ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2.
Available from http://www.i6doc.com/en/.

Improving Light-weight Convolutional Neural Networks for Face Recognition Targeting Resource Constrained Platforms

Iulian Felea and Radu Dogaru

University "Politehnica" of Bucharest
Dept. of Applied Electronics and Information Engineering
Splaiul Independenței 313 - Romania

Abstract. A thorough investigation of the possibility to optimize deep convolutional neural network architectures for face recognition problems is considered, from the perspective of training very compact models to be further deployed on resource-constrained systems. Latencies in recognition phase and memory usage are minimized while recognition accuracies are maintained close to state of the art performance of more complicated deep neural networks. Using two widely used datasets, namely VGG-Face and YouTube Faces, several modifications of a recent light-weight CNN model are proposed, and for a reasonable accuracy the most compact solutions were identified. Experiments on VGG-Face show that our proposed models achieves 95.5% accuracy, with 5.6 times less memory storage when compared to the reference slim model.

1 Introduction

Deep neural networks become more popular and are used to solve major visual recognition tasks, such as object recognition, image segmentation, face recognition. It has been observed that a primary trend to settle image recognition problems is by building DNNs. Especially, larger convolutional neural networks (CNNs) with a significant number of layers achieve high recognition accuracy. Deeper architectures are computationally demanding and presents to be rather challenging for integration into resource constrained environments. A high architectural complexity also demands high power consumption as well as a large memory storage.

In computer vision, face recognition problem [1], consists of identifying and verifying a subject based on an image of their face. Over the years many studies on this matter have been occurred. From algorithms [2] and [3] to neural network approaches [4], [5], [6] have been proposed to address this matter.

The article in this paper has the purpose to study the face recognition problem in the context of light-weight neural network architecture models and constrained resources systems. Starting from known neural network architectures, our aim is to design an improved and optimized neural network, in terms of specific parameter usage, resulting in the best trade-off between accuracy, complexity and inference. Chapter 2 presents an overview of research done in this area. Chapter 3 describes our suggested solutions, presents some quantitative results and finally Chapter 4 includes the concluding remarks.

2 Related work

Florian Schroff et al.[4] proposes an innovative alternative solution for the intermediate bottleneck layer of a deep convolutional neural network architecture. This shows enhanced results of facial recognition problem. For FaceNet architecture they advise an embedding optimization using a triplet loss function at training. As the name imply, it compares the Euclidian space between an anchor, a positive aligned face and a non-matching face. Based on similarities,128-sized embeddings are learned and optimized for recognition, verification and clustering of faces.

A number of computationally efficient, with diminished complexity, CNN architectures for constrained resources has been proposed, e.g., EffNet[9], ShuffleNet[10], MobileNet[7]. Howard et al. [8] proposes an impressive model, designed for mobile environments with a wide range of vision application for classification, object detection, face attributes. ShuffleNet[10] introduces pointwise group convolution and channel shuffle. Zhang et al. declares an acceleration of 13 times over an AlexNet, while preserving accuracy. These models designed for generic object detection and image classification manages to find a balance between latency and accuracy.

Recently, L-CNN (Light-weight CNN) was proposed as a trainable version of the architecture BCONV-ELM [11], a binary weight convolutional neural network designed for low complexity implementations. This model is characterized by the optimization of binary kernels while maximizing the accuracy of an extreme learning machine. Sharma and Foroosh [12], designed a lightweight deep convolutional neural network entitled Slim-Net. It is based on a computationally efficient assembly of Slim Module micro-architecture. A module is composed of depth-wise separable convolutions and point-wise convolution. Several Slim Modules construct a compact DCNN with high accuracy for face attribute prediction, suitable for embedded applications. In this paper we explore a modified version of Slim-Net.

3 Methods

3.1 Datasets

3.1.1 VGG-Face dataset

Omkar Parkhi and Andrew Zisserman have assembled VGG-Face dataset[5], a very large public face database, containing 2.6M images of 2622 subjects. Each identity has an associated text file incorporating URLs for images, respectively face detections. Because of this, a python script was used, to download and preprocess the images found at URLs in order to obtain only faces. The dataset consists of public personalities with various facial occlusions and illuminations.

3.1.2 YouTube Faces Database

YouTube Faces Database[6] (YTFD) contains Youtube videos for face recognition. It incorporates 3425 labeled videos of 1595 different people. Each subject has an average 2.15 videos with an average length of 181.3 frames per video. They may have between 48 and 6070 frames. Each video was broken to frames, for each frame, the same [3] face detection algorithm was applied. An extended bounding box of the face was cropped from each frame and aligned[1].

