Compressed Deep Convolution Neural Network for Face Recognition

Ying Zou[1, a], Xiaohong Liu[1, b]

[1]School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China

[a]zouying_1113@126.com, [b]xiaohongliu@bupt.edu.cn

Keywords: Face recognition, Convolutional Neural Network, model compression.

Abstract. Deep convolution neural network (CNN) has achieved a great success on face recognition techniques. But most of CNN models tend to be much deeper, which are at the expenses of high consumption of computation and storage. So, it is hard for these deep CNNs applied to mobile equipments because of poor computational and memory resources. To alleviate this issue, this paper optimizes a lightened baseline CNN model by adopting an additional contrastive loss to learn more discriminative features. To further reduce the number of parameters, a pruning strategy is tried to compress our model, which slightly improves accuracy on the LFW dataset with the compression ratio of 0.7. Finally, experimental result shows that the proposed method achieve state-of-the-art results with much smaller size and fewer training data.

Introduction

In recent years, face recognition in unconstrained conditions achieves a significant breakthrough. Among the traditional algorithms of face recognition, the first step is to extract features from raw face images, such as SIFT, HOG, LBP features with high dimensions. Then use a classifier to complete our recognition task. While CNN network could learn feature vectors to represent a face image automatically, it seems to be more helpful to use CNN model for face recognition task. Deep CNN achieves high accuracy at the expenses of high consumption of computation and storage with large number of parameters. Although NVIDIA company has developed a series of high performance computing GPU to make deep learning more hopeful, it is less practical for mobile equipment because of their poor computation and storage resource. So it is important to design a CNN model with fast speed, small storage as well as high accuracy.

To achieve this goal, this work utilizes the lightened CNN model and optimize it by adding a contrastive loss to minimize the intra-class distances of deep features. Finally we use the pruning method to convert the dense model to a sparse layer to reduce the size of model storage. The rest of our paper is organized as follows: Section 2 compares our work against recent works on face recognition. The details of our proposed method and experiment are presented in Section 3 and Section 4 respectively. Finally, we draw a conclusion in Section 5.

Related work

Many popular face verification methods design a deep CNN network and extract features as the representation of a face. DeepFace [1] trains a deep CNN model with a large face database which contains 4M face images of more than 4K subjects as training data. DeepFace also uses a 3D alignment as data preprocessing to handle out-of-plane rotations. DeepID [2,3,4] series could be regarded as a set of representative work of face recognition with deep learning. DeepID [2] was firstly proposed by Sun et al. trained 25 CNN models with four convolution layers. To further improve accuracy, DeepID2 [3] combined identification and verification supervisory signals based on DeepID, while DeepID2+ [4] added verification supervisory signal for each layer. Google proposed FaceNet [5] in CVPR 2015, which is a very deep CNN model containing 22 layers and trained on totally about 200M face images with 8M identities and adopted triplet loss as supervisory signal.

Although these methods essentially demonstrate the effectiveness of CNN model for feature learning, there are still some problems. For example, the DeepID series trained multiple CNN models and extracted multi patches features, while both VGG-Face[6] and FaceNet designed very deep network with a large number of parameters. All of these complex network could achieve high recognition ratio with high computational costs and memory consumption. As embedded mobile applications usually have poor computational and memory resources, it is hard for deep CNN applied to mobile equipment. This paper proposes a method to realize a face recognition CNN model with high accuracy and small size, which is potentially suitable and practical for mobile equipment.

Proposed Method

Architecture

We utilize the lightened CNN [7] as our baseline model. It uses MFM activation function spired by maxout [8] network instead of conventional activation functions. The model could extract more discriminative representations in comparison with the activations such as sigmoid or ReLU activations. The lightened CNN contains 6 layers including 4 convolution and 2 fully connected layers. The first convolution layer creates 96 outputs with filter size of 9×9, while the second convolution layer has 192 outputs with 5×5 filters. The third convolution layer creates 256 feature maps with 5×5 filters, while the last convolution layer creates 384 outputs with 4×4 filters. The dimensions of following two fully connected layers are 256 and 10575 with Softmax as loss function.

