Deep Learning Face Representation from Predicting 10,000 Classes

Yi Sun[1] Xiaogang Wang[2] Xiaoou Tang[1,3]

[1]Department of Information Engineering, The Chinese University of Hong Kong

[2]Department of Electronic Engineering, The Chinese University of Hong Kong

[3]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences

sy011@ie.cuhk.edu.hk xgwang@ee.cuhk.edu.hk xtang@ie.cuhk.edu.hk

Abstract

This paper proposes to learn a set of high-level feature representations through deep learning, referred to as Deep hidden IDentity features (DeepID), for face verification. We argue that DeepID can be effectively learned through challenging multi-class face identification tasks, whilst they can be generalized to other tasks (such as verification) and new identities unseen in the training set. Moreover, the generalization capability of DeepID increases as more face classes are to be predicted at training. DeepID features are taken from the last hidden layer neuron activations of deep convolutional networks (ConvNets). When learned as classifiers to recognize about 10,000 face identities in the training set and configured to keep reducing the neuron numbers along the feature extraction hierarchy, these deep ConvNets gradually form compact identity-related features in the top layers with only a small number of hidden neurons. The proposed features are extracted from various face regions to form complementary and over-complete representations. Any state-of-the-art classifiers can be learned based on these high-level representations for face verification. 97.45% verification accuracy on LFW is achieved with only weakly aligned faces.

1. Introduction

Face verification in unconstrained conditions has been studied extensively in recent years [21, 15, 7, 34, 17, 26, 18, 8, 2, 9, 3, 29, 6] due to its practical applications and the publishing of LFW [19], an extensively reported dataset for face verification algorithms. The current best-performing face verification algorithms typically represent faces with over-complete low-level features, followed by shallow models [9, 29, 6]. Recently, deep models such as ConvNets [24] have been proved effective for extracting high-level visual features [11, 20, 14] and are used for face verification [18, 5, 31, 32, 36]. Huang et al. [18] learned a generative deep model without supervision. Cai et al. [5] learned deep nonlinear metrics. In [31], the deep models are supervised by the binary face verification target. Differently, in this paper we propose to learn high-level face identity features with deep models through face identification, i.e. classifying a training image into one of n identities (n 10,000 in this work). This high-dimensional prediction task is much more challenging than

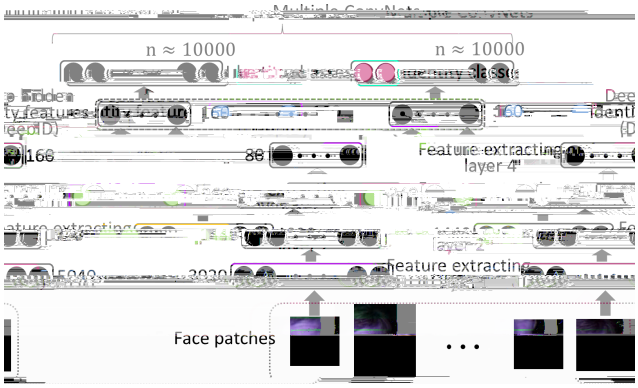


Figure 1. An illustration of the feature extraction process. Arrows indicate forward propagation directions. The number of neurons in each layer of the multiple deep ConvNets are labeled beside each layer. The DeepID features are taken from the last hidden layer of each ConvNet, and predict a large number of identity classes. Feature numbers continue to reduce along the feature extraction cascade till the DeepID layer.

Deep hidden IDentity features or DeepID). Each ConvNet takes a face patch as input and extracts local low-level features in the bottom layers. Feature numbers continue to reduce along the feature extraction cascade while gradually more global and high-level features are formed in the top layers. Highly compact 160-dimensional DeepID is acquired at the end of the cascade that contain rich identity information and directly predict a much larger number (*e.g.*, 10*;* 000) of identity classes. Classifying all the identities simultaneously instead of training binary classifiers as in [21, 2, 3] is based on two considerations. First, it is much more difficult to predict a training sample into one of many classes than to perform binary classification. This challenging task can make full use of the super learning capacity of neural networks to extract effective features for face recognition. Second, it implicitly adds a strong regularization to ConvNets, which helps to form shared hidden representations that can classify all the identities well. Therefore, the learned high-level features have good

