Learning Robust Deep Face Representation

Xiang Wu
University of Science and Technology Beijing
Beijing, China
alfredxiangwu@gmail.com

Abstract

*With the development of convolution neural network, more and more researchers focus their attention on the advantage of CNN for face recognition task. In this paper, we propose a deep convolution network for learning a robust face representation. The deep convolution net is constructed by 4 convolution layers, 4 max pooling layers and 2 fully connected layers, which totally contains about 4M parameters. The Max-Feature-Map activation function is used instead of ReLU because the ReLU might lead to the loss of information due to the sparsity while the Max-Feature-Map can get the compact and discriminative feature vectors. The model is trained on CASIA-WebFace dataset and evaluated on LFW dataset. The result on LFW achieves **97.77%** on unsupervised setting for **single net**.*

1. Introduction

In the past years, with the development of convolution neural network, numerous vision tasks benefit from a compact representation learning via deep model from image data. The performance in various computer vision applications, such as image classification[3], object detection[14], face recognition[11, 15, 16] and so on, achieved great progress.

For the face verification task, the accuracy on LFW, a hard benchmark dataset, has been improved from 97%[15] to 99%[11] in recent year via deep learning model. The main frameworks for face verification are based on multi-class classification[13, 15] to extract face feature vectors and then the vectors are further processed by classifiers or patch model ensembles. However, the probability models such as Joint Bayesian[1] and Gaussian Processing[8] are based on strong assumptions which may not make effect on various situations. Other methods[5, 10] are proposed to optimize verification loss directly for matching pairs and non-matching pairs. The disadvantage of these verification based methods is that it is difficult to select training dataset for negative pairs and the threshold in verification loss function is set manually. Moreover, the joint identification and verification constraint is used for optimizing the deep face model in [11, 16] and it is also difficult to set the trade-off parameter between identification and verification loss for multi-task optimization.

In this paper, we propose a deep robust face representation learning framework. We utilize convolution networks and propose a Max-Feature-Map activation function, which the model is trained on CASIA-WebFace dataset[1] and evaluated on LFW dataset.

The contributions of this paper are summarized as follows:

(1) We propose a Max-Feature-Map activation function whose values are not sparse while the gradients are sparse instead. The activation function can also be treated as the sparse connection to learn a robust representation for deep model.

(2) We build a **shallower single** convolution network and get better performance than DeepFace[15], DeepID2[11] and WebFace[16].

The paper is organized as following. Section 2 briefly describes our convolution network framework and Max-Feature-Map activation function. We present our experimental results in Section 3 and conclude in Section 4.

2. Architecture

In this section, we describe the framework of our deep face representation model and the compact Max-Feature-Map activation function.

2.1. Compact Activation Function

Sigmoid or Tanh is a nonlinear activation for neural network and often leads to robust optimization during DNN training[4]. But it may suffer from vanishing gradient when lower layers have gradients of nearly 0 because higher layer units are nearly saturate at -1 or 1. The vanishing gradient may lead to converge slow or poor local optima.

[1] http://www.cbsr.ia.ac.cn/english/CASIA-WebFace-Database.html

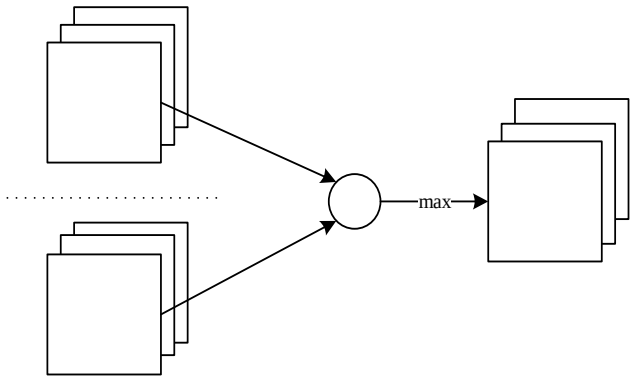

Figure 1. Operation performed by Max-Feature-Map activation function

To overcome vanishing gradient, the Rectified linear unit(ReLU)[9] offers a sparse representation. However, ReLU is at a potential disadvantage during optimization because the value is 0 if the unit is not active. It might lead to loss of some information especially for the first several convolution layers because these layers are similar to Gabor filter which both positive and negative responses are respected. To alleviate this problem, PReLU is proposed and it makes good effect on ImageNet classification task[3].

