Memory Optimized Deep Learning based Face Recognization

Mr. Amit Kumar
Research Scholar, Uttaranchal University, Dehradun
e-mail: amitk_msg@hotmail.com

Dr. Minakshi Memoria
Associate Professor, Uttaranchal University, Dehradun
e-mail: minakshimemoria@gmail.com

Dr. Vinod Kumar
Associate Professor, DTU. Delhi
e-mail: vinod_k@dtu.ac.in

Abstract - Machine learning powerful deep learning model is difficult to apply on a low segment embedded system or low segment mobile due to memory-constrained and battery-constrained platforms. This is a major problem, deep learning models have not been yet deployed on very low segment embedded systems such as smartphones (low price), smart wearables, traffic signals, and embedded systems. We are going to present a new way of doing Face Recognition with CNN to solve this problem in this paper. With the help of this, we can easily deploy a trained CNN based model on memory constrained platform such as low segments embedded system or low memory mobile devices/smart wearables, etc.

We present a novel and "very efficient" Network Architecture in this paper that includes MobileNetV2 with center loss and training tricks for "deep face recognition" for an embedded system that is memory constrained. In our tests, we ran our proposed Network Architecture on the network at some very memory constrained systems (as low as 2 MB). The result shows that when it is runs on a very low memory segment platform as compared to standard CNN based models, It produces very good results on various face verification datasets and model size less than 1MB for a memory-limited embedded device

***Keywords*:** Face recognition, CNN, memory-constrained devices

1. Introduction

Facial recognition is the "method of identifying or verifying the identity of subjects in images" [8]. It captures, analyzes, and compares the pattern based on user facial details. "Face recognition" is a method of "biometric identification" that uses users face to verify the identity [35] and It is contactless and non-intrusive nature which make more appealing and user-friendly than other biometric modalities like fingerprint recognition (need user finger on a sensor), speaker recognition (need a user to speak out loud), iris recognition (need user very close to the camera) while in "face recognition" systems it is only needed that the user are withing the reasonable distance and "field of view" of the camera [35]. The spectrum of possible facial recognition applications, such as access control, fraud detection, identity authentication, surveillance systems (users are not required to comply with the system), and social media, is much broader.

Over the years, face recognition methods have been moved fundamentally. Traditional approaches were focused on design elements using hand like texture descriptors and edges as well as integrated with machine learning methods like SVM, or "linear discriminant analysis" (LDA) [36]. Current techniques in face recognition have recently been overtaken by Convolutional neural networks (CNN) which is a deep learning technique. In case of face recognition CNN is one of the widely used and very popular deep learning techniques. Deep learning techniques have the biggest benefit of being trained easily with the vast value of data and it can adapt the face pattern which is resilient to the difference found in the training data. In this manner, instead of building strategic applications that are responsive to various forms of modifications in intra-class such as age, facial expression, pose illumination. CNN can be benefited from the training data. "Face recognition" methods which are based on CNN and trained with these types of datasets have performed extremely well in terms of accuracy because of their ability to learn and adapt the features which are responsive to the "real-world" significant differences available in the usage of "face images during training" [22]. In addition, the growth in the prevalence of computer vision deep learning approaches have increased face recognition work, as CNN is used to solve several other complex machine learning tasks such as optical character recognition, age estimation, segmentation, facial expression analysis, object detection, and recognition, etc. Training the "deep learning techniques" with a huge set of data which can involve adequate variations generalizing unseen samples is the major disadvantage of using deep learning techniques. Neural networks as well as learning discriminative

characteristics are able to minimize dimensionality and they can be trained by approaches based on metric learning or as a classifier. Three key factors influencing the performance of CNN based techniques used for "face recognition" are "CNN architecture, training data, and loss function" [22]. To avoid overfitting of the model, a large dataset (training set) is required in deep learning methods. When there are more samples per cases then the CNN based classification, models achieve more accuracy. It is because as more intra-class distinctions are revealed, the CNN model acquires more robust functionality. However, we are focused on feature extraction in face recognition which extends to subjects not available in the "training set". Therefore, the datasets which are used in the "face recognition" models should include a wide range of subjects so that the algorithm becomes subjected to further distinctions within groups.