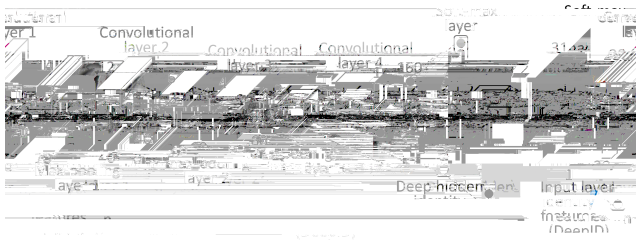


Figure 2. ConvNet structure. The length, width, and height of each cuboid denotes the map number and the dimension of each map for all input, convolutional, and max-pooling layers. The inside small cuboids and squares denote the 3D convolution kernel sizes and the 2D pooling region sizes of convolutional and max-pooling layers, respectively. Neuron numbers of the last two fully-connected layers are marked beside each layer.

31 k for rectangle patches, and 31 31 k for square patches, where $k = 3$ for color patches and $k = 1$ for gray patches. Figure 2 shows the detailed structure of the ConvNet which takes 39 31 1 input and predicts n (*e.g.*, $n = 10;000$) identity classes. When the input sizes change, the height and width of maps in the following layers will change accordingly. The dimension of the DeepID layer is fixed to 160, while the dimension of the output layer varies according to the number of classes it predicts. Feature numbers continue to reduce along the feature extraction hierarchy until the last hidden layer (the DeepID layer), where highly compact and predictive features are formed, which predict a much larger number of identity classes with only a fewn/s0 -11.9557(of)-dimension of the output layer varies

3.2. Feature extraction

We detect five facial landmarks, including the two eye centers, the nose tip, and the two mouth corners, with the facial point detection method proposed by Sun *et al*. [30]. Faces are globally aligned by similarity transformation according to the two eye centers and the mid-point of the two mouth corners. Features are extracted from 60 face patches with ten regions, three scales, and RGB or gray channels. Figure 3 shows the ten face regions and the three scales of two particular face regions. We trained 60 ConvNets, each of which extracts two 160-dimensional DeepID vectors from a particular patch and its horizontally flipped counterpart. A special case is patches around the two eye centers and the two mouth corners, which are not flipped themselves, but the patches symmetric with them (for example, the flipped counterpart of the patch centered on the left eye is derived by flipping the patch centered on the right eye). The total length of DeepID is 19*;* 200 (160 2 60), which is ready for the final face verification.

3.3. Face verification

We use the Joint Bayesian [8] technique for face verification based on the DeepID. Joint Bayesian has been highly successful for face verification [9, 6]. It represents the extracted facial features x (after subtracting the mean) by the sum of two independent Gaussian variables

$$x = \quad + \quad , \tag{5}$$

where $N(0; S\ )$ represents the face identity and $N(0; S\ )$ the intra-personal variations. Joint Bayesian models the joint probability of two faces given the intra- or extra-personal variation hypothesis, $P(x_1; x_2 \mid H_I)$ and $P(x_1; x_2 \mid H_E)$. It is readily shown from Equation 5 that these two probabilities are also Gaussian with variations

$$_I = \begin{matrix} S \ + S & S \\ S & S \ + S \end{matrix} \tag{6}$$

and

$$_E = \begin{matrix} S \ + S & 0 \\ 0 & S \ + S \end{matrix} \ , \tag{7}$$

respectively. S and S can be learned from data with EM algorithm. In test, it calculates the likelihood ratio

$$r(x_1; x_2) = \log \frac{P(x_1; x_2 \mid H_I)}{P(x_1; x_2 \mid H_E)} , \tag{8}$$

which has closed-form solutions and is efficient.

