

AdaFace: Quality Adaptive Margin for Face Recognition

Minchul Kim, Anil K. Jain, Xiaoming Liu

Department of Computer Science and Engineering,
Michigan State University, East Lansing, MI, 48824

{kimminc2, jain, liuxm}@cse.msu.edu

Abstract

Recognition in low quality face datasets is challenging because facial attributes are obscured and degraded. Advances in margin-based loss functions have resulted in enhanced discriminability of faces in the embedding space. Further, previous studies have studied the effect of adaptive losses to assign more importance to misclassified (hard) examples. In this work, we introduce another aspect of adaptiveness in the loss function, namely the image quality. We argue that the strategy to emphasize misclassified samples should be adjusted according to their image quality. Specifically, the relative importance of easy or hard samples should be based on the sample's image quality. We propose a new loss function that emphasizes samples of different difficulties based on their image quality. Our method achieves this in the form of an adaptive margin function by approximating the image quality with feature norms. Extensive experiments show that our method, AdaFace, improves the face recognition performance over the state-of-the-art (SoTA) on four datasets (IJB-B, IJB-C, IJB-S and TinyFace). Code and models are released in Supp.

1. Introduction

Image quality is a combination of attributes that indicates how faithfully an image captures the original scene [32]. Factors that affect the image quality include brightness, contrast, sharpness, noise, color constancy, resolution, tone reproduction, etc. Face images, the focus of this paper, can be captured under a variety of settings for lighting, pose and facial expression, and sometimes under extreme visual changes such as a subject's age or make-up. These parameter settings make the recognition task difficult for learned face recognition (FR) models. Still, the task is achievable in the sense that humans or models can often recognize faces under these difficult settings [37]. However, when a face image is of low quality, depending on the degree, the recognition task becomes infeasible. Fig. 1 shows examples of both high quality and low quality face images. It is not possible to recognize the subjects in the last column of Fig. 1.

Recognizability Image Quality	Easy to Recognize	Hard to Recognize	Impossible to Recognize
High Quality			
Low Quality			

■ : Images contain enough clues to identify the subject
■ : Images do not have enough clues to identify the subject

Figure 1. Examples of face images with different qualities and recognizabilities. Both high and low quality images contain variations in pose, occlusion and resolution that sometimes make the recognition task difficult, yet achievable. Depending on the degree of degradation, some images may become impossible to recognize. By studying the different impacts these images have in training, this work aims to design a novel loss function that is adaptive to a sample's recognizability, driven by its image quality.

Low quality images like the bottom row of Fig. 1 are increasingly becoming an important part of face recognition datasets because they are encountered in surveillance videos and drone footage. Given that SoTA FR methods [7, 8, 16, 21] are able to obtain over 98% verification accuracy in relatively high quality datasets such as LFW or CFP-FP [14, 31], recent FR challenges have moved to lower quality datasets such as IJB-B, IJB-C and IJB-S [17, 26, 41]. Although the challenge is to attain high accuracy on low quality datasets, most popular training datasets still remain comprised of high quality images [7, 11]. Since only a small portion of training data is low quality, it is important to properly leverage it during training.

One problem with low quality face images is that they tend to be unrecognizable. When the image degradation is too large, the relevant identity information vanishes from the image, resulting in *unidentifiable images*. These unidentifiable images are detrimental to the training procedure since a model will try to exploit other visual characteristics, such as clothing color or image resolution, to lower the training loss. If these images are dominant in the distribution of low quality images, the model is likely to perform poorly on low quality datasets during testing.

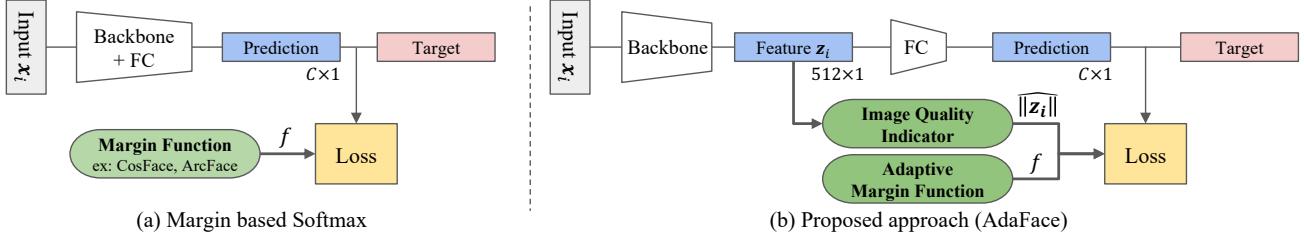


Figure 2. Conventional margin based softmax loss vs our AdaFace. (a) A FR training pipeline with a margin based softmax loss. The loss function takes the margin function to induce smaller intra-class variations. Some examples are SphereFace, CosFace and ArcFace [7,24,39]. (b) Proposed adaptive margin function (AdaFace) that is adjusted based on the image quality indicator. If the image quality is indicated to be low, the loss function emphasizes easy samples (thereby avoiding unidentifiable images). Otherwise, the loss emphasizes hard samples.

Motivated by the presence of unidentifiable facial images, we would like to design a loss function which assigns different importance to samples of different difficulties according to the image quality. We aim to emphasize hard samples for the high quality images and easy samples for low quality images. Typically, assigning different importance to different difficulties of samples is done by looking at the training progression (curriculum learning) [4, 16]. Yet, we show that the sample importance should be adjusted by looking at both the difficulty and the image quality.

The reason why importance should be set differently according to the image quality is that naively emphasizing hard samples always puts a strong emphasis on unidentifiable images. This is because one can only make a random guess about unidentifiable images and thus, they are always in the hard sample group. There are challenges in introducing image quality into the objective. This is because image quality is a term that is hard to quantify due to its broad definition and scaling samples based on the difficulty often introduces ad-hoc procedures that are heuristic in nature.

In this work, we present a loss function to achieve the above goal in a seamless way. We find that 1) feature norm can be a good proxy for the image quality, and 2) various margin functions amount to assigning different importance to different difficulties of samples. These two findings are combined in a unified loss function, AdaFace, that adaptively changes the margin function to assign different importance to different difficulties of samples, based on the image quality (see Fig. 2).

In summary, the contributions of this paper include:

- We propose a loss function, AdaFace, that assigns different importance to different difficulties of samples according to their image quality. By incorporating image quality, we avoid emphasizing unidentifiable images while focusing on hard yet recognizable samples.
- We show that the angular margin scales the learning signal (gradient) based on the training sample’s difficulty. This observation motivates us to change margin function adaptively to emphasize hard samples if the image quality is high, and ignore very hard samples (unidentifiable

images) if the image quality is low.

- We demonstrate that feature norm can serve as the proxy of image quality. It bypasses the need for an additional module to estimate image quality. Thus, adaptive margin function is achieved without additional complexity.
- We verify the efficacy of the proposed method by extensive evaluations on 9 datasets (LFW, CFP-FP, CPLFW, AgeDB, CALFW, IJB-B, IJB-C, IJB-S and TinyFace) of various qualities. We show that the recognition performance on low quality datasets can be hugely increased while maintaining performance on high quality datasets.

2. Related Work

Margin Based Loss Function. The margin based softmax loss function is widely used for training face recognition (FR) models [7, 16, 24, 39]. Margin is added to the softmax loss because without the margin, learned features are not sufficiently discriminative. SphereFace [24], CosFace [39] and ArcFace [7] introduce different forms of margin functions. Specifically, it can be written as,

$$\mathcal{L} = -\log \frac{\exp(f(\theta_{y_i}, m))}{\exp(f(\theta_{y_i}, m)) + \sum_{j \neq y_i}^n \exp(s \cos \theta_j)}, \quad (1)$$

where θ_j is the angle between the feature vector and the j^{th} classifier weight vector, y_i is the index of the ground truth (GT) label, and m is the margin, which is a scalar hyper-parameter. f is a margin function, where

$$f(\theta_j, m)_{\text{SphereFace}} = \begin{cases} s \cos(m\theta_j) & j = y_i \\ s \cos \theta_j & j \neq y_i \end{cases}, \quad (2)$$

$$f(\theta_j, m)_{\text{CosFace}} = \begin{cases} s(\cos \theta_j - m) & j = y_i \\ s \cos \theta_j & j \neq y_i \end{cases}, \quad (3)$$

$$f(\theta_j, m)_{\text{ArcFace}} = \begin{cases} s \cos(\theta_j + m) & j = y_i \\ s \cos \theta_j & j \neq y_i \end{cases}. \quad (4)$$

Sometimes, ArcFace is referred to as an *angular* margin and CosFace is referred to as an *additive* margin. Here, s is a

hyper-parameter for scaling. P2SGrad [46] notes that m and s are sensitive hyper-parameters and proposes to directly modify the gradient to be free of m and s .

Our approach aims to model the margin m as a function of the image quality because $f(\theta_{y_i}, m)$ has an impact on which samples contribute more gradient (*i.e.* learning signal) during training.

Adaptive Loss Functions. Many studies have introduced an element of adaptiveness in the training objective for either hard sample mining [22, 40], scheduling difficulty during training [16, 35], or finding optimal hyperparameters [45]. For example, CurricularFace [16] brings the idea of curriculum learning into the loss function. During the initial stages of training, the margin for $\cos \theta_j$ (negative cosine similarity) is set to be small so that easy samples can be learned and in the later stages, the margin is increased so that hard samples are learned. Specifically, it is written as

$$f(\theta_j, m)_{\text{Curricular}} = \begin{cases} s \cos(\theta_j + m) & j = y_i \\ N(t, \cos \theta_j) & j \neq y_i \end{cases}, \quad (5)$$

where

$$N(t, \cos \theta_j) = \begin{cases} \cos(\theta_j) & s \cos(\theta_{y_i} + m) \geq \cos \theta_j \\ \cos(\theta_j)(t + \cos \theta_j) & s \cos(\theta_{y_i} + m) < \cos \theta_j \end{cases}, \quad (6)$$

and t is a parameter that increases as the training progresses. Therefore, in CurricularFace, the adaptiveness in the margin is based on the training progression (curriculum).

On the contrary, we argue that the adaptiveness in the margin should be based on the image quality. We believe that among high quality images, if a sample is hard (with respect to a model), the network should learn to exploit the information in the image, but in low quality images, if a sample is hard, it is more likely to be devoid of proper identity clues and the network should not try hard to fit on it.

