Deep Boosting Multi-Modal Ensemble Face Recognition with Sample-Level Weighting

Sahar Rahimi Malakshan, Mohammad Saeed Ebrahimi Saadabadi,
Nima Najafzadeh, and Nasser M. Nasrabadi
sr00033, me00018, nn00008@mix.wvu.edu, nasser.nasrabadi@mail.wvu.edu

Abstract

Deep convolutional neural networks have achieved remarkable success in face recognition (FR), partly due to the abundant data availability. However, the current training benchmarks exhibit an imbalanced quality distribution; most images are of high quality. This poses issues for generalization on hard samples since they are underrepresented during training. In this work, we employ the multi-model boosting technique to deal with this issue. Inspired by the well-known AdaBoost, we propose a sample-level weighting approach to incorporate the importance of different samples into the FR loss. Individual models of the proposed framework are experts at distinct levels of sample hardness. Therefore, the combination of models leads to a robust feature extractor without losing the discriminability on the easy samples. Also, for incorporating the sample hardness into the training criterion, we analytically show the effect of sample mining on the important aspects of current angular margin loss functions, i.e., margin and scale. The proposed method shows superior performance in comparison with the state-of-the-art algorithms in extensive experiments on the CFP-FP, LFW, CPLFW, CALFW, AgeDB, TinyFace, IJB-B, and IJB-C evaluation datasets.

1. Introduction

The classical Face Recognition (FR) frameworks are based on extracting hand-crafted features [1, 41]. Nuisance factors such as head pose, resolution, blur, occlusion, and illumination variance in expression severely affect FR performance [1]. Since the advent of deep Convolutional Neural Networks (CNN) and the introduction of large-scale FR datasets, deep CNN-based FR have gained popularity [42]. The introduction of ResNet architecture and seminal works of [3, 15, 41] have revolutionized FR into the challenge of finding robust and suitable loss functions [16, 7]. The general goal is to force the model to learn discriminative representations with a minimal intra-class distance and a maximal inter-class disparity [30]. As a metric learning task, there are two main approaches to designing the loss function: 1) multi-class and 2) pair-wise supervision.

Figure 1: Illustrating the resolution and quality disparity between training (WebFace4M) and testing benchmarks (TinyFace).

Due to the availability of large-scale labeled datasets, the regular choice of training objective is Softmax. However, as an open-set recognition task, the discriminability of feature representation matters [53]. Despite being separable, the representation yielded through Softmax loss exhibits poor discriminability [30, 53]. The pioneering works in [42, 54] improved discriminability by using deep metric learning loss functions. Most recently, using angular distance (instead of Euclidean) and angular penalty in the Softmax improved the discriminability power of feature representations [50, 30, 51, 7]. In the state-of-the-art (SOTA) deep FR framework, a model is trained using angular margin objective until convergence [30, 51, 7]. However, current SOTA performance demonstrates limited generalization on low-quality and Low-Resolution (LR) inputs, such as images captured by surveillance cameras or captured from long ranges [57].

Large-scale datasets such as WebFce260M [63], or MS-Celeb-1M [12], mainly contain high-resolution (HR) instances that have significant statistical disparity from real-world surveillance images, as shown in Fig. 1 [18]. In other words, difficult samples, i.e., including LR instances, are under-represented. However, an implicit assumption in conventional angular margin methods is that all samples are equally important [26, 24]. Consequently, a manually se-

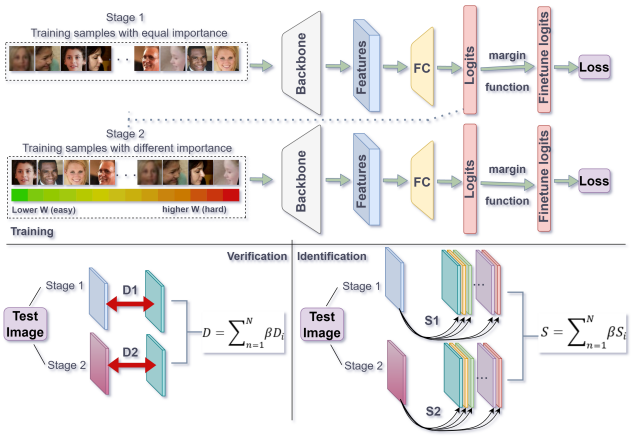

Figure 2: Schematic diagram of the proposed method, based on a CNN transfer learning for $K = 1$ and $K = 2$.

lected margin squeezes all intra-class variations uniformly, which can be sub-optimal [26, 8, 60, 51, 7, 38]. That being said, leveraging FR frameworks with sample hardness has attracted the researcher's attention to increase the model's discriminability power [33].

Methods have been developed to de-emphasize or over-emphasize the instances based on selected characteristics as the proxy of sample hardness, such as feature norm or uncertainty [52, 33, 45]. However, the performance improvement is inconsistent as they ignore hard samples [26, 24]. Also, convergence is not guaranteed due to the complex essence of the FR problem when solely trained on hard instances [26]. Data augmentation can enhance the frequency of difficult images and the diversity of training benchmarks [47, 24]. However, due to the tied angular margin in conventional FR methods, they suffer from convergence problems and cannot fit well with data augmentations [59]. Methods have been proposed to adaptively tune the margin based on the difficulty of the sample [58, 29, 33]. Although promising improvements have been gained via combining augmentations and adaptive margin, the performance still needs to be improved in testing benchmarks with a large distribution gap, implying that augmented samples cannot mimic the actual distribution of in-the-wild images.