We also train a neural network for verification and compare it to Joint Bayesian to see if other models can also learn from the extracted features and how much the features and a good face verification model contribute to the performance, respectively. The neural network contains one input layer

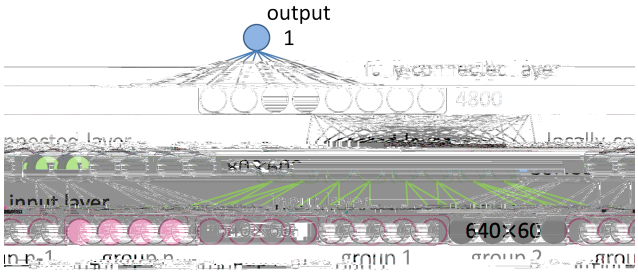


Figure 4. The structure of the neural network used for face verification. The layer type and dimension are labeled beside each layer. The solid neurons form a subnetwork.

taking the DeepID, one locally-connected layer, one fully-connected layer, and a single output neuron indicating face similarities. The input features are divided into 60 groups, each of which contains 640 features extracted from a particular patch pair with a particular ConvNet. Features in the same group are highly correlated. Neurons in the locally-connected layer only connect to a single group of features to learn their local relations and reduce the feature dimension at the same time. The second hidden layer is fully-connected to the first hidden layer to learn global relations. The single output neuron is fully connected to the second hidden layer. The hidden neurons are ReLUs and the output neuron is sigmoid. An illustration of the neural network structure is shown in Figure 4. It has 38*;* 400 input neurons with 19*;* 200 DeepID features from each patch, and 4*;* 800 neurons in the following two hidden layers, with every 80 neurons in the first hidden layer locally connected

[31] and tested on LFW (Section 4.1 - 4.3). CelebFaces contains 87;628 face images of 5436 celebrities from the Internet, with approximately 16 images per person on average. People in LFW and CelebFaces are mutually exclusive.

We randomly choose 80% (4349) people from CelebFaces to learn the DeepID, and use the remaining 20% people to learn the face verification model (Joint Bayesian or neural networks). For feature learning, ConvNets are supervised to classify the 4349 people simultaneously from a particular kind of face patches and their flipped counterparts. We randomly select 10% images of each training person to generate the validation data. After each training epoch, we observe the top-1 validation set error rates and select the model that provides the lowest one.

In face verification, our feature dimension is reduced to 150 by PCA before learning the Joint Bayesian model. Performance almost retains in a wide range of dimensions. In test, each face pair is classified by comparing the Joint Bayesian likelihood ratio to a threshold optimized in the training data.

To evaluate the performance of our approach at an even larger training scale in Section 4.4, we extend CelebFaces to the CelebFaces+ dataset, which contains 202;599 face images of 10;177 celebrities. Again, people in LFW and CelebFaces+ are mutually exclusive. The ConvNet structure and feature extraction process described in the previous section remains unchanged.

4.1. Multi-scale ConvNets

We verify the effectiveness of directly connecting neurons in the third convolutional layer (after max-pooling) to the last hidden layer (the DeepID layer), such that it sees both the third and fourth convolutional layer features, forming the so-called multi-scale ConvNets. It also results in reducing feature numbers from the convolutional layers to the DeepID layer (shown in Figure 1), which helps the latter to learn higher-level features in order to well represent the face identities with fewer neurons. Figure 5 compares the top-1 validation set error rates of the 60 ConvNets learned to classify the 4349 classes of identities, either with or without the skipping layer. The lower error rates indicate the better hidden features learned. Allowing the DeepID to pool over multi-scale features reduces validation errors by an average of 4:72%. It actually also improves the final face verification accuracy from 95:35% to 96:05% when concatenating the DeepID from the 60 ConvNets and using Joint Bayesian for face verification.

