EdgeFace : Efficient Face Recognition Model for Edge Devices

Anjith George, *Member, IEEE,* Christophe Ecabert, *Member, IEEE,*
Hatef Otroshi Shahreza, *Graduate Student Member, IEEE,* Ketan Kotwal, *Senior Member, IEEE,*
Sébastien Marcel, *Senior Member, IEEE*

Abstract—In this paper, we present EdgeFace - a lightweight and efficient face recognition network inspired by the hybrid architecture of EdgeNeXt. By effectively combining the strengths of both CNN and Transformer models, and a low rank linear layer, EdgeFace achieves excellent face recognition performance optimized for edge devices. The proposed EdgeFace network not only maintains low computational costs and compact storage, but also achieves high face recognition accuracy, making it suitable for deployment on edge devices. The proposed EdgeFace model achieved the top ranking among models with fewer than 2M parameters in the IJCB 2023 Efficient Face Recognition Competition. Extensive experiments on challenging benchmark face datasets demonstrate the effectiveness and efficiency of EdgeFace in comparison to state-of-the-art lightweight models and deep face recognition models. Our EdgeFace model with 1.77M parameters achieves state of the art results on LFW (99.73%), IJB-B (92.67%), and IJB-C (94.85%), outperforming other efficient models with larger computational complexities. The code to replicate the experiments will be made available publicly.

Index Terms—Efficient Face Recognition, Edge Devices, Face Recognition

◆

1 INTRODUCTION

Face recognition has become an increasingly active research field, achieving significant recognition accuracy by leveraging breakthroughs in various computer vision tasks through the development of deep neural networks [1], [2], [3] and margin-based loss functions [4], [5], [6], [7], [8], [9], [10]. In spite of remarkable improvements in recognition accuracy, state-of-the-art face recognition models typically involve a deep neural network with a high number of parameters (which requires a large memory) and considerable computational complexity. Considering memory and computational requirements, it is challenging to deploy state-of-the-art face recognition models on resource-constrained devices, such as mobile platforms, robots, embedded systems, etc.

To address the issue of memory and computational complexity of state-of-the-art deep neural networks, researchers have been focusing on designing lightweight and efficient neural networks for computer vision tasks that can achieve a better trade-off between recognition accuracy, on one side, and required memory and computational resources, on the other side [11], [12], [13], [14]. Recently, some works have attempted to utilize lightweight convolutional neural network (CNN) architectures, such as MobileNets [15], [16], ShuffleNet [17], [18], VarGNet [19], and MixNets [20], for face recognition tasks [13], [21], [22], [23], reducing model parameters as well as computational complexity and meanwhile maintaining high levels of accuracy. However, with the recent emergence of vision transformers (ViTs) [24] and their ability in modeling global interactions between pixels, there is

an opportunity to further improve the efficiency and performance of face recognition models by leveraging both CNNs and ViTs capabilities.

In this paper, we present EdgeFace , a novel lightweight face recognition model inspired by the *hybrid* architecture of EdgeNeXt [25]. We adapt the EdgeNeXt architecture for face recognition and also introduce a Low Rank Linear (LoRaLin) module to further reduce the computation in linear layers while providing a minimal compromise to the performance of the network. LoRaLin replaces a high-rank matrix in a fully connected layer with two lower-rank matrices, and therefore reduces the number of parameters and required number of multiply adds (MAdds). EdgeFace effectively combines the advantages of both CNNs and ViTs, utilizing a split depth-wise transpose attention (STDA) encoder to process input tensors and encode multi-scale facial features, while maintaining low computational costs and compact storage requirements. Through extensive experimentation on challenging benchmark face datasets, including LFW, CA-LFW, CP-LFW, CFP-FP, AgeDB-30, IJB-B, and IJB-C, we demonstrate the effectiveness and efficiency of EdgeFace in comparison to state-of-the-art lightweight models and deep face recognition models, showing its potential for deployment on resource-constrained edge devices. Variants of the proposed EdgeFace model achieved the top ranking among models with fewer than 2M parameters in the IJCB 2023 Efficient Face Recognition Competition [26]. The main contributions of our work can be summarized as follows:

- We propose an efficient lightweight face recognition network, called EdgeFace , based on a hybrid network architecture that leverages CNN and ViT capabilities. We adapt the hybrid network architecture of EdgeNeXt for the face recognition task. To the best of our knowledge, this is the first work that uses a hybrid CNN-transformer for efficient face recognition.
- We introduce a Low Rank Linear (LoRaLin) module to further reduce the computation in linear layers while providing

- *All authors are with Idiap Research Institute, Martigny, Switzerland. Hatef Otroshi Shahreza is also affiliated with École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, and Sébastien Marcel is also affiliated with Université de Lausanne (UNIL), Lausanne, Switzerland. E-mail: {anjith.george, christophe.ecabert, hatef.otroshi, ketan.kotwal, sebastien.marcel}@idiap.ch*

a minimal compromise to the performance of the network. LoRaLin module replaces a high-rank matrix in a fully connected layer with two lower-rank matrices, and therefore reduces the number of parameters and required computations.

- We provide extensive experimental results on various challenging face recognition datasets, demonstrating the superior performance of EdgeFace in comparison to existing lightweight models. Our experiments also highlight the model's robustness under different conditions, such as pose variations, illumination changes, and occlusions.

The source code will be made available publicly [1].

The remainder of this paper is organized as follows. Section 2 provides a brief overview of related works, discussing the limitations of existing lightweight face recognition models and the potential advantages of hybrid architectures. Section 3 presents a detailed description of the proposed EdgeFace model and the overall hybrid architecture. Section 4 outlines the experimental setup, datasets, and evaluation metrics used to assess the performance of EdgeFace , followed by a comprehensive analysis of the results in Section 5. Finally, Section 6 concludes the paper and outlines potential future directions for this research.

2 RELATED WORK

Over the past decade, face recognition (FR) has been regarded as one of the most prominent and widely deployed applications of deep learning. However, as the handheld mobile devices and edge computing became prevalent, the researchers directed efforts towards developing lightweight FR models without compromising their accuracy. Since then the design of lightweight CNNs has emerged as an active research area in general machine learning; and various lightweight architectures have been developed for common ML applications, such as object recognition. Inspired by the success of smaller ML models, the biometric community has adapted some of these lightweight architectures for FR tasks. We begin with an overview of some of the general-purpose lightweight architectures and then discuss advances in lightweight and efficient FR.

2.1 Lightweight Neural Architectures

With the introduction of MobileNets [15], [16], the use of depthwise separable convolutions became a major factor in further improving the aspects of model parameters and FLOPs. The family of MobileNet architectures splits the typical convolution layer into depthwise (or channelwise) convolutions, followed by pointwise convolutions. These architectures are small and low-latency, and thus well-suited for deployment on handheld or embedded applications. Iandola *et al.* proposed SqueezeNet architecture that achieved state-of-the-art recognition accuracy then, despite having $50\times$ fewer parameters than the contemporary models [27]. The SqueezeNet architecture uses a 'fire' module as its core block. This module consists of squeeze and expand operations that are performed by 1×1 convolution filters and a mix of 1×1 and 3×3 filters, respectively. Zhang *et al.* constructed ShuffleNet architectures– that employ pointwise group convolutions and channel shuffling– for efficient processing on mobile devices [17]. Several variants of ShuffleNet can be defined by applying a scale factor to the number of channels.

The vanilla depth-wise convolutions are extended by incorporating multiple kernel sizes in a single convolution to design MixConvNets [20]. Kernels with different sizes simplify capturing multiple patterns from the input using a layer of depthwise convolutions. By replacing depthwise convolutions in MobileNets (v1 and v2) with the MixConv feature maps, the MixConvNets are able to achieve better accuracy with a slight drop in the number of parameters and FLOPs compared to the baseline MobileNet architectures. The ShiftNet, proposed in [28], is a family of CNNs with a 'shift' block that is a FLOP-free alternative to expensive convolution operation. The shift block, along with pointwise convolutions, efficiently mixes spatial information across channels and helps attain a competitive performance for several tasks, such as classification and style transfer.

In [19], variable group convolutions were proposed specifically for deploying neural networks on embedded systems such as FPGA or ASIC. The variable group operations are targeted at balancing the computational complexity inside a network block (primarily depthwise separable convolutions). A series of linear transformations is used to generate *ghost* feature maps with cheap computation in [29]. By stacking such Ghost modules, Han *et al.* proposed a GhostNet architecture that requires fewer parameters and computations compared to the architectures using vanilla convolutional networks. Different variants of GhostNets can be generated by controlling hyperparameters related to the number of intrinsic feature maps and kernel size.