An optimal FR model must generalize across different data distributions, such as off-angle, LR, and distorted images, to accommodate distribution shifts from training to real-world applications. Ensemble learning is an effective method that trains a set of models on the whole or subsets of a dataset to mitigate the sensitivity to the distribution shift in the data [36]. Most recently, combining boosting strategies with CNNs for object classification and image denoising resulted in promising improvements [13, 34, 4, 5]. Boosting, as an ensemble learning technique, is capable of generalizing over various distributions and good interpretability [32, 22]. In contrast to methods that try to compensate for the scarcity of low-quality samples by introducing a sampling strategy [38, 21, 58, 29, 33], in boosting framework, distinct models are designed which are experts for samples with different hardness. To achieve this, the optimization path changes in accordance with the sample's hardness, and as an ensemble of models, they can enhance the overall generalization power [25].

We propose a method to take advantage of Adaptive Boosting (AdaBoost) to deal with the distribution shift on FR applications. To this aim, we hypothesize that in large-scale FR training benchmarks (i.e., LR, HR, long-range, and distorted images) are available (an imbalanced distribution concerning sample hardness). However, high-quality images dominate in these datasets (imbalance sample hardness). We propose an ensemble learning framework such that each learner adjusts the weight of the training samples for the consecutive learners based on the samples' hardness. As a result, the subsequent learners will concentrate on distinct samples compared to their predecessors, which changes the optimization path and leads to a more diverse feature representation. Consequently, we can: 1) explore currently available training data more effectively and 2) increase the generalizability of the resulting ensembled model on hard instances while maintaining the performance on easy facial images. Contributions of this work can be summarized as follows:

- We move beyond class-level imbalance to propose a novel sample-level objective function inspired by AdaBoost that better compensates for data distribution imbalance and give more importance to the misclassified samples from the earlier trained model.
- We empirically and theoretically show the relationship between sample mining and angular margin penalty.
- We propose a method to relieve the convergence issue of the current FR training paradigm when there is more emphasis on hard samples.

2. Related Work

2.1. Boosting

Boosting has been used in ensemble models to enhance performance by cascading several sub-models [13, 34]. Out of the many methods used for boosting, AdaBoost and Gradient Boosting are two of the most commonly surfed techniques [10, 11]. Boosting, also known as forward stagewise additive modeling, was originally proposed to improve the performance of classification trees. It has been recently incorporated into deep learning models to improve their performance further. Schwenk and Bengio [43] improved the ensemble accuracy of neural networks by using AdaBoost. Kawana et al. [23] have proposed an ensemble of CNNs for

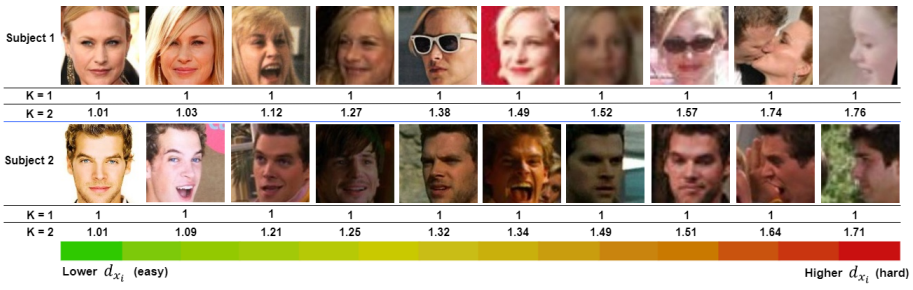

Figure 3: Various examples (including easy and hard samples) from two subjects and their assigned weights d_{x_i} by our method for ($K = 1$) and ($K = 2$).

human pose estimation, where each CNN in the ensemble model is optimized for a range of poses. They integrated the output of each individual CNN by feeding them as input to an integration module. Instead of averaging, the boosted CNN method in [34] uses the least square objective function to incorporate the boosting weights into training. Since boosting increases the networks' complexity, dense connections were adopted in a deep boosting framework to tackle the problem of vanishing gradient [4]. Yang et al. [56] used CNN to generate high-level features, followed by a boosted Forest classifier. Boosting techniques have been studied in classical machine learning methods and some limited areas of deep learning; however, they have yet to be explored for deep FR.

2.2. Margin-based Face Recognition

In recent years, most of the studies in the area of deep FR have been dedicated to enhancing performance by devising novel training criteria. The standard Softmax loss does not provide sufficient discriminability in embedding, i.e., intra-class compactness and inter-class separability [17]. The pioneer works of FaceNet [41] introduced a novel loss function that simultaneously uses positive (same identities) and negative (different identities) samples to improve the deep representations' discriminative power. This is achieved by bringing the anchor and positive sample closer together in the embedding space and pushing the anchor away from the negative sample [54, 49]. Several studies on the characteristics of the Softmax embeddings found an angular distribution in the representations. Therefore, the most recent methods proposed to increase the discriminability power of feature representation by mapping the Euclidian similarity of standard Softmax to angular space. As a result, by incorporating the angular penalty into the angular Softmax function, SOTA results have been achieved in various studies, including CosFace [51], ArcFace [7], and AdaFace [24].

