

Lightweight Backbone Evaluation for Face Recognition with ArcFace

Carlos Ito, Atoany Fierro

Tecnologico de Monterrey

Abstract - Resource-constrained environments require lightweight face recognition models, but the true performance trade-offs across different efficient backbones are obscured by varying training protocols. This study provides a unified, head-to-head comparison of several leading lightweight architectures (including MobileFaceNet, ShuffleFaceNet, and MixFaceNet variants) and IResNet baselines. Using a standardized ArcFace training pipeline on VGGFace2, we assess accuracy on LFW, CFP-FP, AgeDB-30, IJB-B, and IJB-C. We conclude that MobileFaceNet achieves the most favorable efficiency-accuracy balance under unified conditions, while the smallest models demonstrate clear performance degradation. These results highlight the critical importance of architecture-specific optimization strategies for maximizing the potential of highly compressed networks.

Index Terms - Face Recognition, ArcFace, Lightweight Backbone Networks, Efficient CNNs, Deep Learning

1. INTRODUCTION

Face recognition has become one of the most prominent applications of deep learning in computer vision, achieving remarkable accuracy across large-scale verification and identification tasks. Deep convolutional neural networks (CNNs) have advanced the field by learning highly discriminative facial feature embeddings that exhibit strong intra-class compactness and inter-class separability. This progress has been driven primarily by the development of margin-based softmax loss functions such as SphereFace [6], CosFace [7], and ArcFace [1]. Among these, ArcFace introduced an additive angular margin penalty that directly optimizes the geodesic distance between features, resulting in significant improvements in recognition performance.

While deep CNNs have achieved state-of-the-art accuracy, this often comes at the cost of high computational complexity and memory requirements. Large-scale backbones such as ResNet-100 [8] and iResNet-100 [5] contain tens of millions of parameters, making them impractical for real-time or resource-

constrained scenarios such as mobile devices, drones, and embedded systems. This trade-off between accuracy and efficiency has motivated increasing research into lightweight architectures that preserve strong discriminative power while reducing computational overhead.

Recent studies have introduced efficient face recognition backbones specifically optimized for constrained environments. Architectures such as MobileFaceNet [2], ShuffleFaceNet [3], and MixFaceNet [4] employ techniques like depthwise separable convolutions, multi-scale kernel operations, and channel attention to achieve high performance with significantly fewer parameters and floating-point operations (FLOPs). These compact models demonstrate that efficient architectures can approach the accuracy of large-scale networks when trained with robust margin-based loss functions like ArcFace. However, a systematic evaluation of these lightweight backbones under identical ArcFace-based training conditions is still lacking.

This work addresses this gap by conducting a comprehensive comparative study of lightweight backbone architectures trained using ArcFace loss. All models are trained on the VGGFace2 dataset [9] under identical conditions to ensure fair comparison. Performance is evaluated on standard benchmarks including LFW [10], CFP-FP [11], AgeDB-30 [12], IJB-B [13], and IJB-C [14]. The evaluation aims to analyze the trade-offs between model efficiency and recognition accuracy, providing practical insights for selecting suitable architectures in real-world, resource-constrained applications.

The main contributions of this paper are as follows:

Comprehensive Evaluation: A unified comparison of multiple lightweight face recognition backbones trained with ArcFace under identical conditions is presented.

Efficiency–Accuracy Analysis: Trade-offs between model complexity, parameter count, and recognition performance across different architectures are assessed.

Practical Insights: Models that achieve an optimal balance between efficiency and accuracy, suitable for deployment in mobile and embedded environments, are identified.

2. RELATED WORK

2.1 Deep Learning for Face Recognition

Deep convolutional neural networks have revolutionized face recognition, replacing traditional hand-crafted feature methods such as Local Binary Patterns (LBP) [15] and Fisherfaces [16]. Early CNN-based approaches like DeepFace [17] and DeepID [18] demonstrated the potential of deep representations for face verification, paving the way for large-scale, high-accuracy systems.

The introduction of margin-based softmax loss functions further enhanced discriminative power by improving intra-class compactness and inter-class separability. SphereFace [6] introduced an angular margin, CosFace [7] added a cosine margin, and ArcFace [1] refined this idea with an additive angular margin directly in normalized space. ArcFace has since become a standard in modern face recognition

pipelines due to its simplicity and superior generalization.