In order to make the representation compact instead of sparsity in ReLU, we propose the Max-Feature-Map(MFM) activation function which is inspired by [2]. Given an input convolution layer $C \in \mathbb{R}^{h \times w \times 2n}$, as is shown in Fig.1, the Max-Feature-Map activation function can be written as

$$f = C_{ij}^{k'} = \max_{1 \le k \le n} (C_{ij}^{k}, C_{ij}^{k+n}) \tag{1}$$

where the number of convolution feature map C is $2n$. The gradient of this activation function can be shown as

$$\frac{\partial f}{\partial C^k} = \begin{cases} 1, & \text{if } C^k \ge C^{k+n} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

The Max-Feature-Map activation function is not a normal single-input-single-output function such as sigmoid or ReLU, while it is the maximum between two convolution feature map candidate nodes. This activation function can not only select competitive nodes for convolution candidates, but also make the 50% gradients of convolution layers are 0. Moreover, the Max-Feature-Map activation function can also treated as the sparse connection between two convolution layers, which can encode the information sparsely onto a feature space.

2.2. Convolution Network Framework

The deep face convolution network is constructed by four convolution layers, 4 max pooling layers, Max-Feature-Map activation functions and 2 fully connected layers as is shown in Fig.2.

Table 1. The architecture of the proposed deep face convolution network.

Name	Type	Filter Size /Stride	Output Size
input	-	-	$144 \times 144 \times 1$
crop	-	-	$128 \times 128 \times 1$
conv1_1	convolution	$9 \times 9/1$	$120 \times 120 \times 48$
conv1_2	convolution	$9 \times 9/1$	$120 \times 120 \times 48$
mfm1	MFM	-	$120 \times 120 \times 48$
pool1	max pooling	$2 \times 2/2$	$60 \times 60 \times 48$
conv2_1	convolution	$5 \times 5/1$	$56 \times 56 \times 96$
conv2_2	convolution	$5 \times 5/1$	$56 \times 56 \times 96$
mfm2	MFM	-	$56 \times 56 \times 96$
pool2	max pooling	$2 \times 2/2$	$28 \times 28 \times 96$
conv3_1	convolution	$5 \times 5/1$	$24 \times 24 \times 128$
conv3_2	convolution	$5 \times 5/1$	$24 \times 24 \times 128$
mfm3	MFM	-	$24 \times 24 \times 128$
pool3	max pooling	$2 \times 2/2$	$12 \times 12 \times 128$
conv4_1	convolution	$4 \times 4/1$	$9 \times 9 \times 192$
conv4_2	convolution	$4 \times 4/1$	$9 \times 9 \times 192$
mfm4	MFM	-	$9 \times 9 \times 192$
pool4	max pooling	$2 \times 2/2$	$5 \times 5 \times 192$
fc1	fully connected	-	256
fc2	fully connected	-	10575
loss	softmax	-	10575

The input image is 144×144 gray-scale face image from CASIA-WebFace dataset. The detail parameters setting is presented in Table.1. We crop each input image randomly into 128×128 patch as the input of the first convolution layer. The network include 4 convolution layers that each convolution layer is combined with two independent convolution parts calculated from the input. The Max-Feature-Map activation function and max pooling layer are used later. The fc1 layer is a 256-dimensional face representation since we usually consider that the face images usually lie on a low dimensional manifold and it is effective to reduce the complexity of the convolution neural network. The fc2 layer is used as the input of the softmax cost function and is set to the number of WebFace identities(10575). Besides, the proposed network has 4153K parameters which is smaller than DeepFace and WebFace net.

