

QuantFace: Towards Lightweight Face Recognition by Synthetic Data Low-bit Quantization

Fadi Boutros^{1,2}, Naser Damer^{1,2}, Arjan Kuijper^{1,2}

¹Fraunhofer Institute for Computer Graphics Research IGD, Darmstadt, Germany

²Department of Computer Science, TU Darmstadt, Darmstadt, Germany

Email: fadi.boutros@igd.fraunhofer.de

Abstract—Deep learning-based face recognition models follow the common trend in deep neural networks by utilizing full-precision floating-point networks with high computational costs. Deploying such networks in use-cases constrained by computational requirements is often infeasible due to the large memory required by the full-precision model. Previous compact face recognition approaches proposed to design special compact architectures and train them from scratch using real training data, which may not be available in a real-world scenario due to privacy concerns. We present in this work the QuantFace solution based on low-bit precision format model quantization. QuantFace reduces the required computational cost of the existing face recognition models without the need for designing a particular architecture or accessing real training data. QuantFace introduces privacy-friendly synthetic face data to the quantization process to mitigate potential privacy concerns and issues related to the accessibility to real training data. Through extensive evaluation experiments on seven benchmarks and four network architectures, we demonstrate that QuantFace can successfully reduce the model size up to 5x while maintaining, to a large degree, the verification performance of the full-precision model without accessing real training datasets. All training codes are publicly available ¹.

I. INTRODUCTION

Recent high performing deep neural networks (DNN) rely on over-parameterized networks with high computational cost [1], [2]. State-of-the-art (SOTA) face recognition (FR) models followed this common trend by relying on over-parameterized DNN [3]–[5]. However, deploying such extremely large models with hundreds of megabytes (MB) of memory requirements on embedded devices and other use-cases constrained by the computational capabilities and high throughput requirements is still challenging [6], [7].

Enabling FR on domains limited with computational capabilities requires designing a special architecture or compressing the current solutions to meet the computational requirements of the deployment environments. Several efficient FR models have been proposed in the literature [8]–[11]. The core idea of many of these works depended on utilizing efficient architecture designed for common computer vision tasks such as MixNets, [12], MobileNetV2 [13], ShuffleNet [14], VarGNet [15] for FR [8], [9], [11], [16]. Very recently, few of works [10], [17] proposed architectures based on face-specific neural architecture search (NAS) for efficient FR.

However, none of all previous efficient FR works [8]–[11], [16], [17] explored the potential of using model quantization to reduce the computational cost of existing widely used FR architectures, e.g. ResNet [1], [3] or SE-ResNet [2], [5].

Model quantization approaches compress the DNN by reducing the number of bits required to represent each weight, e.g., using a lower precision format than full-precision (FP) floating-point, such as 4-bit [18], [19] or 8-bit [20], [21] signed integer. Such methods have shown great success in reducing the computation cost of DNN and are supported by most deep learning accelerators [22], [23]. Model quantization enables performance gains in different areas. First, it reduces the model size, which can be directly measured using the number of bits required to represent each parameter. For example, applying model quantization using 8bit bandwidth on ResNet100 [1], [3] (65.2M parameters) reduces the model size from 261.2MB to 65.3 MB. Second, many deep learning accelerators such as Pytorch [22] and TensorFlow [23] can run a quantized model faster than a FP one. For example, Pytorch [22] can run a quantized model 2-4x faster than the FP model and reduce the required memory bandwidth by 2-4x [20], [22]. However, the exact inference speed and memory bandwidth are highly dependent on the underlying hardware and deep learning accelerator [22]. Once the model is quantized, the model weights and quantization parameters need to be tuned and calibrated. These processes commonly require access to the training data, either entirely or partially [24].

This need for data if quantization is used on FR networks follows that of deep FR models reliance on large-scale training datasets [3], [8], [16] such as MS1M [25], VGGFace2 [5] and CASIA-WebFace [26]. Existing efficient FR solutions are not different as they also require face image databases, whether for conventional training and/or knowledge distillation (KD) from teacher networks [8]–[11], [16], [17]. Most of the recent face datasets used in the literature have been collected from the web [27]. According to [27], 45 of face datasets have been created after 2014, and around 78% of these datasets are derived from the web. However, it is not a trivial task, and it may not be feasible to further collect face datasets for biometric processing from the web due to legal privacy issues in some countries [28]–[31]. Privacy regulations, just as the GDPR [30], gives individuals the right to withdraw their consent to store or use their private data [30], [32], a process that can practically be

¹<https://github.com/fdbtrs/QuantFace>

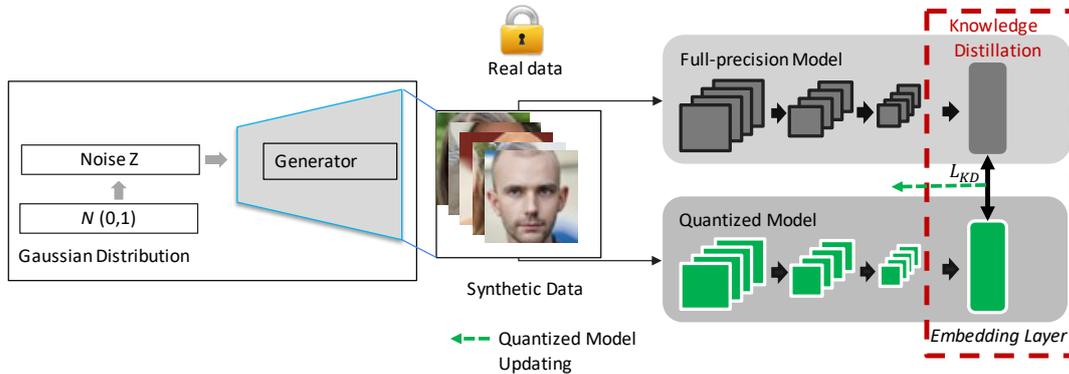


Fig. 1: An overview of the proposed QuantFace framework. Given a Gaussian noise Z from a normal distribution, the pretrained generator produces a fake data sample and feeds it into the FP and the b-bit quantized model. The KD loss is then computed between the normalized feature embeddings of the FP model and the quantized model.

