from Figure 6(a) that false positives mainly concentrate at the bottom right of the heatmap, *i.e.*, when both query and matching faces are of extreme yaw angles. In comparison to the baseline, our approach yields much fewer false positives. From Figure 6(b), we observe that a majority of false negatives take place when one of the faces is frontal and the other one is profile. Again, our approach performs superior over the baseline in terms of false negative rates.

4.4. Examining the Architecture of DREAM Block

In this subsection we study the effectiveness of different architectures of the DREAM block. The architecture of DREAM block could differ in two aspects: 1) the location to apply the block, and 2) the design. For the first part, we compare different locations of inserting a DREAM block. 'After pooling layer' means we insert the DREAM block directly after the average pooling layer of ResNet. Recall that we introduced a new fully connected layer between the average pooling layer and the original fully connected layer of ResNet (see Sec. 3.2). 'After feature layer' refers to inserting the DREAM block after this feature layer. For the second part, we try to build the DREAM block with only one fully connected layer or replace one fully connected layer with a 1D convolution layer. We found that the best location to insert DREAM block is at the top of the network, where the feature is deep and feature dimension is compact. We believe the two aspects will benefit the learning of DREAM block. From Table 3 we observe that non-linearity plays a role in DREAM block (the setting that uses 2 fc layers outperforms linear mapping).

4.5. An Ablation Study on the Soft Gate

The soft gate produces a yaw coefficient to control the amount of residuals to be added to the initial representation. As described in the methodology section, the coefficient can be linear or nonlinear w.r.t. the degree of face yaw. In this experiment, we examine the effectiveness of different settings of the soft gate. In addition, we also report a baseline in which the soft gate is consistently closed, *e.g.*, $\mathcal{Y}(\mathbf{x}) = 1$. It is noteworthy that this baseline still keeps the same number of parameters but it degenerates to a conventional residual branch that losses the capability of distinguishing frontal and profile faces.

Table 4. Comparative analysis of different gate settings. Evaluation is conducted on MS-Celeb-1M with ResNet-18. Results are reported as EER error.

Gate Setting	With DREAM Block
Consistently Close, $\mathcal{Y}(\mathbf{x}) = 1$	9.31
Linear	8.82
Nonlinear	**8.45**

From Table 4, we observe that the performance of face verification significantly drops if the soft gate is closed consistently. The results suggest the improved performance attained by the DREAM block insertion is not merely due to the additional parameters, but the effective mechanism inherently brought by the residual branch and soft gate. From Table 4, we also observe that nonlinear mapping yields superior performance over the linear setting. This observation ascertains our design of exerting a higher degree of correction to a face pose larger than 45°, as this pose range is harder to be handled by the stem CNN (as observed from Figure 6).

5. Conclusion

We have presented a Deep Residual EquivAriant Mapping (DREAM) block to improve the performance of face recognition on profile faces. Our method is novel in that we take a radically different approach to handle profile faces. Specifically, we bridge the discrepancy between profile and frontal faces through performing equivariant mapping in the deep feature space. The mapping is achieved through the light-weight DREAM block that is easy to implement. Extensive results on CFP, IJB-A, and MS-Celeb-1M datasets demonstrate the applicability of the block on different types of stem CNNs, including ResNet-18, ResNet-50, and Center-Loss models. Interestingly, we observed that performing frontalization in the feature space is more fruitful than the image space for the task of face verification. It is noteworthy that the proposed block is not limited to face recognition with pose variation, it is suitable for other problems, *e.g.* cross-age face recognition, of which performance also suffers from uneven distribution of training data.

Acknowledgement: This work is supported by SenseTime Group Limited and the General Research Fund sponsored by the Research Grants Council of the Hong Kong SAR (CUHK 14241716, 14224316. 14209217).

References

[1] W. AbdAlmageed, Y. Wu, S. Rawls, S. Harel, T. Hassner, I. Masi, J. Choi, J. Lekust, J. Kim, P. Natarajan, et al. Face recognition using deep multi-pose representations. In *WACV*, 2016. 7

[2] X. Cao, D. Wipf, F. Wen, G. Duan, and J. Sun. A practical transfer learning algorithm for face verification. In *ICCV*, 2013. 5

[3] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun. Bayesian face revisited: A joint formulation. In *ECCV*, 2012. 3, 5, 6

[4] J.-C. Chen, V. M. Patel, and R. Chellappa. Unconstrained face verification using deep cnn features. In *WACV*, 2016. 2, 7