MagFace [27] explores the idea of applying different margins based on recognizability. It applies large angular margins to high norm features on the premise that high norm features are easily recognizable. Large margin pushes features of high norm closer to class centers. Yet, it fails to emphasize hard training samples, which is important for learning discriminative features. A detailed contrast with MagFace can be found in the supplementary B.1. It is also worth mentioning that DDL [15] uses the distillation loss to minimize the gap between easy and hard sample features.

Face Recognition with Low Quality Images. Recent FR models have achieved high performance on datasets where facial attributes are discernable, *e.g.*, LFW [14], CFP-FP [31], CPLFW [49], AgeDB [29] and CALFW [50]. Good performance on these datasets can be achieved when the FR model learns discriminative features invariant to lighting, age or pose variations. However, FR in unconstrained scenarios such as in surveillance or low quality videos [42] brings more problems to the table. Examples of datasets

in this setting are IJB-B [41], IJB-C [26] and IJB-S [17], where most of the images are of low quality, and some do not contain sufficient identity information, even for human examiners. The key to good performance involves both 1) learning discriminative features for low quality images and 2) learning to discard images that contain few identity cues. The latter is sometimes referred to as *quality aware fusion*.

To perform quality aware fusion, probabilistic approaches have been proposed to predict uncertainty in FR representation [5, 21, 30, 33, 51]. They assume the features are distributions and the variance can be used to calculate the certainty in prediction. However, probabilistic approaches often resort to learning mean and variance separately, which is not simple during training and suboptimal as the variance is optimized with a fixed mean. Our work, however, is a modification to the conventional softmax loss, making the framework easy to use. And we use the feature norm as a proxy for quality during quality-aware fusion.

QSub-PM [47] and UGG [48] also show good performances in LQ video recognition by using rich subspace (matrix) representation for comparison and using auxiliary context (such as a body) to aid feature fusion respectively.

Synthetic data or augmentations can be used to mimic low quality data. [10, 34] adopts 3D reconstruction to generate faces. Extra steps complicate the training procedure, making it hard to generalize to other domains. We adopt easily applicable crop, blur and photometric augmentations.

3. Proposed Approach

The cross entropy softmax loss of a sample x_i can be formulated as follows,

$$\mathcal{L}_{CE}(x_i) = -\log \frac{\exp(\mathbf{W}_{y_i} \mathbf{z}_i + b_{y_i})}{\sum_{j=1}^C \exp(\mathbf{W}_j \mathbf{z}_j + b_j)}, \quad (7)$$

where $\mathbf{z}_i \in \mathbb{R}^d$ is the x_i 's feature embedding, and x_i belongs to the y_i th class. \mathbf{W}_j refers to the j th column of the last FC layer weight matrix, $\mathbf{W} \in \mathbb{R}^{d \times C}$, and b_j refers to the corresponding bias term. C refers to the number of classes.

During test time, for an arbitrary pair of images, x_p and x_q , the cosine similarity metric, $\frac{\mathbf{z}_p \cdot \mathbf{z}_q}{\|\mathbf{z}_p\| \|\mathbf{z}_q\|}$ is used to find the closest matching identities. To make the training objective directly optimize the cosine distance, [24, 38] use normalized softmax where the bias term is set to zero and the feature \mathbf{z}_i is normalized and rescaled with s during training. This modification results in

$$\mathcal{L}_{CE}(x_i) = -\log \frac{\exp(s \cdot \cos \theta_{y_i})}{\sum_{j=1}^C \exp(s \cos \theta_j)}, \quad (8)$$

where θ_j corresponds to the angle between \mathbf{z}_i and \mathbf{W}_j . Follow-up works [7, 39] take this formulation and introduces a margin to reduce the intra-class variations. Generally, it can be written as Eq. 1 where margin functions are defined in Eqs. 2, 3 and 4 correspondingly.

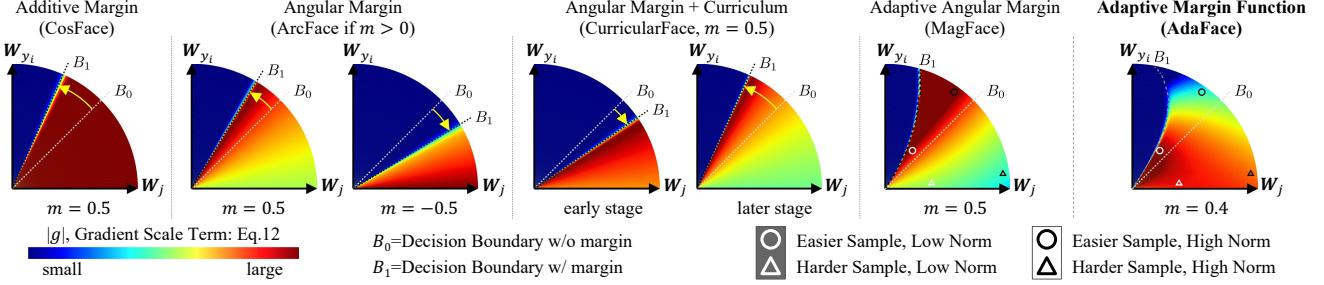


Figure 3. Illustration of different margin functions and their gradient scaling terms on the feature space. B_0 and B_1 show the decision boundary with and without margin m , respectively. The yellow arrow indicates the shift in the boundary due to margin m . In the arc, a well-classified sample will be close to (in angle) the ground truth class weight vector, \mathbf{W}_{y_i} . A misclassified sample will be close to \mathbf{W}_j , the negative class weight vector. The color within the arc indicates the magnitude of the gradient scaling term g (Eq. 12). Samples in the dark red region will contribute more to learning. Note that additive margin shifts the boundary toward \mathbf{W}_{y_i} , without changing the gradient scaling term. However, positive angular margin not only shifts the boundary, but also makes the gradient scale high near the boundary and low away from the boundary. This behavior de-emphasizes very hard samples, and likewise MagFace has similar behavior. On the other hand, negative angular margin induces an opposite behavior. CurricularFace adapts the boundary based on the training stage. Our work adaptively changes the margin functions based on the norm. With high norm, we emphasize samples away from the boundary and with low norm we emphasize samples near the boundary. Circles and triangles in the arc show example scenarios in the right most plot (AdaFace).

3.1. Margin Form and the Gradient

Previous works on margin based softmax focused on how the margin shifts the decision boundaries and what their geometric interpretations are [7, 39]. In this section, we show that during backpropagation, the gradient change due to the margin has the effect of scaling the importance of a sample relative to the others. In other words, angular margin can introduce an additional term in the gradient equation that scales the signal according to the sample’s difficulty. To show this, we will look at how the gradient equation changes with the margin function $f(\theta_{y_i}, m)$.

Let $P_j^{(i)}$ be the probability output at class j after softmax operation on an input \mathbf{x}_i . By deriving the gradient equations for \mathcal{L}_{CE} w.r.t. \mathbf{W}_j and \mathbf{x}_i , we obtain the following,

$$P_j^{(i)} = \frac{\exp(f(\cos \theta_{y_i}))}{\exp(f(\cos \theta_{y_i})) + \sum_{j \neq y_i}^n \exp(s \cos \theta_j)}, \quad (9)$$

$$\frac{\partial \mathcal{L}_{CE}}{\partial \mathbf{W}_j} = \left(P_j^{(i)} - \mathbb{1}(y_i = j) \right) \frac{\partial f(\cos \theta_j)}{\partial \cos \theta_j} \frac{\partial \cos \theta_j}{\partial \mathbf{W}_j}, \quad (10)$$

$$\frac{\partial \mathcal{L}_{CE}}{\partial \mathbf{x}_i} = \sum_{k=1}^C \left(P_k^{(i)} - \mathbb{1}(y_i = k) \right) \frac{\partial f(\cos \theta_k)}{\partial \cos \theta_k} \frac{\partial \cos \theta_k}{\partial \mathbf{x}_i}. \quad (11)$$

In Eqs. 10 and 11, the first two terms, $\left(P_j^{(i)} - \mathbb{1}(y_i = j) \right)$ and $\frac{\partial f(\cos \theta_j)}{\partial \cos \theta_j}$ are scalars. Also, these two are the only terms affected by parameter m through $f(\cos \theta_{y_i})$. As the direction term, $\frac{\partial \cos \theta_j}{\partial \mathbf{W}_j}$ is free of m , we can think of the first two scalar terms as a gradient scaling term (GST) and denote,

$$g := \left(P_j^{(i)} - \mathbb{1}(y_i = j) \right) \frac{\partial f(\cos \theta_j)}{\partial \cos \theta_j}. \quad (12)$$

For the purpose of the GST analysis, we will consider the class index $j = y_i$, since all negative class indices $j \neq y_i$ do not have a margin in Eqs. 2, 3, and 4. The GST for the normalized softmax loss is

$$g_{\text{softmax}} = (P_{y_i}^{(i)} - 1)s, \quad (13)$$

since $f(\cos \theta_{y_i}) = s \cdot \cos \theta_{y_i}$ and $\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}} = s$. The GST for the CosFace [39] is also

$$g_{\text{CosFace}} = (P_{y_i}^{(i)} - 1)s, \quad (14)$$

as $f(\cos \theta_{y_i}) = s(\cos \theta_{y_i} - m)$ and $\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}} = s$. Yet, the GST for ArcFace [7] turns out to be

$$g_{\text{ArcFace}} = (P_j^{(i)} - 1)s \left(\cos(m) + \frac{\cos \theta_{y_i} \sin(m)}{\sqrt{1 - \cos^2 \theta_{y_i}}} \right). \quad (15)$$

The derivation can be found in the supplementary. Since the GST is a function of θ_{y_i} and m as in Eq. 15, it is possible to use it to control the emphasis on samples based on the difficulty, i.e., θ_{y_i} during training.

To understand the effect of GST, we visualize GST w.r.t. the features. Fig. 3 shows the GST as the color in the feature space. Note that for the angular margin, the GST peaks at the decision boundary but slowly decreases as it moves away towards \mathbf{W}_j and harder samples receive less emphasis. If we change the sign of the angular margin, we see an opposite effect. Note that, in the 6th column, MagFace [27] is an extension of ArcFace (positive angular margin) with larger margin assigned to high norm feature. Both ArcFace and MagFace fail to put high emphasis on hard samples (green area near \mathbf{W}_j). We combine all margin functions (positive and negative angular margins and additive margins) to emphasize hard samples when necessary.