2. Realted Work

As we know in face recognition methods an individual's identity is recognized or checked by using their faces and can be used in images, videos or in real-time to recognize individuals. The most representative research works in the field of face recognition is done on.

a. Geometry-based Methods

b. Feature-based methods

c. Holistic Methods

d. Hybrid Methods

e. Deep Learning Methods.

Usually "Face recognition systems" are composed of the following four building blocks:

2.1. *Face detection*

"Face detection" is a process of detection and locates human faces in given images or videos.

2.2. *Face alignment*

In "face alignment" aim is to crop and scale face photos while utilizing a series of data points placed in the photos at defined positions. This method usually involves identifying a collection of facial images using a hallmark detector and determining the best "affine transition" that matches the reference points in the case of a specific 2D orientation.

2.3. *Face representation*

In "face representation" "pixel values" of a face picture are converted into a portable and 'discriminative vector' of the function at the face representation level which is also called template. Ideally, all dimensions of the same subject will project to identical vectors of characteristics.

2.4. *Face matching*

Two models are compared in the face matching block to generate a "similarity score" showing the possibility that they relate to the same object.

"Face detection" is now a standard for products. Several apps like "Adobe Photoshop elements" "Picasa", Microsoft windows "live photo gallery or "Apple iPhoto", use facial recognition to help users identifying their images.

3. Our methodology

Standard Convolutional Neural Networks based models for computer vision with high accuracy comes with a high cost, like they need high compute power, consume lots of memory and need more battery power. It also means that these powerful "deep learning-based models" are very difficult to deploy on memory-constrained platforms and battery-constrained embedded systems like mobile devices. This is a major problem, deep learning models have not yet been deployed on very low segment embedded systems such as smartphones (low price), smart wearables, traffic signals, and embedded systems. To keep this problem in mind, we present a new way of doing Face Recognition with CNN. With the help of this method, we can easily deploy a "trained CNN model" on any memory constrained platform such as low segments embedded system or memory-constrained mobile devices, smart wearables, etc. During experiments, we have run our proposed Network Architecture in some extremely small memory sizes (as low as 2 MB) platform. The result shows that our method runs on a very low memory segment platform as compared to a "traditional CNN based model". We achieve very good results on various "face verification datasets" and model size less than 1MB for a memory-limited embedded device.

Different tools were produced in the last few years using the new AI / ML that are very helpful in a wide range of legal issues. The CNN (Convolutional Neural Networks) is at the core of almost all the devices. The use of CNN and AI / ML is not the new thing in Face recognition. Convolutional Neural Networks (CNN) algorithms developed over the past years, using a machine and deep learning, represent a significant step forward compared to text extraction algorithms previously used. Early versions used "strong" decision trees if-then rules that are,

by definition, inflexible and lack essential contextual information to understand meaning and variations. Machine learning algorithms abandon the decision tree model of real-value weights attached to each input in favour of "soft" probabilistic decisions. Some machine learning methods used in the Face recognition for different uses are:

Machine Learning-Based:

- Linear models: Linear Regression, Logistic Regression,
- Support Vector Machines, K-Nearest Neighbours
- Ensemble models: Random Forest, Gradient Boosting Trees, Adaboost.

Deep Neural Network Based:

- Long Short-Term Memory
- Convolutional Neural Networks
- Recurrent Neural Networks

Hybrid Systems:

- Machine learning and a rule-based system, which is used to further improve the results.

