Pose-Aware Face Recognition in the Wild

Iacopo Masi[1] Stephen Rawls[2] Gérard Medioni[1] Prem Natarajan[2]
[1] USC Institute for Robotics and Intelligent Systems (IRIS), Los Angeles, CA
[2] USC Information Sciences Institute (ISI), Marina Del Rey, CA
{iacopo.masi,medioni}@usc.edu {srawls,pnataraj}@isi.edu

Abstract

We propose a method to push the frontiers of unconstrained face recognition in the wild, focusing on the problem of extreme pose variations. As opposed to current techniques which either expect a single model to learn pose invariance through massive amounts of training data, or which normalize images to a single frontal pose, our method explicitly tackles pose variation by using multiple pose-specific models and rendered face images. We leverage deep Convolutional Neural Networks (CNNs) to learn discriminative representations we call Pose-Aware Models (PAMs) using 500K images from the CASIA WebFace dataset. We present a comparative evaluation on the new IARPA Janus Benchmark A (IJB-A) and PIPA datasets. On these datasets PAMs achieve remarkably better performance than commercial products and surprisingly also outperform methods that are specifically fine-tuned on the target dataset.

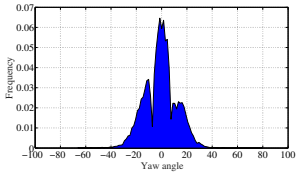
1. Introduction

There has been a flurry of advances in face recognition in recent years, with some techniques claiming to have met [29] or even surpassed [18, 25] human face verification performance. It is common to see saturated accuracy under certain conditions on the standard LFW benchmark.

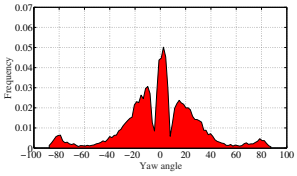
Recognizing that current face verification systems still have shortcomings under real-world conditions, a new benchmark (IJB-A) has been proposed in [13]. This is a publicly available benchmark to encourage researchers to focus on novel issues in face recognition in the wild. The IJB-A dataset focuses especially on variations in pose and represents a more challenging benchmark compared to LFW. Fig. 1 shows the distribution of yaw angles in LFW and in the newly released IJB-A dataset. As can be seen, the IJB-A dataset encompasses a wider variety of face poses than LFW and for this reason in this work we propose a method that is entirely designed to overcome variations in pose. In addition, the IJB-A dataset introduces a new testing protocol which more closely matches real-world use cases.

(a) LFW faces

(b) IJB-A faces



(c) Pose distribution in LFW



(d) Pose distribution in IJB-A

Figure 1: *Top*: images with increasing yaw angle from left to right in LFW (blue) and IJB-A (red). *Bottom*: Pose distributions (yaw angle) in the two datasets.

Instead of evaluating image pairs, the IJB-A protocol evaluates template pairs, where a template is a set of one or more images. This could represent the more realistic use-case where end-users have multiple images of a single subject. Since templates can contain images of a subject over multiple poses, it becomes important to consider how to handle template matching and how the pose variation takes part in the matching process. To the best of our knowledge, as also pointed out in [23], none have addressed pose variation in IJB-A.

The contribution of this paper is that we propose to take into account pose variability by training multiple pose-specific models, and exploiting those models when matching images of varying poses. While most previous approaches rely only on a single frontal-pose model [30, 5], possibly normalizing images via frontalization [29], we propose to handle pose variability by learning Pose-Aware Models (PAMs) for frontal, half-profile and full-profile

poses. We partition and augment the training dataset considering the training pose distribution and then co-train deep Convolutional Neural Networks (CNNs) to learn Pose-Aware Models. PAMs are then used to perform pose-aware face recognition in the wild. Moreover, differently from [33] that uses multi-task learning and from the multi-view perceptron (MVP) method [36], we use rendering technique to generate a synthetic view, instead of training the network to interpolate between views. PAMs outperform a single pose-agnostic model and yield state-of-the-art results on IJB-A. Moreover PAMs outperform DeepFace [29] on the new PIPA dataset despite having significantly less training data.

The paper is organized as follows: in Sect. 2 we review papers that address pose-invariance and face recognition in the wild. In Sect. 3 we present the overview about PAMs and in Sect. 4 we discuss how to learn Pose-Aware Models. In Sect. 5 we show how to use PAMs for face recognition. Sect. 6 presents experimental results, while we draw conclusions and discuss future research in Sect. 7.

2. Related work

Researchers have long acknowledged that matching techniques struggle to handle variations in pose. According to this, we review techniques to handle pose variation regarding state-of-the-art face recognition. Seminal papers in the past introduced the idea of training face classifiers at different poses, for example, the effort in [21] that extends the basic Eigenface approach to multiple poses. Recently, methods for pose-invariance have focused primarily in dealing with controlled datasets, such as Multi-PIE [9]. The authors in [17] are the first to introduce a 3D average face model rather than relying on 3D cylindrical or ellipsoid model. Prabhu *et al.* in [22] proposed an efficient way to estimate a 3D model from a single frontal image using a Generic Elastic Model (GEM). The method has been further improved considering diverse average values of depth per ethnic group [11]. It was noteworthy because they were one of the first to match frontal versus profile images using rendering. Moreover, 3D data have been used for matching rendered images with 2D face imagery in the wild, accounting for small pose variations [20] and, additionally, other researchers introduced the idea of rotating a face to get different training poses [7]. Recent papers focus on normalizing the pose from a profile face to a canonical frontal view. The paper in [2] is the first paper that reports the improvement in face recognition by rendering a profile face to frontal and others have also followed this approach, proposing different methods for the same underlying idea of *"frontalization"* [31, 29, 19, 10, 8]. In contrast to these methods, Sharma *et al.* [26] proposed the Discriminant Multiple Coupled Latent Subspace framework to address pose variations. Regarding methods that trained a CNN to recognize faces, one approach to obtain pose-invariance is to train a single CNN with a large enough dataset covering a diverse set of poses so that the CNN in principle could learn some degree of pose invariance automatically: FaceNet [25] shows that it is possible to learn a compact embedding for faces with an end-to-end learning system trained on 260 million images. DeepID [27] uses a large ensemble of networks trained on different patches of the face along with Joint Bayesian metric learning, showing remarkable performance. This work has been extended in [28] to show how the CNN is learning sparse features that implicitly encode attribute informations such as the gender.