very challenging when a database is widely distributed, which puts the privacy rights of individuals in jeopardy. Following such concerns, datasets such as VGGFace2 [5] and CASIA-WebFace [26] are not anymore publicly accessible in many countries. Companies like Facebook announced that they will shut down their FR system due to such privacy concerns [33]. This motivated several recent works to explore the potential of using synthetically generated face data [28], [29], [31], [34].

This work presents contributions towards enhancing the compactness of FR models while maintaining high accuracy in a privacy-friendly process that does not require real face databases. Towards that, this work is, to the best of our knowledge, the first to regulate the FR computational cost by applying quantization-aware training. We additionally propose a training paradigm involving KD that uses synthetically generated face data to fine-tune the quantized model and adapt the quantization operator parameters. We empirically prove the success of the proposed approach in reducing the bit bandwidth up to 5x of the evaluated models while maintaining, to a large degree, the model verification performance. Additionally, the use of synthetic data within the proposed training paradigm proved to be highly effective and produced quantized models that performed competitively with models quantized using real data, even outperforming them in many experimental setups. Our quantized models based on synthetic data did outperform other full precision models with higher memory footprints, as will be demonstrated later in this work.

II. RELATED WORKS

There is a large body of works that proposed efficient FR models based on designing compact convolution building blocks [8], [9], [11], [16]. Such efficient FR models followed those of deep image classification [12]–[14], which in terms evolved from the depthwise separable convolution [35]. Additionally, efficient FR models [8], [9], [11], [16] opted to replace the fully connected (FC) layer on the top of CNN with a global depthwise convolution to reduce the large number of parameters caused by the FC layer. MobileFaceNets [8] is a popular network architecture that has been widely adopted in different compact FR solutions [7], [36]. MobileFaceNets contains around one million trainable parameters with 443M

FLOPs. MobileFaceNets architecture is based on the residual bottlenecks proposed by MobileNetV2 [13] and depth-wise separable convolutions layer. VarGFaceNet [11] deployed the variable group convolutional network proposed by VarGNet [15] to design a compact FR model with 5m trainable parameters. ShuffleFaceNet [9] is a compact FR model based on ShuffleNet-V2 [14]. MixFaceNets [16] proposed a family of efficient FR models by extending the MixConv block [12] with a channel shuffle operation [14] aiming at increasing the discriminative ability of MixFaceNets.

An alternative to previous handcrafted DNN designs is utilizing NAS, KD, or a combination of different model compression techniques. KD transfers the acquired knowledge learned by a larger network to a smaller one [37], [38]. KD has shown great success in improving the verification performance of compact FR models [10], [11], [39]. Furthermore, the combination of KD with other model compression techniques such as compact model design [11], or NAS [10], [17] demonstrated very promising accuracies in FR.

One must note that all the discussed FR models are built with FP single floating-point, and none has adopted model quantization techniques. Additionally, all these works required the privacy-sensitive use of real face data, either during their conventional training or during the KD from a larger pre-trained network. This stresses the main contributions of this work, i.e., the use of unlabeled synthetic data in the proposed quantization paradigm.

III. METHODOLOGY

This work presents a privacy-friendly framework to minimize the computational complexity of deep learning-based FR models. Towards that, given an FR model with a FP floating point, we propose to quantize the weights and activations to b-bit precision format through uniform quantization aware training. The quantization process commonly requires fine-tuning/retraining the quantized model to adjust the quantization operator parameters and recover the model accuracy after applying the quantization process. Quantization usually requires the original model training data, or part of it, for this calibration or fine-tuning [40]. This original data may not be accessible after training the model due to restrictions

related to privacy, security, data ownership, and data availability restrictions (e.g., model shared from a data owner to a third party). To mitigate this challenge and promote privacy-aware solutions, our proposed framework utilizes synthetically generated face data to fine-tune and calibrate the quantized model. Moreover, to maintain the performance and enable the use of unlabeled synthetic data, the proposed framework combines the quantization process with KD during the fine-tuning phase. This step enables fine-tuning the quantized model without accessing the real training dataset or any prior assumption about the training dataset labels. Figure 1 presents an overview of the proposed framework.

This section presents first the quantization process applied to the FP floating-point FR model. Then, it presents the training paradigm utilized to fine-tune the quantized model.