2.3. Sample Mining in Face Recognition

Improving generalization ability is essential, and one way to achieve this is through hard sample mining [28]. Studies in this area have focused on two aspects: 1) finding a measure of sample hardness and 2) incorporating the sample hardness into the training paradigm [52]. Shrivastava et al., in [48], find easy and hard samples based on the loss value. They emphasize hard samples and discard easy samples (HM-SoftMax) to improve the generalization. Lin et al. [27] re-weights all the samples by introducing a soft mining strategy and training the network on a sparse set of hard instances [27]. Recently, MV-Softmax [52] has emerged as a framework that integrates margin and mining techniques. This approach defines hard samples as misclassified and emphasizes them by applying a predetermined constant on their negative cosine similarities. CurricularFace [20] employs the Curriculum Learning strategy to focus on easy samples at the beginning of training and then shifts the emphasis toward hard instances. Furthermore, Liu et al. [28] showed that samples within the same class have varying levels of importance and employed meta-learning to assign weights to each sample based on multiple variation factors. They trained a model with four learnable margins corresponding to ethnicity, pose, blur, and occlusion to achieve this. One major shortcoming of these works is the inconsistency of the improvements, i.e., increasing the performance on the harder benchmarks is gained by sacrificing performance on the easy benchmarks [40]. We seek to train multiple agents, e.g., deep CNNs, and combine their output to obtain consistency across different benchmarks. Also, we strive to mitigate the challenge of finding the hardness measure by directly employing the model score to indicate hardness.

3. Proposed Work

3.1. Overview of Angular FR Objective

The current SOTA FR training framework consists of a stack of non-linear feature extractor layers (backbone) followed by a classifier [2]. The whole architecture is trained using gradient descent with angular penalty criterion:

Table 1: Perfomance (%) comparison of our method with other recent algorithms. 1:1 verification accuracy for LFW, CFP-FP, CPLFW, AgeDB, and closed-set rank retrieval for TinyFace are reported. The backbone used here is Resnet18.

Method	High Quality						Low Quality (TinyFace)		
	LFW	CFP-FP	CPLFW	CALFW	AgeDB	AVG	Rank-1	Rank-5	Rank-20
HM-Softmax [48]	97.77	90.11	83.25	89.55	90.23	90.18	32.21	37.55	39.45
MV-Softmax [52]	98.25	91.36	84.47	91.88	91.15	91.42	36.19	47.14	40.88
CosFace [51]	99.00	91.89	84.99	91.63	91.85	91.87	45.00	55.00	58.00
CurricularFace [20]	98.87	92.05	86.14	92.46	92.24	92.35	45.15	48.14	54.24
ArcFace [7]	99.01	92.76	86.16	92.65	92.70	92.65	52.47	58.63	62.23
AdaFace [24]	99.13	92.82	87.00	92.65	92.717	92.86	56.06	61.45	65.21
Ours (K=1 & 2)	**99.23**	**92.96**	**87.07**	**92.93**	**93.00**	**93.04**	**57.83**	**63.31**	**67.17**

$$L = -\frac{1}{N}\sum_{i=1}^{N} d_{x_i} \log \frac{e^{f(W_{y_i},x_i,M)}}{e^{f(W_{y_i},x_i,M)} + \sum_{\substack{j=1 \\ j\neq y_i}}^{C} e^{f(W_j,x_i,M)}}, \quad (1)$$

where $W_j \in \mathbb{R}^{dim}$ is j-th classifier (center), and dim is the feature dimension, x_i is the learned feature of i-th sample, and y_i is its corresponding ground truth. N and C represent the mini-batch size and the total number of classes, respectively. $M = (m_s, m_c, m_a)$ is the margin hyperparameter, f is a function of W_j, x_i, and margin. d_{x_i} is the indicator function which in angular margin losses is chosen to be one, i.e., the equal importance of samples. Usually, when the feature vectors and class centers are projected to the unit-hypersphere, then f is written as a function of the angle between the feature vector and the j-th center of the classifier, $f(W_j, x_i, M) = f(\theta_{j,i}, M)$:

$$f(\theta_{j,i}, M) = \begin{cases} s\,cos(m_s\theta_{j,i} + m_a) - m_c; & j = y_i, \\ s\,cos(\theta_{j,i}); & j \neq y_i, \end{cases} \quad (2)$$

where the $\theta_{j,i}$ represents the angle between j-th center and i-th sample.

3.2. Integrating Ensemble Boosting to FR Training

AdaBoost was initially designed for a binary classification task in combination with a decision-tree algorithm [9]. Here, we utilize its original idea to fit it in a multi-class paradigm of FR training [14]. The goal is to enhance the representation power on the hard instances while maintaining the performance on the easy images. To do so, we consider K models (K=2 in our experiments) to form our ensemble model. The first model should be trained using the standard FR framework, $d_{x_i} = 1$ in Eq. 1. The classifier's centers can be considered the average of the samples in each class [40]. Therefore, Eq. 2 reflects the similarity of each sample with the average of samples in the specified class. Because the FR training benchmarks are imbalanced concerning the hardness of the samples [24], class centers tend to drift toward more frequent samples, which results in reducing the loss by increasing the similarity between the centers and over-represented instances, see Fig. 4. Consequently, the model is inclined toward forgetting the hard instances during the training, which can result in a lack of generalization over the hard samples.

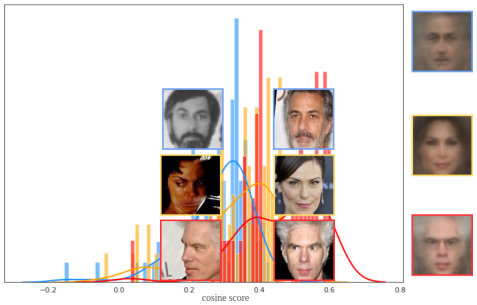

Figure 4: Similarity scores between samples in each class and its corresponding class center (average of samples), higher scores represent the high-quality and most frequent samples and lower scores represent the low-quality and hard samples.

