Sub-byte quantization of Mobile Face Recognition Convolutional Neural Networks

Sebastian Bunda
Faculty of EEMCS, DMB Group
Enschede, Netherlands
s.t.bunda@utwente.nl

Luuk Spreeuwers
Faculty of EEMCS, DMB Group
Enschede, Netherlands
l.j.spreeuwers@utwente.nl

Chris Zeinstra
Faculty of EEMCS, DMB Group
Enschede, Netherlands
c.g.zeinstra@utwente.nl

2022 International Conference of the Biometrics Special Interest Group (BIOSIG) | 978-1-6654-7666-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/BIOSIG55365.2022.9897025

***Abstract*—Converting convolutional neural networks such as MobileNets to a full integer representation is already quite a popular method to reduce the size and computational footprint of classification networks but its effect on face recognition networks is relatively unexplored. This work presents a method to reduce the size of MobileFaceNet using sub-byte quantization of the weights and activations. It was found that 8-bit and 4-bit versions of MobileFaceNet can be obtained with 98.68% and 98.63% accuracy on the LFW dataset which reduces the footprint to 25% and 12.5% of the original weights respectively. Using mixed-precision, an accuracy of 98.17% can be achieved whilst requiring only 10% of the original weight footprint. It is expected that with a larger training dataset, higher accuracies can be achieved.**

***Index Terms*—Resource Limited Face Recognition, Deep Neural Networks, QKeras, Sub-byte Quantization**

I. Introduction

The field of network optimization is dedicated to improving the trade-off between limited resource factors (i.e. size and latency) versus accuracy. A natural split in network optimization can be made into two scopes: architecture optimization and network compression. While the first scope is dedicated to improving the efficiency of network architectures, the second is focused on reducing the computation complexity and footprint of existing network architectures through the use of e.g. quantization (reducing the number of bits to represent the same value[1]) and pruning (minimizing the number of unnecessary calculations). Besides the fact that these methods reduce the memory footprint, integer-only inference has shown to also improve the throughput on hardware optimized (for mobile devices) fixed point operations[2].

Popular quantization methods for neural networks can be found for both Tensorflow Lite and Brevitas for PyTorch. Unfortunately, the official implementations of Tensorflow's quantization methods only support 8-bit quantization up until now. This can mainly be attributed to the fact that most of the current microcontroller instruction sets only support the handling of bytes (8 bits). The expectation is, however, that future instruction sets will get support for mixed-precision words as research shows some great progress in the last few years on both (RISC-V) hardware and software.

Another quantization tooling that works with Tensorflow and does support quantization that is not limited to only 8-bits is called QKeras[3]. Although QKeras is still in development, the tooling does show promising results[4]. By utilizing the sub-byte quantization support, the effect of sub-byte quantization on the accuracy of a (face recognition) neural network can be investigated.

Since face recognition neural networks use a similar basis as image classification neural networks, the same principles can be applied here as well. Popular mobile face recognition networks include ShuffleFaceNet[5] and MobileFaceNet[6], which have the same basis as the popular networks ShuffleNetV2[7] and MobileNetV2[8]. Unfortunately, the quantization of an efficient neural network is a relatively unexplored area in the field of face recognition. Roughly at the same time of the publishing of this paper QuantFace[9] was introduced and seems to show promising results. Quantization significantly reduces the total solution space by reducing the number of representation spaces per feature. It is interesting to explore this by applying network optimization on architectures such as MobileFaceNet and identify possible problems such as the level of discriminability of the feature vector due to the limited solution space.

II. Experimental Setup

In this section, several experiments will be designed to explore the effect of sub-byte quantization on QMobileFaceNet[1]. The first set of experiments will focus on the effect of the level of quantization on the accuracy of evaluating both uniform and individual layer quantization. The experiments will explore various quantization methods using 8-bit, 4-bit and 2-bit quantization. The second set of experiments will delve into the question of to what extent the discriminability of the face recognition system is affected by the (sub-byte) quantization. By using the 8, 4 and 2-bit uniformly quantized networks. 1-bit quantization will not be considered in this work as this requires adaptations to the network and can be considered a work in itself.