Another approach is to apply pose-normalization to a frontal view as a preprocessing step [29, 30, 5]. For instance, DeepFace [29] learns a CNN on 4 million face images using frontalization technique to reduce pose variability. Wang *et al.* [30] showed that is possible to perform accurate face identification on a gallery of 80 million images and on the IJB-A benchmark. Moreover, Chen *et al.* in [5] showed that is possible to get compelling results on IJB-A by using a single CNN trained from scratch on a frontal view, fine-tuning it and learning the metric on the target dataset. Lately, Zhu *et al.* [37] showed that a CNN can be used, not only for classification, but to recover and normalize a near-frontal face to a frontal view. A similar frontalization idea is developed in [10] using a generic 3D model and a rendering framework with a soft-symmetry technique to compensate for self-occlusion, without the use of a neural network.

Researches also tried to let the network disentangle the identity and the view by either performing multi-task learning [33] or multi-view perceptron [36]. The drawback of these latter methods is that are only trained on constrained images on the Multi-PIE dataset in which the pose is manually specified, without reporting performance on benchmarks in the wild such as IJB-A.

3. Pose-Aware Face Models for Recognition

Our method assumes that in general the face pose distribution $p(\mathbf{p}|\mathbf{I})$, given one image $\mathbf{I}$, is not dominated by near-frontal faces and thus we propose to learn multiple pose-specific CNN models as opposed to a single CNN. Assuming detected landmarks on an image, we observe that it is easy to compensate for roll when the face is near-frontal, and for pitch, when the face is near profile by just using in-plane alignment. Thus we focus our models to compensate mainly for yaw variations, by assuming $p(\mathbf{p}|\mathbf{I}) \approx p(\psi|\mathbf{I})$, where ψ represents the face yaw angle. Another observation is that compensating for out-of-plane variations using frontalization [29, 10] could be a noisy process that gets harder as input images move closer to profile. For this reason, we propose a method that extends the concept of frontalization to multiple modes of the pose distribution. In

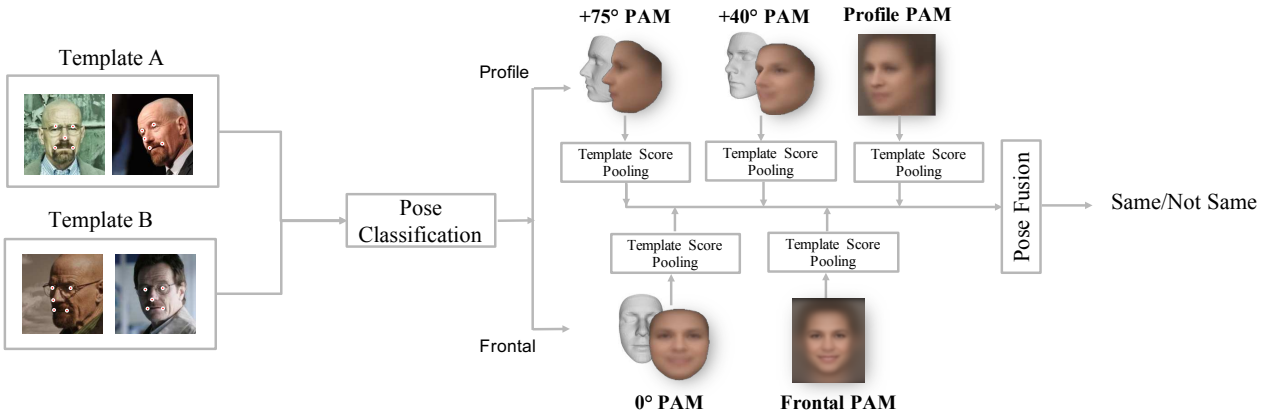


Figure 2: Given a template pair to verify, pose classification is used to forward each image to the corresponding Pose-Aware CNN Model. Given multiple images, each model extracts features, matches them and pools scores at a template level independently. Finally, the contribution of each model is pooled into a single, final score. Note how in our approach we use rendering technique to adjust the pose to a frontal (0°), half-profile (40°) and full-profile view (75°).

addition, our method considers different ways of aligning a face since the latter is very important in face recognition. Our motivation is that in presence of images in the wild, complex methods of alignment such as those compensating out-of-plane rotation could produce weak results when dense landmarks are difficult to localize; in this case, a safer way of alignment (in-plane alignment) could work better, requiring only few anchor points. We have observed that poor quality landmarks can seriously effect the quality of rendered images, and thus cause poor recognition performance. Currently, besides the use of in-plane alignment, we do not do anything to mitigate this, but we plan to investigate methods of incorporating landmark confidence into our approach in the future.

For each model a particular alignment process is applied specifically to the pose we consider. In particular we apply the concept of multi-alignment using the following:

2D in-plane alignment: Image are aligned in plane with a 2D non-reflective similarity that compensates scale, in-plane rotation and translation. In these images important properties of face geometry are preserved along with discriminative part of the head such as hair or ears. The drawback is that these images contain high variability in the pose. We use one model to align images to a frontal reference, while the other to a profile one.