[5] J.-C. Chen, J. Zheng, V. M. Patel, and R. Chellappa. Fisher vector encoded deep convolutional features for unconstrained face verification. In *ICIP*, 2016. 6

[6] N. Crosswhite, J. Byrne, C. Stauffer, O. Parkhi, Q. Cao, and A. Zisserman. Template adaptation for face verification and identification. In *FG*, 2017. 7

[7] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. MS-Celeb-1M: challenge of recognizing one million celebrities in the real world. *Electronic Imaging*, (11):1–6, 2016. 1, 5

[8] T. Hassner, I. Masi, J. Kim, J. Choi, S. Harel, P. Natarajan, and G. Medioni. Pooling faces: template based face recognition with pooled face images. In *CVPRW*, 2016. 7

[9] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, 2015. 4

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[11] C. Huang, Y. Li, C. C. Loy, and X. Tang. Learning deep representation for imbalanced classification. In *CVPR*, 2016. 1

[12] G. B. 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, Technical Report 07-49, University of Massachusetts, Amherst, 2007. 1, 6

[13] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, and A. K. Jain. Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus Benchmark A. In *CVPR*, 2015. 6

[14] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *ICCVW*, 2011. 4

[15] K. Lenc and A. Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *CVPR*, 2015. 2, 3

[16] V. Lepetit, F. Moreno-Noguer, and P. Fua. EPnP: An accurate O(n) solution to the PnP problem. *IJCV*, 81(2):155–166, 2009. 4

[17] D. Lin and X. Tang. Inter-modality face recognition. In *ECCV*, 2006. 5, 6

[18] Y. Liu, J. Yan, and W. Ouyang. Quality aware network for set to set recognition. In *CVPR*, 2017. 7

[19] I. Masi, S. Rawls, G. Medioni, and P. Natarajan. Pose-aware face recognition in the wild. In *CVPR*, 2016. 2, 3, 7

[20] I. Masi, A. T. Trn, T. Hassner, J. T. Leksut, and G. Medioni. Do we really need to collect millions of faces for effective face recognition? In *ECCV*, 2016. 3, 7

[21] A. Nguyen, J. Yosinski, Y. Bengio, A. Dosovitskiy, and J. Clune. Plug & play generative networks: Conditional iterative generation of images in latent space. In *CVPR*, 2017. 5

[22] H. V. Nguyen and L. Bai. Cosine similarity metric learning for face verification. In *ACCV*, pages 709–720. Springer, 2010. 5

[23] S. Sankaranarayanan, A. Alavi, C. D. Castillo, and R. Chellappa. Triplet probabilistic embedding for face verification and clustering. In *BTAS*, 2016. 6

[24] S. Sankaranarayanan, A. Alavi, and R. Chellappa. Triplet probabilistic embedding for face verification and clustering. *arXiv preprint arXiv:1604.05417*, 2017. 7

[25] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 2, 4

[26] S. Sengupta, J.-C. Chen, C. Castillo, V. M. Patel, R. Chellappa, and D. W. Jacobs. Frontal to profile face verification in the wild. In *WACV*, 2016. 1, 2, 3, 5, 6

[27] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958, 2014. 4

[28] Y. Sugano, Y. Matsushita, and Y. Sato. Learning-by-synthesis for appearance-based 3d gaze estimation. In *CVPR*, 2014. 4

[29] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, 2014. 1, 2, 3, 4, 5

[30] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014. 1, 2, 3

[31] L. Tran, X. Yin, and X. Liu. Disentangled representation learning GAN for pose-invariant face recognition. In *CVPR*, 2017. 2, 3, 6

[32] D. Wang, C. Otto, and A. K. Jain. Face search at scale. *TPAMI*, 39(6):1122–1136, 2017. 7

[33] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016. 2, 4, 5

[34] Y. Wu, J. Li, Y. Kong, and Y. Fu. Deep convolutional neural network with independent softmax for large scale face recognition. In *ACM MM*, 2016. 1, 5

[35] X. Yin and X. Liu. Multi-task convolutional neural network for face recognition. *arXiv preprint arXiv:1702.04710*, 2017. 3, 7

[36] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *SPL*, 23(10):1499–1503, 2016. 5

[37] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling. Appearance-based gaze estimation in the wild. In *CVPR*, 2015. 4

[38] Y. Zhong, J. Chen, and B. Huang. Towards end-to-end face recognition through alignment learning. *arXiv preprint arXiv:1701.07174*, 2017. 3

[39] S. Zhu, C. Li, C. C. Loy, and X. Tang. Unconstrained face alignment via cascaded compositional learning. In *CVPR*, 2016. 4, 5