3. Experiments

3.1. Data Pre-processing

CASIA-WebFace dataset is used to train our deep face convolution network. It contains 493456 face images of 10575 identities and all the face images are converted to gray-scale and normalized to 144×144 according to land-

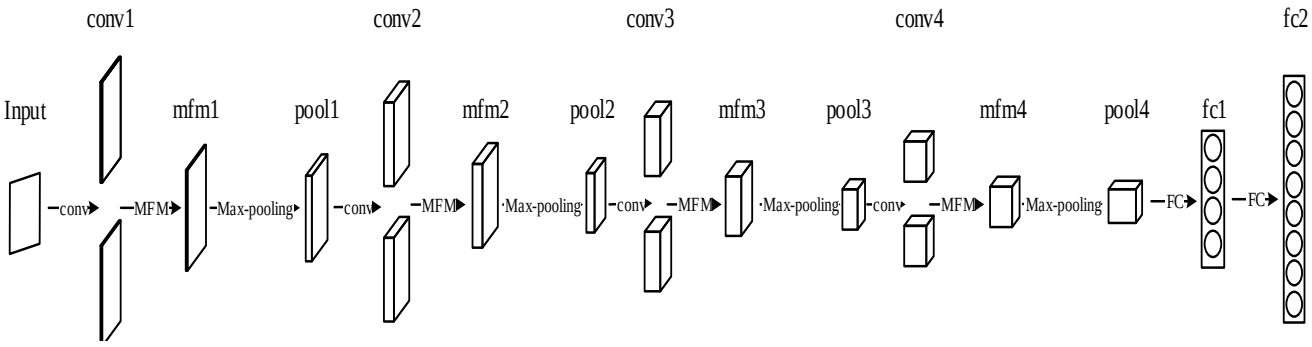

Figure 2. An illustration of the architecture of our deep face convolution networks model.

(a) (b)

Figure 3. Face image alignment for WebFace dataset. (a) is the facial points detection results and (b) is the normalization face image.

marks as is shown in Fig.3(a). According to the 5 facial points extracted by [12] and manually revised, the distance between the midpoint of eyes and the midpoint of mouth is relative invariant to pose variations in yaw angle, therefore, it is fixed to 50 pixels and we also rotate two eye points horizontally to pos variations in roll angle. The normalization face image is shown in Fig.3(b).

3.2. Training Methodology

To train the convolution network, we randomly select one face image from each identity as the validation set and the other images as the train set. The open source deep learning framework *Caffe*[6] is used for training the model.

The input for convolution network is the 144×144 gray-scale face image and we crop the input image into 128×128 and mirror it. These data augmentation method can improve the generalization of the convolution neural network and overcome the overfitting[7]. Dropout is also used for fully connected layer and the ratio is set to 0.7.

Moreover, the weight decay is set to 5e-4 for convolution layer and fully connected layer except the fc2 layer. It is obvious that the fc1 face representation is only used for face verification tasks which is not similar to the image classification and objection task. However, the parameters of fc2 layer is very large. Therefore, it might lead to overfitting for learning the large fully-connected layer parameters. To overcome it, we set the weight decay of fc2 layer to 5e-3.

The learning rate is set to 1e-3 initially and reduce to 5e-5 gradually. The parameters initialization for convolution is Xavier and Gaussian is used for fully-connected layers. Moreover, the deep model is trained on GTX980 and the iteration is set to 2 million.

3.3. Results on LFW benchmark

The evaluation is performed on LFW dataset[2] in detail. LFW dataset contains 13233 images of 5749 people for face verification. And all the images in LFW dataset are processed by the same pipeline as the training dataset and normalized to 128×128.

For evaluation, the face data is divided in 10 folds which contain different identities and 600 face pairs. There are two evaluation setting about LFW training and testing: restricted and unrestricted. In restricted setting, the pre-define image pairs are fixed by author (each fold contains 5400 pairs for training and 600 pairs for testing). And in unrestricted setting, the identities of people within each fold for training is allowed to be much larger.

According to Fig.4, compared with ReLU and Max-Feature-Map, the speed of convergence for Max-Feature-Map network is slower than ReLU due to the complexity of the activation and the randomness of initial parameters. However, with the progress of training, the validation accuracy for Max-Feature-Map net outperforms ReLU.

We test our deep model performance via cosine similarity and ROC curve. The results[3] are shown in Table.2 and the EER on LFW achieves 97.77%, which outperforms DeepFace[15], DeepID2[11] and WebFace[16] for unsupervised setting[4] for **single** net.