For a given sample i, Eq. 2 reflects the similarity of the sample with its class center, which can be interpreted as the sample's hardness [33]. To obtain the samples' hardness, between the range 0 and 1, we explicitly utilize the Softmax output score instead of the Cosine score:

$$p_i = \frac{e^{f(W_{y_i},x_i,M)}}{e^{f(W_{y_i},x_i,M)} + \sum_{\substack{j=1 \\ j\neq y_i}}^{C} e^{f(W_j,x_i,M)}}. \quad (3)$$

In the standard AdaBoost framework, the constraint is that each binary classifier's accuracy is better than random guessing rather than 1/2. However, we want to increase the discrimination power among the hard samples in the FR training paradigm. It is important to mention that solely training on the hard instances can lead to suboptimal solutions and overfitting [37]. Our approach obtains sample weights independent from the visual quality, such as feature norm, and the hardness has directly resulted from the optimization path. To this aim, we chose a simple yet effective weighting scheme that is easy to implement and comprehend, as given by Eq. 4:

$$d_{x_i}{}^{k+1} = d_{x_i}{}^{k} p_i^{-\alpha}. \quad (4)$$

Eq. 4 puts more emphasis on the hard instances in a way that

Table 2: Perfomance (%) comparison of our method with other recent algorithms. True Acceptance Rate (TAR) at a different level of False Acceptance Rate (FAR) are reported for IJB-B and IJB-C. The backbone used here is Resnet18.

Method	Mix Quality (IJB-B)						Mix Quality (IJB-C)					
	1e-06	1e-05	1e-04	0.001	0.01	0.1	1e-06	1e-05	1e-04	0.001	0.01	0.1
HM-Softmax [48]	0.00	0.00	8.06	68.80	87.07	93.12	0.00	0.10	8.76	64.37	88.68	90.76
MV-Softmax [52]	0.00	0.00	0.00	8.90	68.97	94.11	0.00	0.15	10.54	68.15	90.05	94.60
CosFace [51]	0.00	0.11	10.01	70.16	89.69	95.81	0.00	0.80	14.75	69.05	91.58	96.02
CurricularFace [20]	0.00	0.15	10.14	70.95	90.01	95.86	0.01	1.01	15.25	69.18	91.89	96.42
ArcFace [7]	0.01	1.02	12.27	71.89	91.98	97.11	0.13	1.12	15.98	70.19	93.41	97.83
AdaFace [24]	0.11	1.26	13.28	72.81	92.53	97.40	0.13	1.26	17.80	71.23	93.44	97.84
Ours (K=1 & 2)	**1.70**	**7.20**	**40.63**	**82.55**	**93.36**	**97.88**	**1.14**	**6.12**	**41.42**	**83.37**	**94.68**	**98.25**

more challenging samples in each class will receive more weights during training, as shown in Fig. 5. Also, since the easy samples are not completely ignored, they relieve the feature representations from collapsing [39]. In evaluation, the final matching score is the weighted sum over the match score of K backbones:

$$H_{final}(x_i) = \sum_{k=1}^{K} \beta_k H_k(x_i), \tag{5}$$

where β_k hyperparameters are the weights associated to each trained model.

3.3. Sample Hardness as Angular Margin

There are two categories of sample mining methods: over-sampling and weighting schemes [52]. In the context of FR, over-sampling can lead to poor performance by reducing the diversity of samples. To alleviate this issue, one may resort to weighting methods [58, 52]. Here, we propose that naively applying samples' weights to the angular framework can be suboptimal. Although introducing sampling importance aims to compensate for misclassified data, sample weights can have hidden affect in the angular space. Applying sample mining, the term d_{x_i} in Eq. 1 is no longer uniform for all the samples.

In Eq. 1, as the training converges, the denominator can be estimated by a constant value [58]. In convergence, cosine similarity between samples and negative classes is close to zero, i.e., $f(W_j, x_i) = f(\theta_{j,i}) = 0$. Therefore, the denominator can be approximated as:

$$e^{f(W_{y_i},x_i,M)} + \sum_{\substack{j=1 \\ j\neq y_i}}^{C} e^{f(W_j,x_i,M)} \approx e^{f(W_{y_i},x_i,M)} + C - 1, \tag{6}$$

replacing the denominator in Eq. 1 with Eq. 6, then we can rearrange the Eq. 1 as:

$$L = -\frac{1}{N}\sum_{i=1}^{N} \log\left(\frac{e^{f(W_{y_i},x_i,M)}}{e^{f(W_{y_i},x_i,M)} + C - 1}\right)^{d_{x_i}}, \tag{7}$$

consequently, the essential and differentiable component is the $f(W_{y_i}, x_i, M)^{d_{x_i}}$ which can be rewriten as the following:

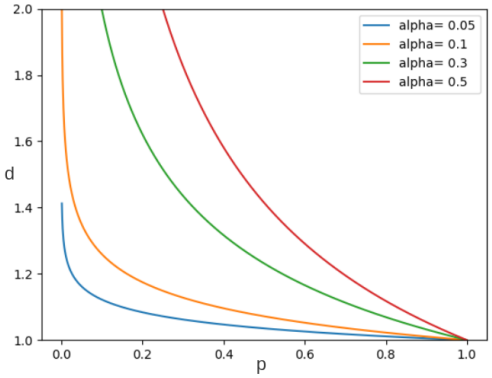

Figure 5: Illustrating the proposed sample hardness, d, against the output probability of the classifier. It is important to consider the impact of the parameter α in Eq. 4. When α has a large value, it can result in the weights of certain samples being overemphasized and easier samples being disregarded; more information is in Table 5.