4.2. Learning effective features

Classifying a large number of identities simultaneously is key to learning discriminative and compact hidden features. To verify this, we increase the identity classes

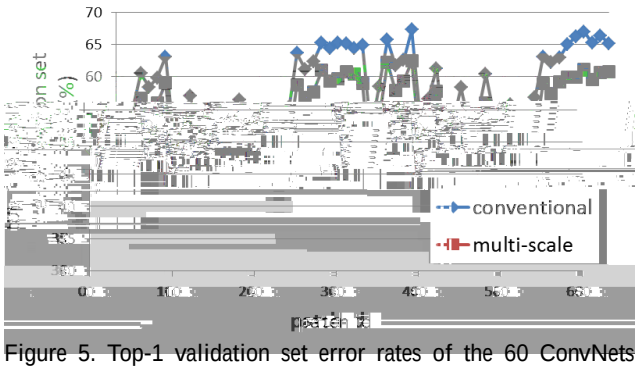


Figure 5. Top-1 validation set error rates of the 60 ConvNets trained on the 60 different patches. The blue and red markers show error rates of the conventional ConvNets (without the skipping layer) and the multi-scale ConvNets, respectively.

for training exponentially (and output neuron numbers correspondingly) from 136 to 4349 while fixing the neuron numbers in all previous layers (the DeepID is kept to be 160 dimensional). We observe the classification ability of ConvNets (measured by the top-1 validation set error rates) and the effectiveness of the learned hidden representations for face verification (measured by the test set verification accuracy) with the increasing identity classes. The input is a single patch covering the whole face in this experiment. As shown in Figure 6, both Joint Bayesian and neural network improve linearly in verification accuracy when the identity classes double. The improvement is significant. When identity classes increase 32 times from 136 to 4349, the accuracy increases by 10:13% and 8:42% for Joint Bayesian and neural networks, respectively, or 2:03% and 1:68% on average, respectively, whenever the identity classes double. At the same time, the validation set error rates drop, even when the predicted classes are tens of times more than the last hidden layer neurons, as shown in Figure 7. This phenomenon indicates that ConvNets can learn from classifying each identity and form shared hidden representations that can classify all the identities well. More identity classes help to learn better hidden representations that can distinguish more people (discriminative) without increasing the feature length (compact). The linear increasing of test accuracy with respect to the exponentially increasing training data indicates that our features would be further improved if even more identities are available. Examples of the 160-dimensional DeepID learned from the 4349 training identities and extracted from LFW test pairs are shown in Figure 8. We find that faces of the same identity tend to have more commonly activated neurons (positive features being in the same position) than those of different identities. So the learned features extract identity information.

We also test the 4349-dimensional classifier outputs as features for face verification. Joint Bayesian only achieves approximately 66% accuracy on these features, while the neural network fails, where it accounts all the face pairs as

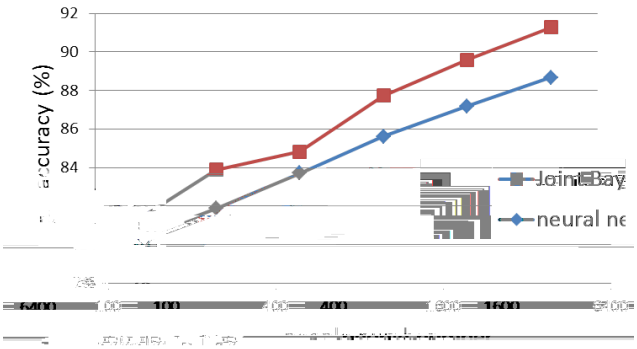


Figure 6. Face verification accuracy of Joint Bayesian (red line) and neural network (blue line) learned from the DeepID, where the ConvNets are trained with 136, 272, 544, 1087, 2175, and 4349 classes, respectively.

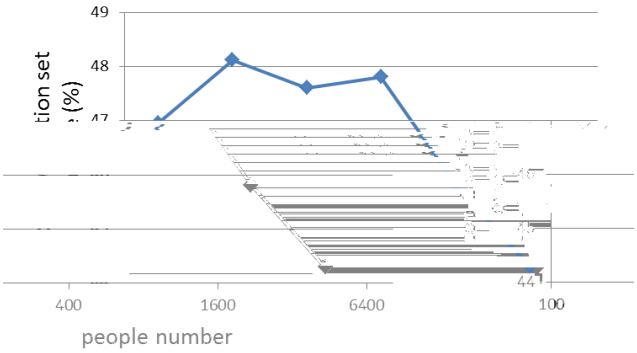


Figure 7. Top-1 validation set error rates of ConvNets learned to classify 136, 272, 544, 1087, 2175, and 4349 classes, respectively.

positive or negative pairs. With so many classes and few samples for each class, the classifier outputs are diverse and unreliable, therefore cannot be used as features.