[2]http://vis-www.cs.umass.edu/lfw/

[3]The model and configuration are released on my Github

[4]The unsupervised setting the model is not trained on LFW in supervised way.

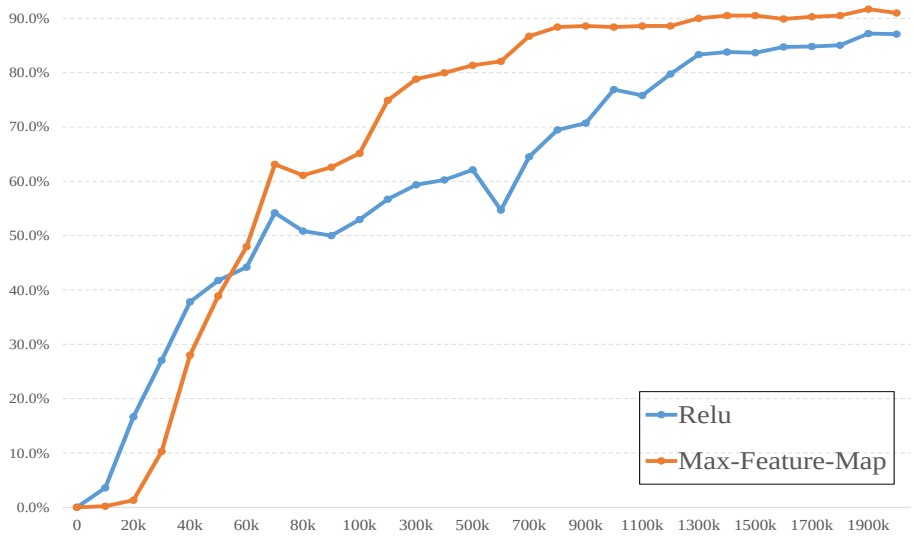

Figure 4. Comparison with ReLU activation function and Max-Feature-Map activation function validation accuracy for CNN training.

Table 2. The performance of our deep face model and compared methods on LFW.

Method	#Net	Accuracy	Protocol
DeepFace	1	95.92%	unsupervised
DeepFace	1	97.00%	restricted
DeepFace	7	97.35%	unrestricted
DeepID2	1	95.43%	unsupervised
DeepID2	2	97.28%	unsupervised
DeepID2	4	97.75%	unsupervised
DeepID2	25	98.97%	unsupervised
WebFace	1	96.13%	unsupervised
WebFace+PCA	1	96.30%	unsupervised
WebFace+Joint Bayes	1	97.30%	unsupervised
WebFace+Joint Bayes	1	97.73%	unrestricted
Our model(ReLU)	1	97.45%	unsupervised
Our model(MFM)	1	**97.77%**	unsupervised

4. Conclusions

In this paper, we proposed a deep convolution network for learning a robust face representation. We use Max-Feature-Map activation function to learn a compact low-dimensional face representation and the results on LFW is 97.77%, which the performance is the state-of-the-art on unsupervised setting for single net as far as we know.

References

[1] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun. Bayesian face revisited: A joint formulation. In *Computer Vision–ECCV 2012*, pages 566–579. Springer, 2012.

[2] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout networks. *arXiv preprint arXiv:1302.4389*, 2013.

[3] 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.

[4] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[5] J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face verification in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1875–1882. IEEE, 2014.

[6] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

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

[8] C. Lu and X. Tang. Surpassing human-level face verification performance on lfw with gaussianface. *arXiv preprint arXiv:1404.3840*, 2014.

[9] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

[10] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *arXiv preprint arXiv:1503.03832*, 2015.

[11] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *Advances in Neural Information Processing Systems*, pages 1988–1996, 2014.

[12] Y. Sun, X. Wang, and X. Tang. Deep convolutional network cascade for facial point detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3476–3483. IEEE, 2013.

[13] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1891–1898. IEEE, 2014.

[14] C. Szegedy, S. Reed, D. Erhan, and D. Anguelov. Scalable, high-quality object detection. *arXiv preprint arXiv:1412.1441*, 2014.

[15] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1701–1708. IEEE, 2014.

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