# DEEP PAIR-WISE SIMILARITY LEARNING FOR FACE RECOGNITION

*Anonymous submission*

Address line 1
Address line 2
Address line 3

## ABSTRACT

Recent advances in deep learning made it possible to build deep hierarchical models capable of delivering state-of-the-art performance in various vision tasks, such as object recognition, detection or tracking. For recognition tasks the most common approach when using deep models is to learn object representations (or features) directly from raw image-input and then feed the learned features to a suitable classifier. Deep models used in this pipeline are typically heavily parameterized and require enormous amounts of training data to deliver competitive recognition performance. Despite the use of data augmentation techniques, many application domains, predefined experimental protocols or specifics of the recognition problem limit the amount of available training data and make training an effective deep hierarchical model a difficult task. In this paper, we present a novel, *deep pair-wise similarity learning* (DPSL) strategy for deep models, developed specifically to overcome the problem of insufficient training data, and demonstrate its usage on the task of face recognition. Unlike existing (deep) learning strategies, DPSL operates on image-pairs and tries to learn pair-wise image similarities that can be used for recognition purposes directly instead of feature representations that need to be fed to appropriate classification techniques, as with traditional deep learning pipelines. Since our DPSL strategy assumes an image pair as the input to the learning procedure, the amount of training data available to train deep models is quadratic in the number of available training images, which is of paramount importance for models with a large number of parameters. We demonstrate the efficacy of the proposed learning strategy by developing a deep model for pose-invariant face recognition, called Pose-Invariant Similarity Index (PISI), and presenting comparative experimental results on the FERET an IJB-A datasets.

***Index Terms***— Deep learning, similarity learning, face recognition

## 1. INTRODUCTION

Deep models represent powerful hierarchical models that have shown immense potential for various computer vision tasks and have pushed the state-of-the-art in many application domains [1], [2], [3]. A typical deep model represents a (complex) highly parameterized neural network trained for solving a specific machine-learning problem, such as object detection, tracking or recognition. When used for recognition tasks, the common approach of deploying deep models is to learn a suitable feature representation from the available training data and then feed the learned feature representation to a classifier of choice. Here, the classifier itself can be another deep model or some other appropriate "shallow" classifier.

Contemporary deep models typically feature a large amount of open parameters (in the order of billions) that need to be learned during training and, therefore, require an enormous amount training data to deliver competitive performance. For many problems it is possible to generate the required amount of training data by scaling, translating, rotating, mirroring or augmentation the training images in other ways, while for other problems data augmentation techniques are not a viable option. The amount of training data available to train deep models is also limited by particularities of different application domains, (e.g., in medical imaging training data is inherently scarce) or experimental protocols, which often forbid using data outside from the predefined training set for training deep models.

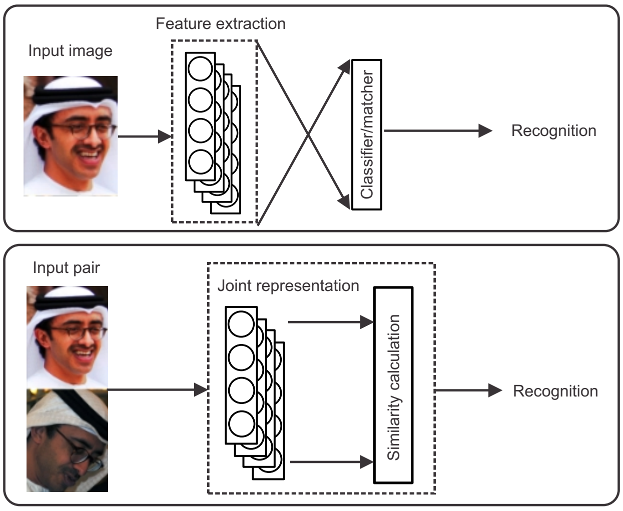


**Fig. 1**. Illustration of the conceptual difference between the typical deep learning strategies used for recognition tasks, where features are first extracted from the input image by the deep model and then fed to a classifier (upper row), and our DPSL strategy, where a pair of images is fed to the model and a similarity index (or score) is then returned by the model (lower row).