3D out-of-plane alignment: Out-of-plane rotation is explicitly compensated by rendering images at a specific yaw value, in order to adjust the pose and remove pose variations. Unlike [29] that uses an adaptive 3D face shape for face modeling, we use an unmodified 3D generic face model, following the idea in [10]. Nevertheless, differently from both [29, 10] that use only frontalization (0°), we also render images to half-profile (40°) and full-profile (75°) views in order to cope for extreme yaw variations.

(a) Single Model (b) Aware of face pose

Figure 3: (a) Average image obtained over the training set considering all the face imagery to learn a single model. Frontal pose absorbs the other poses. (b) Average images on the training set when pose is considered.

Considering these two types of alignment, we trained an ensemble of five CNN models, each of which is "aware" of the face viewpoint by learning specific features for each view. Our models are called Pose-Aware CNN Models (PAMs) and are learned using the CASIA WebFace dataset [32], which is currently the largest publicly available dataset, containing roughly 500K face images. Fig. 2 shows an overview of our approach along with different kinds of alignment used in our method. In the next section we show how PAMs can be learned automatically taking into account the pose distribution of the training data.

4. Learning Pose-Aware CNN Models (PAMs)

Differently from approaches that use just a single, frontal face reference to train a CNN [32, 29, 5], our idea is to learn Pose-Aware CNN Models (PAMs). Firstly, one issue is that we do not have access to millions of data covering all the possible poses: the CASIA WebFace training pose distribution is still biased towards frontal, nevertheless there are still some profile images that could be exploited, as can be seen

from Fig. 3. Secondly, it is necessary to consider that CNN generalization power is usually proportional to the training data size, thus we need to trade-off between data partitioning and clustering when processing the training dataset. In contrast to approaches relying on multi-task learning [33] or [36], that models the identity and face view with the same network, we treat each type of alignment and data independently, that is, we learn a specific model for each type of alignment and mode of the pose distribution. The main motivation for this is that having multiple networks permits co-training them in order to improve transferability of learned features. We have found this particularly important for generalization over other datasets.

4.1. Discovering the training pose distribution

We use the approach in [3] to detect landmarks using a fixed bounding box in CASIA WebFace since most of the faces are centered on the image. From the detected landmarks on CASIA, we estimate the pose of the face in the image putting in correspondence 2D detected landmarks $\mathbf{l} \in \mathbb{R}^{2\times|\mathbf{J}|}$ with 3D labeled landmarks $\mathbf{L} \doteq \mathbf{M}(\mathbf{J}) \in \mathbb{R}^{3\times|\mathbf{J}|}$ on a 3D generic model $\mathbf{M}$, where $\mathbf{J}$ is a set of indices that selects the corresponding landmarks on the 3D model. We can then estimate a perspective camera model mapping the generic 3D model $\mathbf{M}$ on the image such as:

$$\mathbf{l} = \mathbf{p}\,\mathbf{L} \tag{1}$$

where

$$\mathbf{p} = \mathbf{K}\,\left[\mathbf{R}\,\mathbf{t}\right]. \tag{2}$$

We use the PnP method to estimate external camera parameters, assuming the principal point in the image center and then refine the focal length by minimizing landmark reprojection error.

From $\mathbf{p}$, we extract the rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$ that contains the 3D rotation parameters of the model with respect to the image. By decomposing $\mathbf{R}$ we obtain the yaw values ψ of the face across all the dataset w.r.t to the 3D generic model. We accumulate all the $\{\psi_i\}_{i=1}^{N}$ values in order to estimate the training pose distribution $p(\psi|\mathbf{I})$.

Instead of treating all the images as belonging to the same frontal model, irrespective of the yaw distribution, we express the yaw variability. In order to get the main modes of the training yaw distribution $p(\psi|\mathbf{I})$ we run K-means on $\{\psi_i\}_{i=1}^{N}$ to find the main T modes:

$$\Psi = \{\mu_{\psi_t}\}_{t=1}^{T}, \tag{3}$$

along with the hard-assignment of each image to a certain mode. We can interpret the latter as a function that maps each image to a specific mode as:

$$\delta(\mathbf{I}) = t \quad \text{where} \quad t \in [1 \dots T] \tag{4}$$

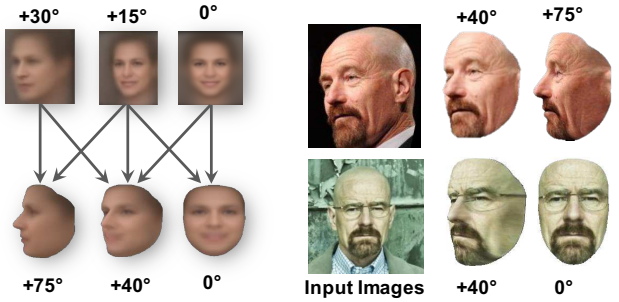
and T represents the five models we want to learn. Each μ_{ψ_t} represents a mode in the yaw distribution and $\delta(\cdot)$ gives the assignments to each t-th mode, for each image. We found the modes pretty balanced each other and to be centered roughly in $\{0°, \pm15°, \pm30°\}$; moreover most of the images are concentrated into the two frontal modes, confirming that this dataset is still biased towards frontal faces.