4.3. Over-complete representation

We evaluate how much combining features extracted from various face patches would contribute to the performance. We train the face verification model with features from k patches (k = 1; 5; 15; 30; 60). It is impossible to numerate all the possible combinations of patches, so we select the most representative ones. We report the best-performing single patch (k = 1), the global color patches in a single scale (k = 5), all the global color patches (k = 15), all the color patches (k = 30), and all the patches (k = 60). As shown in Figure 9, adding more features from various regions, scales, and color channels consistently improves the performance. Combing 60 patches increases the accuracy by 4:53% and 5:27% over best single patch for Joint Bayesian and neural networks, respectively. We achieve 96:05% and 94:32% accuracy using Joint Bayesian and neural networks, respectively. The curves show that the performance may be further improved if more features are extracted.

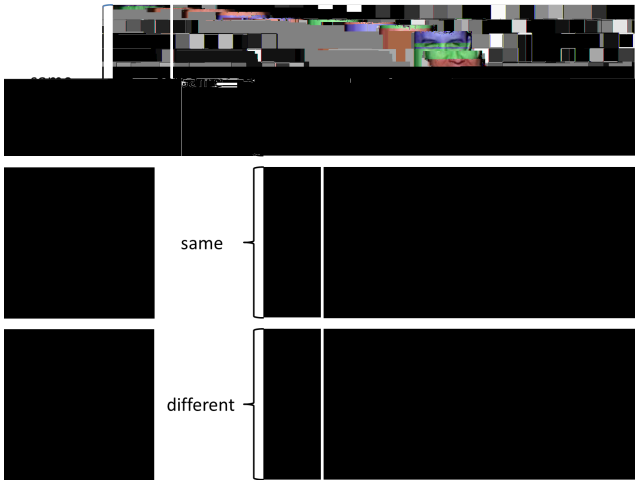
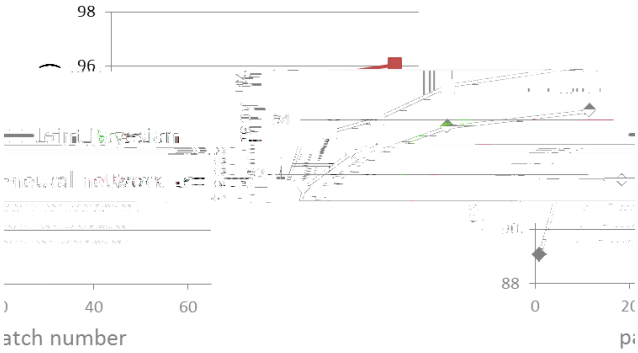


Figure 8. Examples of the learned 160-dimensional DeepID. The left column shows three test pairs in LFW. The first two pairs are of the same identity, the third one is of different identities. The corresponding features extracted from each patch are shown in the right. The features are in one dimension. We rearrange them as 5 32 for the convenience of illustration. The feature values are non-negative since they are taken from the ReLUs. Approximately 40% features have positive values. The brighter squares indicate higher values.



a 32,000-dimensional DeepID feature vector, which is then reduced to 150 dimensions by PCA. Joint Bayesian learned on this 150-dimensional feature vector achieves 97.20% test accuracy on LFW.

Due to the difference in data distributions, models well fitted to CelebFaces+ may not have equal generalization ability on LFW. To solve this problem, Cao et al. [6] proposed a practical transfer learning algorithm to adapt the Joint Bayesian model from the source domain to the target domain. We implemented their algorithm by using the 1477 people from CelebFaces+ as the source domain data and nine out of ten folders from LFW as the target domain data for transfer learning Joint Bayesian, and conduct ten-fold cross validation on LFW. The transfer learning Joint Bayesian based on our DeepID features achieves 97.45% test accuracy on LFW, which is on par with the human-level performance of 97.53%.

