DiscFace: Minimum Discrepancy Learning for Deep Face Recognition

Insoo Kim[1], Seungju Han[1], Seong-Jin Park[1], Ji-won Baek[1], Jinwoo Shin[2],
Jae-Joon Han[1], and Changkyu Choi[1]

[1] Samsung Advanced Institute of Technology (SAIT), South Korea
[2] Korea Advanced Institute of Science and Technology (KAIST), South Korea
{insoo1.kim,sj75.han,sj210.park,jw0328.baek,jae-joon.han,
changkyu_choi}@samsung.com, jinwoos@kaist.ac.kr,

Abstract. Softmax-based learning methods have shown state-of-the-art performances on large-scale face recognition tasks. In this paper, we discover an important issue of softmax-based approaches: the sample features around the corresponding class weight are similarly penalized in the training phase even though their directions are different from each other. This directional discrepancy, i.e., process discrepancy leads to performance degradation at the evaluation phase. To mitigate the issue, we propose a novel training scheme, called minimum discrepancy learning that enforces directions of intra-class sample features to be aligned toward an optimal direction by using a single learnable basis. Furthermore, the single learnable basis facilitates disentangling the so-called class-invariant vectors from sample features, such that they are effective to train under class-imbalanced datasets.

1 Introduction

Recently, deep learning models have been utilized to extract robust and accurate features with state-of-the-art performance for various computer vision tasks. In particular, a multitude of efforts has been devoted to developing a face recognition model that could handle unconstrained variations in large-scale datasets, e.g., variations in pose, illumination, occlusion, facial expression, blur, and low resolution. Convolutional neural networks (CNNs) have shown remarkable face recognition performances by extracting discriminative features. Such breakthroughs were achieved by adopting different effective loss functions tailored for variation-robust face recognition [1–10].

In this paper, we are particularly interested in training on extreme class-imbalanced datasets; the training set consists of an enormous number of classes, with an extremely small number of data per class. In earlier years, deep metric learning methods achieved promising results under the class-imbalanced datasets, by learning face embeddings through local relationships in distances between pairs (or triplets) of samples [1–3, 11]. Deep metric learning has the ability to directly capture more discriminative power by utilizing certain metric losses. However, their performance highly depends on sampling and mining strategies [12].

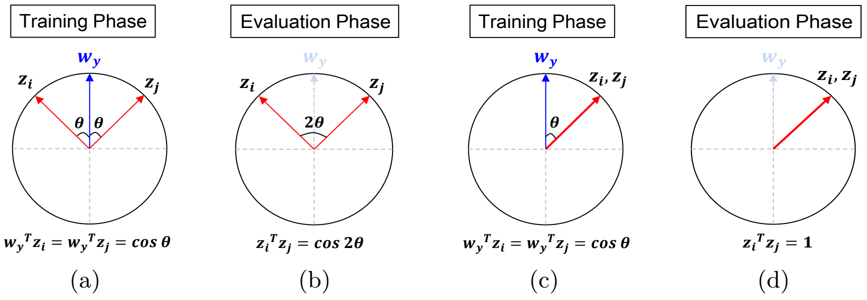


Fig. 1. Conceptual visualization of process discrepancy. Geometric view of a scenario having *process discrepancy* between (a) training phase and (b) evaluation phase. Geometric view of a scenario having *no process discrepancy* between (c) training phase and (d) evaluation phase. Here, z_i and z_j are the normalized intra-class feature vectors, and w_y denotes the corresponding class weight. (a) and (c) incur the same score ($cos\theta$) during the training, but (b) and (d) produce the different scores ($cos2\theta \neq 1$) at the evaluation phase.

Metric learning models typically require time-consuming back-and-forth procedures to train.

In recent years, softmax-based deep learning methods have been more widely used in face recognition tasks. They are easy to train and have achieved state-of-the-art performances in large-scale face recognition tasks. Softmax-based methods consider a classification loss in the training so that the learned features are separable. In contrast to the classification task, learning large-margin discriminative features is essential for face recognition tasks, particularly under the open-set protocol, which is a more realistic yet challenging face recognition protocol [6]. Many works have attempted to revise the softmax loss to obtain effective large-margin discriminative features [5, 6, 13, 7, 8]. Such variants are able to directly optimize the angles between features and the corresponding class weights in the hypersphere manifold.

Nevertheless, our observation is that their evaluation performance can suffer from discrepancy between training and evaluation processes under the open-set protocol: the matching scores between sample features and a softmax class weight are used during the training, while the matching scores are calculated between different sample features (without the class weight) at the evaluation phase. This difference leads to a directional discrepancy between sample features, as shown in Fig.1. We refer to this issue as "process discrepancy".

In this paper, we investigate the fundamental issue of process discrepancy in softmax-based learning methods for face recognition tasks. In particular, we first define displacement vectors which represent feature variations originated from the corresponding class weights in order to address the directional discrepancy between sample features. Then, we propose a new training scheme, called *minimum discrepancy learning* that encourages the directional discrepancy to be minimized by fitting all displacement vectors even with different classes to a single learnable representative vector as described in Fig.2. This single representative vector (e.g., basis) facilitates not only aligning directions of intra-class sample features toward an optimal direction but also disentangling class-invariant displacement vectors from their sample features, such that they are effective to train under the extreme class-imbalanced datasets.

In summary, the proposed scheme is specialized to mitigate process discrepancy between the training and evaluation phase for providing better performance at the evaluation phase. To the best of our knowledge, this is the first softmax-based learning method for face recognition tasks addressing process discrepancy issue, while previous methods focus on discriminative learning only. We demonstrate the superiority of our method under various benchmarks that include a large amount of hard positive examples, such as CPLFW [14], IJB-B [15], IJB-C [16] and QMUL-SurvFace dataset [17]: our regularization method consistently improves previous softmax-based training schemes such as Softmax, CosFace [7] and ArcFace [8].

