A Quality Aware Sample-to-Sample Comparison for Face Recognition

Mohammad Saeed Ebrahimi Saadabadi, Sahar Rahimi Malakshan, Ali Zafari, Moktari Mostofa, and Nasser M. Nasrabadi
me00018, sr00033, az00004, mm0251@mix.wvu.edu, nasser.nasrabadi@mail.wvu.edu

Abstract

Currently available face datasets mainly consist of a large number of high-quality and a small number of low-quality samples. As a result, a Face Recognition (FR) network fails to learn the distribution of low-quality samples since they are less frequent during training (underrepresented). Moreover, current state-of-the-art FR training paradigms are based on the sample-to-center comparison (i.e., Softmax-based classifier), which results in a lack of uniformity between train and test metrics. This work integrates a quality-aware learning process at the sample level into the classification training paradigm (QAFace). In this regard, Softmax centers are adaptively guided to pay more attention to low-quality samples by using a quality-aware function. Accordingly, QAFace adds a quality-based adjustment to the updating procedure of the Softmax-based classifier to improve the performance on the underrepresented low-quality samples. Our method adaptively finds and assigns more attention to the recognizable low-quality samples in the training datasets. In addition, QAFace ignores the unrecognizable low-quality samples using the feature magnitude as a proxy for quality. As a result, QAFace prevents class centers from getting distracted from the optimal direction. The proposed method is superior to the state-of-the-art algorithms in extensive experimental results on the CFP-FP, LFW, CPLFW, CALFW, AgeDB, IJB-B, and IJB-C datasets.

1. Introduction

Recent advances in FR performance can be credited to introduction of novel network architectures, large-scale datasets, and new loss functions [23]. Regarding the architecture, ResNet and its variants are mostly used as the backbone for extracting features from the face images [11]. In terms of datasets, large-scale publicly available training data leads to unprecedented improvement in FR performance [13]. Recent attempts on FR are mainly focused on manipulating the training criteria [25, 40, 11, 32, 39, 42]. In this manner, Softmax with a cross-entropy loss, i.e., sample-to-center comparison, is the most popular criterion for FR, (i.e., classification) [25]. In the classification framework, the weights connecting the penultimate layer output (i.e., feature) to the classification layer represent the centers of Softmax classes [2].

Since FR is an open-set problem, during testing, sample-to-center (dis)similarity of the Softmax is irrelevant and sample-to-sample (dis)similarity matters. In order to unify the train and test similarity metrics, pioneering works [32, 43], devised metric-based loss functions based on sample-to-sample comparison [37, 43]. These metric-based losses try to directly minimize a distance metric when two samples come from the same identity (positive pair); otherwise (negative pair) impose a margin [32]. However, the sample-wise comparison highly depends on the pair selection strategy and requires a sophisticated mining method [22]. Besides, in large-scale datasets [11] with thousands of identities and millions of samples, there is a combinatorial explosion in the number of possible pairs, leading to costly pair selection, unstable training, and slow convergence [36].

Several studies show that projecting features and class...
centers to the unit-hypersphere improves the discriminative power of representation learned by Softmax [40, 39, 25]. In this manner, Softmax classifies images using the angular distance between the feature representations and Softmax centers [39]. An angular margin is then integrated to the Softmax loss to further enforce intra-class compactness and inter-class separability [25, 40, 23, 46]. The angular-based Softmax loss functions are more stable than metric-based, i.e., no pair selection, and the number of centers is much fewer than the number of samples [24]. As a result, angular-based Softmax losses have become the state-of-the-art method for training FR frameworks [11, 28].

In sample-to-center training paradigms, every identity is represented as a deterministic point in high-dimensional latent space [6], i.e., centers. As a result, their performance degrades when there is a large disparity between training and testing data [34]. Although augmentation may narrow the gap between training and evaluation data distribution, it increases the occurrence of unrecognizable samples and overfitting [26], as illustrated in Fig. 2. To alleviate this problem, authors in [28] propose to use feature magnitude as a proxy to measure the image quality. Also, there are methods to estimate the distribution of each class instead of presenting them as a single deterministic point [6, 34]. Despite the performance improvement, these approaches do not propose a solution to the overfitting problem nor guarantee (dis)similarity between (negative)positive samples.