A. Model Quantization

Quantization involves two main operators: Quantize and Dequantize. Let $x \in [\beta, \alpha]$ be a real value and b is the bit width of a low precision format. β and α are minimum and maximum values of x . A 2^b possible integer values can be represented using b-bit format and the value range of a b-bit signed integer precision format is between $[-2^{b-1}, 2^{b-1} - 1]$. A real value of a FP model in this work refers to a 32-bit single-precision floating-point. Quantization maps a real value $x \in [\beta, \alpha]$ to lie within the range value of low precision b-bit $[-2^{b-1}, 2^{b-1} - 1]$. Quantization operator consists of two processes: value transformation and clipping process. Formally, the transformation process that maps x into x_q can be defined as follows:

$$T(x, s, z) = \text{round}\left(\frac{x}{s} - z\right), \quad (1)$$

where $\text{round}()$ is the rounding method that defines the rounding step in which a value is rounded up or down to an integer. z is a constant parameter (zero-point) of the same type as the quantized value. It represents the quantized value x_q corresponding to the real zero value. s is a real-valued scaling factor that divides the real value range into a number of fractions. In asymmetric quantization ($-\alpha \neq \beta$), the scaling factor s and zero-point are defined as follows [40]:

$$s = \frac{\alpha - \beta}{2^b - 1}. \quad (2)$$

$$z = \text{round}\left(\beta \cdot \frac{2^b - 1}{\alpha - \beta} + 2^{b-1}\right). \quad (3)$$

The clipping process clips x_t (the output of Equation 1) to lie within the range $[-2^{b-1}, 2^{b-1} - 1]$. The clip operation can be defined as follows:

$$\text{clip}(x_t, l, u) = \begin{cases} l & \text{if } x_t < l \\ x & \text{if } l \leq x_t \leq u \\ u & \text{if } x > u, \end{cases} \quad (4)$$

where l and u are the minimum and maximum values of the quantization value range. The quantization of real value x to a b-bit precision format is given by:

$$Q(x, b) = \text{clip}(T(x, s, z), -2^{b-1}, 2^{b-1} - 1). \quad (5)$$

Dequantization operation that approximates the real value of the quantized one is defined as follows:

$$D(q) = s \cdot (q + z). \quad (6)$$

a) Tensor Quantization Granularity: Tensor quantization granularity defines how the quantization operator parameters are calculated and shared among the model tensors [41]. Quantization granularity [41] can be categorized into three groups: per-layer [42], per-group of channels [43] and per-channel [40], [42], [44], [45]. In this work, we opt to use the popular choice of per-channel granularity as it provides a high quantization resolution and it has repeatedly led to high accuracy [40], [42], [44], [45]. Per-channel granularity uses a fixed quantization parameter for each channel, independently from other channels.

B. Quantization-Aware Training (QAT)

QAT, utilized in QuantFace, is a common approach to adjust the quantized model parameters [40], [46]. This adjustment is often required for model quantization because rounding the weights of a pre-trained model often results in lower accuracy, especially if the weights and the activation functions have a wide range of values. QAT inserts fake (simulated) quantization operations in the network to emulate inference-time quantization [40]. QAT requires training or fine-tuning the model. In QAT, the forward and backward passes are usually carried out in a floating-point precision, and the model weights and activations are quantized after each gradient update [40]. After each training iteration, the derivatives of the quantized network weights need to be calculated to compute the loss gradients for backpropagation. However, the gradients of the fake quantized operations are predominantly zero [20], making the standard backpropagation not applicable. QuantFace follows the common QAT approaches [20], [40] by addressing this issue by using Straight Through Estimator (STE) [47] to approximate the gradient of fake quantization operators using a threshold, where the derivatives of fake quantization operators are set to one for inputs within the clipping range, i.e. when x is in the range $[\beta, \alpha]$.

C. Training paradigm

QAT requires access to the original labeled training dataset to fine-tune the quantized model [40], which may be infeasible for privacy concerns [48]. Very recently, a number of works proposed to fine-tune a quantized model with generated data from conditional generative models [21], [49]. However, these works [21], [49] required to generate data with labels that are aware of the data labels used to train the FP models. Such knowledge of training class labels is often unobtainable, given only a pre-trained FR model. To solve this, we propose a solution that utilizes unlabeled synthetic data along with a KD-based training paradigm that wavers the requirement of training data labels. Unlike conventional KD [37] that optimizes the classification output (require labeled data), the utilized KD-based training paradigm optimizes the feature representation needed for biometric verification i.e. given a

Model	param	Quantization data	Bits	Size (MB)	LFW	CFP	AgeDb-30	CALFW	CPLFW	IJB-C	IJB-B
					Accuracy (%)						TAR@FAR 1e-4 (%)
ResNet100	65.2M	-	FP32	261.22	99.83	98.40	98.33	96.13	93.22	96.50	95.25
		Real data	w8a8	65.31	99.80	98.31	98.13	96.05	92.92	96.38	95.13
		Synthetic data	w8a8	65.31	99.80	98.14	97.95	96.02	92.90	96.09	94.74
		Real data	w6a6	49.01	99.55	89.14	95.85	95.42	85.63	85.80	84.08
		Synthetic data	w6a6	49.01	99.45	91.00	96.43	95.58	86.60	87.00	85.06
ResNet50	43.6M	-	FP32	174.68	99.80	98.01	98.08	96.10	92.43	95.74	94.19
		Real data	w8a8	43.67	99.78	97.70	98.00	96.00	92.17	95.66	94.15
		Synthetic data	w8a8	43.67	99.78	97.43	97.97	95.87	92.08	95.18	93.67
		Real data	w6a6	32.77	99.70	95.00	97.17	95.87	90.17	91.74	90.07
		Synthetic data	w6a6	32.77	99.68	95.17	97.43	95.70	90.38	90.72	89.44
ResNet18	24.0M	-	FP32	96.22	99.67	94.47	97.13	95.70	89.73	93.56	91.64
		Real data	w8a8	24.10	99.63	94.46	97.03	95.72	89.48	93.56	91.57
		Synthetic data	w8a8	24.10	99.55	94.04	97.07	95.58	89.53	92.87	91.01
		Real data	w6a6	18.10	99.52	93.23	96.55	95.58	88.37	93.03	91.08
		Synthetic data	w6a6	18.10	99.55	93.34	96.62	95.32	89.05	92.36	90.38
MobileFaceNet	1.1M	-	FP32	4.21	99.47	91.59	95.62	95.15	87.98	90.88	88.54
		Real data	w8a8	1.10	99.43	91.40	95.47	95.05	87.95	90.57	88.32
		Synthetic data	w8a8	1.10	99.35	90.84	94.37	94.78	87.73	89.21	86.98
		Real data	w6a6	0.79	98.87	87.69	93.03	93.30	84.57	83.13	80.53
		Synthetic data	w6a6	0.79	99.08	87.64	91.77	93.48	84.85	82.94	80.58