2 Related Works

Metric-based Learning. Metric-based learning methods [1–3] directly learn discriminative features from the relationship between samples. The contrastive loss [1] uses positive and negative pairs of samples to learn the relationship between the two samples. The triplet loss [3] learns that the distance between an anchor and a positive sample is smaller than the distance between an anchor and a negative sample. Even though the metric-based learning is an intuitive way to solve the verification problem, the main drawback of metric-based learning resides on the difficulty of data sampling. It is hard to train all possible pairs or triplets, and the performance highly depends on the mining strategies.

Softmax-based Learning. Many approaches have been studied for softmax-based discriminative feature learning in various applications [18–25]. In face recognition task, several approaches have discussed to make more discriminative features based on softmax loss. Center loss [4] proposed a method to minimize intra-class variance. This method computes the centroid of samples for each class and minimizes the intra-class distances between feature vectors and their corresponding centroids. Crystal loss [26] introduced a constraint to enforce the norms of feature vectors to be a certain value. Ring loss [27] makes the norms to be a trainable parameter and encourages the norm of feature vectors to be optimally trained. NormFace [5] is a scheme to learn features on the hypersphere manifold such that the discrimination between classes can be done by angles. Sphereface [6] introduced a multiplicative angular margin loss to make features more discriminative. In a similar way, the effectiveness of the angular margin was demonstrated from CosFace [7] and ArcFace [8]. They used the angular margin in different ways.

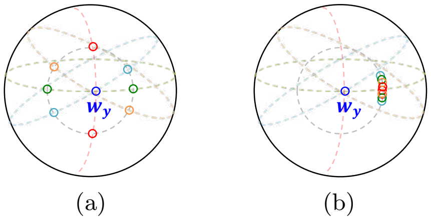


Fig. 2. A geometric view of process discrepancy. Note that all features around the class weight w_y belong to the same class. (a) All features produce the same cosine similarity with the corresponding class weight w_y, while they are separately placed. (b) All features are in similar positions with the same cosine similarity around w_y, which are forced by the proposed scheme.

3 Minimum Discrepancy Learning

We explain process discrepancy issue of softmax-based learning schemes for face recognition tasks in more detail in Section 3.1. In Section 3.2, we present the proposed training scheme designed to minimize process discrepancy.

3.1 Discrepancy in Face Recognition Schemes

Consider a dataset $\mathcal{D} = \{(x, y)\}$, which contains a sample (e.g., image) x and its corresponding label (or class) $y \in \mathcal{C} = \{1, 2, \ldots, C\}$. We are interested in finding a learnable model parameterized by $\{\theta, w\}$ that outputs a learned feature $\varphi_\theta(x)$ and a classification score $w_k^T \varphi_\theta(x)$ for each class k. This can be done by minimizing the following softmax loss with respect to $\{\theta, w\}$:

$$L_{\texttt{softmax}}(\theta, w; \mathcal{D}) = -\frac{1}{|\mathcal{D}|} \sum_{(x,y)\in\mathcal{D}} \log \frac{e^{w_y^T \varphi_\theta(x)}}{\sum_{k\in\mathcal{C}} e^{w_k^T \varphi_\theta(x)}}, \qquad (1)$$

where $|\mathcal{D}|$ is the number of samples. Note that the class weights not only contribute to inter-class variations by the denominator term $w_k^T \varphi_\theta(x)$ of (1), but intra-class variations by the numerator term $w_y^T \varphi_\theta(x)$ of (1). Once the model is trained, the evaluation phase in face recognition takes two input images x_i and x_j whose corresponding classes may not be in $\mathcal{C}$. Then, the cosine similarity between normalized feature vectors, i.e., $z_i = \frac{\varphi_\theta(x_i)}{\|\varphi_\theta(x_i)\|_2}$, $z_j = \frac{\varphi_\theta(x_j)}{\|\varphi_\theta(x_j)\|_2}$, is measured to identify whether they are in the same class or not.

As described in Fig.2 (a), the features around the corresponding class weight w_y lie on a hypersphere manifold. In the training phase, these features are able to produce the same scores by $w_y^T \varphi_\theta(x)$ of the softmax function (1) even though their directions are different from each other. On the other hand, this directional discrepancy is attributed to different displacement vectors $(\varphi_\theta(x) - w_y)$ and leads to an undesirable effect at the evaluation stage. Namely, the directional relationship between features are important in the evaluation phase, but it is not directly considered during the training. We call this issue, *process discrepancy* that might not appear in deep metric-based learning because the underlying

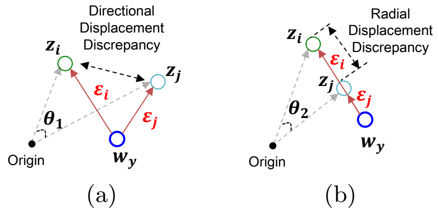


Fig. 3. Intra-class variations when considering two types of discrepancies: (a) Directional displacement discrepancy and (b) Radial displacement discrepancy. Note that the directional and radial displacement discrepancies cause the angle difference between features.

functions in both the training and evaluation process are identical. However, as aforementioned in Section 1, deep metric-based learning is not widely used due to the performance sensitivity to sampling and mining methods. In fact, process discrepancy comes from the different underlying functions in both the training and evaluation process. Furthermore, process discrepancy directly influences the evaluation performance, as described in Fig.1. This means that even a well-trained model under the softmax loss could not ensure the high performance in the evaluation phase (see Section 4.3 for more details).