Usually, there are three different types of samples in FR datasets [28]. First, samples that are easy to learn by the model. These easy samples usually have good quality-related factors such as high resolution [28]. Second, images that have low-quality, but recognizable (hard-samples) [7]. Third, unrecognizable samples that even humans cannot correctly recognize their identity [28, 7]. As shown in Fig. 1, unrecognizable samples have larger angular disparity, \( \theta_2 \) and \( \theta_4 \), compared to low-quality instances, \( \theta_2 > \theta_1 \) and \( \theta_4 > \theta_3 \). The primary idea in recent works is to increase the margin constraints as the angular disparity between sample and the Softmax center increases [46, 23, 18]. However, the recognizability of samples and sample-wise (dis)similarity are not considered by any of the mentioned methods. Therefore, the model tries to reduce the training loss by overfitting on unrecognizable samples, which harms model generalization [13].

Recently, authors in [12] integrate sample-to-sample comparison to the Softmax via injecting sample representations to the centers. Although VPL [12] brings sample-to-sample comparison to the Softmax framework, the prior assumption is that the unrecognizable samples do not distract the learning. Due to the larger angular disparity, injecting without considering the recognizability of samples puts more emphasis on unrecognizable samples during injection, see Fig. 1. Adding variations toward the unrecognizable samples harms the model learning paradigm and distract the Softmax centers from the optimal direction. The injection process directly changes centers. Therefore, it is important to push the centers to a valid direction.

Sample selection strategy is indispensable in every sample-wise FR training paradigm [32, 23]. In this work, we try to weigh samples based on their recognizability and quality. In this manner, the sample-wise part of the proposed method (QAFace), injection, benefits from recognizable low-quality (hard) samples. During training, QAFace effectively ignores the unrecognizable samples and prevents the class centers from being distracted from the optimal direction. At the same time, QAFace emphasizes on low-quality samples considering them as hard samples. Moreover, since high-quality samples are being well explored by sample-to-center part of the training, the proposed method puts less attention on high-quality samples during their injection. Compared to [12], our method adds no extra memory consumption and sampling strategy to the training. The presented model queues sample representation using MOCO [16] to maintain both samples and centers on the same embedding space. Contributions of this work can be summarized as follows:

- we use informative hard samples (low-quality instances) to introduce sample-wise comparison to the angular-margin Softmax loss.
- we propose a new quality-based weighting function that can effectively de-emphasize unrecognizable samples based on the magnitude of their feature representation as a proxy of image quality.
- we leverage hard samples to add uncertainty to the Softmax centers toward the direction of hard samples.

2. Related Works

2.1. FR Loss Functions

Most of the previous FR methods were established on a metric-learning loss function, such as triplet [32] or contrastive loss [8, 30]. These loss functions were based on sample-to-sample comparison in Euclidean space. Then [43] enhanced the intra-class similarity via proposing a new loss to directly minimize intra-class distance while doing classification. In this manner, the main challenges toward general FR were the necessity of sample mining, lack of generalization, and feature collapsing problem [40, 32, 15]. More recently, studies showed that applying Softmax to the angular space enhances the discriminability of features [25, 40, 11, 43]. Consequently, pioneering works of [39, 40, 11], introduced intuitive loss functions by applying three different types of margin to the angular space of Softmax: 1) multiplicative angular margin, 2) cosine margin,
Figure 2. Same Augmentation, i.e., down-sampling and random cropping, results in different recognizability among samples. Green border shows the recognizable samples, and red shows unrecognizable samples.

and 3) additive angular margin, resulting to state-of-the-art performance.

2.2. Angular Margin Variations

Recent studies explore the effects of adaptive angular margin on the learning paradigm of the network [46, 23, 26]. Liu et al. [23] propose to adaptively tune the margin value to put more constraint on the tail classes. The role of negative samples in obtaining more discriminative features is investigated in [18]. Authors of [14], add new term to the angular-margin loss function to supervise the uniformly spreading of class centers on the unit hyper-sphere. MagFace [28] establishes the norm of features as a proxy of sample recognizability. Image recognizability increases as the feature norm increase [28]. MagFace assigns high angular margin on the high-norm feature in the premises of pushing those samples to be closer to their class center. The drawback is that it fails to put emphasis on the valuable hard samples. Furthermore, none of the mentioned methods guarantee the samples-wise similarity. Also, representing each identity with a single deterministic point, i.e., center, in high-dimensional space results in the performance drop when testing data has a large disparity with training samples [34].