2.2 Lightweight Architectures for Face Recognition

Despite their success, high-performing CNN-based systems typically rely on heavy backbones such as ResNet-100 [8] or iResNet-100 [5], which are unsuitable for real-time deployment in constrained environments. To address this limitation, research has focused on lightweight and efficient architectures that maintain high accuracy with significantly fewer parameters and operations.

The development of compact models was initially driven by efficient general-purpose CNNs such as MobileNet [19], ShuffleNet [20], and MixNet [21], which introduced optimization techniques including depthwise separable convolutions and channel shuffling. These concepts were subsequently adapted for face recognition tasks, resulting in architectures such as MobileFaceNet [2], ShuffleFaceNet [3], MixFaceNet [4], and ShuffleMixFaceNet [4]. These models integrate various efficiency mechanisms with discriminative loss functions (e.g., ArcFace) to generate compact yet highly discriminative facial embeddings.

Each of these architectures represents a unique approach to balancing accuracy and computational efficiency. MobileFaceNet employs a depthwise separable convolutional design optimized for facial embeddings [2]. ShuffleFaceNet applies channel shuffling and grouped convolutions to achieve fast inference [3]. MixFaceNet leverages multi-scale kernels to enhance feature diversity with minimal computational overhead [4]. Finally, ShuffleMixFaceNet combines design principles from ShuffleNet and MixNet to achieve an improved trade-off between performance and efficiency [4].

While these lightweight networks have demonstrated strong results, previous studies often differ in training datasets, loss functions, and evaluation protocols, making direct comparison challenging. This paper contributes a unified evaluation framework to systematically assess these architectures under consistent ArcFace-based training and testing settings.

3. APPROACH

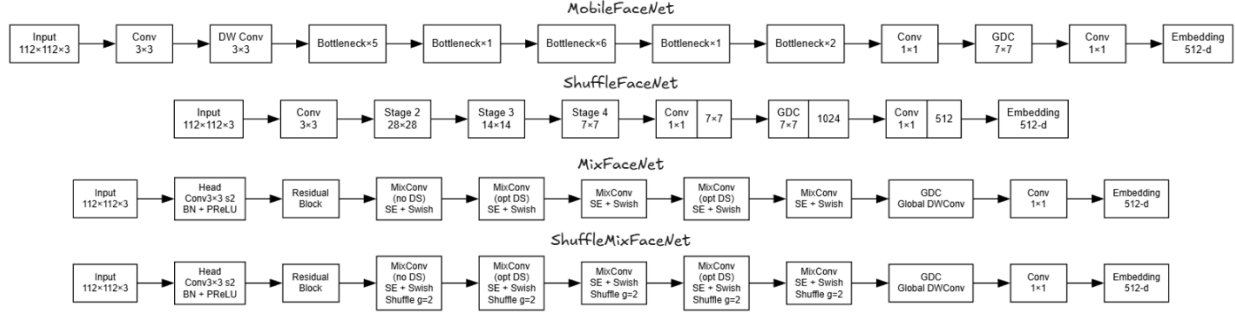


Fig. 1. Block diagrams of the lightweight backbone architectures evaluated in this study. The figure illustrates the overall computational pipelines of MobileFaceNet, ShuffleFaceNet, MixFaceNet, and ShuffleMixFaceNet. Each diagram highlights the major structural components used by the respective networks.

3.1 Overview

To ensure a consistent and fair comparison across lightweight face recognition architectures, all experiments were conducted under a unified training and evaluation pipeline based on the official ArcFace implementation [1]. This included standardized face alignment and resizing, but notably, no data augmentation was applied. The training framework, hyperparameters, datasets, and evaluation protocols were kept identical for all models, with the only difference being the network backbone. This setup allows a direct assessment of the trade-offs between model size, computational cost, and recognition performance.

3.2 Model Architectures

Figure 1 provides a high-level overview of the four lightweight backbone architectures evaluated in this study (MobileFaceNet, ShuffleFaceNet, MixFaceNet, and ShuffleMixFaceNet) highlighting the structural differences that define their computational and representational characteristics.