In this paper, we introduce a novel, *deep pair-wise similarity learning* (DPSL) strategy for deep models designed specifically to mitigate the problem of insufficient training data. With the DPSL strategy, the given deep model is not trained on individual input images, but on image-pairs, which are combined into synthetic images that serve as inputs for the training procedure. Different from the existing (deep) learning strategies used for recognition tasks, our DPSL strategy does not learn object features from the data, but tries to learn image similarities that can be exploited for recognition directly. An illustration of the conceptual difference between the feature-learning pipeline commonly used for recognition tasks with deep models and our DPSL strategy is shown in Fig. 1 for the problem of face recognition.

To demonstrate the feasibility of the DPSL strategy, we develop a novel deep model for face recognition under large pose variations, named Pose-Invariant Similarity Index (or PISI for short) and assess it's performance on two publicly available datasets (FERET and IJB-A) with encouraging results. During the training stage, the PISI model learns to distinguish between image pairs corresponding to the same subjects and image pairs corresponding to different subjects by producing a large similarity index for the former and a small similarity index for the latter pair of images. In a sense, our PISI model learns a similarity function that can be used with pairs of facial images for recognition purposes.

As the PISI model is based on the DPSL strategy it is considerably different from other deep models developed for unconstrained face recognition (where large pose-variations may commonly be encountered). Deep models, such as the one presented in [4] or [3] typically build a deep hierarchical model comprised of convolutional, max-pooling and fully-connected neural-network layers and exploit the learned feature representations for recognition. Our PISI model, on the other hand, operates on pairs of images and instead of features learns similarities between pairs of facial images that correspond to the same subjects and were captured under different head poses. The first, convolutional layers of our model can still be interpreted as feature extractors, which are trained to produce joint feature representations of the (input) image pairs, while the next, the fully-connected layers can be considered as classifiers/matchers over the extracted (image-pair) features that produce the final similarity index. Another important aspect of our PISI model is the fact that the amount of training samples available to train our model is quadratic in the number of available training images. This is a highly convenient characteristic as the complexity of the model and its large parameter space (the model has approximately $15.6 \times 10^6$ open parameters) would otherwise require large amounts of additional training data or the usage of various data augmentation techniques.

Note that face-recognition under large pose variations is a very active research field. A detailed survey of all existing techniques is beyond the scope of this paper. The reader is referred to [5] for a well written (and condensed) overview of the field.

To summarize, we present the following key contributions in this paper:

- we introduce a deep pair-wise similarity learning (DPSL) strategy that learns image similarities over image-pairs instead of features from single input images and can be used to increase the amount of available training data without data augmentation techniques (Section 3),
- we present a deep network architecture (the PISI model) for face recognition under large pose variations developed based on the DPSL strategy (Section 3), and
- we evaluate the PISI model on the FERET and IJB-A datasets and present encouraging experimental results using our learning strategy and modeling approach (Section 5).

## 2. THEORETICAL BACKGROUND

Traditionally, face recognition systems relied on hand-crafted features and suitable classifiers to achieve competitive recognition performance. As suggested in [2], most of the research effort was spend on the development and/or choice of appropriate image features, which were often more important than the classifier itself w.r.t. the final performance of the recognition system. In contrast, contemporary state-of-the-art face recognition systems exploit advances in deep learning and heavily rely on convolutional neural networks [2] that can operate directly on image data and, hence, alleviate the need for hand-crafted features. In this section we briefly survey the most important building blocks of deep models and introduce some terminology used in the remainder of the paper. For a more comprehensive coverage of the topic, the reader is referred to [6].

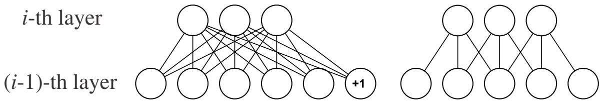


**Fig. 2**. A fully-connected layer with a bias term (left) and a convolutional layer with a filter of size 3 and stride 1 (right).

**Deep models and model layers.** Deep models used for image-based recognition/analysis tasks represent hierarchical models, composed of different types of interconnected layers that jointly process the given input in a feed-forward manner. Among the different layers, fully-connected, convolutional and max-pooling layers are most commonly used in the field of computer vision.