A. Datasets and evaluation metrics

The CASIA-Webface[10] dataset is used to train both the normal and the quantized version of MobileFaceNet networks. The CASIA-Webface dataset consists of 435.779 images of 10.575 different identities build from images found on the internet using a semi-automatic method. The performance

[1]A quantized version of MobileFaceNet

TABLE I
THE NETWORK ARCHITECTURE OF MOBILEFACENET. t IS THE EXPANSION FACTOR, c IS THE CHANNEL DEPTH, n IS THE NUMBER OF REPETITIONS AND s IS THE STRIDE (OF THE FIRST LAYER IF BOTTLENECK)

Architecture MobileFaceNet[6]

Layer	Input	Operator	t	c	n	s
1	$112^2 \times 3$	conv3×3		64	1	2
2	$56^2 \times 64$	depthwise conv3×3		64	1	1
3	$56^2 \times 64$	bottleneck	2	64	5	2
4	$28^2 \times 64$	bottleneck	4	128	1	2
5	$14^2 \times 128$	bottleneck	2	128	6	1
6	$14^2 \times 128$	bottleneck	4	128	1	2
7	$7^2 \times 128$	bottleneck	2	128	2	1
8	$7^2 \times 128$	conv1×1		512	1	1
9	$7^2 \times 512$	linear GDC7×7		512	1	1
10	$1^2 \times 512$	linear conv1×1		128	1	1

is evaluated using the Labeled Faces in the Wild[11] and the AgeDB-30 dataset[12] both containing 6000 frontal face pairs, of which half is a face pair of the same identity. The AgeDB-30 evaluation dataset contains faces with a 30 year age gap and requires the network to compare timeless features to decide the two identities are mated or non-mated. The network is evaluated (as [13]) by calculating the angle between both feature vectors that are calculated for each face. Finally using 10-K folding, the best threshold is found, and for this threshold, the accuracy is determined.

B. QMobileFaceNet

The network architecture of MobileFaceNet is visualized in Table I, showing ten layer blocks that can be quantized individually. The pre-quantized network was trained by first training with a softmax classification header and then continued the training with an ArcFace classification header. For both the softmax and ArcFace header, the models were trained using the Stochastic Gradient Descent (SGD) optimizer function with a momentum of 0.9 and a batch size of 512. The learning rate started with a value of 0.1 for the softmax and 0.01 for the ArcFace header. After 36K, 52K and 58K batches the learning rate was divided by 10. Each training session ended after 60K batches. For the quantization aware training it was found that the Adam optimizer with a learning rate of 0.001 worked better than the SGD optimizer.

QKeras does not support a quantizable version of PReLU (the activation function used by MobileFaceNet), our version of QMobileFaceNet will use the ReLU activation function instead (with 3 bits for the integer representation). Using a preliminary study it was also found that folding the batch normalization of the convolutional layers and using a full fractional representation improved the stability of the quantization. The quantization scheme was implemented using automatic scaling of the channels to minimize the quantization losses.

In the original network architectures of MobileNets the depthwise separable convolution is implemented by applying both batch normalization and the ReLU activation function after the depthwise convolution and the pointwise convolution. Research shows that removing the batch normalization and the ReLU after the depthwise convolution, would increase the accuracy from 1.8% to 68.03% [14]. Additionally it was found that removing the batch normalization of the depthwise convolution in the second layer and the global depthwise convolution also improved the accuracy of the network. Both these additions were made to the version of QMobileFaceNet presented in this work.

C. Quantization Experiments

The first experiment will quantize each layer block of MobileFaceNet individually. The goal is to find the location in the network that is the most affected by the quantization. The expectation is that the first layers will affect the accuracy the most due to the propagation effect. It is also expected that the last layers will not affect the accuracy in a significant way for the same reason.