TABLE I: B-bit precision data vs. performance on LFW , CFP , AgeDb-30 , CALFW, CPLFW (Accuracy %), IJB-C and IJB-B (TAR at FAR 1e-4). The results is reported for FP model (32-bit), quantized models to 8-bit (w8a8) and 6-bit (w6a6) using real and synthetic data. All decimal points are rounded up to two decimal places. The top verification performances under the same quantization settings (network architecture and bit bandwidth) are in bold.

batch of *unlabeled* face images, the quantized model is fine-tuned to learn producing feature representation similar to the one learned by the full precision model as will be detailed in this section. We propose to use synthetically generated data from a Generative Adversarial Network (GAN) [50], [51] to fine-tune the quantized model. We sample noise Z from a Gaussian distribution $N(0,1)$ and feed it into a pretrained generator G to generate unlabeled synthetic data x , as shown in Figure 1. Formally, the synthetically generated data is obtained by:

$$x = G(z), \quad Z \sim N(0,1). \quad (7)$$

Then, the synthetically generated data is aligned and cropped (See Section IV). This data is unlabeled, as the random Z produces random identities unrelated to those used to train the FP model. Thus, the model cannot be directly fine-tuned with the generated data. Given this restriction, we propose in this work to fine-tune the quantized model using KD from the FP model. Specifically, the quantized model is trained to learn feature embedding similar to the ones from the FP model in the normalized embedding space. During the fine-tuning phase, a batch of size M of unlabeled synthetic images X is sampled and fed into the quantized and full precision model to obtain feature embeddings f^q and f^t , respectively. Then, f^q and f^t are normalized ($f^q = \frac{f^q}{\|f^q\|_2}$, $f^t = \frac{f^t}{\|f^t\|_2}$) and used to compute the \mathcal{L}_{kd} loss based on the cosine distance between the normalized features. Finally, the gradient of the \mathcal{L}_{kd} loss function is computed and used to update the weight parameters. Different from supervised classification losses e.g. cross-entropy or conventional KD losses [37] that require class label for calculating the loss value, \mathcal{L}_{kd} loss is calculated based on the feature embedding layers. Thus, it mitigates the need to have identity labels for the input training images. Formally, the \mathcal{L}_{kd} loss is defined as follows:

$$\mathcal{L}_{kd} = 1 - \frac{1}{M} \sum_{i \in M} \frac{f_i^q \cdot f_i^t}{\|f_i^q\| \|f_i^t\|}. \quad (8)$$

Using synthetic data to train FR model might lead to sub-optimal verification performances [29] due to a possible domain gap between the real and synthetic data [52]–[54], especially if the model is trained from scratch to learn identity representation [29], which might require conducting domain adaption e.g. adversarial domain adaptation [29], [52] to reduce such effect. However, our goal in this work is not to learn identity representation from scratch through optimizing classification loss e.g. cross entropy, we rather fine-tune the pretrained model with synthetic data to adjust the model weights and quantization parameters after applying quantization process. Thus, by utilizing the proposed training paradigm, we restrict our use of the synthetic data to ensure that the response of the pretrained quantized model to input $x \in X$ during the fine-tuning phase is similar to the response of the full precision model to the same input $x \in X$. We demonstrate that this process is not largely effected by the potential domain gap between the real and synthetic data by (1) comparing the model responses to real and synthetic data i.e. activation function value ranges of two models fine-tuned with real and synthetic data 2, respectively and by (2) comparing the evaluation results of these models on real data benchmarks (details in Section V-C).

IV. EXPERIMENTAL SETUP

This section presents the baseline models with implementation details, model quantization implementation details, and evaluation benchmarks used in this work.

A. Baselines

The FP models in this work are trained with ArcFace loss [3] on MS1MV2 dataset [3], [25]. The MS1MV2 is a refined version of the MS-Celeb-1M [25] by [3] containing

5.8M images of 85K identities. The baseline backbones are ResNet100 [1], [3], ResNet50 [1], [3], ResNet18 [1], [3], and MobileFaceNet [8]. We follow the training setting of [3] to set the scale parameter s to 64 and the margin m to 0.5. We set the mini-batch size to 512 and trained the presented models on one Linux machine (Ubuntu 20.04.2 LTS) with Intel(R) Xeon(R) Gold 5218 CPU 2.30GHz, 512G RAM, and four Nvidia GeForce RTX-6000 GPUs. The FP models and the quantization operators in this paper are implemented using Pytorch [22]. All models are trained with Stochastic Gradient Descent (SGD) optimizer. We used random horizontal flipping with a probability of 0.5 for data augmentation during the training. We set the momentum to 0.9 and the weight decay to $5e-4$. All the images in the evaluation and training datasets are aligned and cropped to 112×112 , as described in [3]. The initial learning rate of the FP models is 0.1, and it is divided by 10 at 100K and 160K training iterations, following [3]. The training is stopped after 180K iterations.