**Fully-connected layers.** Fig. 2 shows the difference between fully-connected and convolutional layers in a deep model. In a fully-connected model, each neuron in the $i$-th layer has a weighted connection to every neuron in the previous layer as well as an additional bias term. The activation of the $i$-th layer of a fully connected neural network is defined as:

$$\mathbf{y}_i = \sigma(\mathbf{W}_i\mathbf{x}_i + \mathbf{b}_i), \tag{1}$$

where $\sigma(\cdot)$ represents the activation function of the $i$-th layer, $\mathbf{W}_i$ represents the connection matrix between the $(i-1)$-th and $i$-th layers, $\mathbf{x}_i$ is the vector of inputs into the $i$-th layer, and $\mathbf{b}_i$ represents the bias-vector terms of the units in the $i$-th layer. Fully connected neural networks require $n^2 + n$ parameters for a connection between two layers of size $n$ (assuming both layers have the same number of neurons).

**Convolutional layers.** In contrast to fully connected layers, each neuron in a convolutional layer (illustrated in Fig. 2 - right) is only connected to it's neighboring neurons in the previous layer with the neighborhood size determined by the shape (or size) of the convolutional kernels used. As stated in [2], convolutional deep models with interspersed convolutional and max-pooling layers do not only enable machine learning for computer-vision tasks without requiring pre-processing of data, but also have the advantage (compared to fully-connected models) of tolerating variances in object scale and translation.

A convolutional layer operating on two-dimensional (image) input data has $m \times n \times k \times l$ parameters, where $m$ is the number of convolutional filters in the $i$-th layer, $n$ is the number of filters in the $(i-1)$-th layer, and $(k, l)$ are numbers that define the shape (or size) of the local filters in the $i$-th layer. Using valid-border convolution without zero-padding, the filters of size $(k, l)$ reduce an input image of size $a \times b$ pixels to $m$ maps of size $(a-k+1, b-l+1)$. The two-dimensional convolution operation of an input image $f(x, y)$ with a filter $w(x, y)$ that is performed in a convolutional layer is then defined as:

$$o(x, y) = \sum_{u=-\infty}^{\infty} \sum_{v=-\infty}^{\infty} f(u, v)w(x-u, y-v), \tag{2}$$

where $w(x, y)$ represents one of the filter kernels in the convolutional layer, whose parameters (i.e., weights) are learned during training of the CNN, and $o(x, y)$ denotes part of the output of the convolutional layer that corresponds to the given kernel $w(x, y)$.

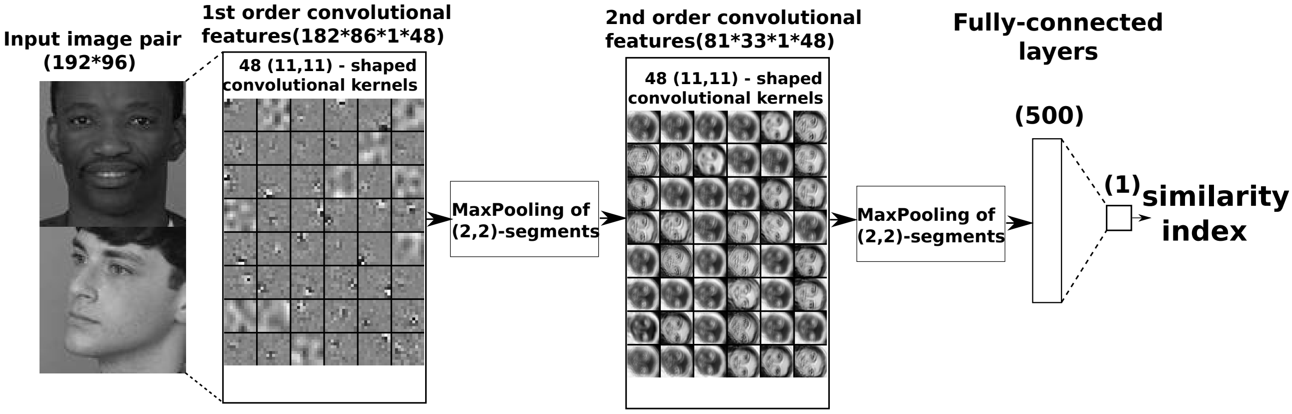


**Fig. 3**. The network architecture of the PISI (pose-invariant similarity index) model. The model exploits our DPSL strategy (deep pair-wise similarity learning) and takes two grayscale facial images with different poses as input and outputs a similarity index. The numbers in the brackets above the layers stand for the dimensionality of the layer outputs. Note that the outputs of the convolutional layers shown in the figure are only of an illustrative nature.