Note that this adaptiveness is also different from approaches that use the training stage to change the relative importance of different difficulties of samples [16]. Fig. 3 shows CurricularFace where the decision boundary and the GST g change depending on the training stage.

3.2. Norm and Image quality

Image quality is a comprehensive term that covers characteristics such as brightness, contrast and sharpness. Image quality assessment (IQA) is widely studied in computer vision [43]. SER-FIQ [36] is an unsupervised DL method for face IQA. BRISQUE [28] is a popular algorithm for blind/no-reference IQA. However, such methods are computationally expensive to use during training. In this work, we refrain from introducing an additional module that calculates the image quality. Instead, we use the feature norm as a proxy for the image quality. We observe that, in models trained with a margin-based softmax loss, the feature norm exhibits a trend that is correlated with the image quality.

In Fig. 4 (a) we show a correlation plot between the feature norm and the image quality (IQ) score calculated with (1-BRISQUE) as a green curve. We randomly sampled 1,534 images from the training dataset (MS1MV2 [7] with augmentations described in Sec. 4.1) and calculate the feature norm using a pretrained model. At the final epoch, the correlation score between the feature norm and IQ score reaches 0.5235 (out of -1 and 1). The corresponding scatter plot is shown in Fig. 4 (b). This high correlation between the feature norm and the IQ score supports our use of feature norm as the proxy of image quality.

In Fig. 4 (a) we also show a correlation plot between the probability output P_{y_i} and the IQ score as an orange curve. Note that the correlation is always higher for the feature norm than for P_{y_i} . Furthermore, the correlation between the feature norm and IQ score is visible from an early stage of training. This is a useful property for using the feature norm as the proxy of image quality because we can rely on the proxy from the early stage of training. Also, in Fig. 4 (c), we show a scatter plot between P_{y_i} and IQ score. Notice that there is a non-linear relationship between P_{y_i} and the image quality. One way to describe a sample’s difficulty is with $1 - P_{y_i}$, and the plot shows that the distribution of the difficulty of samples is different based on image quality. Therefore, it makes sense to consider the image quality when adjusting the sample importance according to the difficulty.

3.3. AdaFace: Adaptive Margin based on Norm

To address the problem caused by the unidentifiable images, we propose to adapt the margin function based on the feature norm. In Sec. 3.1, we have shown that using different margin functions can emphasize different difficulties of samples. Also, in Sec. 3.2, we have observed that the feature norm can be a good way to find low quality images. We

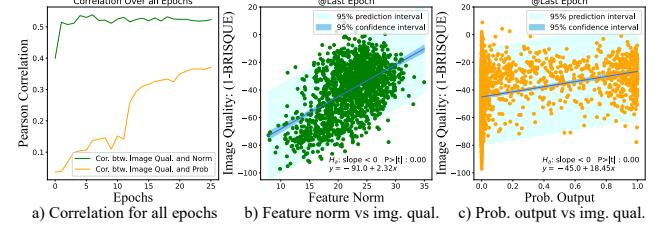


Figure 4. (a) A plot of Pearson correlation with image quality score (1-BRISQUE) over training epochs. The green and orange curves correspond to the correlation plot using the feature norm $\|\mathbf{z}_i\|$ and the probability output for the ground truth index P_{y_i} , respectively. (b) and (c) Corresponding scatter plots for the last epoch. The blue line on the scatter plot and the corresponding equation shows the least square line fitted to the data points.

will merge the two findings and propose a new loss for FR.

Image Quality Indicator. As the feature norm, $\|\mathbf{z}_i\|$ is a model dependent quantity, we normalize it using batch statistics μ_z and σ_z . Specifically, we let

$$\widehat{\|\mathbf{z}_i\|} = \left[\frac{\|\mathbf{z}_i\| - \mu_z}{\sigma_z/h} \right]_{-1}^1, \quad (16)$$

where μ_z and σ_z are the mean and standard deviation of all $\|\mathbf{z}_i\|$ within a batch. And $[\cdot]$ refers to clipping the value between -1 and 1 and stopping the gradient from flowing. Since $\frac{\|\mathbf{z}_i\| - \mu_z}{\sigma_z/h}$ makes the batch distribution of $\widehat{\|\mathbf{z}_i\|}$ as approximately unit Gaussian, we clip the value to be within -1 and 1 for better handling. It is known that approximately 68% of the unit Gaussian distribution falls between -1 and 1 , so we introduce the term h to control the concentration. We set h such that most of the values $\frac{\|\mathbf{z}_i\| - \mu_z}{\sigma_z/h}$ fall between -1 and 1 . A good value to achieve this would be $h = 0.33$. Later in Sec. 4.2, we ablate and validate this claim. We stop the gradient from flowing during backpropagation because we do not want features to be optimized to have low norms.

If the batch size is small, the batch statistics μ_z and σ_z can be unstable. Thus we use the exponential moving average (EMA) of μ_z and σ_z across multiple steps to stabilize the batch statistics. Specifically, let $\mu^{(k)}$ and $\sigma^{(k)}$ be the k -th step batch statistics of $\|\mathbf{z}_i\|$. Then

$$\mu_z = \alpha \mu_z^{(k)} + (1 - \alpha) \mu_z^{(k-1)}, \quad (17)$$

and α is a momentum set to 0.99. The same is true for σ_z .

Adaptive Margin Function. We design a margin function such that 1) if image quality is high, we emphasize hard samples, and 2) if image quality is low, we de-emphasize hard samples. We achieve this with two adaptive terms g_{angle} and g_{add} , referring to angular and additive margins, respectively. Specifically, we let

$$f(\theta_j, m)_{\text{AdaFace}} = \begin{cases} s(\cos(\theta_j + g_{\text{angle}}) - g_{\text{add}}) & j = y_i \\ s \cos \theta_j & j \neq y_i \end{cases} \quad (18)$$



Figure 5. Examples of three categories of test datasets in our study.

where g_{angle} and g_{add} are the functions of $\widehat{\|z_i\|}$. We define

$$g_{\text{angle}} = -m \cdot \widehat{\|z_i\|}, \quad g_{\text{add}} = m \cdot \widehat{\|z_i\|} + m. \quad (19)$$

Note that when $\widehat{\|z_i\|} = -1$, the proposed function becomes ArcFace. When $\widehat{\|z_i\|} = 0$, it becomes CosFace. When $\widehat{\|z_i\|} = 1$, it becomes a negative angular margin with a shift. Fig. 3 shows the effect of the adaptive function on the gradient. The high norm features will receive a higher gradient scale, far away from the decision boundary, whereas the low norm features will receive higher gradient scale near the decision boundary. For low norm features, the harder samples away from the boundary are de-emphasized.

4. Experiments

4.1. Datasets and Implementation Details

Datasets. We use MS1MV2 [7], MS1MV3 [9] and WebFace4M [52] as our training datasets. Each dataset contains 5.8M, 5.1M and 4.2M facial images, respectively. We test on 9 datasets of varying qualities. Following the protocol of [34], we categorize the test datasets into 3 types according to the visual quality (examples shown in Fig. 5).

- **High Quality:** LFW [14], CFP-FP [31], CPLFW [49] AgeDB [29] and CALFW [50] are popular benchmarks for FR in the well controlled setting. While the images show variations in lighting, pose, or age, they are of sufficiently good quality for face recognition.
- **Mixed Quality:** IJB-B and IJB-C [26, 41] are datasets collected for the purpose of introducing low quality images in the validation protocol. They contain both high quality images and low quality videos of celebrities.
- **Low Quality:** IJB-S [17] and TinyFace [6] are datasets with low quality images and/or videos. IJB-S is a surveillance video dataset, with test protocols such as *Surveillance-to-Single*, *Surveillance-to-Booking* and *Surveillance-to-Surveillance*. The first/second word in the protocol refers to the probe/gallery image source. *Surveillance* refers to the surveillance video, *Single* refers to a high quality enrollment image and *Booking* refers to multiple enrollment images taken from different viewpoints. TinyFace consists only of low quality images.

Training Settings. We preprocess the dataset by cropping and aligning faces with five landmarks, as in [7, 44], resulting in 112×112 images. For the backbone, we adopt ResNet [12] as modified in [7]. We use the same optimizer

and a learning rate schedule as in [16], and train for 24 epochs. The model is trained with SGD with the initial learning rate of 0.1 and step scheduling at 10, 18 and 22 epochs. If the dataset contains augmentations, we add 2 more epochs for convergence. For the scale parameter s , we set it to 64, following the suggestion of [7, 39].

Augmentations. Since our proposed method is designed to train better in the presence of unidentifiable images in the training data, we introduce three on-the-fly augmentations that are widely used in image classification tasks [13], *i.e.*, cropping, rescaling and photometric jittering. These augmentations will create more data but also introduce more unidentifiable images. It is a trade-off that has to be balanced. In FR, these augmentations are not used because they generally do not bring benefit to the performance (as shown in Sec. 4.2). We show that our loss function is capable of reaping the benefit of augmentations because it can adapt to ignore unidentifiable images.

Cropping defines a random rectangular area (patch) and makes the region outside the area to be 0. We do not cut and resize the image as the alignment of the face is important. Photometric augmentation randomly scales hue, saturation and brightness. Rescaling involves resizing an image to a smaller scale and back, resulting in blurriness. These operations are applied randomly with a probability of 0.2.

4.2. Ablation and Analysis

For hyperparameter m and h ablation, we adopt a ResNet18 backbone and use 1/6th of the randomly sampled MS1MV2. We use two performance metrics. For High Quality Datasets (HQ), we use an average of 1:1 verification accuracy in LFW, CFP-FP, CPLFW, AgeDB and CALFW. For Low Quality Datasets (LQ), we use an average of the closed-set rank-1 retrieval and the open-set TPIR@FIPR=1% for all 3 protocols of IJB-S. Unless otherwise stated, we augment the data as described in Sec. 4.1.

Effect of Image Quality Indicator Concentration h . In Sec. 3.3, we claim that $h = 0.33$ is a good value. To validate this claim, we show in Tab. 1 the performance when varying h . When $h = 0.33$, the model performs the best. For $h = 0.22$ or $h = 0.66$, the performance is still higher than CurricularFace. As long as h is set such that $\widehat{\|z_i\|}$ has some variation, h is not very sensitive. We set $h = 0.33$.