4.2. PAMs for in-plane alignment

Starting from Eq. (3), we exploit face symmetry property to simplify our representation. By flipping one direction of yaw distribution as $\mu_{\psi_t} \rightarrow |\mu_{\psi_t}|$, we flip the corresponding images along the vertical axes and modify the assignments in Eq. (4) accordingly. In this way we can consider only one side of the distribution $p(\psi|\mathbf{I})$, for example left side, reducing the number of models we need to train. We finally have a new set of modes $\Psi' = \{\mu_{\text{frontal}}, \mu_{\text{near-frontal}}, \mu_{\text{profile}}\}$, corresponding to yaw values centered in $\{0°, +15°, +30°\}$. This allows us to express $p(\psi|\mathbf{I})$ as a bi-modal distribution by partitioning the dataset in two classes: near frontal faces with small variability and profile faces with high variability in pose. In particular we partition the images using the image assignments in Eq. (4), classifying one image as profile if belongs to the third mode μ_{profile} and frontal otherwise. In this way, we are able to partition the CASIA dataset in two different new datasets, that are used to learn two CNN models with in plane alignment namely $\text{PAM}_{\text{in-f}}$ and $\text{PAM}_{\text{in-p}}$. Since we divided frontal images from profile ones, we are able to perform different types of 2D in-plane alignment for each set: the frontal set uses nine most reliable landmarks, while the profile set is aligned using the tip of the nose and the center of the two eyes. For both the alignments we use a non-reflective similarity transformation $\mathbf{S}(s, \theta, t_x, t_y)$. The parameters for scale, in-plane rotation an translation are recovered by solving a linear system of equations using detected and reference landmarks, specific for each alignment.

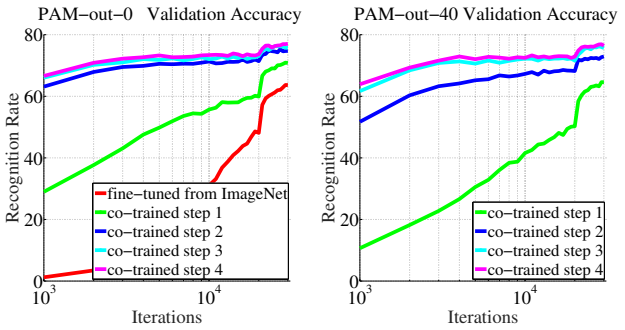
4.3. PAMs for out-of-plane alignment

There are still unresolved issues about the process presented in Sect. 4.2. The first one is that the images associated to each mode discovered in Eq. (3) have still intra-pose variability within the same mode that could be captured by the network and it is well known that pose variability drastically affects face recognition performance. Another issue is that is very hard to find publicly available datasets containing a large amount of full profile faces in order to learn a discriminative CNN model for a full-profile view. For all these reasons, in this section we learn also other models that compensate for out-of-plane rotation in order to minimize pose variability and tackle the lack of training data for profile faces. As in Sect. 4.2, in this case $p(\psi|\mathbf{I})$ is expressed as multi-modal distribution with three prominent modes. We can exploit again Eq. (3) and face symmetry as done in



(a) (b)

Figure 4: (a) The directed graph used to map each mode of CASIA yaw distribution to the desired mode (b) The process helps to render properly one image: if the face is frontal we can render to a frontal and half-profile view. If an image is far from being frontal we avoid frontalizing it.



(a) Co-training $\text{PAM}_{\text{out-0}}$ (b) Co-training $\text{PAM}_{\text{out-40}}$

Figure 5: Steep increase in validation accuracy in function of iterations in the fine-tuning process. Each curve represents a step of co-training. Iterations are shown in log scale.

Sect. 4.2 but here we can apply the concept of pose adjustment to a certain yaw value, as done before for frontalization for near-frontal faces. We would like our models to represent frontal, half-profile and full profile faces but our data are mostly centered on near-frontal faces. We propose to unbias the source, yaw distribution and its modes by transforming them to a new target distribution: we use the 3D generic model $\mathbf{M}$ and the estimated pose $\mathbf{p}$ in Eq. (2) to render a specific face to a new mode of the desired distribution using dense facial landmarks. This new, target distribution is decided a priori to have frontal ($0°$), half-profile ($40°$) and a full profile views ($75°$) in order to be able to learn a discriminative classifier for full-profile faces as well. These values are set in order to trade-off between having good rendered faces and generating side views. In particular we employ the directed graph in Fig. 4 to decide how to render a target face; each edge in the graph represents the rendering process from a certain mode to a target one. The face rendering technique is derived from [10] with soft-symmetry. Besides reducing pose variability for each mode, a benefit that we get also by rendering images is that, for each new target pose, we can increase the number of samples to train our models. The increasing factor is function of the number of images assigned to each source mode and the number of edges entering the target node. Considering this process, we are able learn three additional networks, one for each mode of the new desired pose distribution, namely $\text{PAM}_{\text{out-0}}$, $\text{PAM}_{\text{out-40}}$, $\text{PAM}_{\text{out-75}}$.