**Max-pooling layers.** Max-pooling layers are commonly used between convolutional layers to reduce the dimensionality of the data that is propagated through the deep model. Max-pooling layers operate on the outputs of the previous convolutional layer, sample the filtered outputs with over small non-overlapping image patches ( e.g., of size 2 × 2 or 3 × 3 pixels) and output the element with the highest value from each of the sampled patches. In the context of deep convolutional models, max-pooling layers can be considered automated feature extractors - for each patch of the output of a convolutional filter bank, they propagate forward only the most highly activated feature of that patch. This reduces the dimensionality of the propagated data (e.g., by a factor of 4 for max-pooling over 2 × 2-pixel patches, or by a factor of 9 for max-pooling over 3 × 3 pixel patches) and also preserves the most important features, which significantly reduces the number of parameters required in the following model layers.

## 3. DEEP PAIR-WISE SIMILARITY LEARNING AND THE PISI MODEL

In this section we introduce our *deep pair-wise similarity learning* (DPSL) strategy and *pose-invariant similarity index* (PISI) model for face recognition.

**DPSL and PISI.** The main idea of the deep pair-wise similarity learning strategy is to combine image pairs from the given training set into synthetic images and train a deep model the synthetic images. Instead of learning feature representations from the individual input images, we try to learn image similarities directly and output a single similarity index that can be used for recognition. Given a training set of $M$ images, we are able to produce $N = M(M-1)$ distinct image pairs (i.e., synthetic images) that belong to one of two classes (i.e., match and non-match) and can be used to train a deep model capable of outputing similarity scores for the given input image pair. If we denote two arbitrary images of size $a \times b$ pixels from the training set as $f(x, y) \in \mathbb{R}^{a\times b}$ and $h(x, y) \in \mathbb{R}^{a\times b}$ then an image pair $P \in \mathbb{R}^{a\times 2b}$ that forms the input to the training procedure is formed by a simple concatenation of the two images, i.e., $P = [f(x, y), h(x, y)]$.

Based on the outlined learning strategy we designed a deep model for pose invariant face recognition, called the *pose-invariant similarity index* (PISI) model. The PISI model represents a deep neural architecture as depicted in Fig. 3. The model takes two grey-scale facial images under different poses as input, and outputs their similarity, where a value close to one indicates that the input image pair represents the same subjects, and a value close to zero indicates that the image pair represents different subjects. The model uses two convolutional layers separated by a max-pooling layer for image-pair representation and feeds the outputs of these layers to two fully-connected layers, which combine them into the final similarity index. Non-saturating linear rectifiers (ReLU) are used as activation functions, which speeds up learning and improves gradient-descent learning on deep neural network architectures, as reported by Krizhevsky et al. [1].

**PISI architecture.** The deep network architecture used in the PISI model comprises two structural parts. The first part serves as the feature extractor for the pair of input images and is composed of two convolutional layers, each followed by a max-pooling layer. The convolutional layers try to capture the joint characteristics of the input-image pair, while the max-pooling layers introduce a degree of translation invariance and reduce the size of the model's parameter space.

The second structural part of the PISI model comprises two fully connected layers. The outputs of the first part of the model (which can be seen as joint features of the given image-pair) serve as the input for the fully connected layers. The main goal of these layers is to produce a similarity index based on the feature representation produced by the first structural part of the model.

Tables 1 and 2 summarize the model layers and parameters, respectively. Note that all layers use pre-activation bias terms. Rectified linear units (ReLU) are used as activation functions in most layers, except for the max-pooling and third fully-connected layer, where simple linear activation functions are used.

**Training the PISI model.** Training the PISI model involves finding a suitable set of values for all parameters tabulated in Table 2, such that the model produces values as close as possible to the value of one for image pairs corresponding to the same subjects and values as close as possible to zero for image pairs corresponding to different subjects.

Consider a training set of image pairs $P_1, P_2, ...P_N$, where a given pair of images $P_k = [f(x, y), h(x, y)]$ corresponds either to images of the same subject or images of two different subjects,

**Table 1**. A summary of the model layers

| Layer | Activation | Dimensions |
|---:|:---:|---:|
| Input | linear | $192 \times 96$ |
| Conv2d, stride 1 | ReLU | $182 \times 86 \times 1 \times 48$ |
| $(2, 2)$ MaxPooling | linear | $91 \times 43 \times 1 \times 48$ |
| Conv2d, stride 1 | ReLU | $81 \times 33 \times 1 \times 48$ |
| $(2, 2)$ MaxPooling | linear | $40 \times 16 \times 1 \times 48$ |
| Fully-connected | ReLU | 500 |
| Fully-connected | linear | 1 |