Effect of Hyperparameter m . The margin m corresponds to both the maximum range of the angular margin and the magnitude of the additive margin. Tab. 1 shows that the performance is best for HQ datasets when $m = 0.4$ and for LQ datasets when $m = 0.75$. Large m results in large angular margin variation based on the image quality, resulting in more adaptivity. In subsequent experiments, we choose $m = 0.4$ since it achieves good performance for LQ datasets without sacrificing performance on HQ datasets.

Method	h	m	Proxy	HQ Datasets	LQ Datasets
CurricularFace [16]	-	0.50		93.43	32.92
AdaFace	0.22			93.67	34.92
AdaFace	0.33	0.40	Norm	93.74	35.40
AdaFace	0.66			93.70	35.29
AdaFace	0.33	0.40		93.74	35.40
AdaFace	0.33	0.50	Norm	93.56	35.23
AdaFace		0.75		93.37	35.69
AdaFace	-	0.33	0.40	93.74	35.40
-	-	P_{y_i}	1-BRISQUE	93.43	34.55
-	-			93.46	35.17

Table 1. Ablation of our margin function parameters h and m , and the image quality proxy choice on the ResNet18 backbone. The performance metrics are as described in Sec. 4.2.

Method	p	HQ Datasets	LQ Datasets
CurricularFace [16]	0.0	96.85	41.00
CurricularFace [16]	0.2	96.75	40.84
CurricularFace [16]	0.3	96.59	40.58
AdaFace	0.0	96.72	40.95
AdaFace	0.2	96.88	41.82
AdaFace	0.3	96.78	41.93

Table 2. Ablation of augmentation probability p , on the ResNet50 backbone. The metrics are the same as Tab. 1.

Effect of Proxy Choice. In Tab. 1, to show the effectiveness of using the feature norm as a proxy for image quality, we switch the feature norm with other quantities such as (1-BRISQUE) or P_{y_i} . The performance using the feature norm is superior to using others. The BRISQUE score is precomputed for the training dataset, so it is not as effective in capturing the image quality when training with augmentation. We include P_{y_i} to show that the adaptiveness in feature norm is different from adaptiveness in difficulty.

Effect of Augmentation. We introduce on-the-fly augmentations in our training data. Our proposed loss can effectively handle the unidentifiable images, which are generated occasionally during augmentations. We experiment with a larger model ResNet50 on the full MS1MV2 dataset.

Tab. 2 shows that indeed the augmentation brings performance gains for AdaFace. The performance on HQ datasets stays the same, whereas LQ datasets enjoy a significant performance gain. Note that the augmentation hurts the performance of CurricularFace, which is in line with our assumption that augmentation is a tradeoff between a positive effect from getting more data and a negative effect from unidentifiable images. Prior works on margin-based softmax do not include on-the-fly augmentations as the performance could be worse. AdaFace avoids overfitting on unidentifiable images, therefore it can exploit the augmentation better.

Analysis. To show how the feature norm $\|z_i\|$ and the difficulty of training samples change during training, we plot the sample trajectory in Fig. 6. A total of 1,536 samples are randomly sampled from the training data. Each column in the heatmap represents a sample, and the x-axis is sorted according to the norm of the last epoch. Sample #600 is

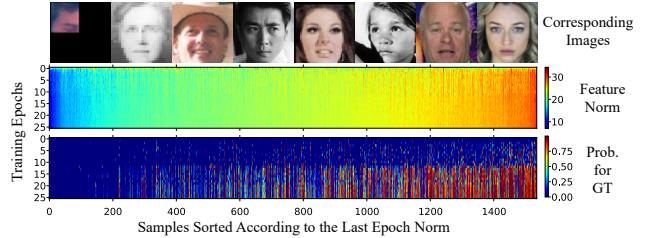


Figure 6. A plot of training samples’ trajectories of feature norm $\|z_i\|$ and the probability output for the ground truth index P_{y_i} . We randomly select 1,536 samples from the training data with augmentations, and show 8 images evenly sampled from them. The features with low norm have a different probability trajectory than others and the corresponding images are hard to identify.

approximately a middle point of the transition from low to high norm samples. The bottom plot shows that many of the probability trajectories of low norm samples never get high probability till the end. It is in line with our claim that low norm features are more likely to be unidentifiable images. It justifies our motivation to put less emphasis on these cases, although they are “hard” cases. The percentage of samples with augmentations is higher for the low norm features than for the high norm features. For samples number #0 to #600, about 62.0% are with at least one type of augmentation. For the samples #600 or higher, the percentage is about 38.5%.

Time Complexity. Compared to classic margin-based loss functions, our method adds a negligible amount of computation in training. With the same setting, ArcFace [7] takes 0.3193s per iteration while AdaFace takes 0.3229s (+1%).

4.3. Comparison with SoTA methods

To compare with SoTA methods, we evaluate ResNet100 trained with AdaFace loss on 9 datasets as listed in Sec. 4.1. For the high quality datasets, Tab. 3 (a) shows that AdaFace performs on par with competitive methods such as BroadFace [20], SCF-ArcFace [21] and VPL-ArcFace [8]. This strong performance in high quality datasets is due to the hard sample emphasis on high quality cases during training. Note that some performances in high quality datasets are saturated, making the gain less pronounced. Thus, choosing one model over the others is somewhat difficult based solely on the numbers. Unlike SCF-ArcFace, our method does not use additional learnable layers, nor requires 2-stage training. It is a revamp of the loss function, which makes it easier to apply our method to new tasks or backbones.

For mixed quality datasets, Tab. 3 (a) clearly shows the improvement of AdaFace. On IJB-B and IJB-C, AdaFace reduces the errors of the second best relatively by 11% and 9% respectively. This shows the efficacy of using feature norms as an image quality proxy to treat samples differently.

For low quality datasets, Tab. 3 (b) shows that AdaFace substantially outperforms all baselines. Compared to the second best, our averaged performance gain over 4 Rank-

Method	Venue	Train Data	High Quality						Mixed Quality	
			LFW [14]	CFP-FP [31]	CPLFW [49]	AgeDB [29]	CALFW [50]	AVG	IJB-B [41]	IJB-C [26]
CosFace ($m = 0.35$) [39]	CVPR18	MS1MV2	99.81	98.12	92.28	98.11	95.76	96.82	94.80	96.37
ArcFace ($m = 0.50$) [7]	CVPR19	MS1MV2	99.83	98.27	92.08	98.28	95.45	96.78	94.25	96.03
AFRN [18]	ICCV19	MS1MV2	99.85	95.56	93.48	95.35	96.30	96.11	88.50	93.00
MV-Softmax [40]	AAAI20	MS1MV2	99.80	98.28	92.83	97.95	96.10	96.99	93.60	95.20
CurricularFace [16]	CVPR20	MS1MV2	99.80	98.37	93.13	98.32	96.20	97.16	94.80	96.10
URL [34]	CVPR20	MS1MV2	99.78	98.64	-	-	-	-	-	96.60
BroadFace [20]	ECCV20	MS1MV2	99.85	98.63	93.17	98.38	96.20	97.25	94.97	96.38
MagFace [27]	CVPR21	MS1MV2	99.83	98.46	92.87	98.17	96.15	97.10	94.51	95.97
SCF-ArcFace [21]	CVPR21	MS1MV2	99.82	98.40	93.16	98.30	96.12	97.16	94.74	96.09
DAM-CurricularFace [23]	ICCV21	MS1MV2	-	-	-	-	-	-	95.12	96.20
AdaFace ($m = 0.4$)	CVPR22	MS1MV2	99.82	98.49	93.53	98.05	96.08	97.19	95.67	96.89
VPL-ArcFace [8]	CVPR21	MS1MV3	99.83	99.11	93.45	98.60	96.12	97.42	95.56	96.76
AdaFace ($m = 0.4$)	CVPR22	MS1MV3	99.83	99.03	93.93	98.17	96.02	97.40	95.84	97.09
ArcFace* [7]	CVPR19	WebFace4M	99.83	99.19	94.35	97.95	96.00	97.46	95.75	97.16
AdaFace ($m = 0.4$)	CVPR22	WebFace4M	99.80	99.17	94.63	97.90	96.05	97.51	96.03	97.39

(a) A performance comparison of recent methods on high and mixed quality datasets.

Method	Train Data	Low Quality (IJB-S [17] and TinyFace [6])										
		Surveillance-to-Single [17]			Surveillance-to-Booking [17]			Surveillance-to-Surveillance [17]			TinyFace [6]	
		Rank-1	Rank-5	1%	Rank-1	Rank-5	1%	Rank-1	Rank-5	1%	Rank-1	Rank-5
PFE [33]	MS1MV2 [7]	50.16	58.33	31.88	53.60	61.75	35.99	9.20	20.82	0.84	-	-
ArcFace [7]	MS1MV2 [7]	57.35	64.42	41.85	57.36	64.95	41.23	-	-	-	-	-
URL [34]	MS1MV2 [7]	59.79	65.78	41.06	61.98	67.12	42.73	-	-	-	63.89	68.67
CurricularFace* [16]	MS1MV2 [7]	62.43	68.68	47.68	63.81	69.74	47.57	19.54	32.80	2.53	63.68	67.65
AdaFace ($m = 0.4$)	MS1MV2 [7]	65.26	70.53	51.66	66.27	71.61	50.87	23.74	37.47	2.50	68.21	71.54
AdaFace ($m = 0.4$)	MS1MV3 [9]	67.12	72.67	53.67	67.83	72.88	52.03	26.23	40.60	3.28	67.81	70.98
ArcFace* [7]	WebFace4M [52]	69.26	74.31	57.06	70.31	75.15	56.89	32.13	46.67	5.32	71.11	74.38
AdaFace ($m = 0.4$)	WebFace4M [52]	70.42	75.29	58.27	70.93	76.11	58.02	35.05	48.22	4.96	72.02	74.52

(b) A performance comparison of recent methods on low quality datasets.

Table 3. Comparison on benchmark datasets, with the ResNet100 backbone. For high quality and mixed quality datasets, 1:1 verification accuracy and TAR@FAR=0.01% are reported respectively. For IJB-S, open-set TPIR@FPIR=1% and closed-set rank retrieval (Rank-1 and Rank-5) are reported. Rank retrieval is also used for TinyFace. [KEYS: **Best**, **Second best**, * = our evaluation of the released model]

1 metrics is 3.5%, and over 3 TPIR@=FPIR=1% metrics is 2.4%. These results show that AdaFace is effective in learning a good representation for the low quality settings as it prevents the model from fitting on unidentifiable images.