2.3. Probabilistic Face Modeling

Probabilistic face modeling is well-established in face template/video matching [5, 3]. In these works, a series of samples is used as the input rather than a single face image. Shi et al. for the first time integrate uncertainty into a single image FR [34]. PFE [34] represents each image as a Gaussian distribution. The mean and variance of the Gaussian reflect the “most likely latent feature”, and “the uncertainty in the feature values”, respectively [34]. The goal is to add uncertainty to the model to boost performance for unseen data with large disparity [6]. Instead of adding uncertainty to each image representation, VPL [12] assigns a distribution to each class within the classification framework. Specifically, VPL injects the class instances to the corresponding classifier to bring more uncertainty to centers and, at the same time, integrate sample-to-sample comparison to the classification paradigm. However, it fails to consider the image recognizability measure. Considering the whole memorizing process in [12], projecting all the representations to the unit hyper-sphere results in an equal contribution of different instances in the memory. Therefore, because of large angular disparity with centers, unrecognizable samples distract centers from their optimal direction, see Fig. 1 (b).

3. Proposed Method

In this section, we begin by analyzing the Softmax-based loss functions. Then, we further explain the integration of Softmax-based classifier with the sample-to-sample comparison. We devise a new injection function to integrate a quality-aware sample-to-sample comparison to the classification framework (QAFace). Finally, we investigate the capability of our method to ignore the unrecognizable samples and the complementary role of our quality-aware injection to the learning signal of Softmax-based loss function.

3.1. Preliminaries

Most of the deep visual recognition modules, including FR, can be regarded as the stack of non-linear feature extractor layers (backbone), together with a classifier which is usually a Softmax layer [4]. Both the backbone and classifier will be trained end-to-end using a back-propagation algorithm. The Softmax training criterion can be formulated as follows [1, 20]:

$$L = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{W^T_i x_i + b_j}}{\sum_{j=1}^{C} e^{W^T_j x_i + b_j}}, \quad (1)$$

where $W_j \in \mathbb{R}^d$ is $j$-th classifier (center), $d$ is the feature dimension, and $b_j$ is the bias for $j$-th Softmax output. $x_i$ is the learned representation 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.

The angular distribution of representations learned via the Softmax loss, $x_i$, suggests using cosine distance as the metric rather than Euclidean distance [39]. Hence, a modified Softmax loss was defined by projecting both centers and representations to the unit-hypersphere [39, 25].

$$L' = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{s (\cos(\theta_{y_i}))}}{e^{s (\cos(\theta_{y_i}))} + \sum_{j=1}^{C} e^{s (\cos(\theta_j))}} \quad (2)$$

where $\cos(\theta_{y_i})$ reflects the cosine similarity between $x_i$ and $w_{y_i}$, and $\cos(\theta_j)$ denote similarity between $x_i$ and $w_j$ (negative centers). $s$ is introduced as the scaling hyper-parameter which affects the curves of the output [46], see Fig. 6. In common FR practice, biases are removed from Eq. 1 because they are learned for close-set recognition and cannot be generalized to open-set testing.
To enhance the intra-class compactness and inter-class separability, authors in [39, 40, 11] developed intuitive loss functions by applying three different types of margins to Eq. 2.

\[
L' = \frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{s(\cos(m_S \theta_i + m_A) - m_C)}}{\sum_{j=1}^{C} e^{s(\cos(m_S \theta_i + m_A) - m_j)}} + \sum_{j=1}^{C} \lambda M e^{s(\cos(\theta_j))},
\]