**Table 2**. A summary of the model parameters

| Parameter $p$ | Dimensions |
|---:|---:|
| $w_1$ | $48 \times 1 \times 11 \times 11$ |
| $b_1$ | 48 |
| $w_2$ | $48 \times 48 \times 11 \times 11$ |
| $b_2$ | 48 |
| $w_3$ | $(40 \times 16 \times 48) \times 500$ |
| $b_3$ | 500 |
| $w_4$ | $500 \times 1$ |
| $b_4$ | 1 |

$N$ stands for the number of all image pairs in the training set and $k = 1, 2, ..., N$. A target similarity index $y_k \in \{0, 1\}$ of 1 is then assigned to all image pairs corresponding to the same subject and a target similarity index of $(y_k = 0)$ is assigned to all image pairs corresponding to different subjects, where $k = 1, 2, ..., N$. The loss function used for the training procedure is the mean absolute deviation $L(\tilde{y}_k) = 1/N \sum_{k=1}^{N} |\tilde{y}_k - y_k|$ between the target similarity indices and the similarity indices produced by the model with the current values of the parameters, where $\tilde{y}_k \in \mathbb{R}$ and $0 \leq \tilde{y}_k \leq 1$ The complete training rule is defined by Eqs. (3), (4), and (5), i.e.:

$$\epsilon_{i+1} \leftarrow \epsilon_0 \times (1 + 0.0005(i + 1))^{-1} \tag{3}$$

$$v_{i+1} \leftarrow 0.9 \times v_i - \epsilon_i \times \left\langle \left. \frac{\partial L}{\partial p} \right|_{p_i} \right\rangle_{D_i} \tag{4}$$

$$p_{i+1} \leftarrow p_i + v_{i+1} \tag{5}$$

As can be seen from the equations, mini-batch stochastic gradient descent with the momentum of 0.9, the decay of 0.0005 and the initial learning rate of 0.01 is used. In the above equations $\epsilon_{i+1}$ denotes the learning rate, $v_i$ represents the velocity variable, $p$ represents a parameter of the PISI model (see Table 2), $i$ represents the mini-batch index and $\left\langle \partial L / \partial p|_{p_i} \right\rangle_{D_i}$ stands for the average over the $i$-th mini-batch $D_i$ of the derivative of the objective with respect to $p$, evaluated at $p_i$ [1]. Mini-batches of 128 samples were used for training.

For the convolutional layers, the parameters were initialized as proposed by He et al. [7] - through the random normal distribution with the mean value of 0 and the variance of $\sigma^2 = \frac{2}{n_l}$, where $n_l$ represents the number of model parameters for the $l$-th layer. Bias terms of the convolutional layers were initialized with the constant value of 0. Fully-connected layers were initialized from the random normal distribution with the mean value of 0 and the standard deviation of 0.01, and their bias terms were initialized with the constant value of 1, as proposed by Krizhevsky et al. [1]. Furthermore, the dropout rate of 0.5 was used on the fully-connected layers.

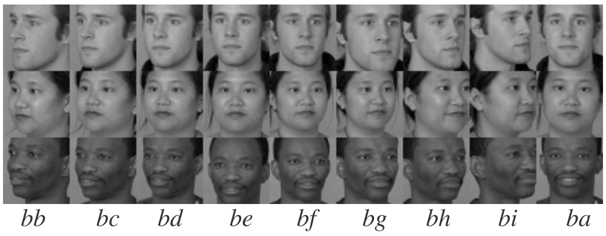


**Fig. 4**. Sample images from the FERET image subsets used for experimentation. Each row represents images of one subject, while each column represents one of the image subsets.

**Relation to previous work.** Our DPSL strategy and PISI model are related to deep metric learning techniques, such as [8, 9, 10, 11, 12, 13].

Similar to these techniques we also try to learn a measure of similarity between input image pairs, but we do not require the learned measure to be metric. Furthermore, deep metric learning techniques typically rely on Siamese networks at the lower model layers for feature extraction and uses additional layers on top of the Siamese architecture to learn a metric useful for recognition over the image-pair features. The feature representation used with these techniques is still learned from individual images and requires a large amount of training data. The trained deep model (i.e., the feature extractor) is then simply duplicated for the second image, which produces two independent processing pipelines (with shared parameters) that are applied to each image in the input image pair. Our learning strategy and corresponding model, on the other hand, produce a single processing pipeline and can, therefore, benefit from the increase of training data due to the joint processing of all image-pairs.