We further train on a refined dataset, MS1MV3 [9] for a fair comparison with a recent work VPL-ArcFace [8]. The performance using MS1MV3 is higher than MS1MV2 due to less noise in MS1MV3. We also train on newly released WebFace4M [52] dataset. While one method might shine on one type of data, it is remarkable to see that collectively Adaface achieves SOTA performance on test data with a wide range of image quality, and on various training sets.

5. Conclusion

In this work, we address the problem arising from unidentifiable face images in the training dataset. Data collection processes or data augmentations introduce these images in the training data. Motivated by the difference in recognizability based on image quality, we tackle the problem by 1) using a feature norm as a proxy for the image quality and 2) changing the margin function adaptively based on the feature norm to control the gradient scale assigned to different quality of images. We evaluate the efficacy of the proposed adaptive loss on various qualities of datasets and achieve SoTA for mixed and low quality face datasets.

Limitations. This work addresses the existence of unidentifiable images in the training data. However, a noisy label is also one of the prominent characteristics of large-scale facial training datasets. Our loss function does not give special treatment to mislabeled samples. Since our adaptive loss assigns large importance to difficult samples of high quality, high quality mislabeled images can be wrongly emphasized. We believe future works may adaptively handle both unidentifiability and label noise at the same time.

Potential Societal Impacts. We believe that the Computer Vision community as a whole should strive to minimize the negative societal impact. Our experiments use the training dataset MS1MV*, which is a by-product of MS-Celeb [25], a dataset withdrawn by its creator. Our usage of MS1MV* is necessary to compare our result with SoTA methods on a fair basis. However, we believe the community should move to new datasets, so we include results on newly released WebFace4M [52], to facilitate future research. In the scientific community, collecting human data requires IRB approval to ensure informed consent. While IRB status is typically not provided by dataset creators, we assume that most FR datasets (with the exceptions of IJB-S) do not have IRB, due to the nature of collection procedures. One direction of the FR community is to collect large datasets with informed consent, fostering R&D without societal concerns.

References

- [1] InsightFace. <https://github.com/deepinsight/insightface.git>. Accessed: 2021-9-1. 7
- [2] InsightFacePytorch. https://github.com/TreBleN/InsightFace_Pytorch.git. Accessed: 2021-9-1. 7
- [3] TFace. <https://github.com/Tencent/TFace.git>. Accessed: 2021-10-3. 7
- [4] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 41–48, 2009. 2
- [5] Jie Chang, Zhonghao Lan, Changmao Cheng, and Yichen Wei. Data uncertainty learning in face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5710–5719, 2020. 3
- [6] Zhiyi Cheng, Xiatian Zhu, and Shaogang Gong. Low-resolution face recognition. In *Asian Conference on Computer Vision*, pages 605–621, 2018. 6, 8, 7
- [7] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 1, 2, 3, 4, 5, 6, 7, 8
- [8] Jiankang Deng, Jia Guo, Jing Yang, Alexandros Lattas, and Stefanos Zafeiriou. Variational prototype learning for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11906–11915, 2021. 1, 7, 8
- [9] Jiankang Deng, Jia Guo, Debing Zhang, Yafeng Deng, Xiangju Lu, and Song Shi. Lightweight face recognition challenge. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 6, 8
- [10] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3d face reconstruction and dense alignment with position map regression network. In *European Conference on Computer Vision*, pages 534–551, 2018. 3
- [11] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. MS-Celeb-1M: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision*, pages 87–102, 2016. 1
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 6
- [13] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2019. 6
- [14] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric 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. 1, 3, 6, 8, 7
- [15] Yuge Huang, Pengcheng Shen, Ying Tai, Shaoxin Li, Xiaoming Liu, Jilin Li, Feiyue Huang, and Rongrong Ji. Improving face recognition from hard samples via distribution distillation loss. In *European Conference on Computer Vision*, pages 138–154, 2020. 3
- [16] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. CurricularFace: adaptive curriculum learning loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5901–5910, 2020. 1, 2, 3, 5, 6, 7, 8
- [17] Nathan D Kalka, Brianna Maze, James A Duncan, Kevin O'Connor, Stephen Elliott, Kaleb Hebert, Julia Bryan, and Anil K Jain. IJB-S: IARPA Janus Surveillance Video Benchmark. In *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–9, 2018. 1, 3, 6, 8, 7
- [18] Bong-Nam Kang, Yonghyun Kim, Bongjin Jun, and Daijin Kim. Attentional feature-pair relation networks for accurate face recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5472–5481, 2019. 8
- [19] Insoo Kim, Seungju Han, Ji-won Baek, Seong-Jin Park, Jae-Joon Han, and Jinwoo Shin. Quality-agnostic image recognition via invertible decoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12257–12266, 2021. 7
- [20] Yonghyun Kim, Wonpyo Park, and Jongju Shin. BroadFace: Looking at tens of thousands of people at once for face recognition. In *European Conference on Computer Vision*, pages 536–552, 2020. 7, 8
- [21] Shen Li, Jianqing Xu, Xiaqing Xu, Pengcheng Shen, Shaoxin Li, and Bryan Hooi. Spherical confidence learning for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15629–15637, 2021. 1, 3, 7, 8
- [22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2980–2988, 2017. 3
- [23] Jiaheng Liu, Yudong Wu, Yichao Wu, Chuming Li, Xiaolin Hu, Ding Liang, and Mengyu Wang. DAM: Discrepancy alignment metric for face recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3814–3823, 2021. 8
- [24] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. SphereFace: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 212–220, 2017. 2, 3
- [25] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015. 8
- [26] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggle, Janet Anderson, Jordan Cheney, and Patrick Grother.

- IARPA Janus Benchmark-C: Face dataset and protocol. In *2018 International Conference on Biometrics (ICB)*, pages 158–165, 2018. 1, 3, 6, 8, 5, 7
- [27] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. MagFace: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14225–14234, 2021. 3, 4, 8, 1
- [28] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012. 5
- [29] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. AGEDB: the first manually collected, in-the-wild age database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 51–59, 2017. 3, 6, 8, 7
- [30] Necmiye Ozay, Yan Tong, Frederick W. Wheeler, and Xiaoming Liu. Improving face recognition with a quality-based probabilistic framework. In *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 134–141, 2009. 3
- [31] Soumyadip Sengupta, Jun-Cheng Chen, Carlos Castillo, Vishal M Patel, Rama Chellappa, and David W Jacobs. Frontal to profile face verification in the wild. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, 2016. 1, 3, 6, 8, 7
- [32] Hamid R Sheikh and Alan C Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, 2006. 1
- [33] Yichun Shi and Anil K Jain. Probabilistic face embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6902–6911, 2019. 3, 8
- [34] Yichun Shi, Xiang Yu, Kihyuk Sohn, Manmohan Chandraker, and Anil K Jain. Towards universal representation learning for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6817–6826, 2020. 3, 6, 8
- [35] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 761–769, 2016. 3
- [36] Philipp Terhorst, Jan Niklas Kolf, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Ser-fiq: unsupervised estimation of face image quality based on stochastic embedding robustness. In 2020 ieee. In *CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 13–19, 2020. 5
- [37] Luan Tran, Xi Yin, and Xiaoming Liu. Disentangled representation learning GAN for pose-invariant face recognition. In *Proceeding of IEEE Computer Vision and Pattern Recognition*, pages 1415–1424, 2017. 1
- [38] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. NormFace: L2 hypersphere embedding for face verification. In *Proceedings of the 25th ACM International Conference on Multimedia*, pages 1041–1049, 2017. 3
- [39] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. CosFace: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5265–5274, 2018. 2, 3, 4, 6, 8, 1
- [40] Xiaobo Wang, Shifeng Zhang, Shuo Wang, Tianyu Fu, Hailin Shi, and Tao 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. 3, 8
- [41] Cameron Whitelam, Emma Taborsky, Austin Blanton, Brianna Maze, Jocelyn Adams, Tim Miller, Nathan Kalka, Anil K Jain, James A Duncan, Kristen Allen, et al. IARPA Janus Benchmark-B face dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 90–98, 2017. 1, 3, 6, 8, 7
- [42] Xi Yin, Ying Tai, Yuge Huang, and Xiaoming Liu. FAN: Feature adaptation network for surveillance face recognition and normalization. In *Proceedings of the Asian Conference on Computer Vision*, pages 301–319, 2020. 3
- [43] Guangtao Zhai and Xiongkuo Min. Perceptual image quality assessment: a survey. *Science China Information Sciences*, 63(11):211301, 2020. 5
- [44] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016. 6, 7
- [45] Xiao Zhang, Rui Zhao, Yu Qiao, Xiaogang Wang, and Hongsheng Li. Adacos: Adaptively scaling cosine logits for effectively learning deep face representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10823–10832, 2019. 3
- [46] Xiao Zhang, Rui Zhao, Junjie Yan, Mengya Gao, Yu Qiao, Xiaogang Wang, and Hongsheng Li. P2sGrad: Refined gradients for optimizing deep face models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9906–9914, 2019. 3
- [47] Jingxiao Zheng, Rajeev Ranjan, Ching-Hui Chen, Jun-Cheng Chen, Carlos D Castillo, and Rama Chellappa. An automatic system for unconstrained video-based face recognition. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2(3):194–209, 2020. 3
- [48] Jingxiao Zheng, Ruichi Yu, Jun-Cheng Chen, Boyu Lu, Carlos D Castillo, and Rama Chellappa. Uncertainty modeling of contextual-connections between tracklets for unconstrained video-based face recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 703–712, 2019. 3
- [49] Tianyue Zheng and Weihong 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. 3, 6, 8, 7
- [50] Tianyue Zheng, Weihong Deng, and Jiani Hu. Cross-Age LFW: A database for studying cross-age face recognition in unconstrained environments. *CoRR*, abs/1708.08197, 2017. 3, 6, 8, 7

[51] Shaohua Zhou, Volker Krueger, and Rama Chellappa. Probabilistic recognition of human faces from video. *CVIU*, 91(1-2):214–245, 2003. 3

[52] Zheng Zhu, Guan Huang, Jiankang Deng, Yun Ye, Junjie Huang, Xinze Chen, Jiagang Zhu, Tian Yang, Jiwen Lu, Da-long Du, and Jie Zhou. 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*, pages 10492–10502, 2021. 6, 8

AdaFace: Quality Adaptive Margin for Face Recognition

Supplementary Material

A. Gradient Scaling Term

In Sec. 3.1 of the main paper, the gradient scaling term (GST), g is introduced. Specifically, it is derived from the gradient equation for the margin-based softmax loss and defined as

$$g := \left(P_j^{(i)} - \mathbb{1}(y_i = j) \right) \frac{\partial f(\cos \theta_j)}{\partial \cos \theta_j}, \quad (1)$$

where

$$P_j^{(i)} = \frac{\exp(f(\cos \theta_{y_i}))}{\exp(f(\cos \theta_{y_i})) + \sum_{j \neq y_i}^n \exp(s \cos \theta_j)}. \quad (2)$$

This scalar term, g affects the magnitude of the gradient during backpropagation from the margin-based softmax loss. The form of g depends on the form of the margin function $f(\cos \theta_j)$. In Tab. 1, we summarize the margin function $f(\cos \theta_j)$ and the corresponding GST when $j = y_i$, the ground truth index.