A diverse set of lightweight backbones was evaluated, each representing a distinct design philosophy for efficient convolutional networks. The evaluated architectures include ShuffleFaceNet [3] with width multipliers of 0.5×, 1×, 1.5×, and 2×; MobileFaceNet [2] following the original design proposed by Chen et al.; MixFaceNet [4] in XS, S, and M configurations; and ShuffleMixFaceNet [4] in XS, S, and M variants. Additionally, IResNet architectures [5] with 18, 50, and 100 layers were included as non-lightweight baseline models for reference.

All backbones were implemented based on their respective papers [2, 3, 4, 5] and verified against official or reference repositories. Minor modifications were made to ensure compatibility with mixed-precision (FP16) training and the ArcFace loss formulation [1]. All embeddings were projected to a 512-dimensional feature space.

3.3 Training Pipeline

The training followed the standard ArcFace configuration using the VGGFace2 dataset [9], containing approximately 3.1M images across 8,631 identities. Each image was aligned using five facial landmarks and resized to 112×112 resolution using the same preprocessing pipeline as the original ArcFace implementation [1]. No further data augmentation techniques (such as random cropping, horizontal flipping, or color jittering) were applied. All models were trained from scratch without pretraining on ImageNet or other datasets.

Training was performed using PyTorch with the Automatic Mixed Precision (AMP) module for FP16 optimization. A single NVIDIA RTX 4070 GPU was used for all experiments. The training ran for 20 epochs with a batch size of 128. The optimizer was SGD with momentum 0.9 and weight decay 5×10^{-4} . The initial learning rate was 0.1, scheduled by a quadratic (polynomial) decay policy without warm-up.

The ArcFace hyperparameters were kept constant across models, with a scale parameter $s = 64$ and an angular margin $m = 0.5$, corresponding to a margin_list = (1.0, 0.5, 0.0) configuration in the official codebase [1].

3.4 Evaluation Protocol

Each model was evaluated under identical conditions on the standard verification benchmarks LFW [10], CFP-FP [11], and AgeDB-30 [12], using the ArcFace evaluation scripts [1]. Validation on these datasets was performed automatically every 2,000 training steps to monitor convergence and generalization. After training completion, the models were further tested on the large-scale benchmarks IJB-B [13] and IJB-C [14] for comprehensive evaluation of verification and identification performance. Performance metrics include verification accuracy and TAR@FAR=1e-1 to 1e-6, following the ArcFace evaluation protocol [1].

3.5 Computational Analysis

To quantify efficiency, FLOPs and parameter counts for all models were computed using the ptflops library [22] with an input resolution of 112×112. This provided a consistent measure of computational complexity across networks.

4. RESULTS

This section evaluates all backbones using the unified ArcFace training pipeline [1]. Each subsection references its associated figures, which illustrate convergence behavior, loss dynamics, ROC characteristics, and computational efficiency.

4.1 Convergence on LFW, CFP-FP, and AgeDB-30

Figures 2, 3, and 4 illustrate the convergence behavior on LFW [10], CFP-FP [11], and AgeDB-30 [12]. As expected, all models rapidly saturate on the relatively easy LFW benchmark most surpassing 0.99 accuracy but CFP-FP and AgeDB-30 expose much clearer separations in model capacity.

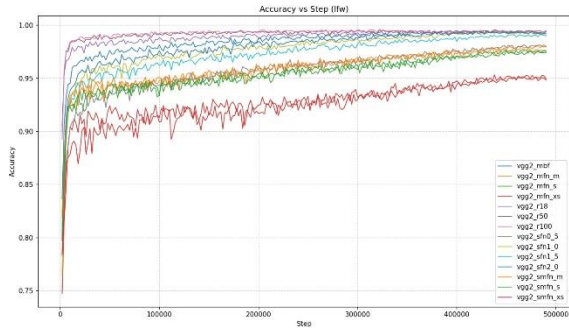


Fig. 2. LFW Convergence. Accuracy vs. Steps on the LFW dataset [10].