Vision Transformer (ViT) architectures [3] have achieved excellent results for various recognition tasks, but their high computational costs have restricted the usage of vision transformers in a low-resource environment. To address this shortcoming, Chen *et al.* combined local processing in CNNs (such as MobileNet) and global interaction in transformers to design a new architecture, Mobile-Former [30]. Mehta and Rastegari introduced MobileViT architecture, based on local-global image context fusion, to build a lightweight and low latency network for general vision tasks [31]. While both aforementioned architectures attempt to leverage the benefits of CNNs and transformers for vision-classification tasks, the computational complexity of their MHA (multi-head attention) blocks still remains a bottleneck for the inference time on edge devices.

2.2 Lightweight FR Architectures

MobileFaceNets are a family of efficient CNN models, based on MobileNet architecture [15], [16], designed for real-time face verification tasks [21]. It achieved 99.55% accuracy on LFW while using less than 1M parameters. The Efficient Lightweight Attention Networks (ELANet) consist of inverted residual blocks (similar to MobileNetV2), and additionally, employ concurrent channel- and spatial-level attention mechanisms [32]. The ELANets have nearly 1M parameters, and achieve state-of-the-art performance across multiple datasets.

The MixConv concept [20] was used to develop MixFaceNet networks for lightweight FR [13]. The XS configuration of MixFaceNet has been reported to exhibit high recognition performance with as low as 1M parameters.

The FR model using ShiftNet architecture [28] with 0.78M parameters, called ShiftFaceNet, achieves a comparable performance to that of FaceNet in terms of recognition accuracy. Duong *et al.* considered faster downsampling of spatial data/ feature maps and bottleneck residual blocks towards developing lightweight

---

[1] https://gitlab.idiap.ch/bob/bob.paper.tbiom2023_edgeface

FR models [33]. Their MobiFace and Flipped-MobiFace models provide more than 99.70% accurate results on LFW dataset. Inspired from the ShuffleNetV2 [18], the family of lightweight models, referred to as ShuffleFaceNet, for FR was proposed in [22]. The number of parameters in these models vary from 0.5M to 4.5M while verification accuracies of higher than 99.20% have been reported for LFW dataset. Another family of lightweight architectures, ConvFaceNeXt [34] uses enhanced version of ConvNeXt blocks and different downsampling strategies to reduce the number of parameters as well as FLOPs. With about 1M parameters and nearly 400M FLOPs, ConvFaceNeXt networks achieve a comparable performance in FR.

In [14], neural architecture search (NAS) was used to automatically design an efficient network- PocketNet, for face recognition. The PocketNet architecture was learnt using differential architecture search (DARTS) algorithm on CASIA-WebFace dataset [35]. The training of this network also comprises a multi-step knowledge distillation (KD). Another approach involving KD for training a face recognition network was employed in [23]. Their model uses variable group convolutions to handle the unbalance of computational intensity. The corresponding model, called VarGFaceNet, was the winner of Lightweight Face Recognition (LFR) challenge at ICCV 2019 [11]. In [36], authors introduced a distillation framework called SynthDistill, and shown that lightweight models can be trained using synthetic data in an online distillation framework.

Recently, Alansari *et al.* proposed GhostFaceNets (multiple configurations) that exploit redundancy in convolutional layers to create compact networks [37]. In these modules, a certain fixed percentage of the convolutional feature maps are generated using depthwise convolutions that are computationally inexpensive. With configurable hyperparameters, GhostFaceNets can be designed to contain as low as 61M FLOPs with nominal reduction in their recognition performance.

In [36], lightweight networks (called TinyFaR), based on TinyNet structure [38], were suggested and were trained by KD from pretrained FR model using synthetic data. For training the lightweight network within the KD framework, a face generator model was used to generate synthetic face images, and the lightweight network (as a student) was optimized to generate the same embedding as the pretrained face recognition model (as a teacher). This work used dynamic sampling to help the student network focus on difficult images while exploring newer synthetic images.

3 PROPOSED EDGEFACE ARCHITECTURE

In this section, we describe the detailed architecture of the EdgeFace FR model. While most of the works on efficient face recognition networks focus on variants of CNNs, they have two primary constraints due to their convolution operations. Firstly, they possess a local receptive field, making it challenging for them to represent global context. Secondly, the weights learned by CNNs remain static during inference, limiting their adaptability to different input content. Transformers and CNN-Transformer hybrids attempt to address these limitations, despite their higher computational cost. In this work, we introduce a lightweight FR model inspired from the CNN-Transformer hybrid architecture of the EdgeNeXt model introduced in [25]. We adapt this model to make it suitable for the face recognition task with a focus on reducing the parameters and FLOPs.

3.1 EdgeFace Face Recognition Model

The primary focus of this work is to design an efficient network tailored for face recognition on edge devices. Towards this goal, we extend the EdgeNeXt [25] architecture for face recognition. First, we try to reduce the parameters and FLOPs of the model further by replacing the Linear layers in the EdgeNeXt network with the newly introduced low rank *LoRaLin* layers. In addition, we add a classification head composed of Adaptive Average Pooling and layer norms, followed by a *LoRaLin* layer outputting a 512-dimensional representation. The input resolution required for the model is adjusted to be 112×112. To optimally train this adapted model for face recognition, we employ end-to-end training in conjunction with a CosFace [5] classification head. Figure 1 provides a schematic representation of the updated EdgeFace face recognition model. First, we detail the architecture of the EdgeNeXt model designed for image classification, followed by our new additions to make it an efficient face recognition network.

TABLE 1: Model Layer Structure, and output dimensions of intermediate layers for different variants of EdgeFace

Layer (depth)	O/P size	Channels		
		SMALL	X-SMALL	XX-SMALL
Sequential 2-1	28×28	48	32	24
Conv2d 3-1	28×28	48	32	24
LayerNorm2d 3-2	28×28	48	32	24
EdgeFace-Stage 3-3	28×28	48	32	24
EdgeFace-Stage 3-4	14×14	96	64	48
EdgeFace-Stage 3-5	7×7	160	100	88
EdgeFace-Stage 3-6	3×3	304	192	168
AdaptiveAvgPool2d 4-9	1×1	304	192	168
LayerNorm2d 3-8	1×1	304	192	168
Flatten 3-9	-	304	192	168
Dropout 3-10	-	304	192	168
Linear 3-11	-	512	512	512
MPARAMS	-	5.44	2.24	1.24
MFLOPS	-	461.7	196.9	94.7