Methods	$f(\cos \theta_j), j \neq y_i$	$f(\cos \theta_j), j = y_i$	g when $j = y_i$
Softmax	$s \cdot \cos \theta_{y_i}$	$s \cdot \cos \theta_{y_i}$	$(P_{y_i}^{(i)} - 1)s$
Additive Margin (CosFace [39])	$s \cdot \cos \theta_{y_i}$	$s(\cos \theta_{y_i} - m)$	$(P_{y_i}^{(i)} - 1)s$
Angular Margin (ArcFace [7])	$s \cdot \cos \theta_{y_i}$	$s \cdot \cos(\theta_{y_i} + m)$	$(P_{y_i}^{(i)} - 1)s \left(\cos(m) + \frac{\cos \theta_{y_i} \sin(m)}{\sqrt{1-\cos^2 \theta_{y_i}}} \right)$
Adaptive Angular Margin	$s \cdot \cos \theta_{y_i}$	$s \cdot \cos(\theta_{y_i} + m(\ z_i\))$	$(P_{y_i}^{(i)} - 1)s \left(\cos(m(\ z_i\)) + \frac{\cos \theta_{y_i} \sin(m(\ z_i\))}{\sqrt{1-\cos^2 \theta_{y_i}}} \right)$
	$m(\ z_i\)$ = a monotonically inc. function of $\ z_i\ $. In this table, g is derived with $\ z_i\ $ as a constant.		
CurricularFace [16]	$N(t, \cos \theta_j)$	$s \cdot \cos(\theta_{y_i} + m)$	$(P_{y_i}^{(i)} - 1)s \left(\cos(m) + \frac{\cos \theta_{y_i} \sin(m)}{\sqrt{1-\cos^2 \theta_{y_i}}} \right)$
	$N(t, \cos \theta_j) = \cos(\theta_j)(t + \cos \theta_j)$	if $s \cos(\theta_{y_i} + m) < \cos \theta_j$ else $\cos(\theta_j)$	
AdaFace (ours)	$s \cdot \cos \theta_{y_i}$	$s \cdot \cos(\theta_{y_i} + g_{\text{angle}}) - g_{\text{add}}$	$(P_{y_i}^{(i)} - 1)s \left(\cos(g_{\text{angle}}) + \frac{\cos \theta_{y_i} \sin(g_{\text{angle}})}{\sqrt{1-\cos^2 \theta_{y_i}}} \right)$
	$g_{\text{angle}} = -m \cdot \widehat{\ z_i\ }$, $g_{\text{add}} = m \cdot \widehat{\ z_i\ } + m$	$\widehat{\ z_i\ } = \left\lfloor \frac{\ z_i\ - \mu_z}{\sigma_z/h} \right\rfloor_{-1}$	

Table 1. Table of margin functions and their gradient scale terms. The concept of Adaptive Angular Margin is explored in MagFace [27]. However, unlike other works, MagFace is treating $m(\|z_i\|)$ as a term to optimize (*i.e.* $\|z_i\|$ is a function of $\cos \theta_j$), as oppose to treating it as a constant. In this table, we treat $\|z_i\|$ as a constant to highlight the effect of the margin. The exact form of g for MagFace will be different. In Fig. 3 of the main paper, Adaptive Angular Margin is visualized using the equation from this table.

Note that P_{y_i} is also affected by the choice of the margin function $f(\cos \theta_{y_i})$ as in Eqn. 2. So, g is a function of m , except for Softmax, and g is affected by m through $f(\cos \theta_{y_i})$ in P_{y_i} . For Angular Margin, m appears in the equation for g directly. We derive g for Angular Margin below. The term g for the Adaptive Angular Margin and CurricularFace [16] can be obtained using the g from the Angular Margin. The GST term for AdaFace can be obtained by using g for the Angular Margin and the Additive Margin, and replacing m with adaptive terms g_{angle} and g_{add} . This is possible because $\|z_i\|$ is treated as a constant.

A.1. Derivation of Angular Margin

We can rewrite $f(\cos \theta_{y_i})$ as

$$\begin{aligned} f(\cos \theta_{y_i}) &= s \cdot (\cos(\theta_{y_i} + m)) \\ &= s \cdot (\cos \theta_{y_i} \cos m - \sin \theta_{y_i} \sin m) \\ &= s \cdot \left(\cos \theta_{y_i} \cos m - \sqrt{1 - \cos^2 \theta_{y_i}} \sin m \right), \end{aligned} \quad (3)$$

by the laws of trigonometry. Therefore,

$$\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}} = s \left(\cos(m) + \frac{\cos \theta_{y_i} \sin(m)}{\sqrt{1 - \cos^2 \theta_{y_i}}} \right). \quad (4)$$

A.2. Interpretation of g

For Softmax and Additive Margin, we see that $g = (P_{y_i}^{(i)} - 1)s$. Since the softmax operation in $P_{y_i}^{(i)}$ has a tendency to scale the result to be close to either 0 or 1, the first term in g , $(P_j^{(i)} - 1)$ tends to be close to 1 or 0 far away from the decision boundary. In the equation for P_{y_i} , there is also s which is a scaling hyper-parameter, and is often set to $s = 64$ [7, 16, 24, 39]. This high s makes the softmax operation even steeper near the decision boundary. This results in almost equal GST for samples away from the decision boundary, regardless of how far they are from the decision boundary. This is evident in Fig. 1, where the blue curve is flat except near the decision boundary when s is high.

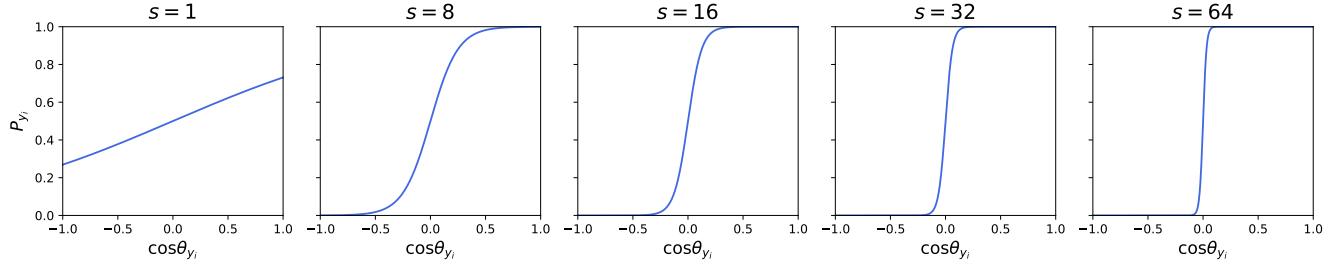


Figure 1. Plot of P_{y_i} for different values of s . In this figure, P_{y_i} is calculated with $f(\cos \theta_j)$ from Softmax (i.e. $m = 0$).

For Softmax and Additive Margin, $\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}} = s$. This term is different for Angular Margin due to $\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}}$ being a function of $\cos \theta_{y_i}$. The exact form of $\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}}$ for Angular Margin is found in Eqn. 4. As shown in Fig. 2, Eqn. 4 is monotonically increasing with respect to $\cos \theta_{y_i}$ when $m > 0$ and vice versa. Note that $\cos \theta_{y_i}$ is how close the sample is to the ground truth weight vector, and it is closely related to the difficulty of the sample during training. Therefore, this partial derivative term from the angular margin, $\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}}$, can be viewed as scaling the importance of sample based on the difficulty.

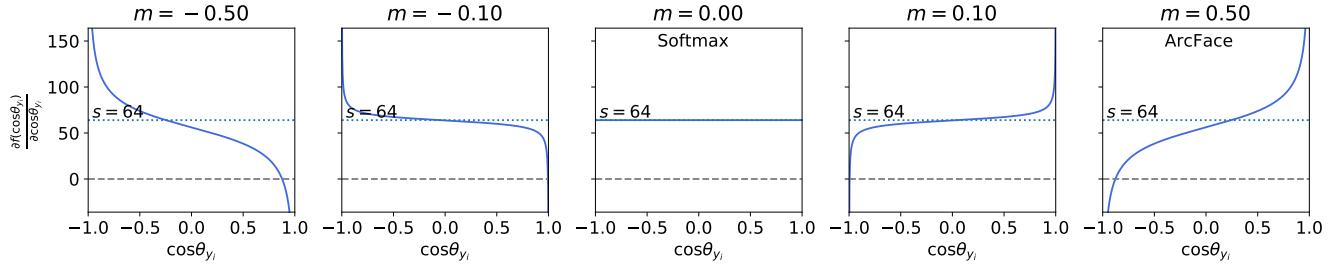


Figure 2. Plot of $\frac{\partial f(\cos \theta_{y_i})}{\partial \cos \theta_{y_i}}$ for different value of m when the margin function is Angluar Margin.

B. Feature Norm Analysis

B.1. Correlation between Norm and BRISQUE during Training

In the Sec. 3.2 of the main paper, we introduce the idea of using the feature norm as a proxy of the image quality. We observe that in models trained with a margin-based softmax loss, the feature norm exhibits a trend that is correlated with the image quality. Here, we show for ArcFace and AdaFace, both loss functions exhibit this trend, in Fig. 3. Regardless of the form of the margin function, the correlation between the feature norm and the image quality is quite similar (green plot in 1st and 2nd columns). We leverage this behavior to design the proxy for the image quality.