The main aim of this research would be to create an AI/ML-based system which will do Face recognition for memory-constrained devices. Already various AI/ML models are present, we will compare the results of these models. After that, we will compare the results of our proposed system with the results of existing systems. Initially, we will use Support Vector Machines (SVM), Logistic Regression, CNN, and after fine-tuning of the parameter, we will compare the results. We are going to study currently used Deep Learning-based techniques and compare the results as well. We will try to find out various parameters and how to tune them so that accuracy of the system can be improved. Finally, we will be proposed a deep learning-based algorithm and compare their results. Below mention metrics will be used for evaluation.

3.1. *Pre-processing:*

a) Face detection and alignment:

"Face detection and alignment" have been done using "Multi-task cascaded CNN" which is proven to be best in the class. A margin of 26pixels is used to give some extra context around the face.

b) Fixed image standardization:

We fixed standardize the image pixels in the range of (-1, 1) instead of going with per image standardization to reduce the computation cost.

c) Image properties:

Applied random crop instead of resizing (i.e., a random portion of 224*224 is selected from 250*250 image).

A random horizontal flip is introduced to make the dataset more generalized.

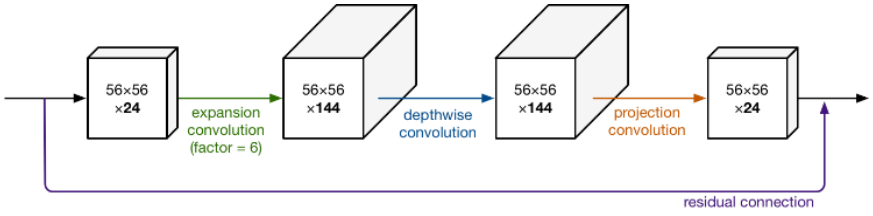
3.2. *Architecture:*

In this section, we proposed our network architecture, which is based on a "Deeper MobileNetV2", so the "residual bottlenecks" proposed in MobileNetV2 will be used as "core building blocks". In our technique we have combine "cross-entropy loss" with a "center loss" on a MobileNetV2. In Fig 4 we have shown our network architecture in details.

The network is built using inverted residual and linear bottleneck blocks inspired by Mobilenet architecture to keep the model computationally efficient.

Inverted bottleneck and linear bottleneck:

The inverted word represents the residual connection is introduced between low-dimensional spaces which are reversed in the case of a conventional residual block



The structure of the block involves 3 steps:

3.2.1Expansion of input space:

The input space will be expanded to a higher dimension by using pointwise convolutions (1*1 filters) followed by activation of Relu6

3.2.2 Feature extraction:

Feature extraction will be carried by using depth wise convolutions followed by Relu6

Note: "Depth wise convolutions a type of factorized convolution that reduces the computational cost as compared to standard convolutions" [20].

3.2.3 Compression of feature space: (projection layer)

The feature space is projected to a low-dimension space using "pointwise convolutions" to reduce the parametric complexity of the network with no activation applied

Note: It has been observed that using activation function after the projection layer resulted in a decrease inaccuracy of the model

[Input space] → [conv2d (1*1 conv)] → [conv2d (depthwise)] → [conv2d (1*1 conv)] (1)

The inverted residual layer has the idea that,

a. "Feature maps can be encoded in low-dimensional subspaces" [14].

b. "Non-linear activations result in information loss in spite of their ability to increase representational complexity" [14].

Loss function:

We have used a loss function which is a combination of "cross-entropy loss" and "center loss". With the help of this loss function discriminative power of the deeply learned features are improved.

$$\boldsymbol{loss = crossentropy\ loss + \lambda * center\ loss} \quad (2)$$

The last fully connected layer functions as a linear classifier when using the softmax loss. In Fig.1 each colour denotes "deep features" from different classes under the softmax loss supervision. We cannot use these deep features explicitly for recognition, since the deep feature involves major intra-class variations. So, we use a center loss to make discriminative enough in deep features.