3.2 EdgeNeXt Architecture

The EdgeNeXt Architecture [25] is a lightweight hybrid design that combines the merits of Transformers [3], [39] and Convolutional Neural Networks (CNNs) for low-powered edge devices. EdgeNeXt models with a smaller number of parameters, model size and multiply-adds (MAdds) and outperforms models such as MobileViT [31] and EdgeFormer [40] in image recognition performance. The EdgeNeXt model builds on ConvNeXt [41] and introduces a new component known as the Split Depth-wise Transpose Attention (STDA) encoder. This encoder works by dividing input tensors into several channel groups. It then uses depth-wise convolution in conjunction with self-attention mechanisms across the channel dimensions. By doing so, the STDA encoder naturally enlarges the receptive field and effectively encodes features at multiple scales. The extensive requirements of the transformer self-attention layer make it impractical for vision tasks on edge devices, primarily due to its high MAdds and latency. To address this issue in SDTA encoder, they utilize transposed query and key attention feature maps [42]. This approach enables linear complexity by performing the dot-product operation of the Multi-Head Self-Attention (MSA) across channel dimensions, instead of spatial dimensions. As a result, cross-covariance across channels can be computed and create attention feature maps that inherently contain global representations. They also introduce adaptive kernel sizes to capture more global information by using smaller kernel
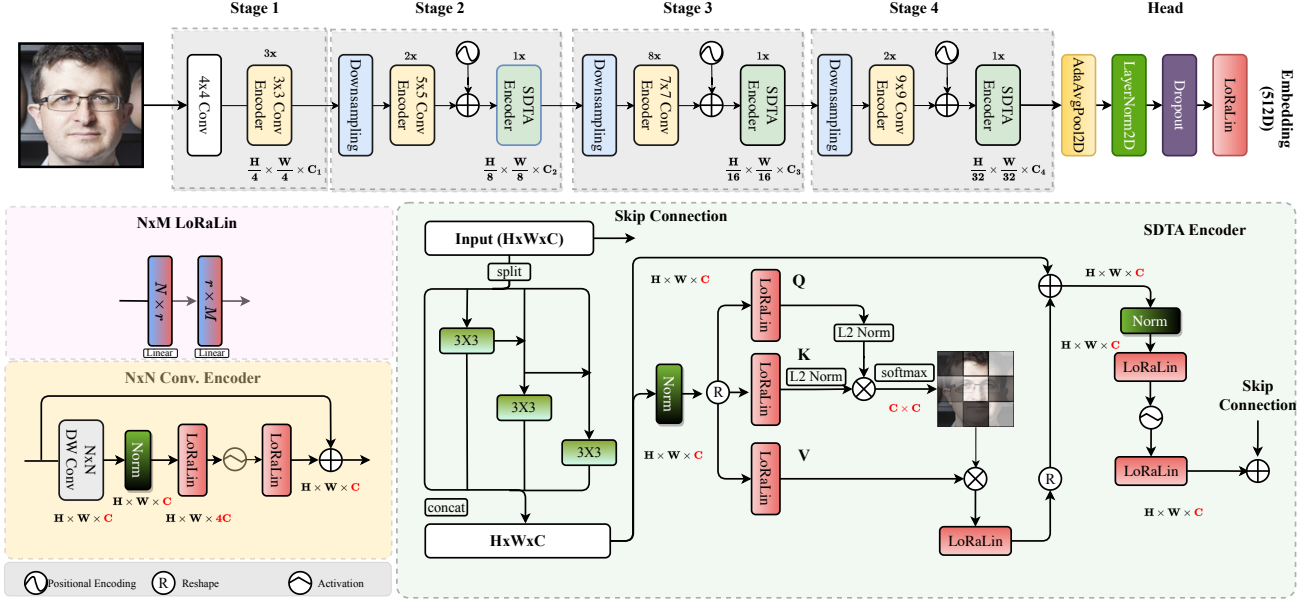
Fig. 1: A schematic diagram of the proposed EdgeFace Face Recognition model. The image is adapted from the EdgeNeXt [25] model to show the additional elements added to convert it to a face recognition network. Specifically, we introduce *LoRaLin* layers and add a head to obtain the 512-dimensional embeddings.

sizes in the initial layers followed by larger kernels for the latter stages in the convolutional encoder stages. These models come in various sizes, offering flexibility based on specific requirements. They include the extra-extra small, extra-small, and small variants. More details about the architecture can be found in [25]. Details of the variants and dimensions of feature maps at different levels are shown in Table 1.

3.3 Low Rank Linear Module (*LoRaLin*)

Despite the considerable optimization offered by the EdgeNeXt architecture, it is observed that a significant portion of both computational and parameter overhead originates from the linear layers. In an attempt to attenuate these parameter demands, we propose the incorporation of a Low Rank Linear Module (*LoRaLin*). This module effectively reduces computational requirements while maintaining minimal compromise to overall performance.

Hu *et al.* [43] proposed an approach termed Low-Rank Adaptation (LoRA) for reducing the number of trainable parameters in large language models during fine-tuning. The LoRA tuning method maintains the weights of the pretrained model unchanged while introducing trainable rank decomposition matrices into every layer of the Transformer architecture. This technique draws inspiration from the concept of 'low intrinsic dimension' observed when adapting a pretrained model to a specific task [44]. The amount of newly introduced parameters are considerably less, even though the original full rank matrices needs to be used at inference time. However, our aim is to reduce the parameter count of the model while accepting a trade-off in terms of model capacity. To accomplish this, we adopt a strategy of factorizing each fully connected layer into two low rank matrices.

Consider a fully connected layer in the network:

$$Y = W_{M \times N} X + b. \tag{1}$$

The weight matrix W in a linear layer of a neural network, which maps an input of size M to an output of size N, has dimensions $M \times N$.

This matrix can be represented as the product of two low rank matrices as follows:

$$W_{M \times N} = W_{M \times r} \cdot W_{r \times N}, \tag{2}$$

where, $W_{M \times r}$ and $W_{r \times N}$, are low rank matrices with a rank r.

Now, the original linear layer can be implemented as :

$$Y = W_{r \times N}(W_{M \times r}(X)) + b. \tag{3}$$

Essentially as two linear layers with lower ranks, this reduces the number of parameters, and the number of multiply adds (MAdds).

This can be implemented using two linear layers instead of one as shown in Fig. 2.

Fig. 2: PyTorch class for a Low-Rank Linear layer (*LoRaLin*).

```
class LoRaLin(nn.Module):
    def __init__(self, in_feat, out_feat, gamma
        ↪ , bias):
        super(LoRaLin, self).__init__()
        rank = max(2,min(in_feat, out_feat) *
            ↪ gamma))
        self.lin1 = nn.Linear(in_feat, rank,
            ↪ bias=False)
        self.lin2 = nn.Linear(rank, out_feat,
            ↪ bias=bias)

    def forward(self, input):
        x = self.lin1(input)
        x = self.lin2(x)
        return x
```

In this context, the rank of each module is determined by a hyper parameter known as Rank-ratio (γ), which governs the ratio

between the ranks. A minimum value of two is employed as the lower limit for the rank in our implementation.

$$rank = \max(2, \gamma * \min(M, N)), \tag{4}$$

By varying the value of γ, both the number of parameters and FLOPS undergo changes. For instance, in the case of the "edgenext-extra-small (XS)" network, the Figure 3 illustrates the reduction in the number of parameters and FLOPS with lower values of γ. The dotted line represents the values associated with the original linear layer. Notably, for $\gamma \leq 0.8$, both parameter count and computational efficiency demonstrate improvements compared to the base model.
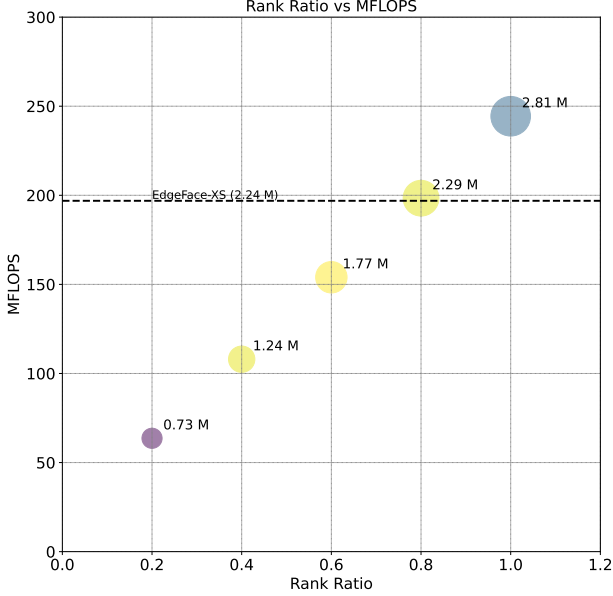

Fig. 3: The figure shows the reduction in Model Parameters (MPARAMS) and Multiply-Accumulate Operations (MFLOPS) as a function of Rank-ratio (γ). The dotted line represents the corresponding values from the default model employing a conventional Linear layer.

3.4 Training details

The dataset used for training the FR models constitutes selected subsets of the Webface260M dataset [45], specifically, the Web-Face 12M and WebFace 4M subsets. These subsets are characterized by an abundance of pre-aligned face images, each with a resolution of 112×112. The initial preprocessing step entails the conversion of these images into tensors, followed by normalization within the -1 to 1 range. We further enhance the data variability through a series of augmentations, including random grayscale conversion, resizing, and blurring. These augmentations are implemented leveraging the capabilities of the DALI [46] library. The models were trained with 4/8 Nvidia RTX 3090 (24GB) GPUs using distributed training strategy. We trained our models using PyTorch with AdamW optimizer [47] and trained the models with CosFace [5] loss function using a polynomial decay learning rate schedule with restarts to achieve the best performance. The batchsize on a single GPU varied from 256 to 512 depending on the size of the model. The embedding size during training is kept as 512. We used the distributed PartialFC algorithm [48] for

faster training and to handle memory issues while dealing with a large number of identities. During inference, the classification head is removed and the resulting 512-D embedding is used for the comparisons. The training settings and hyper parameters for different models were selected for optimal performance.