3.2 Learning Discrepancy-Free Representations

Discrepancy loss. The main idea for minimizing process discrepancy is to enforce the directions of intra-class features to be aligned in a single direction from the perspective of their class weights as illustrated in Fig.2 (b). In essence, process discrepancy occurs because $w_y^T z_i \approx w_y^T z_j$ for the training phase does not guarantee $z_i^T z_j \approx 1$ for the evaluation phase. The proposed idea of directional alignment can minimize the angle between z_i and z_j. To handle the variation of each feature $\varphi_\theta(x)$, we first define the displacement vector ε as follows:

$$\varepsilon(x, y) = \frac{\varphi_\theta(x)}{\|\varphi_\theta(x)\|_2} - \frac{w_y}{\|w_y\|_2}, \tag{2}$$

i.e., it is the difference vector between a feature $\varphi_\theta(x)$ and its class weight w_y as shown in Fig.3. Note that the feature vector $\varphi_\theta(x)$ is normalized in (2) because the evaluation phase calculates the angle difference between features by using normalized features. As mentioned in Section 3.1, the directional discrepancy between features (i.e., process discrepancy) is due to different intra-class displacement vectors of their features, as shown in Fig.2 (a). Inspired by this observation, we introduce an additional learnable representative vector ξ that fits all displacement vectors, which is named here, *deep displacement basis*, in order to minimize the discrepancy between displacement vectors of their features. The discrepancy-free features and model parameters θ, w, ξ are jointly learned by minimizing the following loss with the softmax loss (1):

$$L_{\texttt{discrepancy}}(\theta, w, \xi; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(x,y)\in\mathcal{D}} \|\varepsilon(x, y) - \xi\|_2. \tag{3}$$

The above loss minimizes the directional intra-class discrepancy, such that the training/evaluation process discrepancy can be mitigated. An additional advantage of optimizing (3) is that the class information in the displacement vectors is implicitly forced to be eliminated by fitting all of them to a single displacement basis (regardless of their classes). In other words, it learns to disentangle class-free variations (e.g., displacement vectors) from their features. Hence, manipulating displacement vectors has a negligible effect on inter-class separability, while it helps to minimize process discrepancy.

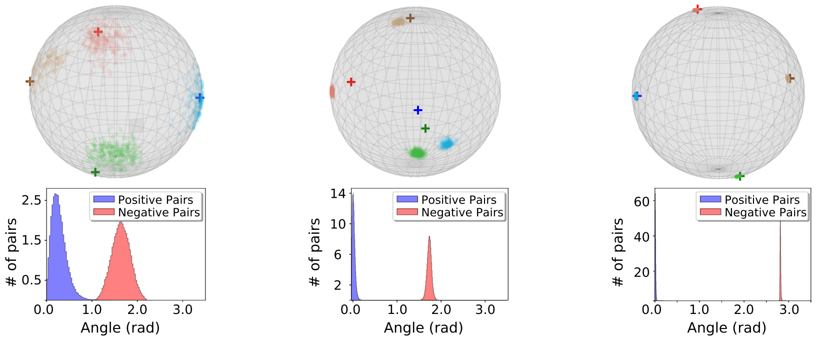
In summary, we suggest to minimize the following loss:

$$L_{\texttt{total}}(\theta, w, \xi; \mathcal{D}) = L_{\texttt{softmax}}(\theta, w; \mathcal{D}) + \lambda L_{\texttt{discrepancy}}(\theta, w, \xi; \mathcal{D}), \tag{4}$$

where $\lambda > 0$ is a hyper-parameter to balance both loss terms. In the above, one can use $L_{\texttt{softmax}}$ as any softmax-based loss for face recognition tasks, e.g., angular-margin losses [8, 7, 6, 13] for improving the performance further.

Directional vs. radial displacement discrepancy. We further remark that process discrepancy can be decomposed into two types as described in Fig.3: *directional displacement discrepancy* and *radial displacement discrepancy.* The directional displacement discrepancy is the angle difference between displacement vectors and is defined by $\frac{\varepsilon(x_i,y_i)}{\|\varepsilon(x_i,y_i)\|_2}^T \frac{\varepsilon(x_j,y_j)}{\|\varepsilon(x_j,y_j)\|_2}$. Also, the radial displacement discrepancy is the norm difference between displacement vectors and is defined by $|\|\varepsilon(x_i, y_i)\|_2 - \|\varepsilon(x_j, y_j)\|_2|$. As shown in Fig.3 (a) and (b), the directional displacement discrepancy causes the angle difference (θ_1) between features, while the radial displacement discrepancy leads to another angle difference (θ_2) between features as well. Namely, both discrepancies in the displacement domain result in directional discrepancies (θ_1 and θ_2) between features, and should simultaneously be suppressed. Since the discrepancy loss (3) is defined by using a non-normalized version of displacement vector $\varepsilon(x, y)$ and basis ξ, it can penalize both directional and radial displacement discrepancies.