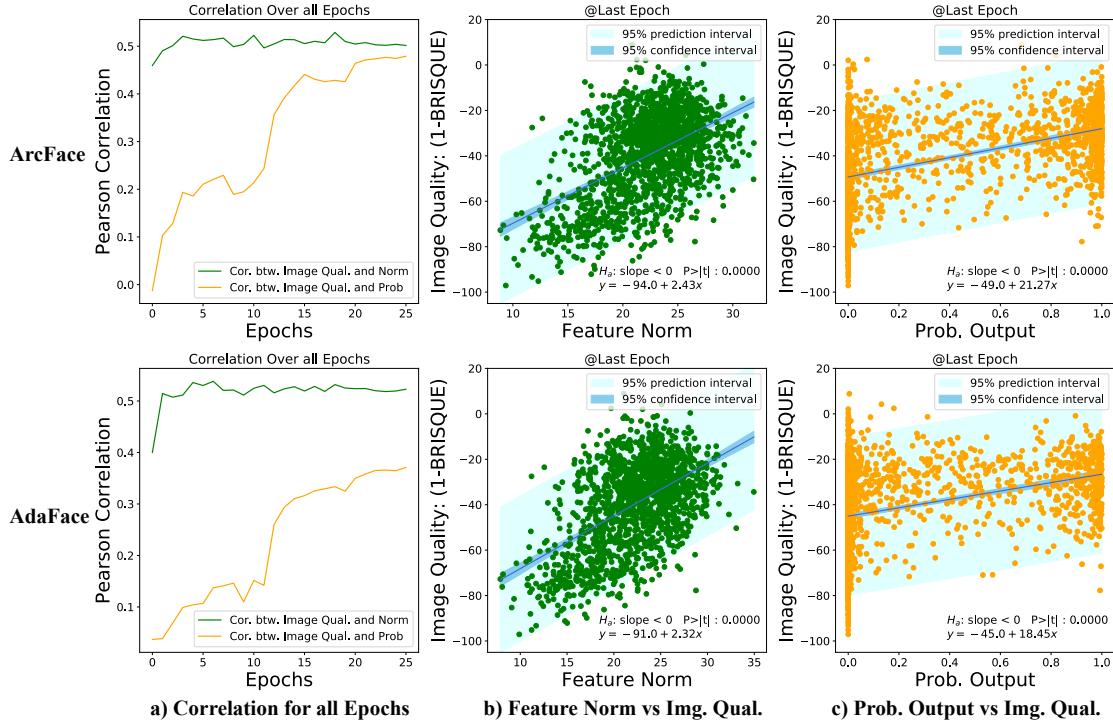


Figure 3. Comparison between ArcFace and AdaFace on the correlation between the feature norm and the image quality. We randomly sampled 1,534 images from the training dataset (MS1MV2 [7]) to show this plot.

We use three concepts (image quality, feature norm and sample difficulty) to describe a sample, as illustrated in Fig. 4. We leverage the correlation between the feature norm and the image quality to apply different emphasis to different difficulty of samples. In contrast, MagFace learns a representation that aligns the feature norm with recognizability. The term, *image quality* in MagFace paper [27] refers to the face recognizability, which is closer in meaning to the sample difficulty than the term, *image quality*, we use in our paper. Please refer to the Fig. 1 (a) and the first contribution claim of the MagFace paper [27]. Also note the difference in gradient flow through the feature norm, $\|z_i\|$. MagFace relies on learning the feature that has $\|z_i\|$ aligned with the recognizability of the sample, requiring the gradient to flow through $\|z_i\|$ during backpropagation. The loss function has the incentive to reduce the margin by reducing $\|z_i\|$. However, our objective is to adaptively change the loss function, itself, so we treat $\|z_i\|$ as a constant. Finally, from Tab. 3 of our main paper, AdaFace substantially outperforms MagFace, *e.g.* reducing the errors of MagFace on IJB-B and IJB-C relatively by 21% and 23% respectively.

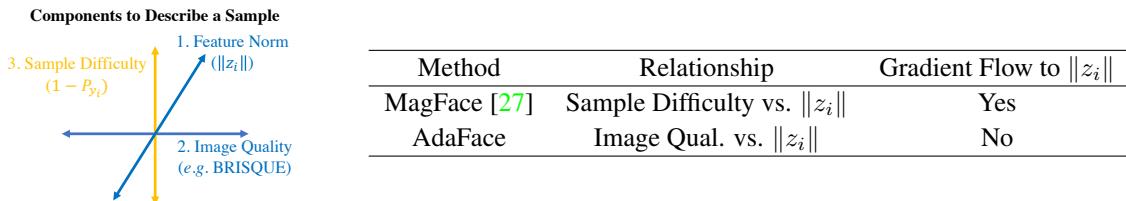


Figure 4. An illustration of different components to describe a sample and their usage in previous works.

B.2. Training Sample Visualization



Figure 5. Actual training data examples corresponding to 6 zones. A pretrained AdaFace model is used as a feature extractor.

We show some visualization of the actual training images. From the randomly sampled 1,534 images from the training dataset (MS1MV2 [7]), we divide the samples into 6 different zones. We plot the samples by $\cos \theta_{y_i}$ (decreasing) as the x-axis and the feature norm $\|z_i\|$ as y-axis in Fig. 5. We divide the plot into 6 zones and sample a few images from each group. Clearly, there are not many samples in the zones highlighted by the gray area (top right and bottom left). This indicates that the sample difficulty distribution is different for each level of feature norm. Furthermore, the samples in the dark green area are mostly unrecognizable images. AdaFace de-emphasizes these samples. Also, the samples in the bright pink area are more difficult samples than the dark pink area. AdaFace puts more emphasis on the harder samples when the feature norm is high. We would like to remind the readers that this figure may serve as *an empirical validation* of the two-dimensional face image categorization we made in Fig. 1 of the main paper.

B.3. Training Samples' Gradient Scaling Term for AdaFace

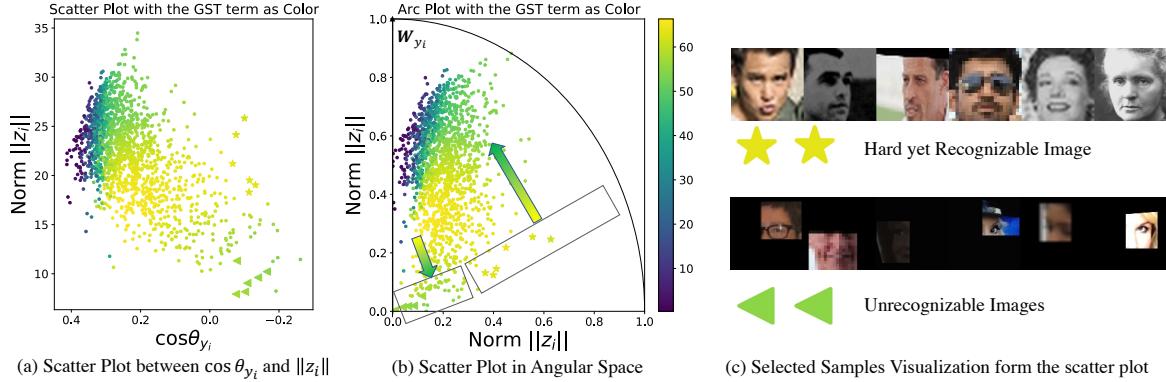


Figure 6. (a) Scatter plot of samples from Fig. 5 with the color as the GST term. (b): Scatter plot of the same 1,534 points in angular space. For each feature, the angle from W_{y_i} is calculated from $\cos \theta_{y_i}$ and the distance from the origin is calculated from $\|z_i\|$. Both terms are normalized for visualization. (c): Sample image visualization from the low norm and high norm regions of similar $\cos \theta_{y_i}$.

In Fig. 6 (a), we plot the actual GST term for AdaFace. We use the same 1,534 images from the training dataset (MS1MV2 [7]) as in Fig. 5. The color of points indicates the magnitude of the GST term. The purple points on the left side of the scatter plot are samples past the decision boundary. Therefore the magnitude of GST term is low. The effective difference in GST term for samples outside the decision boundary can be seen by the color change from green to yellow. Note that AdaFace de-emphasizes samples of low feature norm and high difficulty. This is shown in the lower right region of the plot. In Fig. 6 (b), we warp the plot into the angular space to make a correspondence with the Fig. 3 of the main paper, where we illustrate the GST term for AdaFace. We illustrate how actual training samples are distributed in this angular space. In Fig. 6 (b) and (c), we visualize two groups of images where one is from the low feature norm area (triangle) and the other is from the high feature norm area (star). AdaFace exploits images that are hard yet recognizable, as indicated by the yellow star regions, and lowers the learning signal from the unrecognizable images, as indicated by the green triangle regions.

B.4. Train Samples' Gradient Scaling Term Comparison with ArcFace

In Fig. 7, we compare the GST term placed on training samples. We have two groups of images. One group is comprised of unrecognizable images, shown under the red bar. Another group is comprised of hard yet recognizable images, shown under the green bar. Each bar corresponds to one training sample, and the height of the bar indicates the magnitude of the gradient scaling term (GST). For ArcFace shown on the left, the same level of GST is placed on all samples. However, in AdaFace, unrecognizable samples are less emphasized relative to the recognizable samples.