4.4. Fine-tuning PAMs

For each pose-specific dataset created in previous Sections, we train a Pose-Aware CNN. Since training from scratch a CNN could require millions of annotated images, we learn our Pose-Aware CNNs by fine-tuning state-of-the-art CNN models trained on ImageNet. In our approach, we experiment with a CNN with 8 layers (AlexNet) [15] and one with 19 layers (VGGNet) [4]. We experiment with different network types since we can show that our method is agnostic to the CNN model used and by fusing across pose, we can get improvement, irrespective of the architecture used. All these CNN models end with fully connected layers fc7 and fc8. The output of fc8 is fed to a C-way SoftMax which gives a distribution over the subject labels $\mathcal{C}$. Denoting $x_i(\mathbf{I})$ the i-th output of the network on a given image $\mathbf{I}$, the probability assigned to the i-th class is the output of the SoftMax function $p_i(\mathbf{I}) = \frac{e^{x_i(\mathbf{I})}}{\sum_{l=1}^{C} e^{x_l(\mathbf{I})}}$. The fine-tuning is performed through stochastic gradient descent (SGD) and standard back-propagation [16], minimizing the cross-entropy loss using the SoftMax function and the one-hot vector of ground-truth class $\widehat{c}$ over the entire training set of images. We start from pre-learned weights on ImageNet, initialize from scratch fc8 layer with parameters drawn from a Gaussian distribution with zero mean and standard deviation 0.01. The initial learning rate is set to $\alpha = 0.001$. We fine-tune all the layers with this learning rate but the new fc8 layer which has a learning rate of an order of magnitude greater than α. We learn the the biases two times faster than the weights. Moreover we decrease α of an order of magnitude when a plateau is reached in the validation set.

4.5. Co-training PAMs to improve transferability

The analysis provided in [34] motivates our approach to fine-tune our models as opposed to training them from scratch, since million images are not available in our case. To improve transferability, we propose to co-train our models in order to get a better optimization point in the loss minimization. Differently from [32, 29], that use a single, trained-at-once model, we have different CNN models to address a specific view point of the face and a specific alignment. By exploiting multiple models we opti-

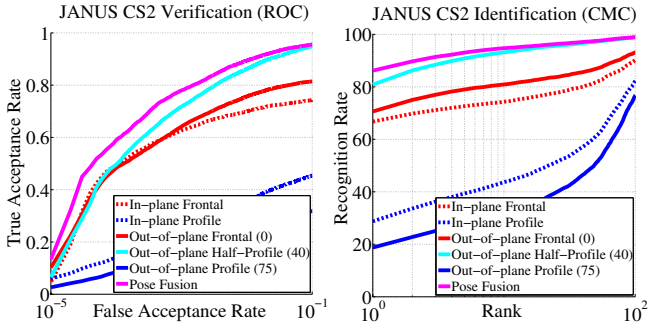


Figure 6: Fusing across pose improves the ROC in the verification protocol and the CMC in the identification protocol on JANUS CS2.

mize the same objective function but from different viewpoints and data. Considering in-plane aligned models, we only fine-tune the frontal network from ImageNet prelearned weights ($\text{PAM}_{\text{in-f}}$); we found experimentally that the $\text{PAM}_{\text{in-p}}$ fine-tuned from ImageNet was performing poorly since we have few profile images, thus we co-train it resuming the optimization from $\text{PAM}_{\text{in-f}}$ weights. Regarding out-of-plane alignment, we start to fine-tune $\text{PAM}_{\text{out-0}}$ from ImageNet. Then we fine-tune $\text{PAM}_{\text{out-40}}$, starting from $\text{PAM}_{\text{out-0}}$ weights. We keep iterating by alternating the co-training until the validation accuracy saturates in both models. Fig. 5 shows the steep increase in the validation accuracy provided by the co-training using AlexNet. After these models are trained, we finally proceed to co-train $\text{PAM}_{\text{out-75}}$ from $\text{PAM}_{\text{out-40}}$. We performed the same process for VGGNet, but this deeper model requires less steps of co-training to get the validation accuracy saturated. To perform recognition, we use the fc7 layer response $\mathbf{x} \in \mathbb{R}^D$.

5. Pose-Aware Face Recognition

The Pose-Aware CNN models learned in Sect. 4 provide a way to perform pose-aware face recognition. We can interpret each CNN as a discriminative classifier explicitly trained at a certain mode of the pose distribution. In the following we describe the general recognition procedure which is applied for each of the ensemble PAMs defined by $\{\ \text{PAM}_{\text{in-f}}, \text{PAM}_{\text{in-p}}, \text{PAM}_{\text{out-0}}, \text{PAM}_{\text{out-40}}, \text{PAM}_{\text{out-75}}\ \}$

Main Matching Process: In the matching process, we exploit the face symmetry in the same way we did in the training set to align or render a face to one side and then flip it back, if needed, to be aligned to the corresponding PAM. The score between a feature pair $(\mathbf{x}_1, \mathbf{x}_2)$ is given by the correlation as:

$$s(\mathbf{x}_1, \mathbf{x}_2) = \frac{(\mathbf{x}_1 - \bar{\mathbf{x}}_1)(\mathbf{x}_2 - \bar{\mathbf{x}}_2)^T}{||\mathbf{x}_1 - \bar{\mathbf{x}}_1||\ ||\mathbf{x}_2 - \bar{\mathbf{x}}_2||} \quad \text{where} \quad \bar{\mathbf{x}} = \frac{1}{D}\sum_d \mathbf{x}.$$

In case of multiple images per template, each single PAM performs template score pooling using a weighted average of the pair-wise scores, where each weight is function of the score using an exponential function as $\exp\big(\gamma\ s(\mathbf{x}_1, \mathbf{x}_2)\big)$. Finally, we average again all the the responses $\forall\, \gamma \in [0..20]$.

Matching with in-plane alignment: Given a testing image $\mathbf{I}$, we firstly detect landmarks and classify the pose using the μ_{profile} corresponding to the mode found for the profile faces in Sect. 4.2. In particular as frontal if $|\psi| \leq \mu_{\text{profile}}$ otherwise as profile. In a general case, if a template contains multiple images (set-to-set matching), we proceed to align and forward each face image either to $\text{PAM}_{\text{in-f}}$ or $\text{PAM}_{\text{in-p}}$, accordingly to the classified pose.