Comparison to other methods. As mentioned in Section 3.1, most softmax-based approaches may generate process discrepancy due to their limited ability in softmax-based loss. Although softmax loss is able to maximize $w_y^T \varphi_\theta(x)$, in practice, some hard features are placed around their class weights as in Fig.2 (a) and it causes process discrepancy. On the other hand, our method at least helps place these features directionally close to each other as in Fig.2 (b), which is not explicitly done by previous works. For example, center loss [4] attempts to minimize intra-class variations by introducing centroids, but process discrepancy is not considered. This means that $w_y^T z_i \approx w_y^T z_j$ for the training phase does not guarantee $z_i^T z_j \approx 1$ for the evaluation phase. Since intra-class features are concentrated only on their centroids (not their class weights), this effect may lead to diminishing inter-class separability as in Fig.4 (b). In contrast, the proposed method introduces the class-free concept of the displacement vector, which enables to minimize directional intra-class variations without hurting inter-class separability as in Fig.4 (c). Alternatively, adding metric losses such as triplet loss [3] and contrastive loss [1] is one of ways to mitigate process discrep-



(a) (b) (c)

Fig. 4. Toy examples on VGGFace2 dataset [28] for 4 classes: (a) CosFace [8], (b) Center Loss [4], (c) Cos-DiscFace (ours). We use ResNet-18 [29] as a baseline network architecture to learn 3 dimensional features. We replace the global average pooling of ResNet-18 by a fully-connected layer. The colored point is a feature vector and the color of points indicates the corresponding class. In figures, "+" indicates a class weight. The first row visualizes the hypersphere manifold with 4 classes. The second row shows the angle distribution of positive and negative pairs. We select 2 classes to generate their angle distributions.

ancy. However, the performance may depend on sampling strategy. On the other hand, our method is the one that minimizes process discrepancy without any sampling method. As a result, our method overcomes the fundamental limitation of softmax-based methods by suppressing process discrepancy and provides performance improvement at the evaluation phase.

4 Analytical Study

4.1 Feature Visualization

We demonstrate a toy example to visualize the feature distributions on methods. To observe intra-class compactness effectively, we choose VGGFace2 dataset [28] which contains an average of 362.6 samples per class. The baselines such as CosFace and Center Loss are performed and their results are illustrated in Fig.4 (a) and (b), respectively. As shown in Fig.4 (c) and (d), our method is conducted to visualize the feature distribution when it uses a single basis and two bases, respectively. The center loss minimizes intra-class variations as depicted in Fig.4 (b), which needs to be more compact. More importantly, Fig.4 (a) shows that ArcFace causes many directions of variations. On the other hand, we observe that the directions of variations are aligned in a single direction by our method. As shown in Fig.4 (c), the position of the feature distribution per class is located to the left side of the corresponding class weight. Moreover, we indeed observe that the proposed method minimizes intra-class variations under preserving inter-

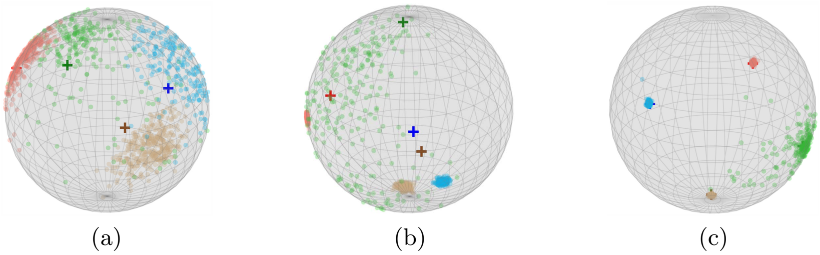


Fig. 5. Visualization results under the extreme class-imbalanced dataset. (a) CosFace [8] (b) Center Loss [4] (c) Cos-DiscFace (ours). Samples from three classes are fed to every mini-batch while only a single sample for the remaining class is fed to a mini-batch every two epochs. In figures, "+" indicates a class weight and the green points are of the class with insufficient samples during the training.

class separability, as the angle distributions are shown in the second row of Fig.4.

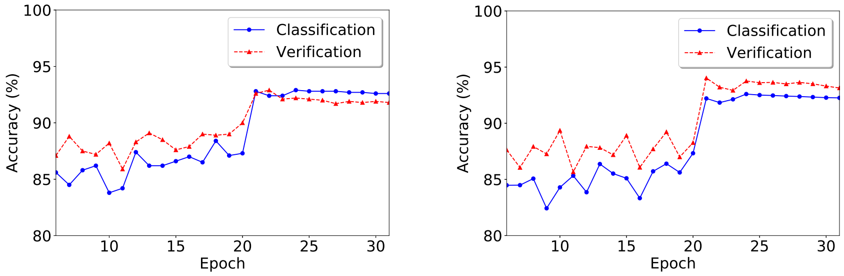
4.2 A Recognition Task under Extreme Class-Imbalanced Dataset

The enormous number of classes, and extreme scarcity of samples per class, i.e., extreme class-imbalanced dataset, makes it difficult to learn features robust to unconstrained variations. Namely, samples for a certain class may appear quite rarely in a mini-batch during the training phase. The proposed method enables us to use all mini-batch samples (even the ones that belong to different classes) as if they were intra-class samples due to class-invariant attributes of displacement vectors. To prove this, we use the following trick to visualize intra-class compactness under the extreme class-imbalanced dataset: a certain class appears in a mini-batch only one (i.e., a single sample) every two epochs. By doing so, the class samples become insufficient to produce intra-class compactness. The remaining experimental setup follows that of the toy example described in Fig.4. In this configuration, we run three methods: CosFace [8], Center Loss [4] and Cos-DiscFace (ours).

The results are presented in Fig.5. Center loss produces intra-class compactness only in three classes, while our method results in intra-class compactness even in the class that appears rarely during the training. Further experiments are conducted to demonstrate the effectiveness of our method under the extreme class-imbalanced datasets as reported in Table 4.