3.2 Architectures

The light weight Slim-CNN (L-Slim CNN) architectures presented here, are modified versions of the original proposed in [12] starting from the suggested implementation in[2]. As stated by Sharma et al., the architecture consists of several Slim Module stacked on a (7,7) initial convolutional layer, followed by a global average pooling layer and a fully connected in the end. Our proposes assumes unbinding the stacked modules while keeping the initial conv layer, the global average pooling and the fully connected layers. Between the last two of them, a dropout (0.5) was added to reduce the overfitting. The combination of 1x1 pointwise convolution and 3x3 depthwise convolution layers from SSE block shows an impressive increase of speed when compared to the standard convolutional layers. The result is thus suitable for constrained resources platforms. We tried to keep as many micro-modules as needed, which consists of two stacked SSE and a depthwise separable conv layer. Experiments for the face detection problem illustrates a great improvement for only two Slim micro-modules, for the above configuration. The L-Slim architecture opens the possibility to explore even lower input dimensions for more restrictive platforms.

While L-Slim model preserves the original squeeze layers, the VL (very light) Slim – CNN (VL-SlimNet) proposes to downsample their values, leading to great parameter reduction, as it can be seen in Tables 1,2,3, obtaining comparable results for lower input image sizes.

3.3 Experiments and Results

Table 1, shows a short comparison between FaceNet[4], a deep neural network (DNN) and different light-weight neural networks (LNN) well known in the field of object and face detection and our proposed architecture from the perspective of trainable parameters and inference time.

Herein we used a restricted version of the VGG Dataset[5] and YTFD[6] dataset with less subjects. In many resource-constrained applications the task is to recognize among a limited set of users, therefore it makes sense to use a restricted classes dataset. Through all the studies from this paper, are reported on an arbitrarily chosen subset of randomly 5 individual classes, from each dataset, for which there were selected the same number of images. For the VGG Dataset, all elected classes had a little over 200 images per class, while for YTFD dataset, they have a little over 600 images per class.

According to the specifications given by Sharma and Foroosh in the original paper[12], SlimNet architecture uses 96 filters with a (7,7) kernel for the first convolutional layer. This configuration exceeds CUDA memory in medium size PC systems (such as the one used by us). For this reason, some of the lines in Table 3 provide limited information (particularly for large image sizes – 256x256 pixels). All neural network models stated above were trained in PyTorch on a computer equipped

[1] https://www.cs.tau.ac.il/~wolf/ytfaces/index.html

[2] https://github.com/gtamba/pytorch-slim-cnn

with an Intel i7-7700HQ CPU of 2.80GHz and an NVIDIA GTX 1050Ti GPU with 4GB memory. For image sizes lower than 128x128, the SlimNet architecture does not perform well. Due to the fact that the neural network model is so deep, and the input size of images is low, it is likely that performance problems are raised by the convolutional layers near the input. Table 2 shows latency and parameters variation for SlimNet, L-SlimNet and VL-SlimNet based on input dimension for YTFD dataset, while Table 3 displays the correspondence between input resolution and filters that influence the parameters, leading to greater or lesser inference time.

NN	Architectures	Input	Params	Latency [ms]
DNN	FaceNet[4]	160	22.8 M	0.198
LNN	MobileNet[7]	128	3.2M	2.45
	ShuffleNet[10]	128	942k	4.93
	EffNet[9]	128	440k	3.75
	SlimNet [12]	128	550k	2.54
	L-CNN[11]	128	661k	1.75
	L-SLIMNet	**128**	**95k**	**0.570**
	VL-SlimNet	**128**	**39.4k**	**0.406**

Table 1: Parameters and latency comparison of DNN (Deep Neural Network) and different LNN (Light-weight Neural Network) architectures

NN	Image Dim	Params	Acc %	Latency [ms]
SlimNet	256	162k	99.10	1.60
	128	157k	99.32	0.491
L-SlimNet	256	109k	99.32	2.45
	128	109k	99.37	0.579
	64	105.6k	99.31	0.264
	32	103.3k	99.38	0.241
VL-SlimNet	256	39.4k	99.42	1.5
	128	39.4k	99.41	0.406
	64	35.9k	99.30	0.227
	32	33.6k	99.48	0.218

Table 2: YouTube Faces Database accuracy and latency results of different input sizes for SlimNet, L-SlimNet and VL-SlimNet with a batch size of 64 and 48 filters

Experiments illustrated on Table 4, on the VGG-Faces show that L-SlimNet achieves an accuracy of 95.50% with at least 4 times fewer parameters than comparably performing methods which reduces the memory storage requirement of the model by at least 82.16%, up to 93.52% for VL-SlimNet.