Matching with out-of-plane alignment: We extend the concept of frontalization by rendering faces to the modes $\{0°, 40°, 75°\}$ we have defined in Sect. 4.3. We render each image always to the half-profile view (40°) and then if the image is classified as near-frontal we frontalize to 0° otherwise we render the image to the profile view (75°). In this case we use the $\text{PAM}_{\text{out-0}}$, $\text{PAM}_{\text{out-40}}$, $\text{PAM}_{\text{out-75}}$ accordingly.

Pose Fusion: Assuming a template pair (or an image pair) to be recognized, then we have at most five scores, each of which is produced by a specific PAM. In our current approach we simply pool the available scores together by average. We found this to provide a good baseline in all our experiments. Fig. 6 shows the performance in terms of ROC and CMC for each single PAM as described in Sect. 5 and the pose fusion (magenta curve) on JANUS CS2 using VGGNet. In our experiments we got improvement as well, by fusing across poses using AlexNet. Note that since we classify the pose, some models could not trigger on some images, that is why in general the performance for profile is low in Fig. 6. We have also tried simply to forward all the images to all the PAMs but we found the proposed pose fusion the more stable result in all our experiments.

6. Experimental Results

In this section we present the experimental results on three datasets in the wild. We are interested in testing our method on imagery containing extreme pose variations, so we report the performance on the new IARPA JANUS Benchmark-A (IJB-A) [13] and People In Photo Albums (PIPA) [35] dataset. We apply PCA and Power Transformation [24] (PT) to the features on the training split of each dataset used in this paper. In all the experiments, no supervised method is applied except for the learning of PAMs on CASIA WebFace. That is, differently from [5], we do not re-train our models using the training splits of each dataset.

Datasets	JANUS CS2				IJB-A			
Networks	AlexNet		VGGNet		AlexNet		VGGNet	
Metrics	TAR	Rank-10	TAR	Rank-10	TAR	Rank-10	TAR	Rank-10
PAMs w/o co-training	0.649	0.869	0.810	0.932	0.494	0.881	0.660	0.923
PAMs	0.792	0.927	0.850	0.941	0.612	0.903	0.701	0.936
PAMs + PCA	0.835	0.928	0.881	0.941	0.666	0.912	0.768	0.938
PAMs + PCA + PT	**0.862**	**0.934**	**0.895**	**0.949**	**0.756**	**0.928**	**0.826**	**0.946**

Table 1: Improvement for each component on CS2 and IJB-A dataset. TAR is reported at FAR=0.01 for verification. Recognition Rate at Rank-10 is reported for identification.

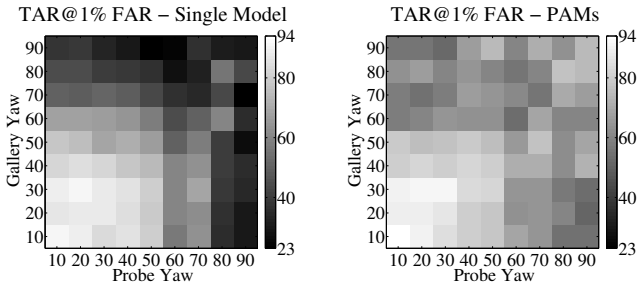


Figure 8: Image-to-Image TAR across poses: PAMs (right) show better pose invariance than the single model (left).

6.1. IARPA Janus Benchmark A (IJB-A)

IJB-A is a new publicly available challenge proposed by IARPA and spread by NIST[1] to push frontiers of face recognition in the wild since lately LFW [12] performance saturated. Both IJB-A and JANUS CS2 share the same 500 subjects under extreme conditions regarding pose, expression and illuminations. IJB-A considers the more difficult pairs compare to the JANUS CS2 splits. The IJB-A evaluation protocol mainly consists of face verification (1:1) and face identification (1:N). The interesting thing about this dataset is that each subject is described by a template containing a set of images or frames extracted from videos. In order to have a fair comparison with the other methods, we removed the overlapping subjects of CASIA WebFace with JANUS CS2 and IJB-A while learning PAMs.

Component analysis: In Tab. 1 we report the improvement for each component of PAMs for two types of CNN models that we used (AlexNet and VGGNet). We show the performance on both JANUS CS2 and IJB-A splits reporting the TAR at FAR=0.01 for the verification protocol and the Recognition Rate at Rank-10 for the identification protocol. The table shows that significant improvement in performance is given by the co-training. The improvement given by the co-training is much bigger with AlexNet model respect to the VGGNet but still, even in this latter case, helps performance. Further improvement is obtained by applying PCA and Power Transformation.

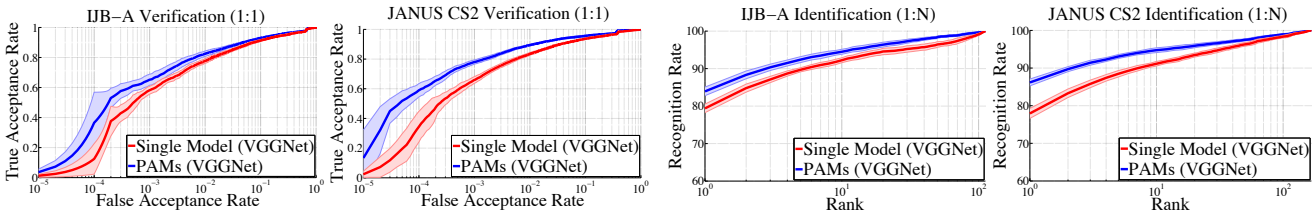
Comparison to a single, frontal model: In Fig. 7 we show the improvement of PAMs respect to learning a single CNN trained starting from CASIA WebFace with standard in-plane alignment (corresponding to $\text{PAM}_{\text{in-f}}$). In this experiment we use VGGNet for both the methods. We show that pose-aware face recognition greatly improves over a single network trained, as previously done before [32], on imagery which pose distribution is dominated by near-frontal faces. Moreover, in order to better factor the effect of pose on performance, we designed an experiment using only image-to-image comparisons of IJB-A data, classifying the pose of each image using our method. In Fig. 8 we report how image-to-image TAR varies by pose. In particular, Fig. 8 shows that while a large part of our gain comes from improved matching of same-pose images, we also make a useful improvement to comparisons of different-pose images. Thus our PAM approach does improve pose-invariance over a single, frontal model.