4.3 Effects on Process Discrepancy

We also analyze process discrepancy via investigating the relationship between classification and verification performance. Once process discrepancy is minimized, the performance improvement in both classification and verification tasks should be consistent. To show this, we use 90% of the CASIA-WebFace [30] as



(a)

(b)

Fig. 6. Accuracy (%) on both classification and verification tasks. (a) CosFace [7], and (b) Cos-DiscFace (Ours). This experiment aims to show the effect of process discrepancy, by demonstrating the inconsistency of performance between classification and verification tasks.

training samples, while 10% of the CASIA-WebFace are used for classification evaluation. For verification evaluation, CFP-FP [31] is chosen. This experiment follows the setup of CASIA-WebFace as described in Section 5.1. CosFace [7] is chosen as a baseline in this analysis.

In particular, we investigate the trends of the relationship between classification and verification performance along training goes. When the classification performance is increased, our method consistently produces the improved verification performance as shown in Fig.6 (b). In the case of the previous method, relatively smaller improvement for verification performance is revealed, compared with classification performance, as shown in Fig.6 (a). As aforementioned in Section 3.1, higher classification performance does not guarantee higher verification performance when process discrepancy occurs, as in Fig.6 (a). The experimental results confirm that the proposed method (i.e., the discrepancy loss) serves as making verification performance consistent with classification performance.

5 Experiments on Various Benchmark Datasets

5.1 Experimental Setup

Training datasets. To verify the robustness of the proposed method, training the models is performed using both small-scale and large-scale datasets. As a small-scale training dataset, CASIA-WebFace dataset [30] is selected. The CASIA-WebFace contains 0.49M face images collected from 10K subjects. For a fair comparison with the state-of-the-art methods on a large-scale dataset, MS1MV2 dataset is utilized. The MS1MV2 is a refined version of MS-Celeb-1M dataset [8], and it comprises over 5.8M face images from 85K different identities. For low-resolution surveillance face recognition task, we use QMUL-SurvFace dataset [17]. The face images for QMUL-SurvFace are re-scaled to 112x112 by using bicubic interpolation while the other training datasets are cropped to 112x112 by MTCNN [32].

Algorithm 1 The Pseudo-code of DiscFace on Pytorch

```
Require: Feature $x$, Class weight $w$, Displacement basis $b$, Label $t$
  $x = \text{F.normalize}(x)$
  $w = \text{F.normalize}(w)$
  $w_{\texttt{batch}} = w[\text{t,:}]$
  $d = x - w_{\texttt{batch}}$
  $b_{\texttt{norm}} = b.\text{norm(p=2,dim=1)}$
  $b_{\texttt{norm}} = \text{torch.clamp}(b_{\texttt{norm}}, \text{min=0, max=0.05})$
  $y = \text{F.linear}(x,w)$
  $L_{\texttt{softmax}} = \text{F.cross\_entropy}(y,t)$
  $L_{\texttt{discrepancy}} = (d - \text{F.normalize}(b)*b_{\texttt{norm}}).\text{norm(p=2,dim=1).mean()}$
  $L_{\texttt{total}} = L_{\texttt{softmax}} + \lambda L_{\texttt{discrepancy}}$
```

Implementation details. As our network architectures, we use ResNet-34 [29] for QMUL-SurvFace, LResNet50E-IR [8] for CASIA-WebFace and LResNet100E-IR [8] for MS1MV2. The learning rate is initially set to 0.1 for all experiments. It is then divided by 10 at the 100K, 160K and 220K iterations for the MS1MV2, the 20K and 28K iterations for the CASIA-WebFace, and the 10K and 15K iterations for the QMUL-SurvFace. The training is complete at the 360K-th iteration for MS1MV2, the 32K-th iteration for the CASIA-WebFace, and 26K-th iteration for QMUL-SurvFace. The batch size is set to 512 for the CASIA-WebFace and MS1MV2, and 128 for the QMUL-SurvFace. The training is performed on 2 GPUs and the network architectures are optimized by using stochastic gradient decent (SGD) algorithm.

Evaluation datasets and protocols. Our method is evaluated on several benchmarks. As small-scale datasets, LFW [33], CPLFW [14], CALFW [34], CFP-FP [31], and AgeDB-30 [35] are employed. CPLFW and CFP-FP contain different pose variations. CALFW and AgeDB-30 include different age variations. To evaluate the proposed method on large-scale test datasets, IJB-B [15], IJB-C [16] and MegaFace dataset [36] with FaceScrub dataset [37] are utilized for open-set face verification. To further verify that the proposed method is robust to hard positive examples, we employ QMUL-SurvFace dataset [17]. All face images are cropped or bicubic-interpolated to 112x112 to make them consistent with the images from the training datasets. Cosine similarity between probe and gallery features is used to measure whether the given features belong to the same identity or not. Performance is assessed by calculating *Accuracy* and *True Acceptance Rate* (TAR) with a fixed *False Acceptance Rate* (FAR).

Methods. As mentioned in Section 3.2, the proposed method could be combined with any softmax-based method. Our method is applied to the standard softmax loss (1), which is referred to as Soft-DiscFace. We also apply our method to the angular-margin softmax schemes, such as ArcFace [8] and CosFace [7]. We refer to these models as Arc-DiscFace and Cos-DiscFace, respectively. Similarly, the other losses such as center loss [4], contrastive loss [1] and triplet loss [3] are combined with ArcFace, which are referred to as

Methods	TAR(%)@FAR			Accuracy (%)
	10%	1%	0.1%	
DeepID2 [38]	60.0	28.2	13.4	76.1
FaceNet [3]	79.9	40.3	12.7	85.3
SphereFace [6]	63.6	34.1	15.6	77.5
Center Loss [4]	86.0	53.3	26.8	88.0
CosFace*	72.0	44.0	14.7	81.3
Cos-DiscFace (Ours)*	74.4	44.7	23.0	82.3
Softmax*	83.3	52.4	17.8	86.5
Soft-DiscFace (Ours)*	**86.8**	**62.9**	**35.9**	**88.6**

Table 1. Face Verification (%) on QMUL-SurvFace dataset. The hyperparameter ($\lambda = 1$) of our method are used in the experiments. Note that "*" indicates our implementations and the best results are indicated in bold.