Equation 3 represents all men-

\[
\text{SphereFace [25] introduced the multiplicative angular margin to modify decision boundaries from } \cos(\theta_1) = \cos(\theta_2) \text{ to } \cos(m_S \theta_1) = \cos(\theta_2). \text{ Where } (\theta_1, \theta_2) \text{ represent angle between } x_i \text{ and } (W_{yi})W_j. \text{ Their modification to Softmax does improve the result; however, the proposed loss function is computed through a series of approximations, which results in unstable training [40]. CosFace modified the decision boundary to } \cos(\theta_1) + m_C = \cos(\theta_2) \text{ and ArcFace changed it to } \cos(\theta_1 + m_A) = \cos(\theta_2). \text{ Eq. 3 represents all mentioned modifications. Where } m_S, m_C, \text{ and } m_A \text{ are margins introduced by SphereFace [25], CosFace [40], and ArcFace [11], respectively. Despite the remarkable improvement, the sample-wise (dis)similarity is not considered in any of these modifications.}

Also, presenting each identity with a single deterministic point in embedding space results in performance degradation when there is a significant disparity between training and testing data [34]. For better illustration, we experiment by manually degrading five high-quality testing datasets. Comparing the results of ArcFace and VPL in Table 1, less performance gap in VPL shows that adding uncertainty to the Softmax centers results in better handling of quality disparity between train and test datasets. Comparing the results of QAFace with VPL, it is shown that our proposed method can further reduce the gap between representation of high and low quality samples by putting more emphasis on the low-quality samples during the injection.

### 3.2. Classification-Based Gradient

We can divide a Softmax-based FR method into its backbone and classifier components. Hence, here we study the updating of the backbone and center of Softmax separately.

For the backbone, we show the gradient with regard to its output (feature), i.e., \( \frac{\partial L}{\partial x_i} \). By omitting bias in the Eq. 1 the derivatives to \( j \)-th class center and \( i \)-th sample’s feature are:

\[
\frac{\partial L}{\partial x_i} = ((p_{i,y_i} - 1)W_{yi}) + \sum_{j=1}^{C} p_{i,j}W_j,
\]

and

\[
\frac{\partial L}{\partial W_j} = \sum_{i=1}^{N} ((p_{i,y_i} - 1)x_i) + \sum_{i=1}^{N} p_{i,j}x_i,
\]

where \( p_{i,j} = \frac{e^{sW_j^T x_i}}{\sum_{j=1}^{C} e^{sW_j^T x_i}} \). Eq. 4 shows that from the backbone perspective, the network is being updated toward increasing the similarity between features and the positive class centers while decreasing similarity with negative centers. Moreover, Eq. 5 demonstrates that centers update toward being more similar to their corresponding class instances and away from samples of other classes. Hence, both backbone and centers are moving toward each other, and sample-wise (dis)similarity is being supervised indirectly.

### 3.3. Sample-Wise Similarity with Softmax

In order to directly supervise sample-wise (dis)similarity, [12] injects samples feature to their corresponding class center. To this end, a memory, \( M \), is constructed, which memorizes the positive features of each class. The memory has the same shape as the Softmax output (feature), i.e., \( f(x_i) \). In each iteration, the class centers represent the centers and accumulated features of the hard samples of previous iterations. The representations for injecting to the centers are obtained from the momentum backbone. Therefore, sample-to-sample (dis)similarity is being directly supervised. \( \lambda \) is a hyperparameter to adjust the amount of the injection and should

![Figure 3. Left: the general architecture of the proposed method. In each iteration, the class centers represent the centers and accumulated features of the hard samples of previous iterations. The representations for injecting to the centers are obtained from the momentum backbone. Right: shows the updating of the Momentum Backbone parameters.](image-url)
3.4. Quality Aware Sample Injection

To address the mentioned shortcomings of the classification framework, prevent Softmax centers from being distracted, and explore recognizable low-quality samples, we propose QAFace, a quality-aware injection procedure. Proposed method ignores unrecognizable samples and, at the same time, uses recognizable low-quality samples to add more valid uncertainty to the centers. The injection process is as follows:

\[
W_{y_i} = W_{y_i} + f(||x_i||) \cdot \frac{x_i}{||x_i||},
\]

where \( ||x_i|| \) is the normalized version of feature magnitude. We normalize the feature magnitude via batch statistics: \( \mu \) and \( \sigma \). To relax \( \mu \) and \( \sigma \) from the batch size, we calculate them in an exponential moving average over the training iterations. \( f(||x_i||) \) projects \( ||x_i|| \) to have zero value for the features that have \( ||x_i|| \) lower than a threshold, \(-\tau\), otherwise positive.