We compare with the state-of-the-art face verification methods on LFW. In the comparison, we report three results. The first two are trained on CelebFaces and CelebFaces+, respectively, without transfer learning, and tested on LFW. The third one is trained on CelebFaces+ with transfer learning on LFW. Table 1 comprehensively compares the accuracies, the number of facial points used for alignment, the number of outside training images (if applicable), and the final feature dimensions for each face (if applicable). Low feature dimensions indicate efficient face recognition systems. Figure 10 compares the ROC curves. Our DeepID learning method achieves the best performance on LFW. The first four best methods compared used dense facial landmarks, while our faces are weakly aligned with only five points. The deep learning work (DeepFace) [32] independently developed by Facebook at the same time of this paper achieved the second best performance of 97.25% accuracy on LFW. It utilized 3D alignment and pose transform as preprocessing, and more than seven million outside training images plus training images from LFW.

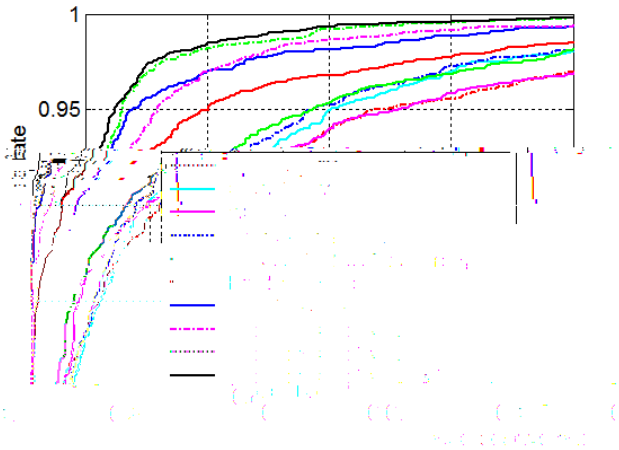


Figure 10. ROC comparison with the state-of-the-art face verification methods on LFW. TL in our method means transfer learning Joint Bayesian.

5. Conclusion and Discussion

This paper proposed to learn effective high-level features revealing identities for face verification. The features are built on top of the feature extraction hierarchy of deep ConvNets and are summarized from multi-scale mid-level features. By representing a large amount of different identities with a small number of hidden variables, highly compact and discriminative features are acquired. The features extracted from different face regions are complementary and further boost the performance. It achieved 97.45% face verification accuracy on LFW, while only requiring weakly aligned faces.

Even more compact and discriminative DeepID can be learned if more identities are available to increase the dimensionality of prediction at the training stage. We look forward to larger training sets to further boost our performance. A recent work [27] reported 98.52% accuracy on LFW with Gaussian Processes and multi-source training sets, achieving even higher than human performance. This could be due to the fact that the nonparametric Bayesian kernel method can adapt model complexity to data distribution. Gaussian processes can also be modeled with deep learning [12].

Method	Accuracy (%)	No. of points	No. of images	Feature dimension
Joint Bayesian [8]	92.42 (o)	5	99,773	2000 4
ConvNet-RBM [31]	92.52 (o)	3	87,628	N/A
CMD+SLBP [17]	92.58 (u)	3	N/A	2302
Fisher vector faces [29]	93.03 (u)	9	N/A	128 2
Tom-vs-Pete classifiers [2]	93.30 (o+r)	95	20,639	5000
High-dim LBP [9]	95.17 (o)	27	99,773	2000
TL Joint Bayesian [6]	96.33 (o+u)	27	99,773	2000
DeepFace [32]	97.25 (o+u)	6 + 67	4,400,000 + 3,000,000	4096 4
DeepID on CelebFaces	**96.05** (o)	5	87,628	150
DeepID on CelebFaces+	**97.20** (o)	5	202,599	150
DeepID on CelebFaces+ & TL	**97.45** (o+u)	5	202,599	150