4 EXPERIMENTS

4.1 Test Datasets

We evaluated performance of the proposed EdgeFace model on seven distinct benchmarking datasets. The datasets selected for assessment include Labeled Faces in the Wild (LFW) [49], Cross-age LFW (CA-LFW) [50], CrossPose LFW (CP-LFW) [51], Celebrities in Frontal-Profile in the Wild (CFP-FP) [52], AgeDB-30 [53], IARPA Janus Benchmark-B (IJB-B) [54], and IARPA Janus Benchmark-C (IJB-C) [55]. To maintain consistency with prior works, we report accuracy values for high-resolution datasets such as LFW, CA-LFW, CP-LFW, CFP-FP, and AgeDB-30. For the IJB-B and IJB-C datasets, we report the True Accept Rate (TAR) at a False Accept Rate (FAR) of 1e-4.

4.2 Comparison with SOTA

Table 2 compares our method with SOTA lightweight face recognition models in the literature on different benchmarking datasets. We categorized models in the literature based on the number of parameters into 2-5 M parameters and < 2M parameters. In each category, we also have a representative version of EdgeFace model. In the category of 2-5 M parameters models, our representative model is EdgeFace-S ($\gamma = 0.5$), and in the second category (< 2M parameters) we can consider EdgeFace-XS ($\gamma = 0.6$) as our representative model. As the results in this table show our EdgeFace models achieve competitive performance with SOTA lightweight models in the literature. For CA-LFW, CP-LFW, IJB-B, and IJB-C datasets, our EdgeFace-S ($\gamma = 0.5$) model achieves the best recognition accuracy compared to SOTA models 2-5 M parameters. It is noteworthy our EdgeFace-S ($\gamma = 0.5$) model is also the most efficient model in terms of FLOPs among the SOTA lightweight models with 2-5 M parameters. For the second category, our EdgeFace-XS ($\gamma = 0.6$) model achieves the best recognition performance for LFW, CP-LFW, IJB-B, and IJB-C datasets. Compared to other models in the same category, our model is the second most efficient model in terms of FLOPs. In this category, we observe that ShuffleFaceNet 0.5x has fewer FLOPs, but it also has the poorest recognition performance in all datasets. The superior performance of our models in terms of FLOPS to performance can be observed in Fig. 4.

4.3 Ablation studies

4.3.1 Ablation with varying values of γ

To evaluate the effectiveness of the *LoRaLin* layers, we conducted a series of experiments using the EdgeFace-XS model. These experiments involved varying the value of γ from 0.2 to 1, with increments of 0.2. All models were trained using the same configuration for 50 epochs. As a point of reference, we also compared these models with the *default* EdgeFace-XS model, which does not include the *LoRaLin* layer.

Figure 3 illustrates the changes in model parameters and FLOPs as the value of γ varies. It is observed that the parameters and FLOPs remain consistent with the EdgeFace-XS model when

TABLE 2: Performance evaluation (TAR) of the proposed EdgeFace model, along with various recent compact FR models, on 7 benchmarking datasets. The models are ordered based on the number of parameters. All decimal points are provided as reported in the respective works. Models are categorized based on the number of parameters into 2-5M parameters and < 2M parameters. For each benchmarking dataset, the best performance in each category is emboldened.

Model	MPARAMS	MFLOPs	LFW (%)	CA-LFW (%)	CP-LFW (%)	CFP-FP (%)	AgeDB-30 (%)	IJB-B (%)	IJB-C (%)
VarGFaceNet [12], [23]	5.0	1022	**99.85**	95.15	88.55	**98.50**	**98.15**	92.9	94.7
ShuffleFaceNet 2× [22]	4.5	1050	99.62	-	-	97.56	97.28	-	-
MixFaceNet-M [13]	3.95	626.1	99.68	-	-	-	97.05	91.55	93.42
ShuffleMixFaceNet-M [13]	3.95	626.1	99.60	-	-	-	96.98	91.47	91.47
MobileFaceNetV1 [12]	3.4	1100	99.4	94.47	87.17	95.8	96.4	92.0	93.9
ProxylessFaceNAS [12]	3.2	900	99.2	92.55	84.17	94.7	94.4	87.1	89.7
MixFaceNet-S [13]	3.07	451.7	99.6	-	-	-	96.63	90.17	92.30
ShuffleMixFaceNet-S [13]	3.07	451.7	99.58	-	-	-	97.05	90.94	93.08
ShuffleFaceNet 1.5x [12], [22]	2.6	577.5	99.7	95.05	88.50	96.9	97.3	92.3	94.3
MobileFaceNet [12]	2.0	933	99.7	95.2	89.22	96.9	97.6	92.8	94.7
PocketNetM-256 [14]	1.75	1099.15	99.58	95.63	90.03	**95.66**	**97.17**	90.74	92.70
PocketNetM-128 [14]	1.68	1099.02	99.65	95.67	90.00	95.07	96.78	90.63	92.63
MixFaceNet-XS [13]	1.04	161.9	99.60	-	-	-	95.85	88.48	90.73
ShuffleMixFaceNet-XS [13]	1.04	161.9	99.53	-	-	-	95.62	87.86	90.43
MobileFaceNets [21]	0.99	439.8	99.55	-	-	-	96.07	-	-
PocketNetS-256 [14]	0.99	587.24	99.66	95.50	88.93	93.34	96.35	89.31	91.33
PocketNetS-128 [14]	0.92	587.11	99.58	95.48	89.63	94.21	96.10	89.44	91.62
ShuffleFaceNet 0.5x [22]	0.5	66.9	99.23	-	-	92.59	93.22	-	-
EdgeFace - S ($\gamma = 0.5$) (**ours**)	3.65	306.11	99.78	**95.71**	**92.56**	95.81	96.93	**93.58**	**95.63**
EdgeFace - XS ($\gamma = 0.6$) (**ours**)	1.77	154	**99.73**	95.28	**91.82**	94.37	96.00	**92.67**	**94.85**

TABLE 3: The comparison of performance of the default and low rank $\gamma = 0.6$ variants of EdgeFace-XS. The % difference in verification accuracy as well as in parameters and FLOPS are provided in the table.

Model	LFW	IJB-B	IJB-C	MPARAMS	MFLOPS
EdgeFace-XS	99.8	92.65	94.75	2.24	196.9
$\gamma = 0.6$	99.7 (0.1% ↓)	92.24 (**0.4%**↓)	94.28 (**0.49%**↓)	**1.77 (21% ↓)**	**153.9 (22% ↓)**

γ is approximately 0.8. For values of γ below 0.8, there is a reduction in model parameters, FLOPs, and size.

To assess the performance of these models, we evaluated them using standard benchmarks. The results are presented in Table 4, which displays the performance across these benchmarks. The performance deteriorates as the value of γ decreases (Fig. 5). However, it is notable that the performance remains satisfactory up to $\gamma = 0.6$, beyond which it starts to decline more sharply.

Figure 6 demonstrates the performance changes of the models on the IJB-B and IJB-C datasets. In both cases, the proposed method achieves good performance up to $\gamma = 0.6$. Additionally, Table 3 provides the percentage points of performance degradation corresponding to the changes in model parameters and FLOPs for the IJB-C and IJB-B datasets. It can be seen that we can obtain around 20% savings in parameters and FLOPS with less than 0.5% drop in accuracy.

The results presented in Table 3 highlight that our approach achieves a significant improvement in parameter and FLOP efficiency while maintaining a minimal reduction in performance. This demonstrates the effectiveness of our approach in achieving a favorable trade-off between efficiency and performance.

4.3.2 Ablation studies with loss functions

In this section, we perform experiments with training the Edge-Face with different loss functions; specifically, we train the same

TABLE 4: The performance of ablation of **XS** variant of EdgeFace on different face recognition datasets. The ablation is performance with respect to the γ parameter. For top 5 rows, all values indicate verification accuracy (TAR) expressed as percentages; while for IJB-B and IJB-C datasets, the values refer to true positive rate (TPR) at false positive rate (FPR) of $1e - 4$.