Contrastive Loss

As our task is face verification, the model is supposed not only to separate features but also to discriminate learned features. The Softmax layer acts as a classifier and is prone to separate extracted features, which is not enough for our recognition model for its weak constraint on features from the same identity. Hence, an additional contrastive loss could encourage features from the same identity more similar. As shown in figure.1, the combination of Softmax loss and contrastive loss could learn more discriminative features.

$$\mathrm{contras}(x_i, x_j, s_{i,j}) = \begin{cases} D(x_i, x_j) & s_{i,j} = 1 \\ \max(0, m - D(x_i, x_j)) & s_{i,j} = 0 \end{cases} \quad (1)$$

Where $D(x_i, x_j) = \| x_i - x_j \|_2^2$. We denote x_i and x_j as the vectors extracted from the two face images in comparison, $s_{i,j} = 1$ means x_i and x_j are from the same identity, otherwise means different identities. In this way, Equation. (1) minimizes the L2 distance between similar pairs, while requires the distance larger than a margin parameter m .The final loss function of our method is

$$loss = \mathrm{softmax}(S, I) + \lambda \mathrm{contras}(P, L) \quad (2)$$

Where S is single face of training data I is the identity of face, $\mathrm{softmax}(X, Y)$ is classification loss and $\mathrm{contras}(P, S)$ means the verification loss. P means the training pairs while L means the label of pairs. In this case, we could learn more discriminative features by combining with the classification loss with our additional contrastive loss.

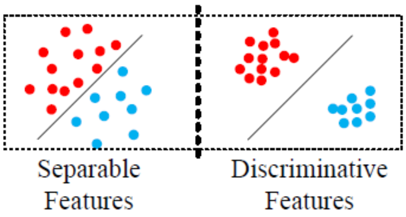

Fig. 1 An illustraion of contrastive loss

Network Pruning

As the weight values of CNN model are always sparse, network pruning has been widely used to compress CNN networks, it is a valid method to avoiding the over fitting by reducing model complexity. We adopt a pruning strategy inspired by [9] to prune the connections of CNN according to their contribution. First of all, we train our model with the methods mentioned above, the weight values are not parameters of final model, but a basis for contribution of connections. Then, the most critical step is pruning, we remove all the less important connections according to the step one. The last step fine-tunes the pruned model to restore accuracy of recognition.

Experiment

We use CASIA-WebFace [10] dataset as training data which contains about 0.5M face images with 10575 identities. We detect faces by extracting 5 facial points, align faces and normalize images to 128×128 gray-scale. We also augment the training dataset by mirroring images. Our method is evaluated on the LFW [11] dataset. All the test images are pre-processed by the same pipeline as training data. Finally, the deep 256-d vector is extracted from the fc1 layer as the representation of a face, and cosine distance is used to measure the similarity of any two faces. We implemented our experiment with Caffe.

Network Pruning

Fig.2 shows the trade-off between compression ratio of fc1 and accuracy. With more connections pruned away, the overall trend of accuracy activity declines. There is little impact on accuracy with compression ratio below 0.5. And it is interesting that the accuracy is better than the original model when compression ratio is around 0.7. The reason is pruning and fine-tuning model could lead to a sparse network that could reduce the complexity of the model and prevent over fitting.

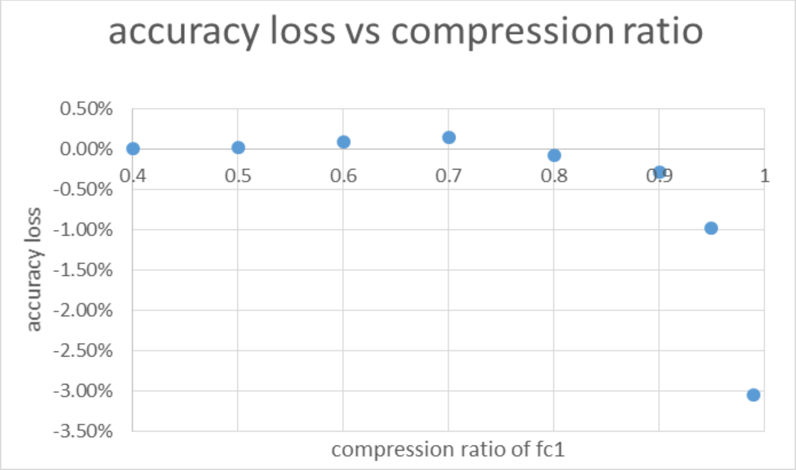

Fig.2 accuracy loss vs compression ratio

Weight value distribution of fc1 layer

Fig.3 shows the weight value distribution of fc1 layer with compression ratio ranging from 0 to 0.95. The original weight value distribution curve is one-humped with the weight value of center peak is zero. After pruning, the curve becomes a bimodal distribution. With the compression ratio increasing, the weight values of center peak are away from zero, and are distributed more uniformly. The more connections are pruned away, the remaining sparse network adjusts parameters to be more representative.