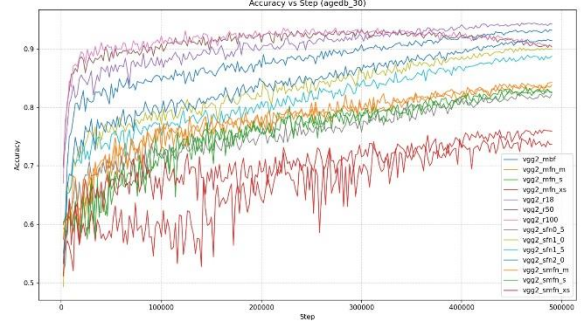


Fig. 3. AgeDB-30 Convergence. Accuracy vs. Steps on the AgeDB-30 dataset [12].

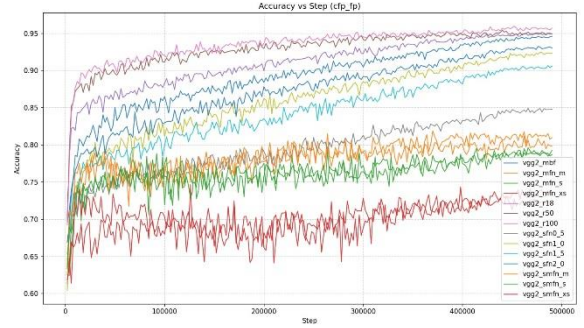


Fig. 4. CFP-FP Convergence. Accuracy vs. Steps on the CFP-FP dataset [11].

Among the lightweight backbones, MobileFaceNet-based variants show the strongest overall results: mbf reaches 0.946 on CFP-FP and 0.932 on AgeDB-30, outperforming the MobileFaceNet (m/s/xs) series, whose accuracy drops sharply as the architectures shrink. A similar trend appears in the ShuffleMixFaceNet models, where the smallest smfn_xs falls to 0.734 on CFP-FP and 0.759 on AgeDB-30.

The ShuffleFaceNet family sits between these groups, with sfn2.0 providing the most balanced lightweight performance (0.930 CFP-FP, 0.914 AgeDB-30). Smaller ShuffleFaceNet variants predictably show reduced robustness to pose and age variation.

The IResNet models [5] remain the most reliable across all benchmarks. r100 attains the best CFP-FP score (0.957), while r18 delivers the strongest AgeDB-30 accuracy (0.943). All three IResNet backbones r18, r50, and r100 remain clustered at the top on LFW and maintain a clear margin over lightweight architectures, reaffirming the advantage of deeper, more expressive networks for challenging face-verification conditions.

4.2 Training Loss Behavior

Figure 5 presents the training loss trajectories for all evaluated architectures, and the final losses reveal a clear separation between the residual [8] and lightweight model families [2, 3, 4]. The IResNet series [5] converges to by far the lowest losses, with r100 reaching 0.45, r50 settling at 0.69, and r18 finishing at 4.26. These values underscore the strong optimization capacity of deeper residual backbones under ArcFace supervision [1].

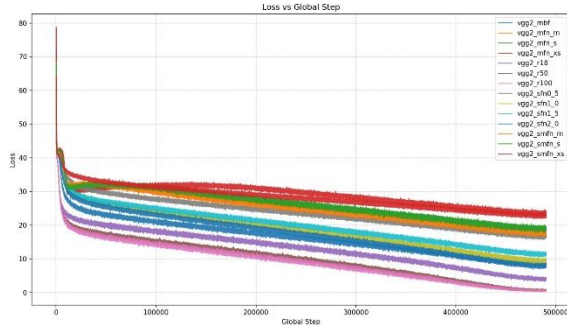


Fig. 5. Loss Convergence. Training Loss vs. Global Steps for all evaluated backbones.

Among the lightweight models, the overall trend is noticeably less favorable. Within this group, the ShuffleFaceNet family [3] shows comparatively better convergence, with sfm2.0 achieving the lowest loss (7.49) and sfm1.0 and sfm1.5 following behind. MobileFaceNet (mbf) [2] also converges more effectively than other compact variants, though still far above the residual baselines. In contrast, the MobileFaceNet-s/xs models and the ShuffleMixFaceNet series [4] display substantially higher final losses exceeding 17–23 in the smallest configurations consistent with their weaker verification accuracy and illustrating the difficulty of optimizing aggressively compressed architectures.