QKeras is designed such that the weights and activations can take specific quantization schemes into account while training. The first step is to directly quantize the weights and activation functions. This is known as post training quantization (PTQ). Quantization Aware Training (QAT) can then be implemented to mitigate some of the quantization errors made by PTQ. The second experiment will apply QAT to the 8, 4 and 2-bit uniform quantized networks and compare them to the original PTQ results. To show the consistency, the mean and standard deviation will be calculated for three separate QAT runs for each bit length.

The third experiment will involve different mixed-precision networks for each layer configuration. The goal is to obtain a graph to show a trade-off between footprint and accuracy. The estimation of the footprint will be made using the number of bits required for the representation of the weights and thus does not include any overhead required for the implementation. For this experiment, the first two layers will remain 8-bits and the rest of the layers are either 4-bit or 2-bit.

The feature vector output of the face recognition system is used to compare the features between two faces to determine whether the features are similar enough to belong to the same identity. In order to investigate the effect of (sub-byte) quantization on this decision making, some small experiments will be done. First, the feature vectors of 10 identities from the MS1M-V2 dataset[13] are calculated. For these identities, a t-distributed stochastic neighbour embedding (t-SNE) analysis [15] will be performed to show the clustering ability between the feature vectors. The resulting figures of the t-SNE will thus not give any numerical results. The second experiment will generate the ROC curves for the LFW and AgeDB-30 datasets across the various uniformly quantized networks.

III. RESULTS & DISCUSSION

The results of the individual layer quantization are shown in Figure 1. This plot shows that the most accuracy degradation can be attributed to the quantization of the layer blocks 3–5. The graph shows that the first three sets of the inverted residual blocks have the largest impact on the accuracy of the AgeDB-30 dataset for the 2-bit quantization. Notice that after some QAT most quantization losses can be somewhat mitigated. The expectation was that the quantization loss had

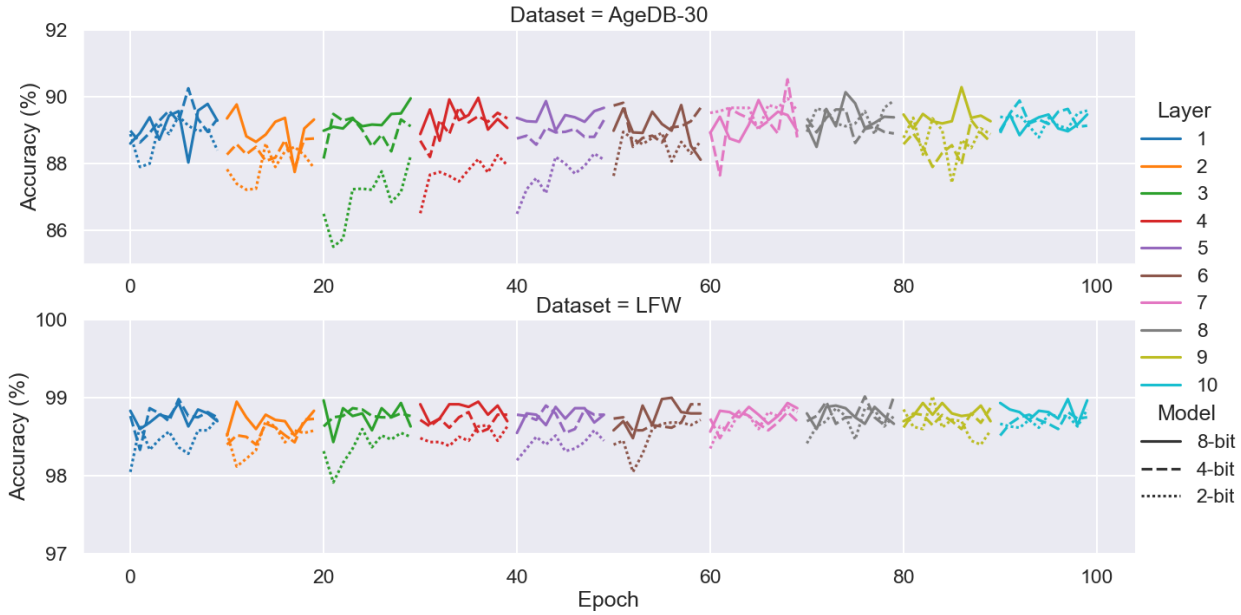


Fig. 1: The accuracy of 8, 4 and 2-bit QMobileFaceNet with folding by quantizing each layer individually for both the LFW and the AgeDB-30 dataset. In bottom plot also shows the number of parameters per layer on a logarithmic scale