\[
\sigma_t = \alpha \sigma_{t-1} + (1 - \alpha) \sigma_{t-1},
\]

\[
\mu_t = \alpha \mu_{t-1} + (1 - \alpha) \mu_{t-1},
\]

\[
||x_i|| = \frac{||x_i|| - (\mu)}{\sigma},
\]

Eqs. 10 and 11 together work in a way that 1) samples with \( ||x|| \) lower than \(-\tau\) would not affect the centers, 2) recognizable but low-quality samples will be emphasized during training, and 3) high-quality samples will receive less attention in comparison to recognizable low-quality samples. Using informative samples in metric-based FR training paradigm has been well-established [32]. Therefore, employing informative samples to add sample-wise comparison to the classification framework is of most importance. Our proposed algorithm adaptively: 1) assigns more weight to the recognizable low-quality (hard) samples, 2) ignores unrecognizable samples, and 3) puts less attention on easy high-quality samples. Hence, our approach can be regarded as a kind of hard-sample mining, but without adding any computational burden on hard sample selection. It is worth mentioning that, \( f(||x_i||) \cdot \frac{x_i}{||x_i||} \) is happening during the memorizing the representations in the memory. Consequently, we can re-write the Eq. 10 as:

\[
W_{y_i} = W_{y_i} + M(||x_i||, x_i).
\]

3.5. Distractor Samples

The major advantage of the QAFace over [12] is the ability to identify the unrecognizable from recognizable samples and emphasize the recognizable low-quality samples.
Table 2. Performance (%) comparison of our method with other recent algorithms. 1:1 verification accuracy is reported on LFW, CFP-FP, CPLFW, AgeDB.

<table>
<thead>
<tr>
<th>Method</th>
<th>Venue</th>
<th>Verification accuracy</th>
<th>TAR@FAR=1e−4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>LFW</td>
<td>CFP-FP</td>
</tr>
<tr>
<td>Wang et al. [40]</td>
<td>CVPR18</td>
<td>99.81</td>
<td>98.12</td>
</tr>
<tr>
<td>Deng et al. [11]</td>
<td>CVPR19</td>
<td>99.83</td>
<td>98.27</td>
</tr>
<tr>
<td>Sun et al. [38]</td>
<td>CVPR20</td>
<td>99.73</td>
<td>96.02</td>
</tr>
<tr>
<td>Deng et al. [9]</td>
<td>ECCV20</td>
<td>99.80</td>
<td>98.80</td>
</tr>
<tr>
<td>Wang et al. [41]</td>
<td>AAAI20</td>
<td>99.80</td>
<td>98.28</td>
</tr>
<tr>
<td>Huang et al. [18]</td>
<td>CVPR20</td>
<td>99.80</td>
<td>98.37</td>
</tr>
<tr>
<td>Kim et al. [22]</td>
<td>ECCV20</td>
<td>99.85</td>
<td>98.63</td>
</tr>
<tr>
<td>Shi et al. [35]</td>
<td>CVPR20</td>
<td>99.78</td>
<td>98.64</td>
</tr>
<tr>
<td>Kim et al. [21]</td>
<td>CVPR20</td>
<td>99.85</td>
<td>98.63</td>
</tr>
<tr>
<td>Chang et al. [6]</td>
<td>CVPR20</td>
<td>99.83</td>
<td>98.78</td>
</tr>
<tr>
<td>Meng et al. [28]</td>
<td>CVPR21</td>
<td>99.83</td>
<td>98.46</td>
</tr>
<tr>
<td>Deng et al. [12]</td>
<td>CVPR21</td>
<td>99.83</td>
<td>99.11</td>
</tr>
<tr>
<td>QAFace</td>
<td></td>
<td>99.85</td>
<td>99.21</td>
</tr>
</tbody>
</table>

Figure 6. Curve of $p_{i,j} = \frac{e^{W^T x_i}}{\sum_{j=1}^{C} e^{W^T x_i} e^{\cos(\theta_j) x_i}}$ when $\cos(\theta_j)$ is fixed and $\cos(\theta_{y_i})$ changes from -1.0 to 1.0.