Dataset	default	$\gamma = 0.2$	$\gamma = 0.4$	$\gamma = 0.6$	$\gamma = 0.8$	$\gamma = 1.0$
LFW	99.8	99.1	99.5	99.7	99.7	99.7
CA-LFW	95.7	92.7	94.7	95.3	95.4	95.6
CP-LFW	92.3	83.7	90.2	91.3	91.8	92.0
AgeDB-30	96.1	89.9	94.6	95.4	95.8	96.1
CFP-FP	95.0	87.0	93.6	94.5	94.9	95.1
IJB-B	92.65	22.97	89.39	92.24	92.80	93.24
IJB-C	94.75	25.13	91.42	94.28	95.01	95.02
MPARAMS	2.24	0.73	1.24	1.77	2.29	2.81
MFLOPS	196.9	63.6	107.9	153.9	198.4	244.4

network with ArcFace [4] and CosFace [5] loss functions. We use the XS variant of the EdgeFace with $\gamma = 0.6$ as selected from the previous section for this set of experiments. We retrained the same model with ArcFace and CosFace for 50 epochs with a batch size of 512. All the hyperparameters were kept the same for both cases for a fair comparison. The results from these experiments are shown in Table 5. In high-resolution benchmarks, ArcFace performs better on LFW, CP-LFW, and CFP-FP, while CosFace performs better on CA-LFW and AgeDB-30. In the evaluations with IJB-B and IJB-C, it can be seen that CosFace slightly performs better than ArcFace at an FPR of $1e - 4$, but ArcFace performs much better than CosFace on very low FPR regions ($1e - 6$).
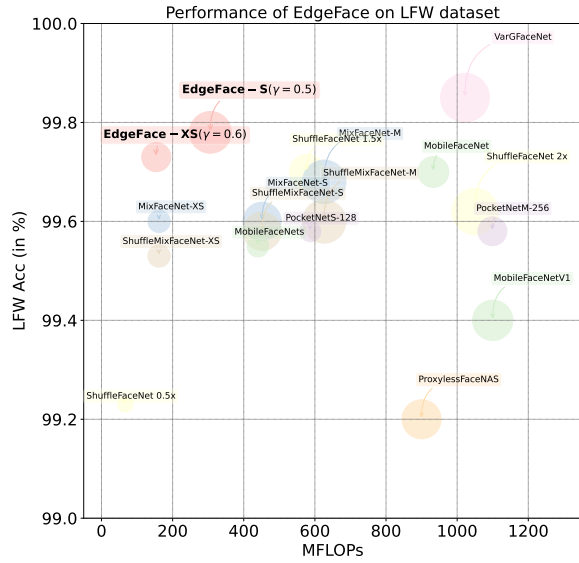
(a) LFW

(b) IJB-B
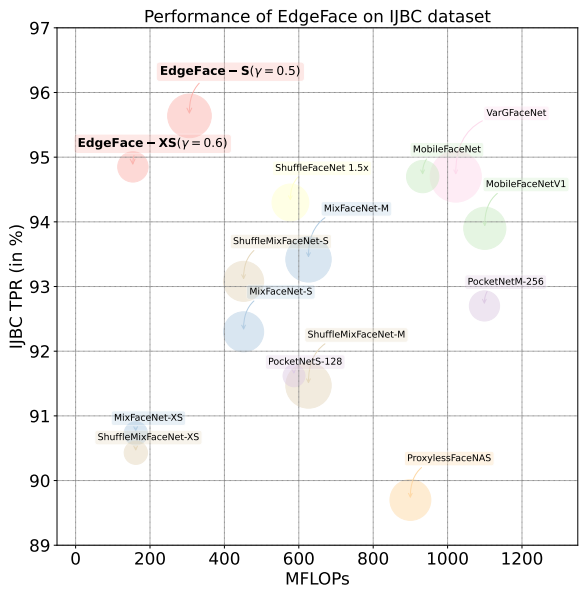
(c) IJB-C

Fig. 4: Performance comparison of different models on (a) LFW, (b) IJB-B, and (c) IJB-C datasets.
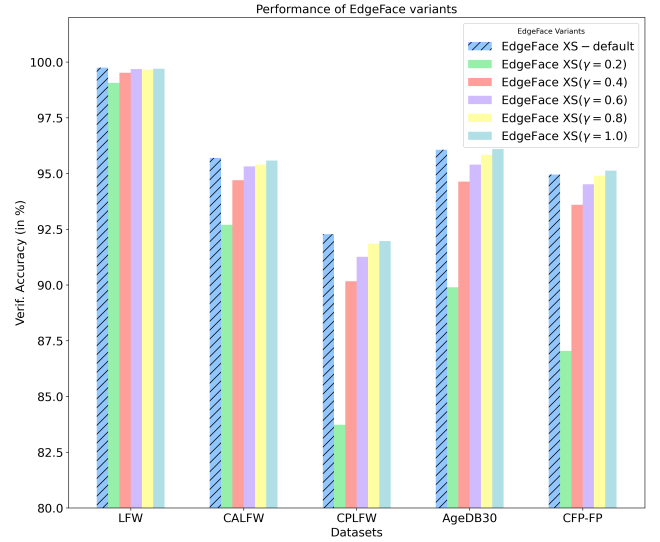
Fig. 5: Ablation study of EdgeFace with respect to low rank parameter (γ). The performance is evaluated as the verification accuracy of 'XS' variant on different face recognition datasets.
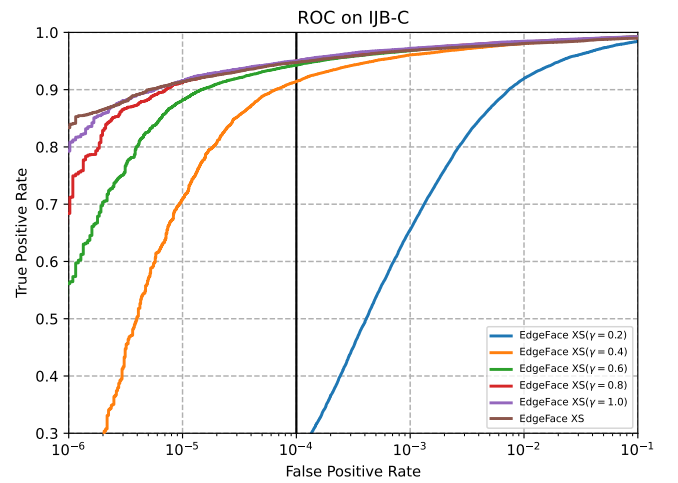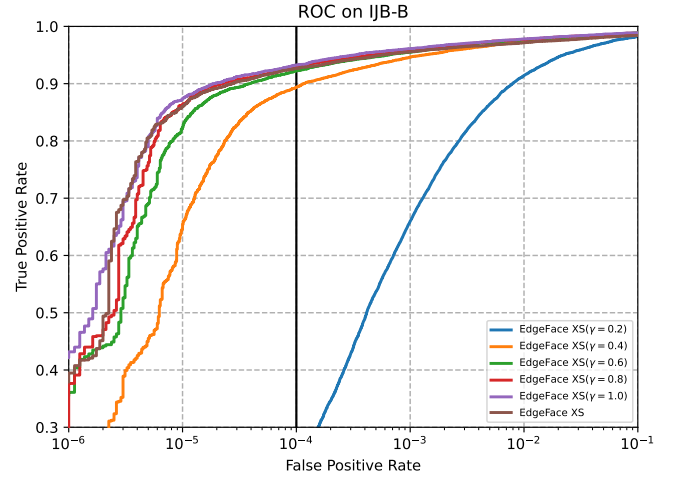
Fig. 6: Performance evolution for EdgeFace-XS across different Rank-ratio γ on IJB-B and IJB-C datasets.
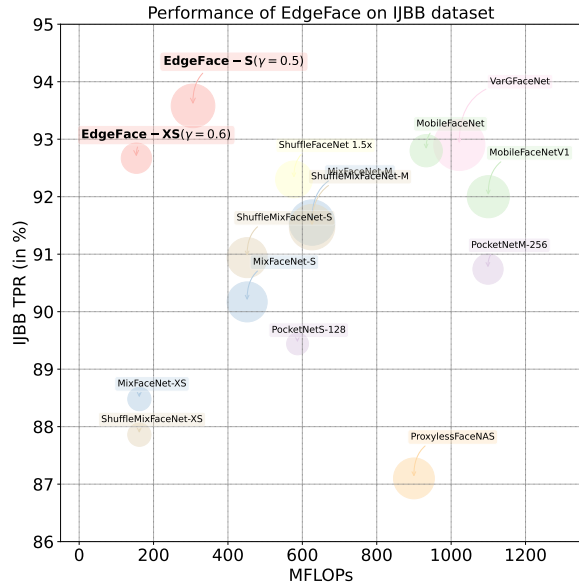
TABLE 5: The performance of ablation of **XS** variant of EdgeFace on different face recognition datasets. The ablation is performance with respect to loss function used in the training. For top 5 rows, all values indicate verification accuracy (TAR) expressed as percentages; while for IJB-B and IJB-C datasets, the values reported are true positive rate (TPR) at false positive rate (FPR) of $1e-4$ and $1e-6$.