$$\begin{aligned}\left(e^{f(W_{y_i},x_i,M)}\right)^{d_{x_i}} &= \left(e^{f(\theta_j,M)}\right)^{d_{x_i}} \\ &= \left(e^{d_{x_i}\ f(\theta_j,M)}\right) \\ &= \left(e^{s\cdot d_{x_i}\ cos(m_s\theta_j+m_a)-d_{x_i}\ m_c}\right).\end{aligned} \tag{8}$$

In other words, the scaling factor and the cosine margin are adaptively tuned with respect to the samples' hardness. To further study the effect of the scale hyperparameter, s, we plot the Softmax output score versus $\theta_{i,j}$ for different s as shown in Fig. 6. When the value of s is too small, such as $s = 10$, it is apparent that the maximum output score cannot reach one. This outcome is not desirable because even if the network is highly confident in the corresponding prediction, the loss function will still penalize the network, leading to poor performance for easy samoples. Conversely, when s is excessively large, the output curve is problematic as it produces a very high probability even when the angle is close to $\frac{\pi}{2}$. Consequently, the loss function with large s may not penalize misclassified samples, resulting in poor performance on hard samples. To alleviate this issue, we propose to tune

Table 3: Perfomance (%) comparison of our method with other recent algorithms. 1:1 verification accuracy for LFW, CFP-FP, CPLFW, AgeDB, closed-set rank retrieval for TinyFace and TAR@FAR=0.01% for IJB-B and IJC-B are reported. The backbone used here is Resnet50.

Method	High Quality						Low Quality (TinyFace)			Mix Quality	
	LFW	CFP-FP	CPLFW	CALFW	AgeDB	AVG	Rank-1	Rank-5	Rank-20	IJB-B	IJB-C
HM-Softmax [48]	97.85	92.85	90.14	91.75	92.33	92.98	46.71	48.21	50.47	89.10	62.96
MV-Softmax [52]	99.08	94.39	93.10	94.01	92.33	94.58	52.36	55.74	58.89	91.39	64.14
CosFace [51]	99.51	95.44	93.90	94.70	94.56	95.62	60.14	63.77	65.77	92.52	65.42
CurricularFace [20]	99.42	96.32	93.85	94.78	94.81	95.84	61.89	65.51	67.86	92.51	65.26
ArcFace [7]	99.70	97.14	94.05	95.14	95.15	96.24	68.99	73.89	76.04	94.39	96.19
AdaFace [24]	**99.78**	97.14	**94.16**	95.98	97.78	96.97	70.25	74.034	76.31	95.44	96.98
Ours (K=1 & 2)	99.75	**97.24**	94.13	**96.05**	**97.85**	**97.01**	**71.01**	**74.54**	**76.80**	**95.47**	**97.00**

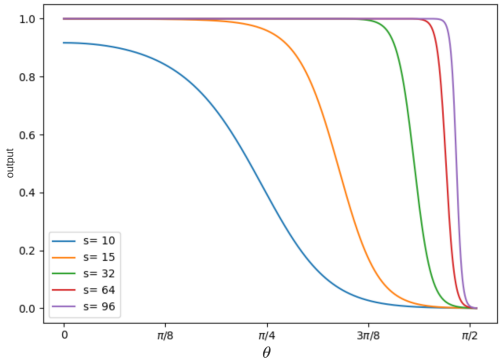

Figure 6: Curves of Softmax output w.r.t. θ_i by choosing different scale values.

the scale value with the normalized hardness score:

$$\widehat{d} = \frac{d - \overline{d}}{std(d)}\,\lambda, \tag{9}$$

where $\overline{d}$ is the moving average over the seen samples' weights, and std represents the standard deviation. $\frac{d-\overline{d}}{std(d)}$ which makes the batch distribution of d close to the normal Gaussian with zero mean and unit standard deviation. To further increase the concentration around the zero, λ is used as a hyperparameter. Then, we use the $\widehat{d}$ to fine-Ftune the scale, s:

$$s' = s - \lfloor \widehat{d} \rceil_{-0.33}^{0.33}\, s, \tag{10}$$

we clip the $\widehat{d}$ to be within $(-0.33, 0.33)$ so the noisy samples do not distract the training [24]. By this adaptivity, we can emphasize the hard samples while maintaining the discrimination on the easy instances [28]. A low value of d results in negative $\widehat{d}$ which increases the value of s (higher output score for easy samples) and vice versa.

4. Experimental Results

4.1. Datasets

We employ publicly available WebFace4M [63] as our training datasets which is a subset of the recently released FR dataset called WebFace260M. WebFace4M contains around 4M samples from 200k identities. Following the protocol of [46], we evaluate our models on five widely applied benchmarks in good quality, including LFW [19], CFP-FP [44], CPLFW [61] AgeDB [35] and CALFW [62]. Also, two mixed-quality datasets from the Janus program, including: the IARPA Janus Benchmark-B (IJB-B) [55] and Benchmark-C (IJB-C) [31] were used in our evaluations. Additionally, we use TinyFace as a challenging low-quality evaluation benchmark [6].

IJB-B and IJB-C: The IJB-B [55] dataset is a collection of face images and videos that is used to benchmark FR systems. It contains 21,800 images (11,800 face and 10,000 non-face images) and 7,000 videos (55,000 frames). The dataset includes 1,845 identities. The experimental protocols for IJB-B follow the standard 1:1 verification protocol, which contains 10,270 positive and 8 million negative matches. A template-based matching process is used, where the global feature vector for each template is obtained by averaging over the instances in the template. IJB-C is an extension of the IJB-B dataset, which contains 31,300 images and 117,500 frames from 3,531 identities. The testing protocol for IJB-C is similar to the protocol for IJB-B.