Figure 6. Curve of $p_{i,j} = \frac{e^{W^T x_i}}{\sum_{j=1}^{C} e^{W^T x_i} e^{\cos(\theta_j) x_i}}$ when $\cos(\theta_j)$ is fixed and $\cos(\theta_{y_i})$ changes from -1.0 to 1.0.

in the injection process. To this end, we employ the magnitude of the feature vector as a proxy for the input sample recognizability [28]. We perform an experiment to demonstrate how the feature magnitude is affected by recognizability and how our method can use hard samples to reduce the gap between the representation of low and high-quality samples. We randomly select a subset of 10K images of the training data and down-sampled them to the different levels (8 x 8, and 16 x 16). Then, at the end of every epoch, we save the magnitude of the representations obtained from original and down-sampled images, see Fig. 4. From the results shown in Fig. 4 (top) we can observe that as the recognizability increases, the magnitude of representations increases as well, i.e., green > blue > orange.

Complete overlapping of the distribution of down-sampled with the original instances is an ideal scenario, which means that the model became quality-agnostic. The model progressively learns to increase the lower bond of feature magnitude to narrow the gap between original and low-quality samples. At the early stages of the training, the full range of feature magnitude is around 25 (from 5 to 30). Then in the final stages, the range narrows to 15 (from 15 to 30). Also, the mean of the original image distribution (green) is always around 23, which shows that our proposed method can effectively involve low-quality samples in training without reducing the performance on the high-quality samples.

Another observation from Fig. 4 (bottom) is the necessity of normalizing the magnitude. Without applying Eq. 10 on the feature magnitude, the threshold for ignoring unrecognizable samples changes as the training progresses. Eq. 10 omits the bias from the distribution of features magnitude caused by the training stage. Therefore, we can choose a fixed $\tau$ for ignoring the unrecognizable and emphasizing the hard samples.

### 3.6. Complement to Angular-Margin Gradient

Unlike triplet and contrastive losses, softmax-based losses are not subject to explicit easy/hard sample mining [42, 31]. In this section, using a simple toy example, we show that the Softmax-based losses implicitly benefit easy/hard sample mining by their gradient. Additionally, we elaborate on the ability of the proposed $f(|x_i|)$ to complement the Softmax learning signal (gradient). Consider a four-identity classification. For a given sample $x_i$ with ground truth identity $y_i = 4$, the logits are $\cos(\theta_1), \cos(\theta_2), \cos(\theta_3), \cos(\theta_4)$. In Fig. 7, we plot the loss value for a fixed $\cos(\theta_j), j \neq 4$ while $\cos(\theta_{y_i}=4)$ changes from -1 to 1. The first observation from Fig. 7 (right) is that the scaling parameter $s$ is tuning the sensitivity of the loss function. As the scaling value increases, the slope of the loss function (gradient) increases [46].

Moreover, $s$ directly influences the point that samples would be recognized as easy. Easy samples would barely experience change, i.e., low slope, while hard samples receive high gradient value, i.e., high slope. Also, in Fig. 7 (left), we demonstrate that the drawback here is the mono-
Figure 7. Curves of $p_{yi} = e^{\alpha \cos(\theta_{yi})}$, when $yi = 4$ and $\cos(\theta_{yi})$ changes from -1 to 1.

Figure 8. Illustration of sample-to-center cosine similarity for two randomly selected subjects. High-norm samples are very similar to the class center. Low-norm samples have lower similarity with Softmax center.

tonicity of the gradient among the hard samples. In other words, some samples are unrecognizable; however, their gradient is equivalent to those of informative but hard samples. Consequently, the model tries to overfit the unrecognizable samples because there is no identity information on those instances [35]. Our proposed method tries to compensate for this effect by ignoring the unrecognizable samples during the injection. It does not further involve the unrecognizable samples in the injection process and ignores them using the proposed feature weighting paradigm. As a result, our method justifies the classifier direction to tolerate more variation toward hard and informative samples and plays a complementary role to the Softmax-based learning signal.