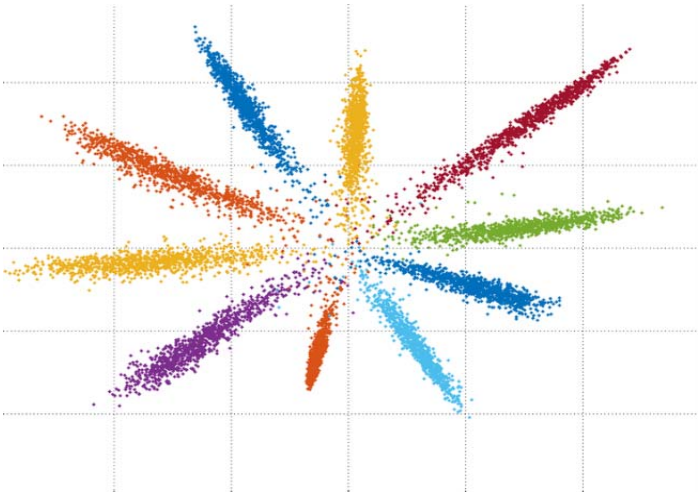


Fig.1. Deeply learned features distribution in softmax loss supervision

Center Loss:

To minimize the intraclass distances while maintaining the "features of different classes separable".

Center loss is defined as:

$$\mathcal{L}_C = \frac{1}{2}\sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{c}_{y_i}\|_2^2 \quad (3)$$

Where xi represents the ith deep feature belonging to the yith class and cyi represents the yith class center of deep features.

Softmax loss and center loss in joint supervision is:

$$\mathcal{L} = \mathcal{L}_S + \lambda \mathcal{L}_C$$
$$= -\sum_{i=1}^{m} \log \frac{e^{W_{y_i}^T \boldsymbol{x}_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T \boldsymbol{x}_i + b_j}} + \frac{\lambda}{2} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{c}_{y_i}\|_2^2 \quad (4)$$

Here λ is a scalar and used to make balance for the two-loss functions

Aftereffect of center loss:

To visualize the effect of joint supervision loss, MNIST data is trained

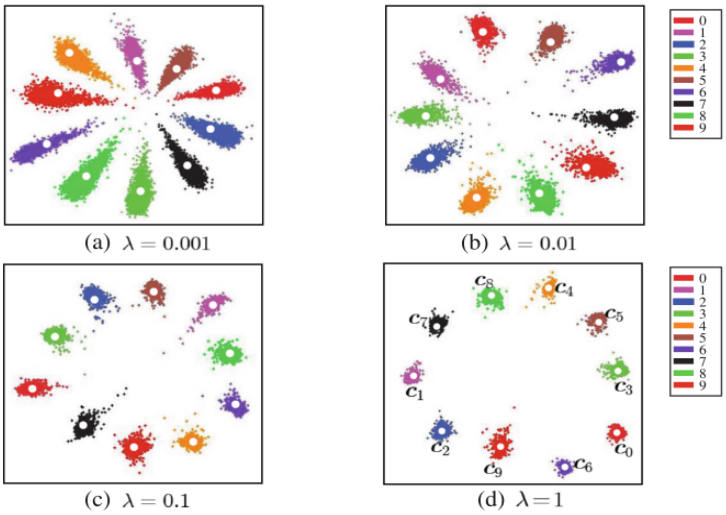


Fig.2. Deeply learned features distribution

Each colour in this image denote deep features from different classes under the joint supervision of softmax loss and center loss.

The centers will be updated by moving average method but not by backpropagation. They are updated as follows,

d_i=(〖1-α)*(c〗 _yi-x_i)

〖 c〗 _yi=c_yi-d_i

Where α is the rate at which the canters update

Block diagram:

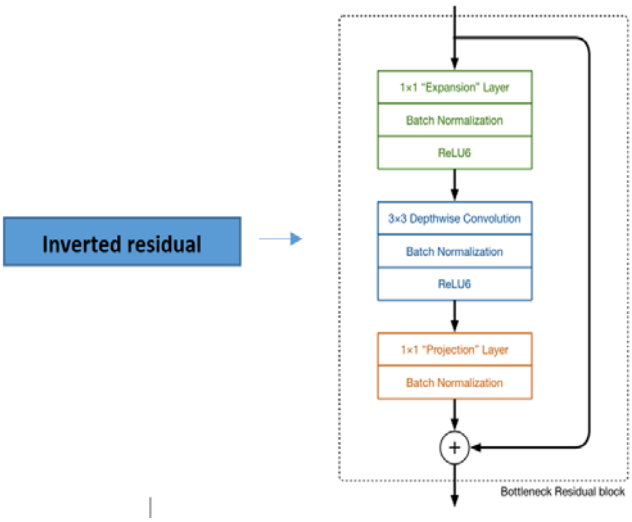


Fig 3. Block Diagram.

Face Recognition Training Tricks

We used combination of Softmax Loss and Center Loss to fine-tune our model.

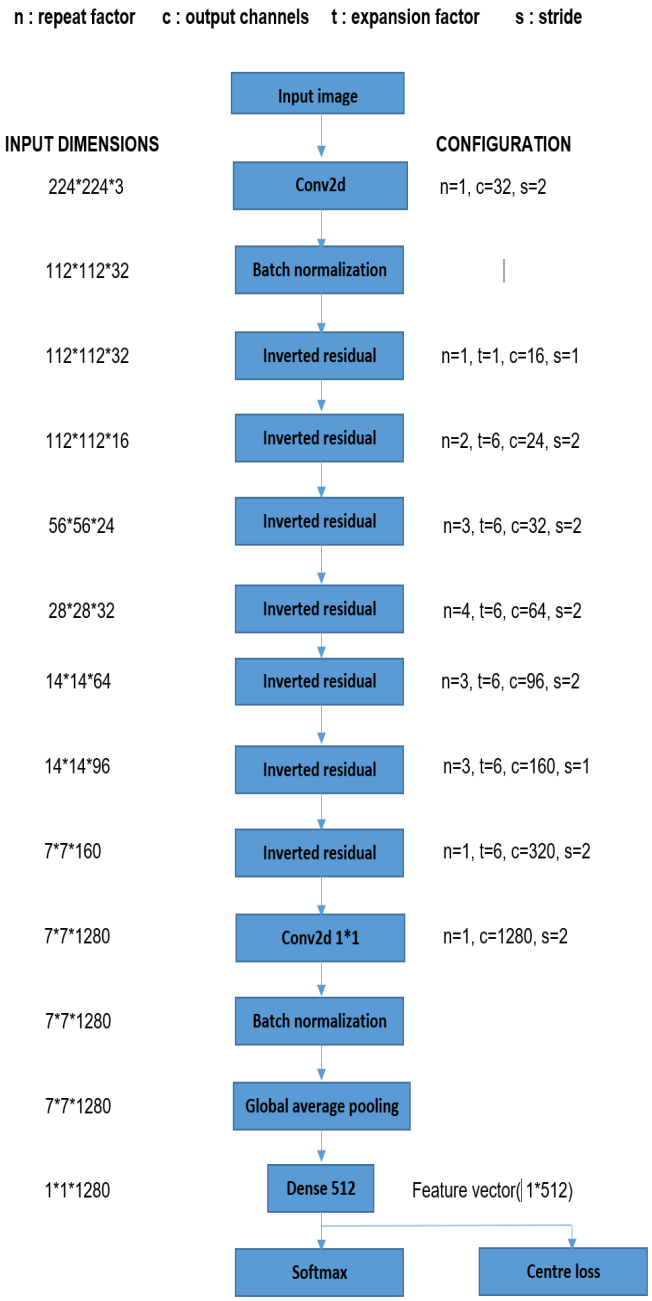


Fig 4. Proposed network architecture

In this architecture n represents the “number of repetitions”, c represents the “output channels”, t represents to the “expansion factor”, and s represents to stride.