the most impact at the beginning of the network. The results seem to be in line with the expectation, although it has to be noted that the inverted residual blocks also account for most of the parameters in the network which could also have led to the quantization loss. Another interesting observation is that the 4-bit quantization has a very similar performance as the 8-bit equivalent, whilst in theory only requires half of the number of bits to describe the weights. This, and the fact that the last four 2-bit quantized layers behave similar to the other bit lengths, is an indication that mixed-precision networks could reduce the size of the network significantly without having to sacrifice the accuracy too much.

The QAT results, found in Table II, are based on the mean and standard deviation of three different quantization training sessions of 20 epochs. This table shows that QAT can significantly improve the quality of the quantized network (for the 8-bit and 4-bit networks), only a few percent points (%p) lower than the original 32-bit floating-point network. Especially the results of the 4-bit network are noteworthy, as it is only 12.5% of the original 32-bit weight footprint with only an accuracy loss of 0.22 %p using the LFW dataset and 2.18 %p using the AgeDB-30 dataset. The 2-bit network also sees an improvement after QAT, but clearly has to sacrifice accuracy for the smaller weight representation. The results shown in the work of QuantFace[9], show a better implementation of MobileFaceNet using the Pytorch framework and a different training dataset. The authors mention that their method did not converge for a 4-bit implementation which is achieved with this work. It is expected that with a more truthful implementation of MobileFaceNet similar results to QuantFace are achievable.

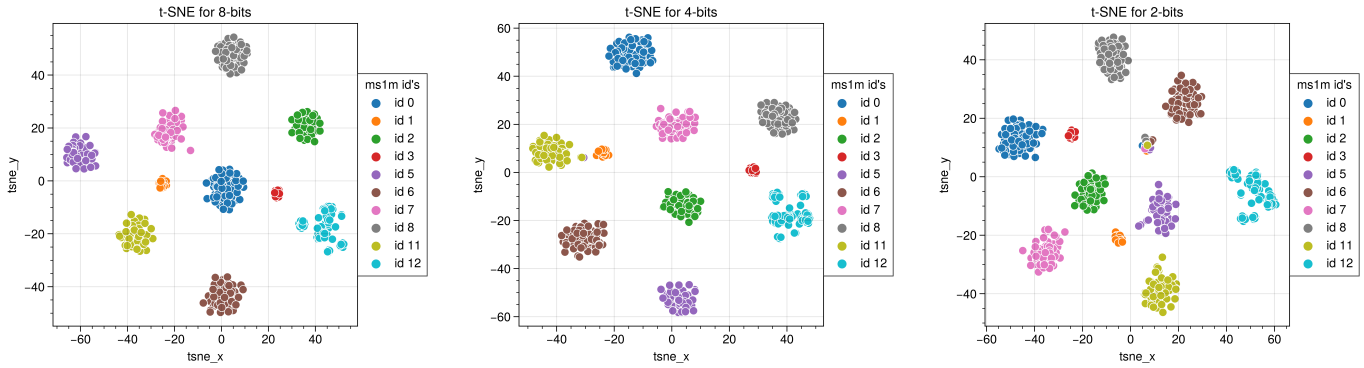
[2]The 32-bit network is without using the quantization adaptations.

Fig. 2: The total weight bits versus accuracy trade-off for the LFW and AgeDB-30 datasets.