Comparison with state-of-the-art: In Tab. 2 we report a comparison with the state-of-the-art. Pose-aware face recognition reports better performance compared to the COTS and GOTS (Commercial and Government Off-the-Shelf) systems and Fisher Vector encoding method using frontalization [6]. If we compare with methods that use deep-learned features, PAMs show better performance than the method in [30] which exploits seven networks and fuses the result with a commercial system. It is worth to mention that PAMs improve in IJB-A verification over [30] of about 9% TAR at FAR=0.01 and 14% TAR at FAR=0.001, showing also better recognition rate at rank-1. Surprisingly, PAMs show even a better ROC with respect to methods that explicitly fine-tuned the network on the IJB-A training set and performed metric-learning on these sets. In particular we improve over [5] of about 2% TAR at FAR=0.01 and 6% TAR at FAR=0.001 for JANUS CS2 splits and 4% TAR at FAR=0.01 for IJB-A. Overall, we sensibly improve over [5] on all the metrics except for the recognition rate in IJB-A identification, in which our method is less effective to rank the gallery. Finally, in Tab. 2 we show also a comparison with the popular method of frontalization, which corresponds to using the frontalized $\text{PAM}_{\text{out-0}}$ on all the images.

6.2. People In Photo Albums (PIPA)

The authors in [35] recently introduced People In Photo Albums (PIPA) dataset, which is composed of public photo albums uploaded to Flickr. One of the characteristics of the dataset is that it contains extreme pose variations. They use the data to perform pose invariant person recognition using multiples cues (face, body, poselets) but they also measure face recognition performance using a subset of the data. We follow the same protocol as in [35], which uses

[1]IJB-A is available under request at http://www.nist.gov/itl/iad/ig/ijba_request.cfm



(a) ROC (b) CMC

Figure 7: Improvement in the ROC (a) and in the CMC (b) comparing a single CNN and the proposed PAMs on the IJB-A challenge and JANUS CS2. Horizontal axes are shown in log scale. Each curve shows also the standard deviation.

Methods ↓	IJB-A Verification (TAR)		IJB-A Identification (Rec. Rate)			JANUS CS2 Verification (TAR)		JANUS CS2 Identification (Rec. Rate)		
Metrics →	@FAR=0.01	@FAR=0.001	@Rank-1	@Rank-5	@Rank-10	@FAR=0.01	@FAR=0.001	@Rank-1	@Rank-5	@Rank-10
COTS	–	–	–	–	–	0.581±0.054	0.37	0.551±0.03	0.694±0.017	0.741±0.017
GOTS	0.406±0.014	0.198±0.008	0.443±0.021	0.595±0.02	–	0.467±0.066	0.25	0.413±0.022	0.571±0.017	0.624±0.018
OpenBR [14]	0.236±0.009	0.104±0.014	0.246±0.011	0.375±0.008	–	–	–	–	–	–
Fisher Vector [6]	–	–	–	–	–	0.411±0.081	0.25	0.381±0.018	0.559±0.021	0.637±0.025
Wang *et al.* [30]	0.733±0.034	0.514±0.060	0.820±0.024	0.929±0.013	–	–	–	–	–	–
Chen *et al.* [5]	0.787±0.043	–	**0.86±0.023**	**0.943±0.017**	**0.962±0.012**	0.876±0.013	0.72	0.838±0.012	0.924±0.009	**0.949±0.006**
PAM_{out-0} (frontal.)	0.733±0.018	0.552±0.032	0.771±0.016	0.887±0.009	0.919±0.009	0.810±0.013	0.638±0.020	0.750±0.012	0.871 ± 0.006	0.905± 0.006
PAMs	**0.826±0.018**	**0.652±0.037**	0.840±0.012	0.925±0.008	0.946±0.007	**0.895±0.006**	**0.780±0.014**	**0.862±0.009**	**0.931±0.005**	**0.949±0.006**

Table 2: Comparative performance analysis on IJB-A benchmark and CS2 for verification (ROC) and identification (CMC). Symbol "–" indicates that the metric is not available for that protocol. Standard deviation is not available for all the methods.

two-fold cross-validation on the face recognition subset of PIPA[2]. Differently from [35], we do not train a classifier for each subject but we classify each subject over the 581 identities as described in Sect. 5. We first run an experiment using only a single-pose frontal model, which matches the reported DeepFace performance of 47.97%. Next we run PAMs which achieved 57.65%. Furthermore, while DeepFace trained SVM classifiers on the PIPA training set, our method made no supervised use of PIPA data. It is interesting to show that PAMs improve also over AlexNet trained on ImageNet but evaluated on body bounding box in the PIPA dataset (56.07%). We are aware that by using multiple cues (face, body, poselets) is possible to achieve higher recognition: we are interested to show that with PAMs there is still useful information to exploit in profile faces.