Table 1. Comparison of state-of-the-art face verification methods on LFW. Column 2 compares accuracy. Letters in the parentheses denote the training protocols used. r denotes the restricted training protocol, where the 6000 face pairs given by LFW are used for ten-fold cross-validation. u denotes the unrestricted protocol, where additional training pairs can be generated from LFW using the identity information. o denotes using outside training data, however, without using training data from LFW. o+r denotes using both outside data and LFW data in the restricted protocol for training. (o+u) denotes using both outside data and LFW data in the unrestricted protocol for training. Column 3 compares the number of facial points used for alignment. Column 4 compares the number of outside images used for training (if applicable). The last column compares the final feature dimensions for each face (if applicable). DeepFace used six 2D points and 67 3D points for alignment. TL in our method means transfer learning Joint Bayesian.

[6] X. Cao, D. Wipf, F. Wen, G. Duan, and J. Sun. A practical transfer learning algorithm for face verification. In *Proc. ICCV*, 2013. 1, 4, 7, 8

[7] Z. Cao, Q. Yin, X. Tang, and J. Sun. Face recognition with learning-based descriptor. In *Proc. CVPR*, 2010. 1, 2

[8] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun. Bayesian face revisited: A joint formulation. In *Proc. ECCV*, 2012. 1, 2, 4, 8

[9] D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In *Proc. CVPR*, 2013. 1, 2, 4, 8

[10] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proc. CVPR*, 2005. 2

[11] D. Ciresan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *Proc. CVPR*, 2012. 1

[12] A. Daminaou and N. Lawrence. Deep gaussian processes. *JMLR*, 31:207–215, 2014. 7

[13] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *Proc. ICML*, 2007. 2

[14] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *PAMI*, 35:1915–1929, 2013. 1

[15] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *Proc. ICCV*, 2009. 1

[16] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, abs/1207.0580, 2012. 4

[17] C. Huang, S. Zhu, and K. Yu. Large scale strongly supervised ensemble metric learning, with applications to face verification and retrieval. *NEC Technical Report TR115*, 2011. 1, 2, 8

[18] G. B. Huang, H. Lee, and E. Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In *Proc. CVPR*, 2012. 1, 2, 3

[19] 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, 2007. 1

[20] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NIPS*, 2012. 1, 3

[21] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *Proc. ICCV*, 2009. 1, 2

[22] H. Larochelle, M. Mandel, R. Pascanu, and Y. Bengio. Learning algorithms for the classification restricted Boltzmann machine. *JMLR*, 13:643–669, 2012. 2

[23] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *Proc. CVPR*, 2011. 2

[24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 1

[25] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proc. ICML*, 2009. 2

[26] P. Li, S. Prince, Y. Fu, U. Mohammed, and J. Elder. Probabilistic models for inference about identity. *PAMI*, 34:144–157, 2012. 1

[27] C. Lu and X. Tang. Surpassing human-level face verification performance on lfw with gaussianface. Technical report, arXiv:1404.3840, 2014. 7

[28] P. Sermanet and Y. Lecun. Traffic sign recognition with multi-scale convolutional networks. In *Proc. IJCNN*, 2011. 3

[29] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman. Fisher vector faces in the wild. In *Proc. BMVC*, 2013. 1, 2, 8

[30] Y. Sun, X. Wang, and X. Tang. Deep convolutional network cascade for facial point detection. In *Proc. CVPR*, 2013. 4

[31] Y. Sun, X. Wang, and X. Tang. Hybrid deep learning for face verification. In *Proc. ICCV*, 2013. 1, 2, 5, 8

[32] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. DeepFace: Closing the gap to human-level performance in face verification. In *Proc. CVPR*, 2014. 1, 7, 8

[33] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on riemannian manifolds. *PAMI*, 30:1713–1727, 2008. 2

[34] Q. Yin, X. Tang, and J. Sun. An associate-predict model for face recognition. In *Proc. CVPR*, 2011. 1

[35] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning identity-preserving face space. In *Proc. ICCV*, 2013. 2

[36] Z. Zhu, P. Luo, X. Wang, and X. Tang. Recover canonical-view faces in the wild with deep neural networks. Technical report, arXiv:1404.3543, 2014. 1, 2