TinyFace: The TinyFace is a low-quality FR evaluation dataset comprising 5,139 labeled identities with 169,403 images. The images are designed for 1:N recognition tests and have an average size of 20×16 pixels. The TinyFace images were collected from public web data and face were captured under various uncontrolled conditions, including different poses, illumination, occlusion, and backgrounds. Fig. 7 illustrates the quality of some samples from TinyFace and IJB-B datasets.

4.2. Metrics

There are two main ways to evaluate the performance of a face recognition paradigm: recognition and verification. Recognition is a 1:N task where the network calculates the similarity score of a given probe image against all the samples in a gallery and identifies the probe image. Verification is a 1:1 task in which the network determines whether a

Figure 7: Left: samples from the IJB-B dataset. Right: sample from TinyFace. IJB-B consists of high-quality and low-quality images, while TinyFace mainly contains low-quality samples.

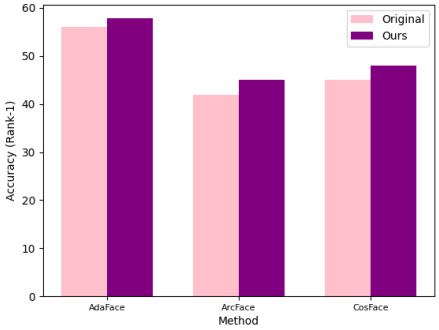

Figure 8: The rank-1 face identification accuracy on the TinyFace dataset using the AdaFace [24], ArcFace [7] and CosFace [51] FR methods when there are just one model (K=1, original) and when our proposed method is applied (i.e., $K = 1$ & $K = 2$, ours).

given pair of images represents the same identity. We report the verification results on the LFW, CFP-FP, AgeDB, CPLFW, IJB-B, IJB-C, and CALFW datasets. The identification results are reported on the TinyFace dataset.

4.3. Implementation

We followed the ArcFace setup for preprocessing [7]. All the images are resized to 112×112, aligned canonical view, and pixel values are normalized to $[-1, 1]$. The experiments are conducted with ResNet18 and Resnet50 as the backbone, and the models, are trained for 24 epochs with AdaFace loss [24]. The optimizer is SGD, with the learning rate starting from 0.1, which is decreased by a factor of 10 at epochs $\{10, 16, 22\}$. The optimizer weight-decay is set to 0.0001, the mini-batch size on each GPU is 512, and the model is trained using two Quadro RTX 8000. Fig. 2 shows the architecture and our proposed method for $K = 1$ and $K = 2$ in both training and evaluation settings. We investigate the impact of varying values of α in Eq. 4 on the performance of our method across different evaluation datasets. Our empirical analysis in Table 5 showed that an alpha value of 0.1 produced the best results. Also, $\beta_1 = 1$ and $\beta_2 = 0.1$ are obtained empirically.

4.4. Performance Comparison

Our proposed method's performance against SOTA studies has been assessed in Tables 1, 2, and 3. According to the results, the gain on the high-quality dataset is less pronounced since these datasets mostly contain high-quality samples. Therefore, the current performance of other competitors is saturated. On the other hand, results show remarkable improvements on the more challenging benchmarks of IJB-B, IJB-C, and TinyFace. In the case of IJB-B and IJB-C, our method achieves over 10 percent improvement using the R18 backbone at $TAR@FAR = 10^{-4}$. Also, over one percent improvement on TinyFace which shows that our method successfully maintains the performance on the high-quality samples and at the same time, it increases the discriminability among the hard instances. It should be noted that the improvement is more sensible when we are using the weaker backbone, Resnet18, as shown in in Tables 1 and 2.

5. Ablation Study

5.1. Training with hard samples

Solely training the K^{th} model on the hard instances can lead to suboptimal solutions and overfitting. Because discarding easy samples completely can be harmful, as they play a crucial role in relieving the representations from collapsing [39]. We investigated this effect by using only misclassified samples (from the training dataset) of the first model for training the second model ($K = 2$). As it is shown in Table 4 (fourth row), the performance of our method degrades severely. This reduction in discriminability can be attributed to 1) the reduction of the diversity of the data and 2) the extremely complex FR task when solely using hard instances.

5.2. Discussion on Individual Model's Performance

Boosting refers to combining different models to improve their overall performance. In this section, we evaluate the performance of individual models extracted from an ensemble model. Our results, presented in Table 4, indicate that the overall performance of the combined models is superior to that of any single model. This is because each model has its expertise in different groups of the training samples. Combining these models provides diverse discriminant information, resulting in a robust feature extractor with higher generalization.

Furthermore, in the classical AdaBoost, each model is trained from scratch, which is unsuitable for CNN and might force the CNN to become overfitted on those samples with higher weight. Transferring the currently learned parameters to the next CNN helps the following CNN preserve the previous knowledge acquired during the learning process and reduces the computational cost. Table 4, shows the comparison between our proposed method when the model corresponding to $K = 2$ trained from scratch (OursV3) or fine-tuned from the previous model (OursV1). The higher

Table 4: Ablation study on model's performance for $K = 1$, $K = 2$, and the combination of them in different settings. OursV1 (best): using all the training samples with assigned weights for training the second model ($K = 2$); OursV2: using hard samples for training the second model and OursV3: training the second model ($K = 2$) from the scratch.