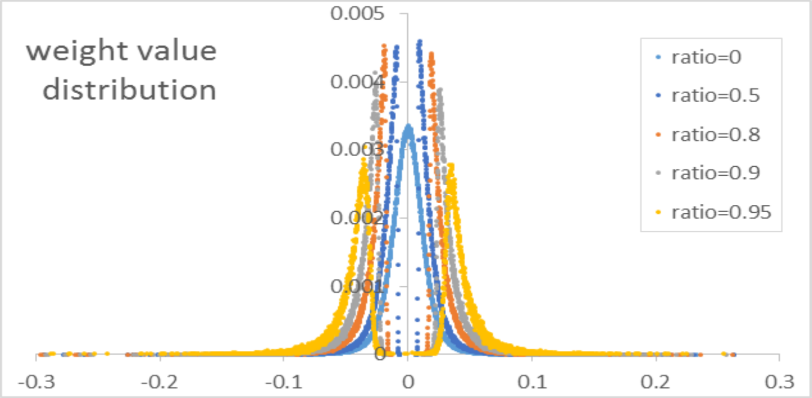

Fig.3 weight value percentage distribution

Comparison With Other Models

Tab.1 shows the comparison with other state-of-the-art methods on the LFW dataset. Our model outperforms the baseline lightened CNN by adding a contrastive loss, which learns more discriminate features. Besides, compared with other stae-of-the-art methods, our model‘s accuracy is higher than DeepFace and DeepID2 for single net. Also, our model outperforms the result of WebFace and other traditional methods, High-dim LBP [12] and Fisher Vector Face [13]. VGG-Face is slightly superior than ours with an extremely complex network which has about 167 times number of parameters than our model. It can be observed that our method achieves comparable results to the state of the art with much less training data, much smaller size.

Tab. 1 Comparison with other state-of-the-art methods on the LFW

Method	Training set	Networks	Params	Acc.
High-dim LBP	-	-	-	95.17%
Fisher Vector Face	-	-	-	93.03%
DeepFace	4.4M	1	28.9M	95.92%
DeepFace	4.4M	7	202.3M	97.35%
DeepID2	-	1	0.39M	95.43%
DeepID2	-	4	1.56M	97.75%
DeepID2	-	25	9.75M	98.97%
WebFace	0.5M	1	5M	96.13%
WebFace+PCA	0.5M	1	5M	96.30%
VGGFace	2.6M	1	134.2M	97.27%
Lightened CNN	0.5M	1	1.25M	96.83%
Lightened CNN +contrastive	**0.5M**	**1**	**1.25M**	**96.97%**
Lightened CNN+contrastive+prune	**0.5M**	**1**	**0.37M**	**97.11%**

Conclusions

This paper proposed a CNN model with small size for face recognition. We optimize the baseline lightened CNN with an additional contrastive loss with 0.14% percentage growth of accuracy. To further reduce the number of parameters, we adopt a pruning method mainly focused on fc1 layer. The size of parameters finally drops to 0.37M with slight accuracy growth due to reduction of over fitting. This results in smaller storage and computation, making it much practical for real time processing on mobile system. For future work, we plan to try to design a faster and more efficient network with better compressing strategy.

References

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

[2] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10, 000 classes. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, pages 1891–1898, 2014.

[3] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In Proceedings of Advances in Neural Information Processing Systems 27, pages 1988–1996, 2014.

[4] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, pages 2892–2900, 2015.

[5] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, pages 815–823, 2015.

[6] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. Proceedings of the British Machine Vision, 2015.

[7] X. Wu, R. He, and Z. Sun. A lightened cnn for deep face representation. In arXiv preprint arXiv:1511.02683v1, 2015.

[8] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio. Maxout networks. In Proceedings of the 30th International Conference on Machine Learning, pages 1319–1327, 2013.

[9] Han, Song, Pool, Jeff, Tran, John, and Dally, William J. Learning both weights and connections for efficient neural networks. In Advances in Neural Information Processing Systems, 2015.

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

[11] Huang G B, Ramesh M, Berg T, et al. Labeled faces in the wild: A database for studying face recognition in unconstrained environments[R]. Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[12] Chen D, Cao X D, Wen F, Sun J. Blessing of dimensionality: Highdimensional feature and its efficient compression for face verification. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2013.

[13] Simonyan K, Parkhi O M, Vedaldi A, Zisserman A. Fisher vector faces in the wild. In: Proceedings of British Machive Vision Conference. 2013.