Arc-CenterLoss, Arc-ContrastiveLoss and Arc-TripletLoss. The hyperparameter of CosFace ($m = 30, s = 0.25$), ArcFace ($m = 64, s = 0.5$), Center Loss ($\lambda = 0.003, \alpha = 0.5$), Contrastive Loss, Triplet Loss ($\lambda = 0.1, m = 0.3$) and ours ($\lambda = 0.2$) are used unless specified. To avoid the dependency on the initial setting of the displacement basis, its norm value is constrained (or clipped) by a certain upper bound (set by 0.05). To implement our method, we provide the pytorch pseudo-code as described in Algorithm 1.

5.2 QMUL-SurvFace Dataset

QMUL-SurvFace dataset [17] has been recently released to verify the robustness of low-resolution surveillance facial images. This dataset is drawn from real surveillance videos, not synthesized by artificial down-sampling of high-resolution images. The dataset is suitable to evaluate realistic performance since it contains the wild environment characteristics such as low-resolution, motion blur, unconstrained poses, poor illumination and background clutters. The QMUL-SurvFace consists of 463,507 facial images with 15,573 unique identities. The positive and negative pairs in the evaluation set are 5,319 pairs, respectively.

According to [17], it is reported that the center loss produces the best performance for the QMUL-SurvFace. Since the standard softmax loss is adopted to the center loss, the standard softmax and CosFace [7] are chosen as baselines. The experimental results are summarized in Table 1. The proposed method achieves the best performance among all tested methods. Since the proposed method is designed to effectively reduce process discrepancy (i.e., directional intra-class variations), its effect can be significant on datasets mainly consisting of hard positive pairs. In the sense, the results from the QMUL-SurvFace are the best demonstration of the effectiveness of the proposed method, where Soft-DiscFace improves TAR for FAR 10^{-3}, from 17.8% to 35.9%, compared with softmax loss only. Interestingly, we found that the standard softmax loss outperforms the angular-margin based softmax loss in the experiments. We conjecture that the margin penalty during the training show an undesirable effect on hard pos-

Methods	Dataset	LFW	CFP-FP	AgeDB-30	CALFW	CPLFW
FaceNet [3]	200M	99.65	-	-	-	-
OE-CNNs [39]	1.7M	99.47	-	-	-	-
Center Loss [4]	0.7M	99.28	-	-	-	-
NormFace [5]	CASIA	99.19	-	-	-	-
SphereFace [6]		99.42	-	-	-	-
RegularFace [9]		99.33	-	-	-	-
AMSoftmax [13]		99.28	94.77	-	-	-
PFE-AMSoftmax [40]		99.55	95.92	-	-	-
CosFace*		99.42	96.29	93.33	92.62	89.28
Cos-DiscFace (Ours)*		**99.62**	**96.54**	**93.63**	**93.30**	**89.73**
Ring Loss [27]	MS1M	99.52	-	-	-	-
AdaptiveFace [41]		99.62	-	-	-	-
CosFace [7]		99.73	-	-	-	-
ArcFace [8]		99.82	98.37	98.15	95.45	92.08
Arc-DiscFace (Ours)*		**99.83**	**98.54**	**98.35**	**96.15**	**93.37**

Table 2. Face verification (%) on the LFW, CFP-FP, Age30-DB, CALFW, and CPLFW. Note that "*" indicates our implementations and the best results are indicated in bold. CosFace and ArcFace are selected as baselines in these implementations since they show the best performance on CASIA-WebFace and MS1M training datasets, respectively.

itive examples such as low-resolution, motion blur and so on. The intra-class compactness methods such as center loss and our method show a positive effect on performance, compared with the result of the softmax loss. Moreover, our method shows better performance, compared to center loss, since it exclusively considers process discrepancy.

5.3 Comparison Results

Results on LFW, CFP-FP, Age30-DB, CALFW and CPLFW benchmarks. In this experiment, we compare our method with state-of-the art methods. LFW [33] contains 13,233 face images collected from 5,749 different identities, forming 6,000 pairs of face images. Other face benchmarks such as AgeDB-30 [35], CPLFW [14], CFP-FP [31] and CALFW [34] are also chosen to compare with state-of-the-art methods. CPLFW contains 6,000 pairs in the profile-profile configuration. CFP-FP contains 7,000 pairs in the frontal-profile configuration. Note that CALFW and AgeDB-30 include age variations and generate 6,000 positive and negative pairs of face images. The experimental results trained on MS1MV2 [42] are reported in Table 2. One can observe that the proposed methods (Cos-DiscFace and Arc-DiscFace) provide improved performance over the previous methods. Specifically, Arc-DiscFace improves the accuracy from 92.08% to 93.37% on CPLFW and 95.45% to 96.15% on CALFW (trained on MS1M). Meanwhile, Cos-DiscFace improves the accuracy from 92.62% to 93.30% on CALFW (trained on CASIA). These results indicate that further improvement is achieved when the proposed method is combined with any softmax-based