| Dataset	ArcFace [4]	CosFace [5]
LFW	**99.65**	99.53
CA-LFW	95.07	**95.38**
CP-LFW	**91.50**	91.35
AgeDB-30	95.58	**95.60**
CFP-FP	**94.80**	94.26
IJB-B [FAR@ $1e-4$]	91.69	**92.41**
IJB-B [FAR@ $1e-6$]	**38.94**	30.20
IJB-C [FAR@ $1e-4$]	94.09	**94.39**
IJB-C [FAR@ $1e-6$]	**75.47**	55.46

4.4 IJCB'23 Efficient Face Recognition (EFaR) competition

The 2023 International Joint Conference on Biometrics (IJCB 2023) featured the Efficient Face Recognition Competition (EFaR), which aimed to stimulate advancements in efficient face recognition techniques. Rankings were determined using a composite score that factored in verification accuracy across a variety of benchmarks, as well as model deployability metrics such as floating-point operations and model size. The contest consists of two categories, differentiated by the model's parameter count. Teams have the flexibility to present two solutions per category. First category is dedicated to highly compact networks possessing fewer than 2 million parameters (denoted as <2 MP). The latter category addresses models that have parameters ranging between 2 and 5 million (2–5 MP). Entries are assessed and ranked within their respective categories. For both categories, a submission's rank is determined by its verification accuracy, computational intricacy, and memory footprint of the model.

For evaluation of face recognition performance the competition organizers utilized accuracy on LFW, CPLFW, CALFW, CFP-FP, and AgeDB30 datasets. Additionally, they also used IJB-C dataset, where they applied the true acceptance rate (TAR) at a false acceptance rate (FAR) of 10^{-4}, denoted as TAR at FAR=10^{-4}. A cumulative ranking across all benchmarks is calculated using the aggregated Borda counts from each dataset, which is then used as the final ranking. To assess a solution's deployability, the competition organizers factor in the model's compactness (measured by the number of parameters), its memory usage (indicated by the model size in MB), and its computational demand (denoted by M FLOPs). For all these aspects, a lower figure suggests superior deployability. Teams were required to provide these metrics (number of parameters, model size, FLOPs) for their entries. Rankings are formulated based on FLOPs and model dimensions. The final team ranking for a track is based on a weighted Borda count incorporating: (a) the standardized Borda count from the evaluated benchmarks, (b) the Borda count associated with the FLOPs metrics, and (c) the Borda count for the model size. The performance on benchmarks holds a weightage of 70%, while both the FLOPs and the model size each carry a weightage of 15%.

Teams had the liberty to present two entries for both tracks. Altogether, teams contributed 17 unique submissions. Out of

these, 9 solutions were for the 2-5 M parameter category, while the < 2 M parameter category received 8 solutions.

Two EdgeFace variants were submitted to each of the categories, they are:

For 2-5M parameters:

1) Idiap EdgeFace-S (γ=0.5) : This is the 'small' variant of EdgeFace model with γ=0.5 in the LoRaLin layers. The parameters are float32 precision.
2) Idiap EdgeFace-XS-Q : This is the 'extra-small' variant of EdgeFace model without LoRaLin layers. The linear layers of the model are quantized to 8-bit in this case.

For less than 2M parameters:

1) Idiap EdgeFace-XS (γ=0.6): This is the 'extra-small' variant of EdgeFace model with γ=0.6 in the LoRaLin layers. The parameters are float32 precision.
2) Idiap EdgeFace-XXS-Q: This is the 'extra-extra-small' variant of EdgeFace model without LoRaLin layers. The linear layers of the model are quantized to 8-bit in this case.

The results from the competition are tabulated in Table 6 (reproduced from the competition paper [26]). For the submitted solutions with < 2 million parameters, Idiap EdgeFace-XS (γ=0.6) **is the overall best performing model**. Idiap EdgeFace-XS(γ=0.6) also ranks first when considering the verification accuracy, including first rank on IJB-C. From the results of the experiment of the models with 2-5 million parameters, although it does not ranked first, highest ranked solution in terms of verification accuracy over the evaluated benchmarks is Idiap EdgeFace-S (γ=0.5). It also achieved the highest TAR at FAR=10^{-4} on the large IJB-C benchmark for all submitted solutions in this category. It can be seen that variants of our EdgeFace achieves superior performance in terms of other models in both categories. Also, for the highly compact networks possessing fewer than 2 million parameters, our model ranks first in the competition, showing the effectiveness of our approach. More details of the competition and evaluation results can be found in the competition paper [26].

5 DISCUSSIONS

Our experiments in Section 4.2 show that our model is very efficient and also achieves competitive recognition accuracy compared to SOTA lightweight models. Among seven benchmarking datasets used in our evaluation, EdgeFace achieves the best recognition performance for four different datasets in each of the categories of models with 2-5 M parameters and < 2M parameters. Achieving such a high recognition accuracy is more particularly impressive considering the computation of different models in terms of FLOPs in Table 2, where we observe that EdgeFace is the most efficient model in the first category (2-5 M parameters) and the second most efficient model in the second category (< 2M parameters).

Among our different benchmarking datasets, five datasets (i.e., LFW, CA-LFW, CP-LFW, CFP-LFW, and AgeDB-30) have higher-quality face images. The results in Table 2 show that our model achieves competitive performance with SOTA models on these benchmarking datasets. In contrast, IARPA Janus Benchmark datasets (i.e., IJB-B and IJB-C) include images with different qualities (including low-quality images) and are among the most challenging face recognition benchmarking datasets. According to the results in Table 2, EdgeFace outperforms all previous lightweight models in both categories of models with 2-5 M

TABLE 6: The results from the IJCB 2023 Efficient Face Recognition Competition reproduced from the competition paper [26], along with the baselines. This includes details on FLOPs, model size, and the number of parameters. The achieved rank for each submission is specified for every dataset. The collective Borda count and rank across all verification benchmarks can be found in the Accuracy column. Rankings are also provided for FLOPs and model size. The final consolidated rank is a weighted Borda count considering the achieved accuracy (70%), FLOPs ranking (15%), and model size (15%). Entries with 2-5M parameters are denoted as "2-5 MP", while those with < 2M parameters are labeled as "2 MP".