B. Quantization implementation details

We quantize the weights and activations of all the baseline FP models to two, 6-bit and 8-bit, precision formats. We reported the results of the quantized models under two settings. First, the quantized models are fine-tuned and calibrated with the original (real) training data, MS1MV2 [3], [25] (described in Section IV-A). Second, the quantized models are fine-tuned and calibrated with the synthetically generated data. We utilized the official open source implementation² of StyleGAN2-ADA to randomly generate 0.5M synthetic face images. These images are then cropped and aligned using the method described in Section IV-A. In both settings, the quantized models are fine-tuned for 11K iterations with a learning rate of $1e-4$.

C. Evaluation benchmarks and metrics

The evaluation results of the FP and quantized models are reported on seven mainstream benchmarks: Labeled Faces in the Wild (LFW) [55], AgeDB-30 [56], Celebrities in Frontal-Profile in the Wild (CFP-FP) [57], Cross-age LFW (CALFW) [58], Cross-Pose LFW (CPLFW) [59], IARPA Janus Benchmark-C and B (IJB-C) [60] and (IJB-B) [61]. We follow the evaluation metrics defined in the utilized benchmarks as follows: LFW (accuracy), CA-LFW (accuracy), CP-LFW (accuracy), CFP-FP (accuracy), AgeDB-30 (accuracy), IJB-C, and IJB-B (true acceptance rate at a false acceptance rate of $1e-4$, noted as TAR at FAR $1e-4$).

V. RESULTS

Table I presents the achieved FR performance results by the FP models (ResNet100, ResNet50, ResNet18, and MobileFaceNet), along with the achieved ones by the quantized models to 8-bit weights and 8-bit activation (noted as w8a8), and to 6-bit weights and 6-bit activation (noted as w6a6) using synthetic or real data quantization. The results are grouped by each network architecture. In each group of rows, the results

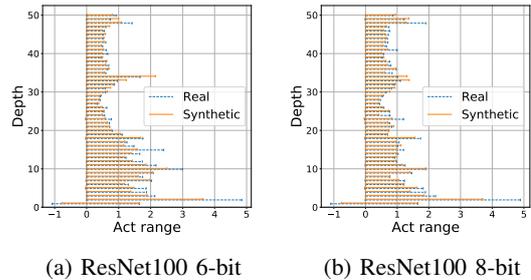


Fig. 2: Correlation between the activation functions (Act) value ranges ($[\beta, \alpha]$) of the ResNet100 quantized using real (solid orange) and synthetic data (dashed black). The y-axis represents the depth of the backbone activation function e.g. depth 1 is the first activation function. These plots represent a fixed-precision quantization bit bandwidth of 6-bit (2a) and 8-bit (2b). Each line in the plot represents the range value of the activation function where the start point is β and end point is α . The high correlation indicates that the quantized model is able to capture sufficient data information from the synthetic data, in comparison to real data.

are first presented for the FP model (baseline), followed by the quantized models. The size (in MB) of the FP models is approximately 4x the number of parameters, i.e., each parameter requires 4 bytes. In both real and synthetic data quantization settings, the reductions in bit bandwidth, and thus the model size, using w8a8 and w6a6, are around 4x and 5.3x, respectively. Also, model quantization enables performance gains in inference speed and memory bandwidth. However, the exact measures of inference speed and memory bandwidth depend on the underlying hardware and deep learning accelerator, as we discussed in Section I. Therefore, the presented results in this section are discussed as a trade-off between FR performance and the bit bandwidth, and thus, model size. Figure 3 presents the trade-off between the model size (in MB) and the achieved verification performance by the FP floating-point 32 models (FP32) and their respective quantized models using synthetically generated data. The model that has the best trade-off between the verification performance and model size tends to be on the top left in the plot 3. The following observations can be made based on the achieved results in Table I:

A. Impact of 8-bit bandwidth quantization

When the model is quantized to 8-bit (w8a8 setting), the achieved verification performances in all experimental settings are slightly degraded. However, the bit bandwidth is significantly reduced (around 4x) when the considered models are quantized. For example, the achieved accuracy by ResNet100 (261.22 MB) on AgeDb-30 is 98.33%. This accuracy slightly dropped to 98.13% and 97.95% when the ResNet-100 is quantized to 8-bit (65.31 MB) and fine-tuned with real and synthetic data, respectively. Similar observations can be made when all considered models are quantized to 8-bit. Another important observation can be drawn from the achieved results:

²<https://github.com/NVlabs/stylegan2-ada>

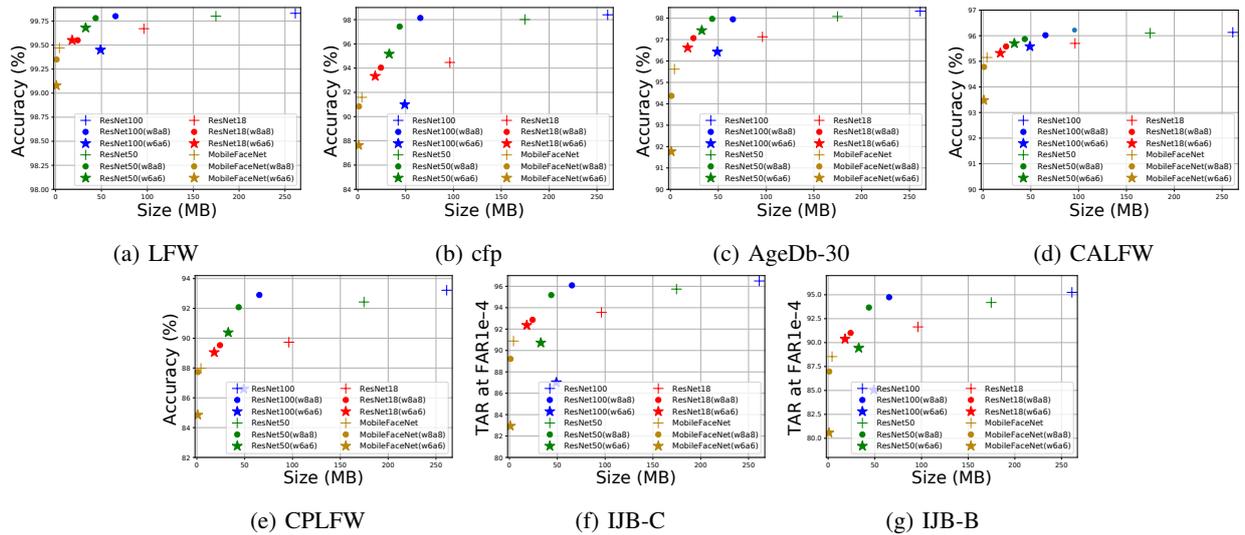


Fig. 3: The model size (in MB) vs. performance on LFW (accuracy), CFP (accuracy), AgeDB-30 (accuracy), CALFW (accuracy), CPLFW (accuracy), IJB-C (TAR at FAR1e-4), and IJB-B (TAR at FAR1e-4). The FP models are marked with a cross. The 8bit and 6bit quantized models using synthetic data are marked with full circles and stars, respectively. The models, after quantization, only lose marginal performance while their sizes are significantly reduced. Quantized models (e.g. ResNet100(w8a8), blue circle) in most cases outperforms larger full precision models (e.g. ResNet50 (green cross) and ResNet18 (red cross)).

when ResNet100 is quantized to 8-bit and fine-tuned with synthetic data (65.31 MB), it significantly outperformed ResNet18 (96.22 MB) on all considered benchmarks, resulting in around 30% less model size. Impressively, the ResNet100 quantized to 8-bit outperformed the FP ResNet50 on most benchmarks while being more than 60% smaller. This observation can be visually seen by comparing the (x,y) positions of blue circle (ResNet100(w8a8)) and green cross (ResNet50) marks in the trade-off plots of Figure 3. On large-scale evaluation benchmarks, the achieved TAR at FAR 1e-4 on IJB-C by quantized models using real and synthetic data is very competitive to the FP model. For example, the achieved verification performance by the FP ResNet19 is 93.56% TAR at FAR 1e-4, and the achieved performance by the quantized models with real and synthetic data are 93.56% and 92.87%, respectively.

B. Impact of 6-bit bandwidth quantization

Using 6-bit bandwidth, the achieved results by quantized models are, as expected, lower than the ones achieved by the 8-bit quantization. However, the reduction in model size is significantly higher than the 8-bit quantization. However, using 6-bit bandwidth can still achieve competitive results to the FP model for use-cases that are extremely limited with computational cost. Moving below 6-bit bandwidth, e.g., 4-bit, our experiments showed that none of the considered models converged during the fine-tuning process.

C. Impact of quantization data source

The quantized models fine-tuned with synthetic data achieved very competitive results to the ones fine-tuned with real data, and even, in many cases, the quantized models fine-tuned with synthetic data outperformed the quantized model with real data. This is especially true for the 6-bit bandwidth,

as shown in Table I. For example, using 6-bit bandwidth, the achieved accuracies on LFW by ResNet100 quantized with real and synthetic data are 99.55% and 99.45%, respectively. To illustrate the correlation between the quantized models using real and synthetic data, Figure 2 presents the activation functions value range variables ($[\beta, \alpha]$) of quantized models using real and synthetic data. The high correlation between the activation functions value range of the quantized model using real and synthetic data can be noticed from the overlap in the value range. This indicates that the quantized model is able to capture sufficient data information from the synthetic data to match the FP model output.

VI. CONCLUSION

This work is the first to explore the potential of regulating the computational cost of existing deep face recognition using low-bit format model quantization in a privacy-friendly process. In particular, once the model is quantized, synthetically generated face data from unconditional GAN is fed into the FP and quantized model. Then, the proposed training paradigm matches the feature embeddings of the FP and quantized model in a normalized embedding space. The reported results pointed out the effectiveness of the presented approach in regulating the computational cost of the face recognition model without accessing the original training data or any prior knowledge about the actual data used to train the FP model.

ACKNOWLEDGMENT

This research work has been funded by the German Federal Ministry of Education and Research and the Hessen State Ministry for Higher Education, Research and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE. This work has been partially funded

by the German Federal Ministry of Education and Research (BMBF) through the Software Campus Project.