Similar as with deep metric learning technique presented in [13], extensions of our learning strategy and model to image-triplets are also possible, but will be presented elsewhere.

## 4. DATASETS AND EXPERIMENTAL PROTOCOLS

To evaluate the performance of the PISI model for pose-invariant face recognition, we selected two datasets for our experiments, i.e., the FERET [14] and IJB-A [15] datasets.

**FERET.** We used image subsets *ba* through *bi* for the experiments, which contain images of 200 subjects taken under various poses. Subsets *be*, *bf*, *bd*, *bg*, *bc*, *bh*, *bb* and *bi* feature head poses with a yaw angles of $15°$, $-15°$, $25°$, $-25°$, $40°$, $-40°$, $60°$ and $-60°$, respectively. Some sample images from these subsets are shown in Figure 4.

All images were preprocessed prior to the experiments by localizing the faces in the images with the approach proposed by Zhu et al. in [16], cropping the facial area from the image and then scaling each cropped (grey-scale) image to a fixed size of 96 × 96 pixels.

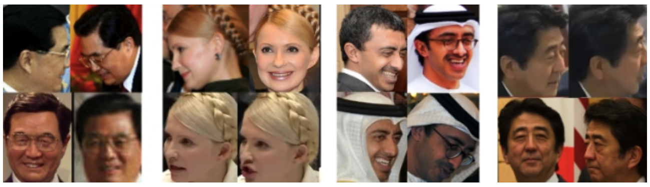
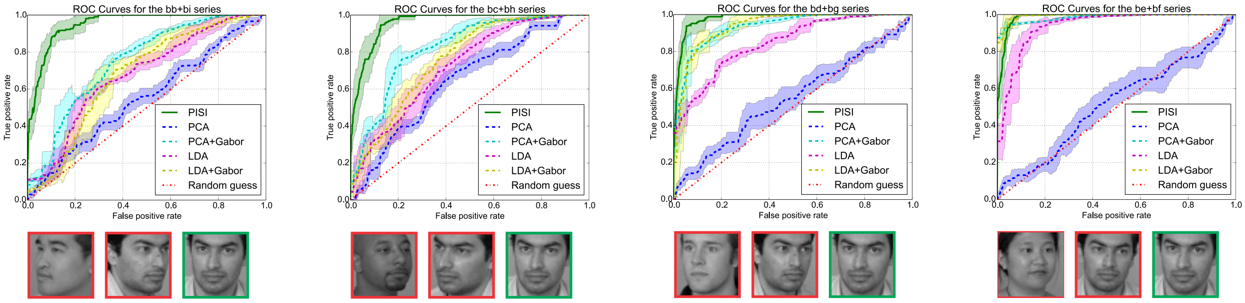


**Fig. 5**. Sample images from the IJB-A dataset. Unlike in this example image, each subject in the dataset is actually represented by a different number of facial images in the dataset.



(a) Subsets *bb* and *bi*

(b) Subsets *bc* and *bh*

(c) Subsets *bd* and *bg*

(d) Subsets *be* and *bf*

**Fig. 6**. ROC curve comparison for the face recognition experiments with different head poses: ±60° yaw angles (a), ±40° yaw angles (b), ±25° yaw angles (c) and ±15° poses (d). The images below the graphs illustrate the visual appearance of the probe and target images. Images with a green bounding box represent examples of the target images, while the images with the red bounding box represent examples of the probe images. The figure is best viewed in color.

The 200 subjects from the FERET subsets (*ba* through *bi*) were divided into disjoint training and testing sets, with 185 subjects (1665 images) in the training set and the remaining 15 subjects (135 images) in the test set. From the images assigned to the training set, 2960 frontal-against-side image pairs were generated, half of which represented positive matches, and the other half represented negative matches. A similar procedure was also adopted to generate image pairs for the experimental assessment of the PISI model. Here, each frontal image from the test set was paired with every other (non-frontal) image from the test set. Thus, a total of 1800 image pairs was generated for the assessment, of which 120 (6.7%) represented positive matches and 1680 (93.3%) represented negative matches.