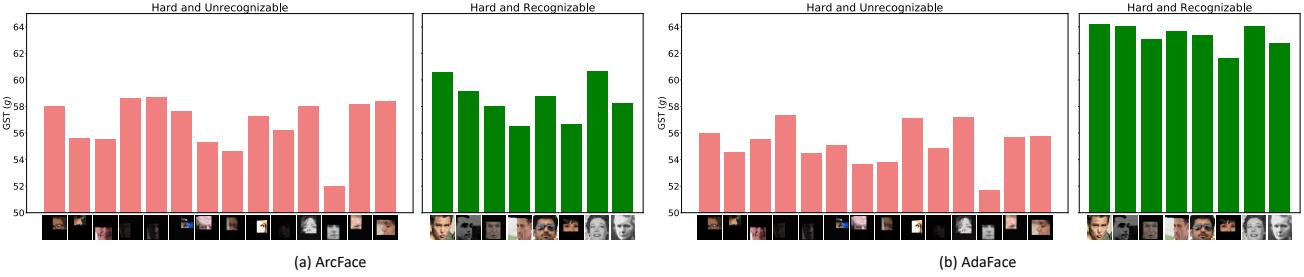
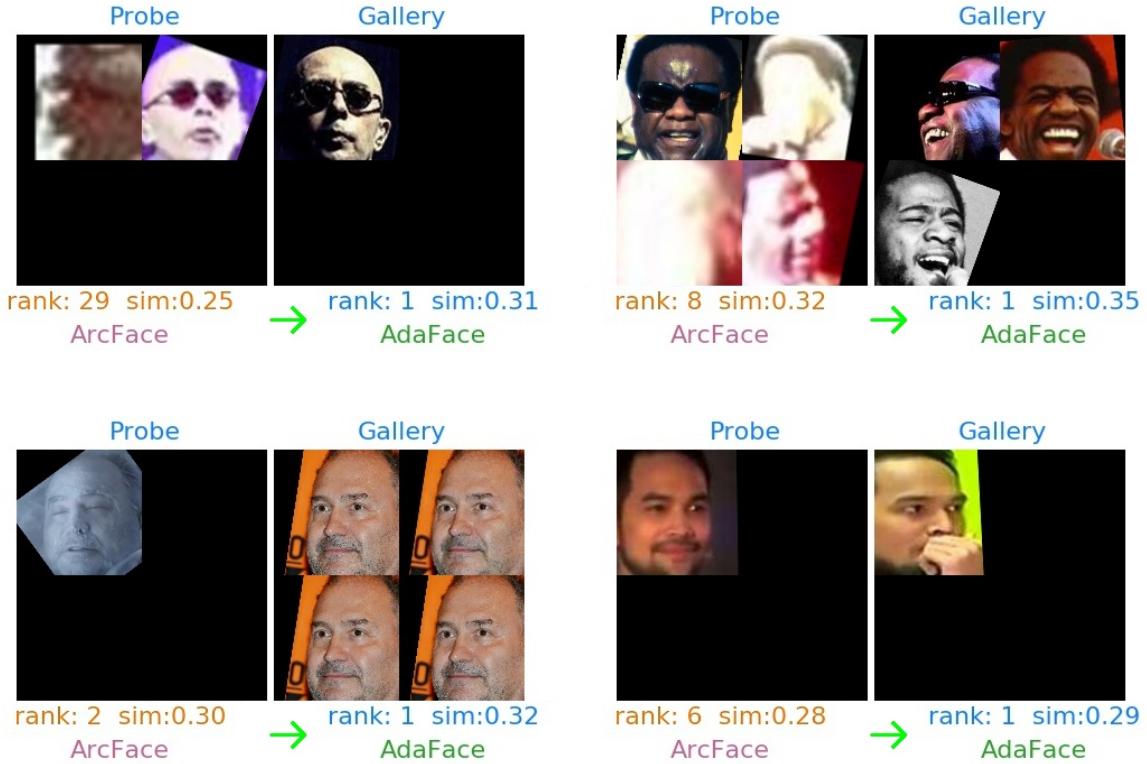


Figure 7. Comparison of the magnitude of GST term between ArcFace and AdaFace.

C. Visualization of Success and Failed Test Images

We show samples from IJB-C [26] dataset to show which samples are correctly classified in AdaFace, compared to ArcFace [7]. In each pair of probe and gallery images, we write the rank and the similarity score for both ArcFace and AdaFace. Rank= 1 is the correct match and a high similarity score is desired. Note that the majority of the cases where AdaFace successfully matches the hard samples for ArcFace are comprised of low quality samples. This shows that indeed AdaFace works well on low quality images.



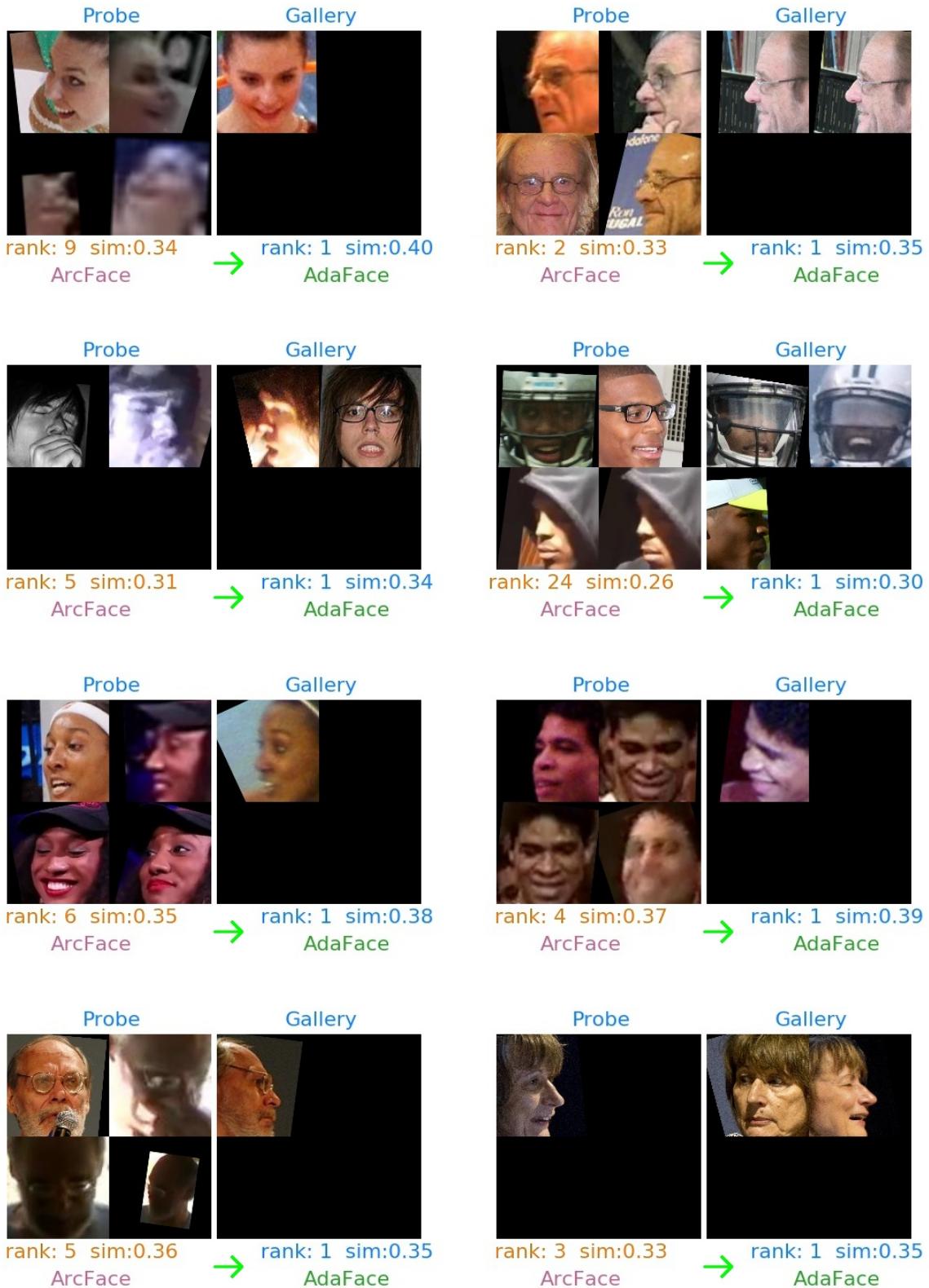


Figure 8. Examples from IJB-C [26] dataset, where ArcFace fails to identify the subject whereas AdaFace successfully finds the correct match between the probe and the gallery. On the left is the set of probe images and on the right is the set of gallery images.

D. Comparison with General Image-Quality Aware Learning Method

We compare our method with QualNet [19] (CVPR21) as a comparison with general image-quality aware learning method. The scope of general image-quality aware learning methods is not limited to face recognition, but the idea is applicable. In Tab. 2, we show the comparison with QualNet with models trained on CASIA-WebFace. AdaFace outperforms QualNet on the TinyFace test set. QualNet aligns the low quality (LQ) image feature distribution to the high quality (HQ) features' distribution via a fixed pretrained decoder. In contrast, AdaFace prevents LQ images from degrading the overall recognition performance by de-emphasizing heavily degraded LQ images. Since LQ facial images can often be devoid of identity, it helps to avoid overfitting on unidentifiable LQ images and learn to exploit the identifiable LQ images. This improves generalization across HQ and LQ.

Method	Training Set	Test set	Rank1	Rank5
QualNet [19] AdaFace	CASIA-Webface	TinyFace	35.54 44.39	44.45 47.23

Table 2. Closed set identification performance (ranked match rate) on TinyFace. For a fair comparison, we adopt the train/test setting of QualNet. QualNet results are directly taken from the CVPR21 paper.

E. Effect of Batch Size

Our image quality proxy $\widehat{\|\mathbf{z}_i\|}$ does not depend on the batch size due to the exponential moving average in Eq.17 of the main paper (rewritten below).

$$\widehat{\|\mathbf{z}_i\|} = \left[\frac{\|\mathbf{z}_i\| - \mu_z}{\sigma_z/h} \right]_{-1}^1, \quad (5)$$

$$\mu_z = \alpha \mu_z^{(k)} + (1 - \alpha) \mu_z^{(k-1)}. \quad (6)$$

To empirically show this, we train R50 model on MS1MV2 with the batch size of 128, 256 and 512 and report their performance on IJB-B TAR@FAR=0.01%. As shown in Tab. 3, the difference due to the batch size is minimal.

Method	Batch size 128	Batch size 256	Batch size 512
AdaFace	94.32	94.42	94.35

Table 3. Performance comparison by varying the batch size. This shows that AdaFace performance not subject to different batch sizes.

F. Implementation Details and Code

The code is released at <https://github.com/mk-minchul/AdaFace>. For preprocessing the training data MS1MV2 [7], we reference InsightFace [1] and InsightFacePytorch [2], for the backbone model definition, TFace [3] and for evaluation of LFW [14], CFP-FP [31], CPLFW [49], AgeDB [29], CALFW [50], IJB-B [41], and IJB-C [26], we use InsightFace [1]. For preprocessing IJB-S [17] and TinyFace [6], we use MTCNN [44] to align faces.