Method	High Quality						TinyFace		Mixed Quality	
	LFW	CFP-FP	CPLFW	CALFW	AgeDB	AVG	Rank-1	Rank-5	IJB-B	IJB-C
AdaFace (K=1)	99.13	92.83	87.00	92.65	92.72	92.87	56.06	61.454	13.28	17.80
Ours (K= 2)	98.90	91.43	84.71	92.35	92.00	91.88	52.60	58.48	29.49	19.28
OursV1 (K=1 & 2)	99.23	92.96	87.07	92.93	93.00	93.04	57.83	63.31	40.63	41.42
OursV2 (K=1 & 2)	99.05	92.20	85.92	92.52	92.17	92.37	53.84	59.33	13.01	15.04
OursV3 (K=1 & 2)	99.12	92.69	85.95	93.00	92.08	92.47	52.55	58.07	13.01	15.04

Table 5: An ablation study to investigate the impact of varying values of α in Eq. 4 on the performance of our proposed method across different evaluation datasets. 1:1 verification average accuracy (Avg) for high-quality datasets, TAR@FAR=0.01% for IJB-B and Rank-1 accuracy for TinyFace are reported. The backbone used here is Resnet18, respectively.

Experiment	α	Avg	TinyFace	IJB-B
1	0.05	92.15	54.45	12.47
2 (Best)	**0.1**	**93.04**	**57.83**	**40.63**
3	0.3	91.71	53.98	15.35
4	0.5	91.59	53.86	14.65

performance of OursV1 demonstrates that transfer learning is significant in our approach.

5.3. Discussion on Re-weighting Samples During Training

Fig. 3 shows the easy and hard samples for two subjects from training dataset and their corresponding weights obtained by our method for $K = 1$ and $K = 2$. As it illustrates, the training of the first model indicates that frontal and high-quality faces are straightforward samples, resulting in lower corresponding weights compared to images with extreme poses or blurriness for training the second model. In the first model ($K = 1$), all samples are assigned the same weight. However, for the second model ($K = 2$), the weights of each sample are changed (based on Eq. 3 and 4), enabling the model to prioritize more challenging and hard samples during the training of the second model.

5.4. Orthogonality to Angular Criterion

Our training framework includes ensemble learning when designing a FR module. We want to evaluate the effectiveness of this approach with various loss functions. Although our main experiments used the AdaFace loss function, our method represents an independent improvement on AdaFace. Specifically, we applied two other SOTA loss functions, including ArcFace and CosFace (each with identical hyperparameters for margin and scale). Our results on the TinyFace dataset, as illustrated in Fig. 8, demonstrates that our approach enhances the feature embedding discriminability in all cases, indicating its independence from the choice of the training criterion.

6. Conclusion

To address the issue of imbalanced quality distribution in face recognition training datasets, we have proposed a novel approach that employs a sample-level weighting technique inspired by the traditional AdaBoost algorithm. By giving higher importance to the underrepresented tail samples during the training of a new model, our method is designed to improve the generalization performance of FR methods on such samples. The training loss function of an earlier model is used to update the sample training weights. If a sample is effectively trained by the first prior model, the weight associated with that sample is exponentially decreased, resulting in a negligible effect on the training of the next model and vice versa. This process results in the subsequent CNN becoming proficient in training samples with high weights. The combination of different models, where each of them is an expert in different groups of training samples, leads to a robust classifier. Our approach successfully outperforms any SOTA FR single model in several challenging face benchmarks as depicted in the experimental section. We believe that our approach could be very helpful for large-scale unbalanced data training in each method.

7. Acknowledgement

This research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA RD Contract No. 2022-21102100001. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006.

[2] F. Boutros, N. Damer, J. N. Kolf, K. Raja, F. Kirchbuchner, R. Ramachandra, A. Kuijper, P. Fang, C. Zhang, F. Wang, et al. Mfr 2021: Masked face recognition competition. In *IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10, 2021.

[3] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pages 67–74, 2018.

[4] C. Chen, Z. Xiong, X. Tian, and F. Wu. Deep boosting for image denoising. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–18, 2018.

[5] C. Chen, Z. Xiong, X. Tian, Z.-J. Zha, and F. Wu. Real-world image denoising with deep boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(12):3071–3087, 2019.

[6] Z. Cheng, X. Zhu, and S. Gong. Low-resolution face recognition. In *Asian Conference on Computer Vision*, pages 605–621. Springer, 2019.

[7] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4690–4699, 2019.

[8] G. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio. Large margin deep networks for classification. *Advances in Neural Information Processing Systems*, 31, 2018.

[9] Y. Freund and R. E. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *Computational Learning Theory: Second European Conference*, pages 23–37. Springer, 1995.

[10] Y. Freund, R. E. Schapire, et al. Experiments with a new boosting algorithm. In *International Conference on Machine Learning (ICML)*, volume 96, pages 148–156. Citeseer, 1996.

[11] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[12] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European Conference, Proceedings, Part III 14*, pages 87–102. Springer, 2016.

[13] S. Han, Z. Meng, A.-S. Khan, and Y. Tong. Incremental boosting convolutional neural network for facial action unit recognition. *Advances in neural information processing systems*, 29, 2016.

[14] T. Hastie, S. Rosset, J. Zhu, and H. Zou. Multi-class adaboost. *Statistics and its Interface*, 2(3):349–360, 2009.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on CVPR*, pages 770–778, 2016.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on CVPR*, pages 770–778, 2016.

[17] L. He, Z. Wang, Y. Li, and S. Wang. Softmax dissection: Towards understanding intra-and inter-class objective for embedding learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10957–10964, 2020.

[18] M. Hong, J. Choi, and G. Kim. Stylemix: Separating content and style for enhanced data augmentation. In *Proceedings of the IEEE conference on CVPR*, pages 14862–14870, 2021.

[19] 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.

[20] 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 conference on CVPR*, pages 5901–5910, 2020.