4. Experimental evaluation

4.1. The Experiment of Center Loss:

In this section different loss function Implementation are taken and comparative study are done on the results. We have used MobileNetV2 network architecture for experiment where 128 embedding size is set and VGGFace2 training data set which contains 3.3M pictures of 9K different individuals. Lastly we used LFW-11500 to compare the performance on the labelled faces.

Table 1. Performance Verification of different loss functions with embedding size 128.

LFW-11500 Dataset, 5749 classes	**Softmax embeddings_128**	**Center_Loss embeddings_128**
Accuracy	86.32%	88.10%
Inference time	400ms on M1, 350ms on M2, 38ms on M3, 100ms on M4	400ms on M1, 350ms on M2, 38ms on M3, 100ms on M4
Model size	1MB	1MB

4.2. Evaluation Results of Network Architecture and Training Tricks

Our Network Architecture contains MobileNetV2 with center loss and training tricks, we call it as a center loss for limited -memory. Under the same training data set VGGFace2 and model constraints, the accuracy of combined (Softmax, Center Loss) reached 88.11%. In combined Softmax, Center Loss has reached incredible efficiency and performance.

5. Conclusion and future scope

We have proposed a novel Network architecture in this paper which is very efficient for face recognition. This Architecture contains MobileNetV2 with “center loss” and training tricks for “deep face recognition” for a memory constrained embedded device. In this carefully designed network architecture, we have explored some very useful training tricks which are used for “deep face recognition”. This proposed architecture solves the problem that normal softmax does not converge for memory constrained embedded devices. If we use combination of Softmax loss & Center loss, anytime it performs better if use Softmax loss or Center loss separately. Using this combination (Softmax loss, Center loss) makes it extremely efficient for “deep face recognition” for memory constrained embedded devices. This model achieves the “State of the arts results” on several face verification datasets and model size less than 1MB for a memory-constrained embedded device.

References

[1] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional lstm and other neural network architectures,” Neural Networks, vol. 18, no. 5-6, p. 602, 2010. View at Publisher View at Google Scholar View at Scopus

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proceedings of the International Conference on Neural Information Processing Systems, pp. 1097–1105, Lake Tahoe, Nevada, December 2012.

[3] C. Szegedy, W. Liu, Y. Jia et al., “Going deeper with convolutions,” in Proceedings of the Computer Vision and Pattern Recognition, pp. 1–9, Boston, MA, USA, June 2015.

[4] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and transferring mid-level image representations using convolutional neural networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1717–1724, Slovenia, Balkans, June 2014.

[5] O. Russakovsky, J. Deng, H. Su et al., “Imagenet large scale visual recognition challenge,” International Journal of Computer Vision, vol. 115, no. 3, pp. 211–252, 2014. View at Publisher View at Google Scholar View at Scopus

[6] S. Chetlur, C. Woolley, P. Vandermersch et al., “cudnn: efficient primitives for deep learning,” 2014, https://arxiv.org/pdf/1410.0759.

[7] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao, "A Discriminative Feature Learning Approach for Deep Face Recognition", 2016, http//vdwen.github.io/apers/WenECCV16.pdf

[8] C. A. Hansen, “Face Recognition”, Institute for Computer Science University of Tromso, Norway.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "Imageries classification with deep convolutional neural networks" In Bartlett et al. [48], pages 1106-1114.

[10] J Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scare image recognition", CoRR, abs/1409.1556, 2014.