Across all networks, the training curves themselves remain smooth and stable, indicating a robust training pipeline and confirming that the ArcFace loss [1] provides consistent optimization behavior even across a wide range of model capacities.

4.3 Evaluation on IJB-B and IJB-C

Figure 6 and Figure 7 present ROC results on IJB-B [13] and IJB-C [14], both of which impose substantial variation in pose, illumination, and image quality.

These conditions amplify the performance gaps already visible in the earlier verification benchmarks.

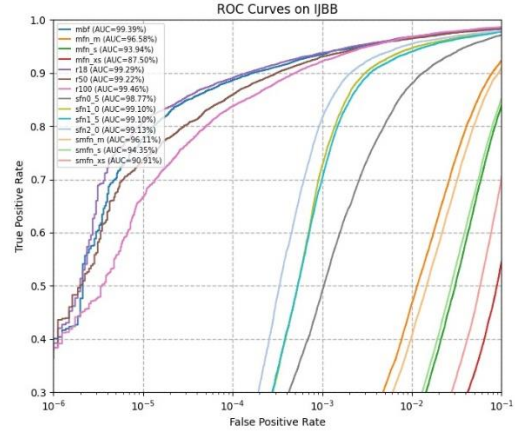


Fig. 6. Verification Performance. ROC curves evaluated on the IJB-B benchmark [13].

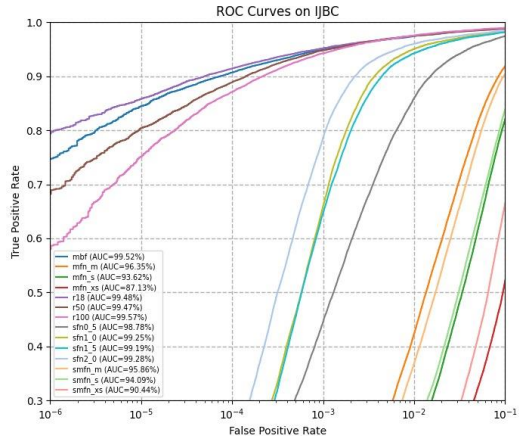


Fig. 7. Verification Performance. ROC curves evaluated on the IJB-C benchmark [14].

The IResNet family [5] remains the strongest overall. All three variants achieve AUC values above 99.2% on IJB-B and 99.4% on IJB-C. Their performance at ultra-low false-accept rates is particularly notable: on IJB-C, r18 reaches a remarkable 79.70% TPR@1e-6, outperforming r50 and r100 despite being the shallowest model. Across both datasets, the residual backbones maintain high TPRs across all operating points, reaffirming their robustness in security-critical scenarios.

Within the lightweight architectures, MobileFaceNet (mbf) [2] stands out as the only compact model that remains competitive at the strictest thresholds. On IJB-

B it achieves 40.09% TPR@1e-6, and on IJB-C it reaches 74.78%, far above all other lightweight models and in some settings approaching residual-network behavior. Its AUC values 99.39% (IJB-B) and 99.52% (IJB-C) also highlight its strong generalization under unconstrained conditions.

Other lightweight backbones, including the MobileFaceNet-m/s/xs series [2] and the ShuffleMixFaceNet variants [4], experience severe degradation at low FAR. Their TPR@1e-6 values hover near zero, and their ROC curves rapidly diverge from those of MobileFaceNet and the residual networks. ShuffleFaceNet models [3] fare moderately better with AUC values up to 99.13% on IJB-B and 99.28% on IJB-C but still struggle at the most stringent operating points, where TPRs often fall below 1%.

These trends emphasize the difficulty that extremely compressed architectures face when deployed in highly unconstrained or security-sensitive environments and highlight the clear advantage of both residual backbones and the more capable lightweight design of MobileFaceNet.

4.4 Computational Efficiency: FLOPs and Parameter Count

Table 1 summarizes the computational characteristics of all evaluated backbones, which span a wide range of FLOPs and parameter counts, reflecting diverse deployment targets and resource constraints.