**IJB-A.** The subset of the IJB-A dataset [15] used for training and evaluation of the PISI model consisted of 5712 images of 500 different subjects, for an average of 11.4 images per subject. Figure 5 shows a few sample images from the IJB-A dataset. The dataset consists of images of famous people gathered from the web. The subjects in the dataset were balanced over different geographic population groups, gender as well as other factors. Compared to FERET, the images in the IJB-A dataset exhibit much larger variability w.r.t. pose, illumination, facial expressions, obstructions and image quality, and is only labelled by subject identity - no pose labels are provided for the dataset. Instead of generating frontal-against-non-frontal image pairs, the image pairs needed to train our PISI model were generated by randomly selecting images from the entire dataset and labelling them according to whether they represent a positive match or not.

To preproces the training and test sets, ground truth information provided with the dataset was used to crop the faces from the images. The cropped facial images were then scaled to a fixed size of $96 \times 96$ pixels and converted to grey-scale. The subjects of the images were again divided into disjoint training and test sets, such that 333 subjects were used for training and 166 subjects were used for testing. 4676 image pairs were generated from the training set to learn the open parameters of the PISI model, and 3320 image pairs were generated from the test set to assess PISI's performance. In both training and test sets, exactly half of the image pairs represented positive matches and the other half represented negative matches.

## 5. RESULTS AND DISCUSSION

The training procedure described in the previous section was used to learn the parameters of the model on each of the described datasets. The training was conducted on the GPU (Nvidia GeForce 670GTX) of a desktop PC with an Intel i7 3770K processor running at 3GHz and 8 GB of DDR3 RAM. The training time was approximately one day for the FERET dataset and one week for the IJB-A dataset.

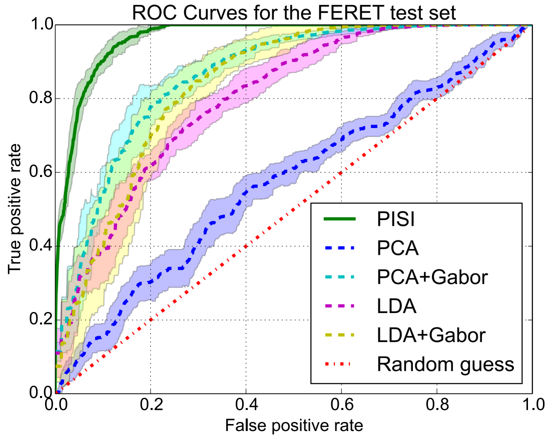


**Fig. 7**. ROC curve comparison for the full feret test set

The performance of the resulting model was evaluated in face verification experiments using ROC curves. Bootstrapping was used to generate variance estimates for the generated curves. The PISI model and DPSL strategy are generally suited best for verification (i.e., two class) problems, but can also be applied for identification (recognition) tasks if the problem is formulated through a series of pair-wise comparisons [17]. We compare the performance of the PISI model to PCA- and LDA-based baselines [18], [19], a combination of Gabor features and PCA as well as the Gabor-Fisher Classifier [20]. All competitor techniques are trained and tested on the same data and with the same experimental protocol as the PISI model. The implementations for the competitor techniques were taken from the Matlab "PhD face recognition toolbox" [21], [22].

**Experiments on the FERET dataset.** In our first series of experiments we assess the PISI model on the FERET dataset. The focus of this series of experiments is solely on pose variability, since all other factors influencing the appearance of the facial images are controlled for this dataset. Fig. 6 shows the experimental results for different yaw angles featured in the different FERET subsets. The PISI model provides significant performance improvements over the chosen baseline techniques for most head posses (i.e., different yaw

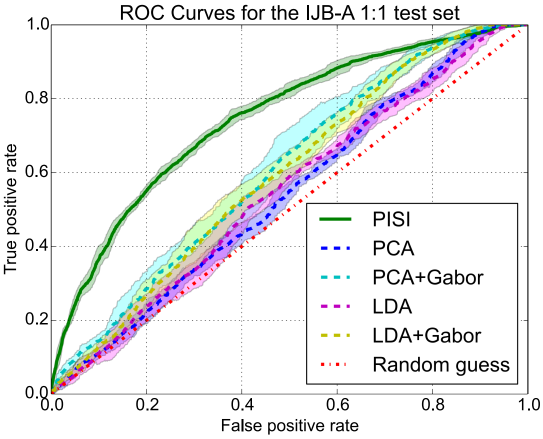


**Fig. 8**. ROC curve comparison for the IJB-A test set

angles) and most operating points on the ROC curves. The biggest difference in performance can be seen at the larger yaw angles (i.e., at yaw angles of $\pm 60^\circ$ and $\pm 40^\circ$), while the performance differences are less significant for the smaller deviations from frontal pose.