The results for the mixed-precision quantization can be found in Figure 2. This figure visualizes the trade-off between the number of bits required for the weight representation versus the accuracy of the LFW and AgeDB-30 datasets. The graph shows a visible degradation of accuracy when the weights can be described in less than 3 million bits. The network just before this drop-off with 3.1 million weight bits achieved 98.17% and 87.37% on the LWF and AgeDB-30 dataset respectively. This network had a configuration where the first two layers blocks were quantized using 8-bit quanti-

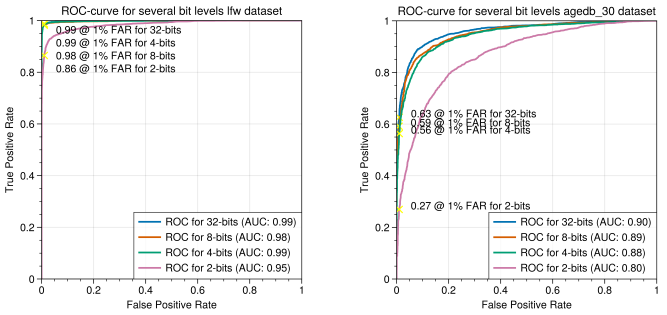
TABLE II
THE MEAN ACCURACY AND STANDARD DEVIATION OF QUANTIZATION AWARE TRAINING OF MOBILEFACENETS COMPARED TO THE POST TRAINING QUANTIZATION RESULTS.

QMobileFaceNet				
LFW	**32-bit**[2]	**8-bit**	**4-bit**	**2-bit**
Post Training Quantization	98.85	94.65	63.15	51.55
Quantization Aware Training	98.85	98.68 ±0.15	98.63 ±0.18	93.45 ±0.66
AgeDB-30	**32-bit**	**8-bit**	**4-bit**	**2-bit**
Post Training Quantization	90.38	73.42	54.98	50.55
Quantization Aware Training	90.38	88.79 ±0.10	88.20 ±0.50	75.62 ±2.83



a t-SNE 8-bit network b t-SNE 4-bit network c t-SNE 2-bit network

Fig. 3: The visualisation of the ability to cluster the identities using the t-SNE algorithm.



a ROC LFW Dataset b ROC AgeDB-30 Dataset

Fig. 4: The ROC curves for several bit lengths. For each curve the location of the False Acceptance Rate (same as FPR) at 1% is highlighted. In the legend also each area under the curve can be found.

zation, layer blocks 3–5 using 4-bit quantization and layer blocks 6–10 using 2-bit quantization. This is 10% of the footprint of the original 32-bit equivalent with an accuracy loss of only 0.68 %p on LFW and 3.01 %p on the AgeDB-30 dataset. These impressive results could bring a toll to the calculation complexity as the system has to support mixed-precision architectures. For future research, it is interesting to investigate the effect of implementing similar mixed-precision networks on the latency in mobile and edge devices.

To analyse the effect of the quantization of the feature vector on the performance of the face recognition system two small experiments have been done. The first experiment was based on the discriminability between different faces. For this experiment, the angle is calculated between two feature vectors from faces found in the MS1M-V2 dataset. The t-SNE clustering, found in Figure 3, shows that the 8-bit and 4-bit quantized networks generate vectors that can easily be clustered and that there is good segregation between the mated and non-mated vectors. In the graph for the 2-bit network, however, a cluster of non-mated vectors can be observed as shown by many dots of different colours around (18, 0). This suggests that the 2-bit network does encounter problems with the discriminability between the vectors and is not a recommended network for a face recognition system.

To show the effect of QAT on the trade-off between the True Positive Rate and the False Positive Rate, the ROC curves can be found in Figure 4. Similar to the results in the first experiment, the curves between the original 32-bit, 8-bit and 4-bit networks are very similar. The 2-bit network performs visibly worse than the other networks. The ROC curve for the LFW dataset still has a comparable AUC to the other networks, which would suggest still a moderate performance. For the AgeDB-30 dataset, however, the curve only converges to almost 100% TPR with an FPR higher than 90%. This is also in line with findings of the earlier experiments on the 2-bit network that it clearly under-performs and should not be considered for a face recognition system.