Model	Category	CPLFW Acc. [%]	Rank	CFP-FP Acc. [%]	Rank	CALFW Acc. [%]	Rank	AgeDB30 Acc. [%]	Rank	LFW Acc. [%]	Rank	IJB-C TAR@10^{-4}	Rank	Accuracy BC	Rank	FLOPS [M]	Rank	Model Size [MB]	Rank	Params [M]	Combined BC	Rank
ResNet-100 ElasticFace (Cos+) [8]	Baseline	93.23	-	98.73	-	96.18	-	98.28	-	99.80	-	96.65	-	-	-	24211.778	-	261.22	-	65.2	-	-
ResNet-100 ArcFace [4], [8]	Baseline	92.08	-	98.27	-	95.45	-	98.15	-	99.82	-	95.60	-	-	-	24211.778	-	261.22	-	65.2	-	-
ResNet-18 Q8-bit [56]	Baseline	89.48	-	94.46	-	95.72	-	97.03	-	99.63	-	93.56	-	-	-	-	-	24.10	-	24.0	-	-
ResNet-18 Q6-bit [56]	Baseline	88.37	-	93.23	-	95.58	-	96.55	-	99.52	-	93.03	-	-	-	-	-	18.10	-	24.0	-	-
MobileFaceNet Q8-bit [56]	Baseline	87.95	-	91.40	-	95.05	-	95.47	-	99.43	-	90.57	-	-	-	-	-	1.10	-	1.1	-	-
MobileFaceNet Q6-bit [56]	Baseline	84.57	-	87.69	-	93.30	-	93.03	-	98.87	-	83.13	-	-	-	-	-	0.79	-	1.1	-	-
PocketNetM-256 [14]	Baseline	90.03	-	95.66	-	95.63	-	97.17	-	99.58	-	92.70	-	-	-	1099.15	-	7.0	-	1.75	-	-
PocketNetM-128 [14]	Baseline	90.00	-	95.07	-	95.67	-	96.78	-	99.65	-	92.63	-	-	-	1099.02	-	6.74	-	1.68	-	-
Idiap EdgeFace-XS(γ=0.6)	2 MP	91.88	1	94.46	3	95.25	1	95.72	2	99.68	1	94.78	1	39	1	153.99	5	7.17	7	1.77	5.0	1
Idiap EdgeFace-XXS-Q	2 MP	89.65	5	93.11	5	94.68	4	93.77	4	99.50	4	92.97	4	22	4	94.72	3	1.73	2	1.24	3.92	4
MobileNet$_{v2\text{-}visteam}$	2 MP	82.90	8	89.39	7	88.63	6	83.65	7	98.58	6	51.60	7	7	7	86.20	2	3.38	3	1.70	2.17	6
SAM-MFaceNet eHWS V1	2 MP	91.35	2	95.01	1	95.10	2	95.57	3	99.55	3	93.07	2	35	2	236.75	7	4.4	5	1.10	4.68	2
SAM-MFaceNet eHWS V2	2 MP	91.28	3	94.73	2	94.90	3	95.72	1	99.65	2	93.06	3	33	3	236.75	6	4.4	6	1.10	4.45	3
SQ-HH	2 MP	84.13	6	91.60	6	87.17	7	84.28	6	98.07	7	63.36	6	10	6	1399.39	8	4.55	8	1.20	1.47	8
ShuffleNetv2x0.5	2 MP	83.48	7	87.76	8	86.00	8	80.33	8	97.72	8	38.57	8	1	8	17.14	1	0.77	1	0.17	1.92	7
ShuffleNetv2x1.5	2 MP	89.73	4	93.44	4	91.08	5	88.78	5	98.95	5	77.11	5	20	5	147.21	4	7.90	4	1.99	2.78	5
VarGFaceNet [23]	Baseline	88.55	-	98.50	-	95.15	-	98.15	-	99.85	-	94.70	-	-	-	1022	-	20.0	-	5.0	-	-
MobileFaceNetV1 [21]	Baseline	87.17	-	95.80	-	94.47	-	96.40	-	99.40	-	93.90	-	-	-	1100	-	13.6	-	3.4	-	-
MixFaceNet-M [13]	Baseline	-	-	-	-	-	-	97.05	-	99.68	-	93.42	-	-	-	626.1	-	15.8	-	3.95	-	-
MixFaceNet-S [13]	Baseline	-	-	-	-	-	-	96.63	-	99.60	-	92.30	-	-	-	451.7	-	12.28	-	3.07	-	-
MixFaceNet-XS [13]	Baseline	-	-	-	-	-	-	95.85	-	99.60	-	90.73	-	-	-	161.9	-	4.16	-	1.04	-	-
ShuffleMixFaceNet-M [13]	Baseline	-	-	-	-	-	-	96.98	-	99.60	-	91.47	-	-	-	626.1	-	15.8	-	3.95	-	-
ShuffleMixFaceNet-S [13]	Baseline	-	-	-	-	-	-	97.05	-	99.58	-	93.08	-	-	-	451.7	-	12.28	-	3.07	-	-
ShuffleFaceNet 1.5x [22]	Baseline	88.50	-	97.26	-	95.05	-	97.32	-	99.67	-	94.30	-	-	-	577.5	-	10.5	-	2.6	-	-
MobileFaceNet [21]	Baseline	89.22	-	96.90	-	95.20	-	97.60	-	99.70	-	94.70	-	-	-	933	-	4.50	-	2.0	-	-
EfficientNet$_{b0\text{-}visteam}$	2-5 MP	87.58	9	91.19	9	93.35	7	90.45	7	99.15	8	85.04	7	7	8	212.50	3	9.18	7	4.60	1.87	8
GhostFaceNetV1-1 KU	2-5 MP	91.70	4	95.97	2	95.77	1	97.20	1	99.62	3	94.93	2	37	2	215.65	4	8.17	4	4.09	5.52	1
GhostFaceNetV1-2 KU	2-5 MP	90.03	7	93.30	7	95.72	2	97.08	2	99.72	2	94.06	4	29	5	60.29	1	8.07	2	4.06	5.33	2
Idiap EdgeFace-S(γ=0.5)	2-5 MP	92.22	3	95.67	3	95.62	3	96.98	3	99.78	1	95.63	1	39	1	306.11	5	14.69	8	3.65	5.15	3
Idiap EdgeFace-XS-Q	2-5 MP	90.92	5	94.26	6	95.03	6	95.22	6	99.50	6	94.40	3	22	6	196.91	2	2.99	1	2.24	4.52	4
MB2-HH	2-5 MP	90.65	6	95.13	4	91.43	8	90.08	8	99.32	7	79.86	9	12	7	741.67	9	8.15	3	2.20	2.15	7
Modified-MobileFaceNet V1	2-5 MP	92.42	1	95.97	2	95.15	5	95.77	5	99.52	5	93.99	5	31	4	456.89	8	8.4	6	2.10	4.22	6
Modified-MobileFaceNet V2	2-5 MP	92.23	2	96.11	1	95.15	4	95.88	4	99.58	4	93.95	4	32	3	456.89	7	8.4	5	2.10	4.33	5
ShuffleNetv2x2.0	2-5 MP	89.27	8	92.71	8	90.88	9	88.08	9	99.03	9	80.92	8	3	9	310.92	6	20.00	9	4.97	0.65	9

parameters and < 2M parameters on these two datasets, which shows the superiority of our model for different quality of images.

Last but not least, we would like to highlight that, as mentioned in Section 4.4, variations of EdgeFace achieved best verification accuracy in both categories of < 2M parameters and 2-5M parameters in the recent efficient face recognition competition in IJCB 2023 [26] amongst all submissions, which used state-of-the-art techniques to train efficient face recognition models. In particular, for the category of models with less than two million parameters, our model not only was the first model in terms of verification accuracy, but also received the first place overall in terms of verification accuracy, computation complexity and memory footprint.

6 CONCLUSIONS

In this paper, we introduced EdgeFace , a highly efficient face recognition model that combines the strengths of CNN and Transformers. By leveraging efficient hybrid architecture and *LoRaLin* layers, the EdgeFace model achieves remarkable performance while maintaining low computational complexity. Our extensive experimental evaluations on various face recognition benchmarks, including LFW, AgeDB-30, CFP-FP, IJB-B, and IJB-C, demonstrate the effectiveness of EdgeFace . Our hybrid design strategy incorporates convolution and efficient self-attention-based encoders, providing an ideal balance between local and global information processing. This enables EdgeFace to achieve superior performance compared to state-of-the-art methods while maintaining low parameters and MAdds. The proposed EdgeFace model secured the first position in general ranking among models having less than 2M parameters in the IJCB 2023 Efficient Face

Recognition Competition [26]. In addition, in both categories of models with 2–5 M and $<$ 2 M parameters, variants of EdgeFace achieved the best verification accuracy in the IJCB 2023 competition. In summary, EdgeFace offers an efficient and highly accurate face recognition model tailored for edge devices. Knowledge distillation strategies can further enhance the model's performance, while exploring different quantization methods holds potential for improving storage and inference, which can be pursued in future research.

ACKNOWLEDGEMENTS

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[2] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning (ICML)*. PMLR, 2019, pp. 6105–6114.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, 2020.

[4] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4690–4699.

[5] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5265–5274.

[6] Y. Huang, Y. Wang, Y. Tai, X. Liu, P. Shen, S. Li, J. Li, and F. Huang, "Curricularface: adaptive curriculum learning loss for deep face recognition," in *proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 5901–5910.

[7] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "Magface: A universal representation for face recognition and quality assessment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 14 225–14 234.

[8] 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)*, 2022, pp. 1578–1587.

[9] M. Kim, A. K. Jain, and X. Liu, "Adaface: Quality adaptive margin for face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 18 750–18 759.

[10] P. Terhörst, M. Ihlefeld, M. Huber, N. Damer, F. Kirchbuchner, K. Raja, and A. Kuijper, "Qmagface: Simple and accurate quality-aware face recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 3484–3494.

[11] J. Deng, J. Guo, D. Zhang, Y. Deng, X. Lu, and S. Shi, "Lightweight face recognition challenge," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[12] Y. Martinez-Diaz, M. Nicolas-Diaz, H. Mendez-Vazquez, 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.

[13] F. Boutros, N. Damer, M. Fang, F. Kirchbuchner, and A. Kuijper, "Mixfacenets: Extremely efficient face recognition networks," in *2021 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2021, pp. 1–8.

[14] F. Boutros, P. Siebke, M. Klemt, 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.

[15] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[17] X. Zhang, X. Zhou, M. Lin, and J. Sun, "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.

[18] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.