Fig. 7 shows the comparison of the PISI model and the baseline techniques for the full FERET test set (all subsets combined). We again observe that the proposed model ensures the best performance of all tested techniques.

**Experiments on the IJB-A dataset.** In our second series of experiments we use the IJB-A dataset, which contains images with other sources of appearance variability next to pose and is considered one of the most challenging datasets for unconstrained face recognition. The results of our experiments on the IJB-A dataset are presented in Fig. 8. Due to the unconstrained recognition problem, all assessed techniques perform visibly worse than on the FERET dataset. However, the performance of all four baseline techniques is only slightly better than chance, while our PISI model ensures significantly better verification results.

## 6. CONCLUSIONS

We have presented a novel deep pair-wise similarity learning strategy for deep models and demonstrated its feasibility by designing and training a deep model for unconstrained face recognition. We have shown that the developed PISI model ensures competitive performance on two challenging datasets. The main shortcoming of the developed model is its complexity, which was limited by the hardware available to us at the time of writing. As part of our future work we plan to extend the PISI model to a more complex architecture that is expected to deliver even better performance. The research presented in this paper has also implications for other fields, where the lack of training data hinders the deployment of deep models. In these fields our DPSL strategy could provide a viable solution for learning deep models.

## 7. REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G.E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[2] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “Deepface: Closing the gap to human-level performance in face verification,” in *CVPR 2014*, 2014, pp. 1701–1708.

[4] Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep learning identity-preserving face space,” in *ICCV 2013*, 2013, pp. 113–120.

[5] Y. Dong, Z. Lei, and S.Z. Li, “Towards pose robust face recognition,” in *CVPR 2013*, 2013, pp. 3539–3545.

[6] Yoshua Bengio, Ian Goodfellow, and Aaron Courville, *Deep Learning*, MIT Press, 2014.

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” *arXiv preprint arXiv:1502.01852*, 2015.

[8] S. Chopra, R. Hadsell, and Y. LeCun, “Learning a similarity metric discriminatively, with application to face verification,” in *CVPR 2005*, 2005, vol. 1, pp. 539–546.

[9] D. Yi, Z. Lei, S. Liao, and S.Z. Li, “Deep metric learning for person re-identification,” in *ICPR 2014*, 2014, pp. 34–39.

[10] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, “Learning fine-grained image similarity with deep ranking,” in *CVPR 2014*, 2014, pp. 1386–1393.

[11] J. Hu, J. Lu, and Y.-P. Tan, “Discriminative deep metric learning for face verification in the wild,” in *CVPR 2014*, 2014, pp. 1875–1882.

[12] J. Lu, G. Wang, W. Deng, P. Moulin, and J. Zhou, “Multi-manifold deep metric learning for image set classification,” in *CVPR 2015*, 2015, pp. 1137–1145.

[13] E. Hoffer and N. Ailon, “Deep metric learning using triplet network,” *ICLR 2015*, 2015.

[14] P.J. Phillips, H. Wechsler, J. Huang, and P.J. Rauss, “The feret database and evaluation procedure for face-recognition algorithms,” *Image and vision computing*, vol. 16, no. 5, pp. 295–306, 1998.

[15] B.F. Klare, 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,” *algorithms*, vol. 13, pp. 4, 2015.

[16] X. Zhu and D. Ramanan, “Face detection, pose estimation, and landmark localization in the wild,” in *CVPR 2012*, 2012, pp. 2879–2886.

[17] A.K. Jain, A.R. Ross, and S. Prabhakar, “An introduction to biometric recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1.

[18] M. Turk and A. Pentland, “Eigenfaces for recognition,” *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[19] W. Zhao, A. Krishnaswamy, R. Chellappa, D.L. Swets, and J. Weng, *Discriminant Analysis of Principal Components for Face Recognition*, pp. 73–85, Springer Verlag Berlin, 1998.

[20] C. Liu, “Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, no. 28.

[21] V. Struc and N. Pavesic, “The complete gabor-fisher classifier for robust face recognition,” *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 31, 2010.

[22] V. Struc, R. Gajsek, and N. Pavesic, “Principal gabor filters for face recognition,” in *BTAS 2009*, 2009, pp. 1–6.