IV. Conclusion & Future Work

This work presents several methods to quantize a face recognition system and an analysis on the effect of quantization on the discriminability of the face feature vectors. It is shown that it is possible to quantize a face recognition system using 8 and 4 bits for the weight representation and activation function with a 0.17 and 0.22 %p loss in accuracy using the LFW evaluation dataset. The research also shows that using mixed precision the size of the footprint can be reduced further. By analysing the discriminability of the feature vector, the conclusion can be drawn that a 2-bit uniformly quantized network has an inadequate performance and is not recommended for a face recognition system. The results of this work suggest that sub-byte and mixed-precision is a promising method to reduce the size of a face recognition system without losing notable accuracy and can thus be used as a basis for future work that has hardware requirements for actual implementation. Additionally, it can be expected that improving the architecture implementation and using a larger dataset to train the network the accuracy can be improved even further.

REFERENCES

[1] M. Nagel, M. Fournarakis, R. A. Amjad, Y. Bondarenko, M. van Baalen, and T. Blankevoort, "A white paper on neural network quantization," *CoRR*, vol. abs/2106.08295, 2021. arXiv: 2106.08295. [Online]. Available: https://arxiv.org/abs/2106.08295.

[2] B. Jacob, S. Kligys, B. Chen, *et al.*, "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2704–2713. DOI: 10.1109/CVPR.2018.00286.

[3] C. Coelho, *Google/qkeras*, 2019. [Online]. Available: https://github.com/google/qkeras (visited on 03/02/2022).

[4] C. N. Coelho, A. Kuusela, S. Li, *et al.*, "Automatic heterogeneous quantization of deep neural networks for low-latency inference on the edge for particle detectors," *Nature Machine Intelligence*, vol. 3, no. 8, pp. 675–686, Jun. 2021. DOI: 10.1038/s42256-021-00356-5. [Online]. Available: https://doi.org/10.1038%5C%2Fs42256-021-00356-5.

[5] Y. Martindez-Díaz, L. S. Luevano, H. Mendez-Vazquez, M. Nicolas-Diaz, L. Chang, and M. Gonzalez-Mendoza, "Shufflefacenet: A lightweight face architecture for efficient and highly-accurate face recognition," in *2019 IEEE/CVF ICCVW*, 2019, pp. 2721–2728. DOI: 10.1109/ICCVW.2019.00333.

[6] S. Chen, Y. Liu, X. Gao, and Z. Han, "Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices," in *Biometric Recognition*, 2018, pp. 428–438. DOI: 10.1007/978-3-319-97909-0_46.

[7] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Computer Vision – ECCV 2018*, Cham: Springer International Publishing, 2018, pp. 122–138, ISBN: 978-3-030-01264-9.

[8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474.

[9] F. Boutros, N. Damer, and A. Kuijper, *Quantface: Towards lightweight face recognition by synthetic data low-bit quantization*, 2022. DOI: 10.48550/ARXIV.2206.10526. [Online]. Available: https://arxiv.org/abs/2206.10526.

[10] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *CoRR*, vol. abs/1411.7923, 2014. arXiv: 1411.7923. [Online]. Available: http://arxiv.org/abs/1411.7923.

[11] G. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," *Tech. rep.*, Oct. 2008.

[12] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, "Agedb: The first manually collected, in-the-wild age database," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 1997–2005. DOI: 10.1109/CVPRW.2017.250.

[13] J. Deng, J. Guo, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," *CoRR*, vol. abs/1801.07698, 2018. arXiv: 1801.07698. [Online]. Available: http://arxiv.org/abs/1801.07698.

[14] T. Sheng, C. Feng, S. Zhuo, X. Zhang, L. Shen, and M. Aleksic, "A quantization-friendly separable convolution for mobilenets," *CoRR*, vol. abs/1803.08607, 2018. arXiv: 1803.08607. [Online]. Available: http://arxiv.org/abs/1803.08607.

[15] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html.