Model	GFLOPs	Params (M)
mbf	0.904	2.060
mfn_m	0.020	1.924
mfn_s	0.018	1.602
mfn_xs	0.007	1.038
r100	24.255	65.156
r50	12.667	43.590
r18	5.252	24.025
sfn0_5	0.023	0.455
sfn1_0	0.212	2.889
sfn1_5	0.098	1.524
sfn2_0	0.396	5.918
smfn_m	0.020	1.924
smfn_s	0.018	1.602
smfn_xs	0.007	1.038

Table 1: Model Efficiency and Complexity. Comparison of the models evaluated, detailing the number of Parameters in millions and GFLOPs required per inference.

The IResNet models [5] anchor the high-accuracy, high-cost end of the spectrum. r100 requires 24.26 GFLOPs and 65.16M parameters, followed by r50 at 12.67 GFLOPs / 43.59M and r18 at 5.25 GFLOPs / 24.03M. These models demand significantly more computation but consistently deliver the strongest recognition performance, particularly in the strictest low-FAR operating conditions.

MobileFaceNet (mbf) [2] provides a notably more efficient balance, requiring only 0.904 GFLOPs and 2.06M parameters. Despite its compact footprint, it remains competitive with the residual models across multiple benchmarks, making it well suited for real-time or edge deployments where both efficiency and robustness are important.

The MixFaceNet (mfn) [4] and ShuffleMixFaceNet (smfn) [4] families are the most computationally lightweight architectures in the study. Their FLOPs range from 0.007–0.020, with parameter counts between 1.04–1.92M. This makes them ideal for severely resource-limited environments, though their performance on challenging benchmarks like IJB-B [13] and IJB-C [14] reflects the substantial trade-offs introduced by aggressive compression.

ShuffleFaceNet (sfn) [3] spans a broader computational range, offering flexible trade-offs. The smallest variant, sfn0.5, uses only 0.023 GFLOPs and 0.455M parameters, while the larger sfn2.0 reaches 0.396 GFLOPs and 5.92M parameters. Their accuracy generally scales with capacity, making this family adaptable across different deployment budgets while remaining competitive within the lightweight category.

Overall, the efficiency–accuracy landscape forms a clear hierarchy: the IResNet models define the upper bound on recognition capability [5]; MobileFaceNet delivers the strongest lightweight balance of accuracy and cost [2]; and the smallest MixFaceNet, ShuffleMixFaceNet, and ShuffleFaceNet variants target scenarios where minimizing compute is the overriding priority [4, 3].

5. DISCUSSION

The unified training pipeline adopted in this study enables a controlled comparison of lightweight and residual backbones under identical conditions, allowing performance differences to be attributed

primarily to architectural design rather than training disparities. This standardization, however, also reveals important nuances regarding how different backbones respond to fixed hyperparameters and optimization settings.

First, the results indicate that residual architectures such as IResNet-50 and IResNet-100 [5] benefit more readily from the chosen hyperparameters. These models exhibit smoother convergence, lower final training losses, and superior generalization across all benchmarks. Their deep residual pathways and high representational capacity make them more resilient to suboptimal training choices, allowing them to fully exploit the ArcFace loss formulation [1] even under relatively short training schedules.

In contrast, lightweight architectures particularly the smallest variants of MixFaceNet [4], ShuffleMixFaceNet [4], and ShuffleFaceNet [3] demonstrate higher sensitivity to the fixed pipeline. Their reduced capacity limits their tolerance to aggressive margins, high learning rates, and short training durations. Therefore, some models converge to higher losses or underfit the training set, resulting in weaker generalization to challenging benchmarks such as CFP-FP and IJB-C. These findings underscore that lightweight models often require more tailored optimization strategies, such as adjusted margin parameters, lower initial learning rates, or extended training schedules, to reach their full potential.

Despite these limitations, MobileFaceNet [2] stands out as a robust lightweight architecture that generalizes well even under a uniform training protocol. Its strong results on IJB-B and IJB-C suggest that its architectural design particularly its bottleneck structure and balance between depth and width is better aligned with the ArcFace training dynamics compared to other compact models.

The evaluation also highlights that the accuracy–efficiency trade-off is not uniform across families. Although ShuffleFaceNet [3] exhibits a clear performance gradient with increasing width multipliers, it still underperforms MobileFaceNet [2] at equivalent computational budgets. ShuffleMixFaceNet [4] and MixFaceNet [4] deliver extremely low computational cost but experience steep accuracy degradation under the shared hyperparameter

schedule, suggesting that highly compressed architectures may require specialized training strategies.