4. Experiments

4.1. Datasets

We employ Webface4M [49] as our training data, which contain 4 million samples of around 200,000 identities, Table 2. For evaluation of our method, we use CFP-FP [33], CPLFW [47], CALFW [48], LFW [17], AgeDB [29], IJB-B [44], and IJB-C [27]. Based on the datasets evaluation protocols, we report 1:1 verification accuracy for CFP-FP, LFW, CPLFW, CALFW, and AgeDB datasets. For IJB-B [44] and IJB-C [27], we report the True Acceptance Rate (TAR) over the False Acceptance Rate of $1e-4$.

4.2. Training Settings

We use [10] to detect five landmarks in each image. Then images are aligned and rescaled to $112 \times 112$, following the setting in [11]. We adopt ResNet [11] for the backbone. The model is trained for 24 epochs with Arcface loss. 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, and the momentum is 0.9. During training, the mini-batch size on each GPU is 512, and the model is trained using two Quadro RTX 8000. Following [16], $\gamma$ in Fig. 3 is 0.99. In calculating the $\mu$ and $\sigma$, $\alpha$ in Eq. 8 and 9, is 0.99. Given a pair of images, the cosine distance between the representations is the metric during inference.

4.3. Ablation Study

4.3.1 Impact of the Memory Length

In [16], the memory is a dynamic queue of representations. The whole memory has a length of $|M|$, and queuing new samples results in de-queuing the oldest samples. Here the length of memory is equal to the number of classes. Therefore, we should memorize the last iteration in that every instance in the memory was updated. In this way, we can prevent from employing outdated representations in the injection. For instance, if the training is on the iteration $I$ and a specific instance in memory was updated on $I - \Delta t$, if $\Delta t$ is larger than a threshold, that specific memory instance cannot be used during the injection. It is shown that in the early epochs, the changes in the feature space are drastic; after that, it is negligible [12]. Therefore, we start the injection after the fourth epoch of training. Table 3 shows the ablation experiments on $\Delta t$. In these experiments, we fixed $\tau$ to 2. We increase $\Delta t$ with the interval of 500 iterations. As illustrated in Table 3, the performance constantly improves from $\Delta t = 0$ to $\Delta t = 1000$, and after that, it starts to degrade.

<table>
<thead>
<tr>
<th>$\Delta t$</th>
<th>Verification Accuracy</th>
<th>TAR @ FAR: $1e-4$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LFW</td>
<td>CFP-FP</td>
</tr>
<tr>
<td>0</td>
<td>99.71</td>
<td>98.40</td>
</tr>
<tr>
<td>500</td>
<td>99.80</td>
<td>98.81</td>
</tr>
<tr>
<td>1000</td>
<td>99.85</td>
<td>99.21</td>
</tr>
<tr>
<td>1500</td>
<td>99.78</td>
<td>99.01</td>
</tr>
<tr>
<td>2000</td>
<td>99.69</td>
<td>98.33</td>
</tr>
</tbody>
</table>
4.3.2 Impact of Threshold (τ)

We fix the memory length to 1000 iterations. Then we investigate different values for the threshold (τ) in Eq. 10. As illustrated in Table 4, the performance on IJB-B and IJB-C constantly increases with changing τ from zero to 2. At τ = 0, only samples with |[ξ]| above the zero are involved in the injection. Consequently, the model performance decreases when the input comes from datasets like IJB-B and IJB-C, which contain low-quality samples [35]. On the other hand, the results in clean datasets like CFP, CPLFW, and LFW are reasonably good [35].

4.4. Impact of Augmentation

For data augmentations, we used random cropping and down-sampling [19, 45]. On-the-fly data augmentation provides more diverse training data. However, as illustrated in Fig. 2, it increases the occurrence of unrecognizable samples. We perform experiments on our method with and without the presence of data augmentation. Accordingly, we can show that our method can effectively ignore unrecognizable samples and, at the same time, benefit from more training instances, see Fig. 5. As shown in Table 6, the model gains performance on the IJB-B and IJB-C datasets by increasing the probability of augmentations. As these datasets contain low-quality samples, down-sampling leads to more performance improvement than random cropping.