7. Conclusion and Discussion

In this paper we proposed a pose-aware method to perform face recognition with imagery containing extreme pose variation. Our approach shows how we can rely not only on a single, frontal model but also on half-profile and full profile models to perform face recognition in the wild. Our approach is agnostic to the underlying CNN used. The main direction for the future is in mitigating the landmark detection errors and to assess pose-invariance in a more controlled dataset such as the Multi-PIE dataset. In particular in order to make the approach more robust to landmark detector failures, we plan to either extend the method by using a confidence value for detected landmarks or having a secondary mechanism to assess the quality of rendered images. Additional future work consists in learning a better pose fusion and developing a single, multi-pose CNN in a unified framework. More details using different landmark detectors and various deep feature responses can be found in [1].

[2]Data and splits are available at http://www.cs.berkeley.edu/~nzhang/piper.html

Acknowledgments

This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA 2014-14071600010. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright annotation thereon. Moreover, we gratefully acknowledge USC HPC for hyper-computing and the support of NVIDIA Corporation with the donation of a NVIDIA Titan X. We would like to thank all the other authors of [1] for their effort on the project. Moreover, thank A. D. Bagdanov, L. Ballan and the three anonymous reviewers for their fruitful comments that improved the manuscript.

References

[1] W. AbdAlmageed, Y. Wu, S. Rawls, S. Harel, T. Hassner, I. Masi, J. Choi, J. Leksut, J. Kim, P. Natarajan, R. Nevatia, and G. Medioni. Face recognition using deep multi-pose representations. In *WACV*, 2016.

[2] A. Asthana, T. Marks, M. Jones, K. Tieu, and M. Rohith. Fully automatic pose-invariant face recognition via 3D pose normalization. In *ICCV*, 2011.

[3] T. Baltrusaitis, P. Robinson, and L.-P. Morency. Constrained local neural fields for robust facial landmark detection in the wild. In *ICCV Workshops*, 2013.

[4] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 2014.

[5] J. Chen, V. M. Patel, and R. Chellappa. Unconstrained face verification using deep CNN features. *arXiv preprint*, arXiv:1508.01722v1, 2015.

[6] J.-C. Chen, S. Sankaranarayanan, V. M. Patel, and R. Chellappa. Unconstrained face verification using fisher vectors computed from frontalized faces. In *BTAS*, 2015.

[7] Y.-C. Chen, V. M. Patel, P. J. Phillips, and R. Chellappa. Dictionary-based face recognition from video. In *ECCV*, 2012.

[8] C. Ferrari, G. Lisanti, S. Berretti, and A. Del Bimbo. Dictionary learning based 3D morphable model construction for face recognition with varying expression and pose. In *3DV*, 2015.

[9] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. *Image Vision Computing*, 2010.

[10] T. Hassner, S. Harel, E. Paz, and R. Enbar. Effective face frontalization in unconstrained images. In *CVPR*, 2015.

[11] J. Heo and M. Savvides. Gender and ethnicity specific generic elastic models from a single 2D image for novel 2D pose face synthesis and recognition. *TPAMI*, 34(12):2341–2350, 2011.

[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 07-49, University of Massachusetts, Amherst, October 2007.

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

[14] J. Klontz, B. Klare, S. Klum, E. Taborsky, M. Burge, and A. K. Jain. Open source biometric recognition. In *BTAS*, 2013.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[16] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, 1998.

[17] A. Li, S. Shan, X. Chen, and W. Gao. Maximizing intra-individual correlations for face recognition across pose differences. In *CVPR*, 2009.

[18] C. Lu and X. Tang. Surpassing human-level face verification performance on LFW with GaussianFace. In *AAAI*, 2015.

[19] I. Masi, C. Ferrari, A. Del Bimbo, and G. Medioni. Pose independent face recognition by localizing local binary patterns via deformation components. In *ICPR*, 2014.

[20] I. Masi, G. Lisanti, A. Bagdanov, P. Pala, and A. Del Bimbo. Using 3D models to recognize 2D faces in the wild. In *CVPR Workshops*, 2013.

[21] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *CVPR*, 1994.

[22] U. Prabhu, J. Heo, and M. Savvides. Unconstrained pose-invariant face recognition using 3D generic elastic models. *TPAMI*, 33(10):1952–1961, 2011.

[23] A. RoyChowdhury, T. Lin, S. Maji, and E. G. Learned-Miller. Face identification with bilinear CNNs. In *WACV*, 2016.

[24] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *IJCV*, 105(3):222–245, 2013.

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

[26] A. Sharma, M. A. Haj, J. Choi, L. S. Davis, and D. W. Jacobs. Robust pose invariant face recognition using coupled latent space discriminant analysis. *CVIU*, 116(11):1095–1110, 2012.

[27] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *CVPR*, 2014.

[28] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In *CVPR*, 2015.

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

[30] D. Wang, C. Otto, and A. K. Jain. Face search at scale: 80 million gallery. *arXiv preprint*, arXiv:1507.07242, 2015.

[31] D. Yi, Z. Lei, and S. Li. Towards pose robust face recognition. In *CVPR*, pages 3539–3545, 2013.

[32] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *arXiv preprint*, arXiv:1411.7923, 2014.

[33] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, and J. Kim. Rotating your face using multi-task deep neural network. In *CVPR*, 2015.

[34] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *NIPS*. 2014.

[35] N. Zhang, M. Paluri, Y. Taigman, R. Fergus, and L. Bourdev. Beyond frontal faces: Improving person recognition using multiple cues. In *CVPR*, 2015.

[36] Z. Zhu, P. Luo, X. Wang, and X. Tang. Multi-view perceptron: a deep model for learning face identity and view representations. In *NIPS*, 2014.

[37] Z. Zhu, P. Luo, X. Wang, and X. Tang. Recover canonical-view faces in the wild with deep neural networks. *arXiv preprint*, arXiv:1404.3543, 2014.