4 Conclusion

Herein we investigated a modified version of the recently proposed light-weight-CNN aiming to reduce both the occupied memory and inference time, thus making the resulting models attractive for embedding into resource-constrained environments. Our proposed architectures find the best trade-off between image resolution and convolutional filters, leading to a high accuracy for the face recognition problem, with an occupied memory 4 times less that previously reported light-weight-CNN implementation and with a good latency.

NN	Image Dim	Batch size	Filters	Params	Accuracy %	Latency [ms]
SlimNet	256	64	32	543.4k	86.52	2.94
		32	96	564.5k	90.21	3.53
			48	550.4k	89.92	3.35
			32	545.7k	90.07	2.51
	128	64	96	553k	85.94	0.78
		32	96	553k	86.37	1.15
L-SlimNet	256	64	48	98.4k	95.43	2.27
			32	93.7k	93.84	2.10
		32	**96**	**112.5k**	**95.50**	**2.73**
			48	98.4k	94.42	2.22
			32	93.7k	93.91	2.12
	128	64	**96**	**112.5k**	**94.63**	**0.66**
			48	98.4k	94.13	0.56
			32	93.7k	92.10	0.52
		32	96	112.5k	94.56	0.71
			48	98.4k	94.13	0.60
			32	93.7k	92.60	0.56
VL-SlimNet	256	32	32	30.4k	93.33	1.36
	128	64	96	44.6k	92.31	0.5
		32	96	44.6k	92.82	0.55
			48	**33.9k**	**93.33**	**0.52**
			32	30.4k	92.82	0.49

Table 3: Reported accuracy and latency on VGG-Faces database for different input image sizes batches and convolutional filters applied on SlimNet, L-SlimNet and VL-SlimNet.

DB	NN	Input size	Train Dim[MB]	Model Dim[MB]
VGG-Faces	SlimNet	256	121.59	6.39
		128	31.99	6.35
	L-SlimNet	**256**	**107.66**	**1.14**
		128	25.04	1.14
		64	5.53	1.14
		32	1.65	1.09
	VL-SlimNet	**256**	**60.34**	**0.410 (420KB)**
		128	14.02	0.410 (420KB)
		64	3.06	0.407 (417KB)
		32	0.94	0.382 (392KB)

Table 4: SlimNet, L-SlimNet and VL-SlimNet architectures estimated training dimension and model final memory storage for VGG-Faces dataset

Acknowlegment

This work has been funded by the Operational Programme Human Capital of the Ministery of European Funds, through the Financial Agreement 51675/09.07.2019, SMIS code 125125.

References

[1] Jain, Anil K., and Stan Z. Li. "*Handbook of face recognition"*. New York: springer, 2011.

[2] Samaria, F.S. and Harter, A.C., "Parameterisation of a stochastic model for human face identification". In *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*, December,1994 (pp. 138-142).

[3] Viola, P. and Jones, M., "Rapid object detection using a boosted cascade of simple features". In *Computer Vision and Pattern Recognition* (CVPR) (1), 1(511-518) , 2001, p.3.

[4] Schroff, F., Kalenichenko, D. and Philbin, J. „Facenet: A unified embedding for face recognition and clustering". In Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815-823.

[5] Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman. "Deep face recognition." In *British Machine Vision Conference* (*BMVC*), vol. 1, no. 3, 2015, p. 6.

[6] Wolf, L., Hassner, T. and Maoz, I."Face recognition in unconstrained videos with matched background similarity", IEEE, 2011, pp.529-534

[7] Howard, Andrew G., et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." *arXiv preprint arXiv:1704.04861*, 2017.

[8] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. and Chen, L.C. "Mobilenetv2: Inverted residuals and linear bottlenecks". In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510-4520.

[9] Freeman, I., Roese-Koerner, L. and Kummert, A. "Effnet: An efficient structure for convolutional neural networks". In *2018 25th IEEE International Conference on Image Processing (ICIP)*, October, 2018 pp. 6-10. IEEE.

[10] Zhang, X., Zhou, X., Lin, M. and Sun, J. Shufflenet: "An extremely efficient convolutional neural network for mobile devices", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6848-6856.

[11] R. Dogaru and I. Dogaru, "BCONV-ELM: Binary Weights Convolutional Neural Network Simulator based on Keras/Tensorflow, for Low Complexity Implementations", ISEEE 2019, October 2019.

[12] Sharma A, Foroosh H. Slim-CNN: A Light-Weight CNN for Face Attribute Prediction. arXiv preprint arXiv:1907.02157. July 2019