Methods	Dataset	CPLFW	IJB-B	IJB-C	MF Id.	MF Ver.
ArcFace*	MS1M	92.70	86.05	92.46	80.77	96.88
Arc-Center Loss*		92.77	86.36	92.34	80.38	96.98
Arc-Contrastive Loss*		92.92	88.16	93.52	80.44	96.98
Arc-Triplet Loss*		92.95	87.25	93.03	**81.25**	97.05
Arc-DiscFace (Ours)*		**93.37**	**88.83**	**93.71**	81.23	**97.44**

Table 3. Comparison results (%) on CPLFW, IJB-B, IJB-C and MegaFace. IJB-B and IJB-C results are based on 1:1 verification for TAR at FAR 10^{-5}. MF Id. and MF Ver. indicate results on MegaFace Challenge 1 using FaceScrub as the probe set. MF Id. refers to the rank-1 face identification accuracy with 1M distractors, and MF Ver. refers to the face verification for TAR at FAR 10^{-6}. Note that "*" indicates our implementations and the best results are indicated in bold.

Method	Dataset	TAR (%) @ FAR on IJB-C				
		10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
ArcFace*	MS1M-LT	98.81	97.59	95.84	93.78	90.48
Arc-Center Loss*		98.87	97.62	96.07	93.79	90.68
Arc-DiscFace (Ours)*		**98.89**	**97.87**	**96.57**	**94.82**	**92.42**

Table 4. Comparison results (%) on IJB-C under the class-imbalanced dataset. IJB-C results are based on 1:1 verification. We train with long-tailed MS1M (MS1M-LT [43]) that has 11.9 mean images per ID. Note that "*" indicates our implementations and the best results are indicated in bold.

losses such as CosFace and ArcFace. Moreover, our method provides the consistency of performance improvement under various training datasets.

Results on challenging benchmarks. The goal of this experiment is to verify the robustness on more challenging benchmarks. MegaFace dataset [36] contains 1M images collected from 690K identities, which is mainly used as a gallery set. The FaceScrub dataset [37] consists of 106,863 images collected from 530 individuals, as a probe set. The IJB-B dataset [15] contains 1,845 subjects with 21.8K still images and 55K frames from 7,011 videos. The IJB-C dataset [16] is an extension of IJB-B, which contains 3,531 subjects with 31.3K still images and 117.5K frames from 11,779 videos. We choose CPLFW as a hard small-scale benchmark. We train LResNet100E-IR [8] on a large-scale training dataset (MS1MV2). For contrastive and triplet loss, anchor, positive and negative images are randomly sampled. The performance result is reported in Table 3. Our method consistently improves the verification performance from 86.05% to 88.83% on IJB-B for TAR at FAR 10^{-5}, 92.46% to 93.71% on IJB-C for TAR at FAR 10^{-5}, and 96.98% to 97.44% on MegaFace for TAR at FAR 10^{-6}. We remark that discrepancy-free methods such as Arc-ContrastiveLoss, Arc-TripletLoss produce performance improvement compared with their counterparts (ArcFace) whereas Arc-CenterLoss seems not apparently to improve performance probably due to process discrepancy. Although the discrepancy-free methods improve performance on some benchmarks, there is still a performance

Benchmarks	$\lambda = 0.0$	$\lambda = 0.2$	$\lambda = 0.4$	$\lambda = 0.6$	$\lambda = 0.8$	$\lambda = 1.0$
LFW	99.23	**99.62**	99.30	99.33	99.38	99.35
AGEDB-30	93.15	93.63	93.65	**93.92**	93.58	93.65
CFP-FP	96.44	96.54	**96.71**	**96.71**	96.59	96.50
CPLFW	89.10	89.73	**90.32**	89.82	89.78	89.23
CALFW	92.63	93.30	92.90	**93.47**	93.02	92.97

Table 5. Ablation study (%) on λ for Cos-DiscFace trained with CASIA-WebFace dataset. Note that Cos-DiscFace with $\lambda = 0.0$ indicates CosFace. Our method is less sensitive to the performance with respect to λ. The best results are indicated in bold.

gap between our method and the discrepancy-free methods. To further improve performance for them, the elaborated sampling strategy should be investigated. On the other hand, our method does not require such a back-and-forth procedure while it provides better performance.

Results on class-imbalanced dataset. As discussed in Section 3.2, all displacement vectors are imposed to be the displacement basis. This means that the feature variation of minor class ideally becomes identical to that of major class. Namely, the relative lack of variations for minor classes is alleviated by forcing all variations to a single variation. To confirm the efficacy, we train on MS1M long-tailed version (i.e., MS1M-LT [43]) and evaluate the performance on IJB-C. The results are shown in Table 4. The results show the effectiveness of our method on the class-imbalanced dataset, resulting in 90.48%@ArcFace, 90.68%@Arc-CenterLoss and 92.42%@Arc-DiscFace for TAR at FAR 10^{-5}. This explains the superiority of the proposed method under the extreme class-imbalanced datasets.

5.4 Ablation Study on λ

We explore the hyperparameter λ in order to investigate performance sensitivity across λ. We train Cos-DiscFace on CASIA-WebFace datasets with λ varying from 0.0 to 1.0 with step 0.2. The results are presented in Table 5. Note that Cos-DiscFace with $\lambda = 0.0$ indicates CosFace. One can observe that our method is less sensitive across the range of the hyperparameter λ. In particular, the higher performance improvement is achieved at $\lambda = 0.4$ and 0.6. In contrast, the greater impact ($\lambda = 1.0$) of discrepancy loss may hinder softmax loss minimization, such that the performance degradation is observed as shown in Table 5.