[19] 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," *arXiv preprint arXiv:1907.05653*, 2019.

[20] M. Tan and Q. V. Le, "Mixconv: Mixed depthwise convolutional kernels," *arXiv preprint arXiv:1907.09595*, 2019.

[21] S. Chen, Y. Liu, X. Gao, and Z. Han, "Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices," in *Biometric Recognition: 13th Chinese Conference, CCBR 2018, Urumqi, China, August 11-12, 2018, Proceedings 13*. Springer, 2018, pp. 428–438.

[22] Y. Martinez-Diaz, 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[23] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[24] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," *ACM computing surveys (CSUR)*, vol. 54, no. 10s, pp. 1–41, 2022.

[25] M. Maaz, A. Shaker, H. Cholakkal, S. Khan, S. W. Zamir, R. M. Anwer, and F. Shahbaz Khan, "Edgenext: efficiently amalgamated cnn-transformer architecture for mobile vision applications," in *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VII*. Springer, 2023, pp. 3–20.

[26] J. N. Kolf, F. Boutros, J. Elliesen, M. Theuerkauf, N. Damer, M. Alansari, O. A. Hay, S. Alansari, S. Javed, N. Werghi *et al.*, "Efar 2023: Efficient face recognition competition," pp. 1–8, 2023.

[27] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.

[28] B. Wu, A. Wan, X. Yue, P. Jin, S. Zhao, N. Golmant, A. Gholaminejad, J. Gonzalez, and K. Keutzer, "Shift: A zero flop, zero parameter alternative to spatial convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9127–9135.

[29] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "Ghostnet: More features from cheap operations," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1580–1589.

[30] Y. Chen, X. Dai, D. Chen, M. Liu, X. Dong, L. Yuan, and Z. Liu, "Mobile-former: Bridging mobilenet and transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5270–5279.

[31] S. Mehta and M. Rastegari, "Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer," *arXiv preprint arXiv:2110.02178*, 2021.

[32] P. Zhang, F. Zhao, P. Liu, and M. Li, "Efficient lightweight attention network for face recognition," *IEEE Access*, vol. 10, pp. 31 740–31 750, 2022.

[33] C. N. Duong, K. G. Quach, I. Jalata, N. Le, and K. Luu, "Mobiface: A lightweight deep learning face recognition on mobile devices," in *2019 IEEE 10th international conference on biometrics theory, applications and systems (BTAS)*. IEEE, 2019, pp. 1–6.

[34] S. C. Hoo, H. Ibrahim, and S. A. Suandi, "Convfacenext: Lightweight networks for face recognition," *Mathematics*, vol. 10, no. 19, p. 3592, 2022.

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

[36] H. O. Shahreza, A. George, and S. Marcel, "Synthdistill: Face recognition with knowledge distillation from synthetic data," in *2023 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2023, pp. 1–8.

[37] M. Alansari, O. A. Hay, S. Javed, A. Shoufan, Y. Zweiri, and N. Werghi, "Ghostfacenets: Lightweight face recognition model from cheap operations," *IEEE Access*, 2023.

[38] K. Han, Y. Wang, Q. Zhang, W. Zhang, C. Xu, and T. Zhang, "Model rubik's cube: Twisting resolution, depth and width for tinynets," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19 353–19 364, 2020.

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[40] H. Zhang, W. Hu, and X. Wang, "Edgeformer: Improving lightweight convnets by learning from vision transformers," *arXiv preprint arXiv:2203.03952*, 2022.

[41] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," 2022.

[42] A. Ali, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek *et al.*, "Xcit: Cross-covariance image transformers," 2021.

[43] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[44] A. Aghajanyan, L. Zettlemoyer, and S. Gupta, "Intrinsic dimensionality explains the effectiveness of language model fine-tuning," *arXiv preprint arXiv:2012.13255*, 2020.

[45] Z. Zhu, G. Huang, J. Deng, Y. Ye, J. Huang, X. Chen, J. Zhu, T. Yang, J. Lu, D. Du *et al.*, "Webface260m: A benchmark unveiling the power of million-scale deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 492–10 502.

[46] "DALI: Data Loading Library for Deep Learning," https://github.com/NVIDIA/DALI, accessed on 2023-05-25.

[47] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[48] X. An, J. Deng, J. Guo, Z. Feng, X. Zhu, J. Yang, and T. Liu, "Killing two birds with one stone: Efficient and robust training of face recognition cnns by partial fc," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4042–4051.

[49] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[50] T. Zheng, W. Deng, and J. Hu, "Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments," *arXiv preprint arXiv:1708.08197*, 2017.

[51] 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*, vol. 5, no. 7, 2018.

[52] S. Sengupta, J.-C. Chen, C. 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)*. IEEE, 2016, pp. 1–9.

[53] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, "Agedb: the first manually collected, in-the-wild age database," in *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 51–59.

[54] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. Adams, T. Miller, N. Kalka, A. K. Jain, J. A. Duncan, K. Allen *et al.*, "Iarpa janus benchmark-b face dataset," in *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 90–98.

[55] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney *et al.*, "Iarpa janus benchmark-c: Face dataset and protocol," in *2018 international conference on biometrics (ICB)*. IEEE, 2018, pp. 158–165.

[56] F. Boutros, N. Damer, and A. Kuijper, "Quantface: Towards lightweight face recognition by synthetic data low-bit quantization," in *26th International Conference on Pattern Recognition, ICPR 2022, Montreal, QC, Canada, August 21-25, 2022*. IEEE, 2022, pp. 855–862. [Online]. Available: https://doi.org/10.1109/ICPR56361.2022.9955645

Hatef Otroshi Shahreza received the B.Sc. degree (Hons.) in electrical engineering from the University of Kashan, Iran, in 2016, and the M.Sc. degree in electrical engineering from the Sharif University of Technology, Iran, in 2018. He is currently pursuing the Ph.D. degree with the École Polytechnique Fédérale de Lausanne (EPFL), Switzerland, and is a Research Assistant with the Biometrics Security and Privacy Group, Idiap Research Institute, Switzerland, where he received H2020 Marie Skłodowska-Curie Fellowship (TReSPAsS-ETN) for his doctoral program. During his Ph.D., Hatef also experienced 6 months as a visiting scholar with the Biometrics and Internet Security Research Group at Hochschule Darmstadt, Germany. He is also the winner of the European Association for Biometrics (EAB) Research Award 2023. His research interests include deep learning, computer vision, and biometrics.

Ketan Kotwal received the M.Tech and Ph.D. degrees in electrical engineering from the Indian Institute of Technology Bombay (IIT Bombay, Mumbai, India). His current research interests include various topics in image processing, machine learning, and data processing. He has also been actively involved in technology consulting in relevant areas. He received the Excellence in Ph.D. Thesis Award from the IIT Bombay, and the Best Ph.D. Thesis Award from the Computer Society of India for his doctoral work. Dr. Kotwal is a co-author of research monograph "Hyperspectral Image Fusion" (Springer, US). At present, he is a member of the Biometrics Security and Privacy Group at the Idiap Research Institute.

Anjith George has received his Ph.D. and M-Tech degree from the Department of Electrical Engineering, Indian Institute of Technology (IIT) Kharagpur, India in 2012 and 2018 respectively. After Ph.D, he worked in Samsung Research Institute as a machine learning researcher. Currently, he is a research associate in the biometric security and privacy group at Idiap Research Institute, focusing on developing face recognition and presentation attack detection algorithms. His research interests are real-time signal and image processing, embedded systems, computer vision, machine learning with a special focus on Biometrics.

Christophe Ecabert received his Ph.D. and M.Sc degrees in electrical engineering from the École Polytechnique Fédérale de Lausanne (EPFL) in 2021 and 2014 respectively. He is currently a Post-Doctoral Assistant with the Biometrics Security and Privacy Group at Idiap Research Institute. His current work aims at exploring the use of synthetic data in the context of face recognition.

Sébastien Marcel heads the Biometrics Security and Privacy group at Idiap Research Institute (Switzerland) and conducts research on face recognition, speaker recognition, vein recognition, attack detection (presentation attacks, morphing attacks, deepfakes) and template protection. He received his Ph.D. degree in signal processing from Université de Rennes I in France (2000) at CNET, the research center of France Telecom (now Orange Labs). He is Professor at the University of Lausanne (School of Criminal Justice) and a lecturer at the École Polytechnique Fédérale de Lausanne. He is also the Director of the Swiss Center for Biometrics Research and Testing, which conducts certifications of biometric products.