REFERENCES

- [1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 770–778. [Online]. Available: <https://doi.org/10.1109/CVPR.2016.90>
- [2] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018, pp. 7132–7141.
- [3] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 4690–4699.
- [4] F. Boutros, N. Damer, F. Kirchbuchner, and A. Kuijper, "Elasticface: Elastic margin loss for deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2022, pp. 1578–1587.
- [5] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *13th IEEE International Conference on Automatic Face & Gesture Recognition, FG 2018, Xi'an, China, May 15-19, 2018*. IEEE Computer Society, 2018, pp. 67–74. [Online]. Available: <https://doi.org/10.1109/FG.2018.00020>
- [6] Y. Martínez-Díaz, M. Nicolás-Díaz, H. Méndez-Vázquez, L. S. Luevano, L. Chang, M. Gonzalez-Mendoza, and L. E. Sucar, "Benchmarking lightweight face architectures on specific face recognition scenarios," *Artificial Intelligence Review*, pp. 1–44, 2021.
- [7] J. Deng, J. Guo, D. Zhang, Y. Deng, X. Lu, and S. Shi, "Lightweight face recognition challenge," in *2019 IEEE/CVF ICCV, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*. IEEE, 2019, pp. 2638–2646.
- [8] S. Chen, Y. Liu, X. Gao, and Z. Han, "Mobilefacenet: Efficient cnns for accurate real-time face verification on mobile devices," in *CNBR 2018, Urumqi, China, August 11-12, 2018, Proceedings*, ser. Lecture Notes in Computer Science, vol. 10996. Springer, 2018, pp. 428–438. [Online]. Available: https://doi.org/10.1007/978-3-319-97909-0_46
- [9] Y. Martínez-Díaz, L. S. Luevano, H. M. Vazquez, M. Nicolás-Díaz, L. Chang, and M. González-Mendoza, "Shufflefacenet: A lightweight face architecture for efficient and highly-accurate face recognition," in *2019 IEEE/CVF ICCV, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*. IEEE, 2019, pp. 2721–2728.
- [10] F. Boutros, P. Siebke, M. Klemm, N. Damer, F. Kirchbuchner, and A. Kuijper, "Pocketnet: Extreme lightweight face recognition network using neural architecture search and multistep knowledge distillation," *IEEE Access*, vol. 10, pp. 46 823–46 833, 2022.
- [11] M. Yan, M. Zhao, Z. Xu, Q. Zhang, G. Wang, and Z. Su, "Vargfacenet: An efficient variable group convolutional neural network for lightweight face recognition," in *2019 IEEE/CVF ICCV, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, 2019, pp. 2647–2654. [Online]. Available: <https://doi.org/10.1109/ICCV.2019.00323>
- [12] M. Tan and Q. V. Le, "Mixconv: Mixed depthwise convolutional kernels," in *30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, September 9-12, 2019*, 2019, p. 74.
- [13] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, 2018, pp. 4510–4520.
- [14] N. Ma, X. Zhang, H. Zheng, and J. Sun, "Shufflenet V2: practical guidelines for efficient CNN architecture design," in *ECCV 2018, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV*, ser. Lecture Notes in Computer Science, vol. 11218. Springer, 2018, pp. 122–138.
- [15] Q. Zhang, J. Li, M. Yao, L. Song, H. Zhou, Z. Li, W. Meng, X. Zhang, and G. Wang, "Vargnet: Variable group convolutional neural network for efficient embedded computing," *CoRR*, vol. abs/1907.05653, 2019. [Online]. Available: <http://arxiv.org/abs/1907.05653>
- [16] F. Boutros, N. Damer, M. Fang, F. Kirchbuchner, and A. Kuijper, "Mixfacenets: Extremely efficient face recognition networks," in *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, 2021, pp. 1–8. [Online]. Available: <https://doi.org/10.1109/IJCB52358.2021.9484374>
- [17] X. Wang, "Teacher guided neural architecture search for face recognition," in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, 2021, pp. 2817–2825.
- [18] R. Banner, Y. Nahshan, and D. Soudry, "Post training 4-bit quantization of convolutional networks for rapid-deployment," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 2019, pp. 7948–7956. [Online]. Available: <https://proceedings.neurips.cc/paper/2019/hash/c0a62e133894cdce435bcb4a5df1db2d-Abstract.html>
- [19] Z. Yao, Z. Dong, Z. Zheng, A. Gholami, J. Yu, E. Tan, L. Wang, Q. Huang, Y. Wang, M. W. Mahoney, and K. Keutzer, "HAWQ-V3: dyadic neural network quantization," in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, 2021, pp. 11 875–11 886. [Online]. Available: <http://proceedings.mlr.press/v139/yao21a.html>
- [20] R. Krishnamoorthi, "Quantizing deep convolutional networks for efficient inference: A whitepaper," *arXiv preprint arXiv:1806.08342*, 2018.
- [21] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko, "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2704–2713.
- [22] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035.
- [23] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. A. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016, Savannah, GA, USA, November 2-4, 2016*, 2016, pp. 265–283.
- [24] Y. Choukroun, E. Kravchik, F. Yang, and P. Kisilev, "Low-bit quantization of neural networks for efficient inference," in *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, 2019, pp. 3009–3018. [Online]. Available: <https://doi.org/10.1109/ICCV.2019.00363>
- [25] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, ser. Lecture Notes in Computer Science, vol. 9907. Springer, 2016, pp. 87–102.
- [26] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *CoRR*, vol. abs/1411.7923, 2014. [Online]. Available: <http://arxiv.org/abs/1411.7923>
- [27] I. D. Raji and G. Fried, "About face: A survey of facial recognition evaluation," *CoRR*, vol. abs/2102.00813, 2021. [Online]. Available: <https://arxiv.org/abs/2102.00813>
- [28] F. Boutros, M. Huber, P. Siebke, T. Rieber, and N. Damer, "Sface: Privacy-friendly and accurate face recognition using synthetic data," *arXiv preprint*, 2022.
- [29] H. Qiu, B. Yu, D. Gong, Z. Li, W. Liu, and D. Tao, "Synface: Face recognition with synthetic data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 10 880–10 890.
- [30] P. Voigt and A. v. d. Bussche, *The EU General Data Protection Regulation (GDPR): A Practical Guide*, 1st ed. Springer, 2017.
- [31] N. Damer, C. A. F. López, M. Fang, N. Spiller, M. V. Pham, and F. Boutros, "Privacy-friendly synthetic data for the development of face morphing attack detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2022, pp. 1606–1617.
- [32] B. Meden, P. Rot, P. Terhörst, N. Damer, A. Kuijper, W. J. Scheirer, A. Ross, P. Peer, and V. Struc, "Privacy-enhancing face biometrics: A comprehensive survey," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 4147–4183, 2021.