[11] Tom Veniat and Liidovic Denoyer, "Learning Time/Memory-Efficient Deep Architectures with Budgeted Super Networks", CoRR, abs/l706.00046, 2017

[12] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, Kevin Murphy, "Speed/acuracy traie-offs for modern convolutional object detectors" In CVPR, 2017

[13] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, Jian Sun. "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices" CoRR, abs/ 1707.01083 (2017)

[14] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks" in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4510-4520

[15] Ping Luo, Zhenyao Zhu, Ziwei Liu, Xiaogang Wang and Xiaoou Tang, "Face Model Compression by Distilling Knowledge from Neurons" ln: AAAI (2016)

[16] Xianbiao Qi, Lei Zhang,"Face Recognition via Centralized Coordinate Learning", arXiv:1801.05678, 2018

[17] Dongyoon Han, Jiwhan Kim, Junmo Kim, "Deep Pyramidal Residual Networks". arXiv:1610.029l5, 2016

[18] Sergey Ioffe, Christian Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift" In ICML, 2015

[19] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, Wei Liu, "CosFace: Large Margin Cosine Loss for Deep Face Recognition" In CVPR, 2018.

[20] Rajeev Ranjan, Carlos D. Castillo, Rama Chellappa, "L2-constrained Softmax Loss for Discriminative Face Verification" arXiv:1703.09507, 2017.

[21] Jonathan Huang, Vivek Rathod, Derek Chow, Chen Sun, and Menglong Zhu, "Tensorflow object detection api 2017"
[22] Sheng Chen1,2, Yang Liu2, Xiang Gao2, and Zhen Han1, "MobileFaceNets: Efficient CNNs for Accurate Real-Time Face Verification on Mobile Devices"
[23] Jiankang Deng, Jia Guo, Niannan Xue, Stefanos Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition", arXiv:1801.07698v3
[24] Xianyang Li, Feng Wang, Qinghao Hu, Cong Leng, "AirFace:Lightweight and Efficient Model for Face Recognition", arXiv:1907.12256v3
[25] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface Deep hypersphere embedding for face recognition", In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 212–220, 2017
[26] Deng, J., Guo, J., Zafeiriou, S.: ArcFace: Additive Angular Margin Loss for Deep Face Recognition. arXiv preprint, arXiv: 1801.07698 (2018)
[27] Kemelmacher-Shlizerman, I., Seitz, S. M., Miller, D., Brossard, E. "The megaface benchmark: 1 million faces for recognition at scale", In: CVPR (2016)
[28] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L. "ImageNet: a large-scale hierarchical image database" In: CVPR. IEEE (2009)
[29] Russakovsky, O., Deng, J., Su, H., et al., "Imagenet large scale visual recognition challenge". in Large Scale Visual Recognition Challenge (ILSVRC) 115, 211–252 (2015)
[30] Taigman, Y., Yang, M., Ranzato, M., et al. "DeepFace: closing the gap to human-level performance in face verification", in CVPR (2014)
[31] Stanford cs class cs231n: Convolutional neural networks for visual recognition. http://cs231n.github.io/neural-networks-case-study/
[32] K. He, X. Zhang, S. Ren, and J. Sun., "Deep residual learning for image recognition", arXiv:1512.03385v1
[33] J. Liu, Y. Deng, T. Bai, Z. Wei, and C. Huang., "Targeting ultimate accuracy: Face recognition via deep embedding", in arXiv:1506.07310, 2015.
[34] W. Liu, Y. Wen, Z. Yu, and M. Yang., "Large-margin softmax loss for convolutional neural networks", in arXiv:1612.02295v4
[35] [35] T.Sabhanayagam, Dr. V. Prasanna Venkatesan and Dr. K. Senthamaraikannan, "A Comprehensive Survey on Various Biometric Systems", in International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 5 (2018)
[36] [36] By Waldemar Wójcik, Konrad Gromaszek and Muhtar Junisbekov, in "Face Recognition: Issues, Methods and Alternative Applications"
[37] W. Zhao, R. Chellappa, P. J. Phillips & A. Rosenfeld, “Face recognitions literature survey”, ACM Computing Surveys, Vol. 35, No. 4, December 2003, pp. 399–458.
[38] Daniel Saez Trigueros, Li Meng, Margaret Hartnett, "Face Recognition: From Traditional to Deep Learning Methods", in arXiv:1811.00116v1