[21] M. A. Jamal, M. Brown, M.-H. Yang, L. Wang, and B. Gong. Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In *Proceedings of the IEEE Conference on CVPR*, pages 7610–7619, 2020.

[22] J. Jiang, B. Cui, C. Zhang, and F. Fu. Dimboost: Boosting gradient boosting decision tree to higher dimensions. In *Proceedings of International Conference on Management of Data*, pages 1363–1376, 2018.

[23] Y. Kawana, N. Ukita, J.-B. Huang, and M.-H. Yang. Ensemble convolutional neural networks for pose estimation. *Computer Vision and Image Understanding*, 169:62–74, 2018.

[24] M. Kim, A. K. Jain, and X. Liu. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE Conference on CVPR*, pages 18750–18759, 2022.

[25] A. Krogh and J. Vedelsby. Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, 7, 1994.

[26] X. Li, F. Wang, Q. Hu, and C. Leng. Airface: Lightweight and efficient model for face recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[28] C. Liu, X. Yu, Y.-H. Tsai, M. Faraki, R. Moslemi, M. Chandraker, and Y. Fu. Learning to learn across diverse data biases in deep face recognition. In *Proceedings of the IEEE Conference on CVPR*, pages 4072–4082, 2022.

[29] H. Liu, X. Zhu, Z. Lei, and S. Z. Li. Adaptiveface: Adaptive margin and sampling for face recognition. In *Proceedings of the IEEE Conference on CVPR*, pages 11947–11956, 2019.

[30] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on CVPR*, pages 212–220, 2017.

[31] 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 *International Conference on Biometrics (ICB)*, pages 158–165. IEEE, 2018.

[32] P. Melville and R. J. Mooney. Creating diversity in ensembles using artificial data. *Information Fusion*, 6(1):99–111, 2005.

[33] Q. Meng, S. Zhao, Z. Huang, and F. Zhou. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE Conference on CVPR*, pages 14225–14234, 2021.

[34] M. Moghimi, S. J. Belongie, M. J. Saberian, J. Yang, N. Vasconcelos, and L.-J. Li. Boosted convolutional neural networks. In *British Machine Vision Conference (BMVC)*, volume 5, page 6, 2016.

[35] 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 CVPR workshops*, pages 51–59, 2017.

[36] D. Opitz and R. Maclin. Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, 11:169–198, 1999.

[37] M. Opitz, G. Waltner, H. Possegger, and H. Bischof. Bier-boosting independent embeddings robustly. In *Proceedings of the IEEE international conference on computer vision*, pages 5189–5198, 2017.

[38] M. Ren, W. Zeng, B. Yang, and R. Urtasun. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pages 4334–4343. PMLR, 2018.

[39] W. Robbins and T. E. Boult. On the effect of atmospheric turbulence in the feature space of deep face recognition. In *Proceedings of the IEEE Conference on CVPR*, pages 1618–1626, 2022.

[40] M. S. E. Saadabadi, S. R. Malakshan, A. Zafari, M. Mostofa, and N. M. Nasrabadi. A quality aware sample-to-sample comparison for face recognition. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 6129–6138, 2023.

[41] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on CVPR*, pages 815–823, 2015.

[42] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on CVPR*, pages 815–823, 2015.

[43] H. Schwenk and Y. Bengio. Training methods for adaptive boosting of neural networks. *Advances in neural information processing systems*, 10, 1997.

[44] 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 WACV*, pages 1–9. IEEE, 2016.

[45] Y. Shi and A. K. Jain. Probabilistic face embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6902–6911, 2019.

[46] Y. Shi, X. Yu, K. Sohn, M. Chandraker, and A. K. Jain. Towards universal representation learning for deep face recognition. In *Proceedings of the IEEE Conference on CVPR*, pages 6817–6826, 2020.

[47] S. Shin, J. Lee, J. Lee, Y. Yu, and K. Lee. Teaching where to look: Attention similarity knowledge distillation for low resolution face recognition. In *ECCV*, pages 631–647. Springer, 2022.

[48] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on CVPR*, pages 761–769, 2016.

[49] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. *Advances in neural information processing systems*, 27, 2014.

[50] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille. Normface: L2 hypersphere embedding for face verification. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1041–1049, 2017.

[51] 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 CVPR*, pages 5265–5274, 2018.

[52] X. Wang, S. Zhang, S. Wang, T. Fu, H. Shi, and T. Mei. Mis-classified vector guided softmax loss for face recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12241–12248, 2020.

[53] Y. Wen, W. Liu, A. Weller, B. Raj, and R. Singh. Sphereface2: Binary classification is all you need for deep face recognition. *arXiv preprint arXiv:2108.01513*, 2021.

[54] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 499–515. Springer, 2016.

[55] 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 CVPR workshops*, pages 90–98, 2017.

[56] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Convolutional channel features. In *Proceedings of the IEEE international conference on computer vision*, pages 82–90, 2015.

[57] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[58] X. Zhang, R. Zhao, Y. Qiao, X. Wang, and H. Li. Adacos: Adaptively scaling cosine logits for effectively learning deep face representations. In *Proceedings of the IEEE Conference on CVPR*, pages 10823–10832, 2019.

[59] Y. Zhang, S. Herdade, K. Thadani, E. Dodds, J. Culpepper, and Y.-N. Ku. Unifying margin-based softmax losses in face recognition. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 3548–3557, 2023.

[60] Z. Zhang and M. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in Neural Information Processing Systems*, 31, 2018.

[61] 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*, 5(7), 2018.

[62] 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.

[63] 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 Conference on CVPR*, pages 10492–10502, 2021.