4.5. Comparison with state-of-the-art

Table 2 shows the proposed method’s performance compared to the state-of-the-art algorithms. For better clarification, we explain our observation in two parts. For the results on LFW, CPLFW, CALFW, CFP-FP, and AgeDB, it is essential to mention that QAFace is built upon putting more emphasis on the low-quality samples and making these samples’ representation more similar to the high-quality samples’ features. Consequently, the performance gain in these datasets is marginal, as they contain almost high-quality samples [35]. Although the performance is saturated in most of these datasets, our method strives to increase the 1:1 verification accuracy for the CFP-FP and CPLFW datasets. The IJB-B and IJB-C datasets are more challenging and have images/frames with diverse quality. Results on the IJB-B and IJB-C datasets show the superiority of our approach in more general face recognition. As these datasets contain low-quality and high-quality images, the performance gain in these datasets is more evident. In IJB-B, compared to VPL, QAFace improves the TAR at FAR=1e − 4.

5. Conclusion

This work argues the importance of integrating sample-wise similarity to the Softmax framework. Also, we showed that existing angular-margin-based loss functions could be distracted by the unrecognizable samples in the dataset. Inspired by the well-established idea of hard sample mining in the sample-to-sample comparison framework, we proposed a weighting scenario to ignore unrecognizable samples and emphasize recognizable low-quality samples during the injection. We empirically showed the effect of ignoring unrecognizable samples by improving the similarity score between positive samples in the IJB-C dataset. Also, We analyzed the proposed function for weighting. Our proposed approach is based on the simple idea of using the norm of features as the proxy for the recognizability of face images. Furthermore, we empirically showed the effect of the quality of face images on the magnitude of features. We demonstrated that there is a direct proportion between the face image quality and the magnitude of its representation. Our approach could successfully outperform all of its competitors in five out of seven evaluation benchmarks, including the IJB-B and IJB-C datasets.

References


Table 4. Ablation of τ. The metrics are the same as Table 2.

<table>
<thead>
<tr>
<th>τ</th>
<th>Verification Accuracy</th>
<th>TAR@FAR=1e-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>LFW</td>
<td>CFP-FP</td>
<td>CPLFW</td>
</tr>
<tr>
<td>0</td>
<td>99.86</td>
<td>99.23</td>
</tr>
<tr>
<td>1</td>
<td>99.83</td>
<td>99.11</td>
</tr>
<tr>
<td>2</td>
<td>99.85</td>
<td>99.21</td>
</tr>
<tr>
<td>3</td>
<td>99.83</td>
<td>99.02</td>
</tr>
<tr>
<td>4</td>
<td>98.76</td>
<td>98.45</td>
</tr>
</tbody>
</table>

Table 5. Ablation of augmentation probability, TAR@FAR=1e-4.

<table>
<thead>
<tr>
<th>probability</th>
<th>cropping</th>
<th>down-sampling</th>
<th>IJB-B</th>
<th>IJB-C</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>-</td>
<td>-</td>
<td>95.34</td>
<td>96.60</td>
</tr>
<tr>
<td>0.1</td>
<td>-</td>
<td>✓</td>
<td>95.56</td>
<td>96.89</td>
</tr>
<tr>
<td>0.1</td>
<td>✓</td>
<td>-</td>
<td>95.41</td>
<td>96.65</td>
</tr>
<tr>
<td>0.2</td>
<td>✓</td>
<td>✓</td>
<td>95.57</td>
<td>96.95</td>
</tr>
<tr>
<td>0.2</td>
<td>✓</td>
<td>✓</td>
<td>95.63</td>
<td>96.91</td>
</tr>
<tr>
<td>0.2</td>
<td>✓</td>
<td>✓</td>
<td>95.45</td>
<td>96.67</td>
</tr>
<tr>
<td>0.2</td>
<td>✓</td>
<td>✓</td>
<td>95.67</td>
<td>97.20</td>
</tr>
</tbody>
</table>