- [33] J. Pesenti. (2021) An update on our use of face recognition. [Online]. Available: <https://about.fb.com/news/2021/11/update-on-use-of-face-recognition/>
- [34] C. Rong, X. Zhang, and Y. Lin, "Feature-improving generative adversarial network for face frontalization," *IEEE Access*, vol. 8, pp. 68 842–68 851, 2020. [Online]. Available: <https://doi.org/10.1109/ACCESS.2020.2986079>
- [35] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017, pp. 1800–1807.
- [36] X. Li, F. Wang, Q. Hu, and C. Leng, "Airface: Lightweight and efficient model for face recognition," in *2019 IEEE/CVF ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, 2019, pp. 2678–2682. [Online]. Available: <https://doi.org/10.1109/ICCVW.2019.00327>
- [37] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *CoRR*, vol. abs/1503.02531, 2015. [Online]. Available: <http://arxiv.org/abs/1503.02531>
- [38] M. Huber, F. Boutros, F. Kirchbuchner, and N. Damer, "Mask-invariant face recognition through template-level knowledge distillation," in *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, 2021, pp. 1–8.
- [39] F. Boutros, N. Damer, K. Raja, F. Kirchbuchner, and A. Kuijper, "Template-driven knowledge distillation for compact and accurate periocular biometrics deep-learning models," *Sensors*, vol. 22, no. 5, 2022. [Online]. Available: <https://www.mdpi.com/1424-8220/22/5/1921>
- [40] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. G. Howard, H. Adam, and D. Kalenichenko, "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018, pp. 2704–2713.
- [41] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, "A survey of quantization methods for efficient neural network inference," *CoRR*, vol. abs/2103.13630, 2021. [Online]. Available: <https://arxiv.org/abs/2103.13630>
- [42] R. Krishnamoorthi, "Quantizing deep convolutional networks for efficient inference: A whitepaper," *CoRR*, vol. abs/1806.08342, 2018. [Online]. Available: <http://arxiv.org/abs/1806.08342>
- [43] S. Shen, Z. Dong, J. Ye, L. Ma, Z. Yao, A. Gholami, M. W. Mahoney, and K. Keutzer, "Q-BERT: hessian based ultra low precision quantization of BERT," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 2020, pp. 8815–8821.
- [44] Q. Huang, D. Wang, Z. Dong, Y. Gao, Y. Cai, T. Li, B. Wu, K. Keutzer, and J. Wawrzyniec, "Codenet: Efficient deployment of input-adaptive object detection on embedded fpgas," in *FPGA '21: The 2021 ACM/SIGDA International Symposium on Field Programmable Gate Arrays, Virtual Event, USA, February 28 - March 2, 2021*, 2021, pp. 206–216.
- [45] D. Zhang, J. Yang, D. Ye, and G. Hua, "Lq-nets: Learned quantization for highly accurate and compact deep neural networks," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, 2018, pp. 373–390.
- [46] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 4820–4828.
- [47] Y. Bengio, N. Léonard, and A. C. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," *CoRR*, vol. abs/1308.3432, 2013. [Online]. Available: <http://arxiv.org/abs/1308.3432>
- [48] Y. Choi, J. P. Choi, M. El-Khamy, and J. Lee, "Data-free network quantization with adversarial knowledge distillation," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, 2020, pp. 3047–3057.
- [49] S. Xu, H. Li, B. Zhuang, J. Liu, J. Cao, C. Liang, and M. Tan, "Generative low-bitwidth data free quantization," in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XII*, 2020, pp. 1–17.
- [50] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, 2014*, pp. 2672–2680.
- [51] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.
- [52] M. Xu, J. Zhang, B. Ni, T. Li, C. Wang, Q. Tian, and W. Zhang, "Adversarial domain adaptation with domain mixup," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 2020, pp. 6502–6509.
- [53] S. Sankaranarayanan, Y. Balaji, A. Jain, S. Lim, and R. Chellappa, "Learning from synthetic data: Addressing domain shift for semantic segmentation," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 3752–3761.
- [54] S. Lee, E. Park, H. Yi, and S. H. Lee, "Strdan: Synthetic-to-real domain adaptation network for vehicle re-identification," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 2590–2597.
- [55] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.
- [56] 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 CVPRW, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 1997–2005.
- [57] S. Sengupta, J. Chen, C. D. Castillo, V. M. Patel, R. Chellappa, and D. W. Jacobs, "Frontal to profile face verification in the wild," in *2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016, Lake Placid, NY, USA, March 7-10, 2016*. IEEE Computer Society, 2016, pp. 1–9.
- [58] T. Zheng, W. Deng, and J. Hu, "Cross-age LFW: A database for studying cross-age face recognition in unconstrained environments," *CoRR*, vol. abs/1708.08197, 2017. [Online]. Available: <http://arxiv.org/abs/1708.08197>
- [59] T. Zheng and W. Deng, "Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments," Beijing University of Posts and Telecommunications, Tech. Rep. 18-01, February 2018.
- [60] B. Maze, J. C. Adams, J. A. Duncan, N. D. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, and P. Grother, "IARPA janus benchmark - C: face dataset and protocol," in *2018 International Conference on Biometrics, ICB 2018, Gold Coast, Australia, February 20-23, 2018*. IEEE, 2018, pp. 158–165.
- [61] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. C. Adams, T. Miller, N. D. Kalka, A. K. Jain, J. A. Duncan, K. Allen, J. Cheney, and P. Grother, "IARPA janus benchmark-b face dataset," in *2017 IEEE CVPRW, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 592–600. [Online]. Available: <https://doi.org/10.1109/CVPRW.2017.87>