Overall, the findings emphasize a dual conclusion: using a unified training pipeline is essential for fair comparison, but it also exposes architectural biases in how different models respond to identical hyperparameters. Thus, while the reported results accurately reflect relative performance under controlled conditions, they may underestimate the potential of certain lightweight networks when tuned individually for their architectural characteristics.

6. CONCLUSION

This study presented a systematic evaluation of lightweight face recognition backbones trained with ArcFace [1] under a fully unified pipeline. By standardizing dataset preparation, hyperparameters, optimization settings, and evaluation protocols, the comparison isolates the influence of architectural design on recognition performance. The results show a clear hierarchy: residual models [5] provide the strongest overall accuracy and robustness, MobileFaceNet [2] offers the most favorable balance between efficiency and performance, and extremely lightweight variants of MixFaceNet [4], ShuffleMixFaceNet [4], and ShuffleFaceNet [3] target ultra-constrained environments where computational cost dominates over absolute accuracy.

A key insight from this study is that the use of identical hyperparameters across all architectures has a measurable impact on model performance. While essential for fairness, this uniformity disproportionately favors architectures with high representational capacity and strong gradient propagation, such as deep residual networks. Conversely, the smallest lightweight models although designed for efficiency, may require architecture-specific tuning to fully leverage margin-based objectives like ArcFace. As a result, the present results reflect each model’s performance under controlled, equal conditions rather than its absolute best-case performance.

Future work may extend this analysis by exploring backbone-specific optimization schedules, margin adjustments, or longer training durations to determine

how much additional performance headroom exists for each architecture. Nonetheless, the findings presented here provide a practical and evidence-based reference for selecting appropriate face recognition backbones based on the computational constraints and accuracy requirements of real-world deployment scenarios.

REFERENCES

- [1] 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.
- [2] S. Chen, Y. Liu, X. Gao, and Z. Han, "Mobilefacenet: Efficient cnns for accurate real-time face verification on mobile devices," in *Biometric Recognition: 13th Chinese Conference, CCBIR 2018*, 2018.
- [3] Y. Martinez-Diaz, L. S. Luevano, H. Mendez-Vazquez, M. NicolasDiaz, 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.
- [4] F. Boutros, N. Damer, M. Fang, F. Kirchbuchner, and A. Kuijper, "Mixfacenet: Extremely efficient face recognition networks," in *2021 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2021.
- [5] I. C. Duta, L. Liu, F. Zhu, and L. Shao, "Improved residual networks for image and video recognition," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021.
- [6] W. Liu, Y. Wen, Z. Yu et al., "SphereFace: Deep hypersphere embedding for face recognition," in *Proc. CVPR*, 2017.
- [7] 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.
- [8] 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.
- [9] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. BMVC*, vol. 1, 2015.
- [10] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," in *Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition*, 2008.
- [11] 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.
- [12] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, "Agedb: the first manually collected, in-the-wild age database," in *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017.
- [13] C. Whitlam et al., "IARPA Janus Benchmark-B Face Dataset," in *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017.
- [14] B. Maze et al., "IARPA Janus Benchmark-C: Face dataset and protocol," in *2018 international conference on biometrics (ICB)*. IEEE, 2018.
- [15] Z. Sedaghatjoo, H. Hosseinzadeh, & B. S. Bigham, "Local binary pattern (lbp) optimization for feature extraction," *arXiv preprint arXiv:2407.18665*, 2024.
- [16] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, 1997.
- [17] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. CVPR*, 2014.

[18] D. Yi, Z. Lei, S. Liao, and S. Z. Li, “Learning face representation from scratch,” arXiv preprint arXiv:1411.7923, 2014.

[19] 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.

[20] 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.

[21] M. Tan and Q. V. Le, “Mixconv: Mixed depthwise convolutional kernels,” arXiv preprint arXiv:1907.09595, 2019.

[22] V. Sovrasov, “ptflops: a flops counting tool for neural networks in pytorch framework,” Online Resource, 2023.