6 Conclusion

In this paper, we propose a new training loss to address the fundamental issue of process discrepancy in softmax-based learning methods for face recognition tasks. The proposed method is particularly effective for minimizing the intra-class variations under the extreme class-imbalanced dataset. We demonstrate its superiority on various benchmarks when it combines with existing softmax-based losses. We think it would be interesting to apply our idea to other related tasks such as speaker verification [44] and imbalanced classification [45].

References

1. Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q.: Deep learning face representation by joint identification-verification. Advances in Neural Information Processing Systems (NIPS) (2014) 1988–1996
2. Hu, J., Lu, J., Tan, Y.P.: Discriminative deep metric learning for face verification in the wild. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014) 1875–1882
3. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
4. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. The European Conference on Computer Vision (ECCV) (2016)
5. Wang, F., Xiang, X., Cheng, J., Yuille, A.L.: Normface: L2 hypersphere embedding for face verification. Proceedings of the 25th ACM international conference on Multimedia (2017) 1041–1049
6. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
7. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
8. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
9. Zhao, K., Xu, J., Cheng, M.M.: Regularface: Deep face recognition via exclusive regularization. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
10. Yin, X., Yu, X., Sohn, K., Liu, X., Chandraker, M.: Feature transfer learning for face recognition with under-represented data. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
11. Han, C., Shan, S., Kan, M., Wu, S., Chen, X.: Face recognition with contrastive convolution. The European Conference on Computer Vision (ECCV) (2018)
12. Wang, M., Deng, W.: Deep face recognition: A survey. arXiv preprint arXiv:1804.06655 (2018)
13. Wang, F., Cheng, J., Liu, W., Liu, H.: Additive margin softmax for face verification. IEEE Signal Processing Letters **25** (2018) 926–930
14. Zheng, T., Deng, W.: Cross-pose lfw: A database for studying crosspose face recognition in unconstrained environments. In technical Report (2018)
15. Whitelam, C., Taborsky, E., Blanton, A., Maze, B., Adams, J., Miller, T., Kalka, N., Jain, A.K., Duncan, J.A., Allen, K., Cheney, J., Grother, P.: Iarpa janus benchmark-b face dataset. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2017) 592–600
16. Maze, B., Adams, J., Duncan, J.A., Kalka, N., Miller, T., Otto, C., Jain, A.K., Niggel, W.T., Anderson, J., Cheney, J., et al.: Iarpa janus benchmark-c: Face dataset and protocol. International Conference on Biometrics (ICB) (2018) 158–165
17. Cheng, Z., Zhu, X., Gong, S.: Surveillance face recognition challenge. arXiv preprint arXiv:1804.09691 (2018)

18. Chen, B., Deng, W., Shen, H.: Virtual class enhanced discriminative embedding learning. Advances in Neural Information Processing Systems (NIPS) (2018) 1942–1952
19. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. International Conference on Learning Representations (ICLR) (2018) 6438–6447
20. Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., Bengio, Y.: Manifold mixup: Better representations by interpolating hidden states. The International Conference on Machine Learning (ICML) (2019) 6438–6447
21. Kim, I., Kim, K., Kim, J., Choi, C.: Deep speaker representation using orthogonal decomposition and recombination for speaker verification. The IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2019)
22. Jang, E., Gu, S., Poole, B.: Categorical reparameterization with gumbel-softmax. International Conference on Learning Representations (ICLR) (2017)
23. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. The International Conference on Machine Learning (ICML) (2017)
24. Maddison, C.J., Mnih, A., Teh, Y.W.: The concrete distribution: A continuous relaxation of discrete random variables. International Conference on Learning Representations (ICLR) (2017)
25. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
26. Ranjan, R., Bansal, A., Xu, H., Sankaranarayanan, S., Chen, J.C., Castillo, C.D., Chellappa, R.: Crystal loss and quality pooling for unconstrained face verification and recognition. arXiv preprint arXiv:1804.01159 (2018)
27. Zheng, Y., Pal, D.K., Savvides, M.: Ring loss: Convex feature normalization for face recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
28. Cao, Q., Shen, L., Xie, W., Parkhi, O.M., Zisserman, A.: Vggface2: A dataset for recognising faces across pose and age. The IEEE Conference on Automatic Face & Gesture Recognition (FG 2018) (2018) 67–74
29. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
30. Dong Yi, Zhen Lei, S.L., Li, S.Z.: Learning face representation from scratch. arXiv preprint arXiv:1411.7923 (2014)
31. Sengupta, S., Chen, J.C., Castillo, C., Patel, V.M., Chellappa, R., Jacobs, D.W.: Frontal to profile face verification in the wild. The IEEE Winter Conference on Applications of Computer Vision (WACV) (2016)
32. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters (2016) 1499–1503
33. Huang, G.B., Mattar, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In technical Report 07-49, University of Massachusetts, Amhertst (2007)
34. Zheng, T., Deng, W., Hu, J.: Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. arXiv:1708.08197 (2017)
35. Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., Zafeiriou, S.: Agedb: the first manually collected, in-the-wild age database. The IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2017) 51–59

36. Kemelmacher-Shlizerman, I., Seitz, S.M., Miller, D., Brossard, E.: The megaface benchmark: 1 million faces for recognition at scale. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 4873–4882
37. Ng, H.W., Winkler, S.: A data-driven approach to cleaning large face datasets. IEEE International Conference on Image Processing (ICIP) (2014) 343–347
38. Sun, Y., Wang, X., Tang, X.: Deeply learned face representations are sparse, selective, and robust. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015) 2892–2900
39. Wang, Y., Gong, D., Zhou, Z., Ji, X., Wang, H., Li, Z., Liu, W., Zhang, T.: Orthogonal deep features decomposition for age-invariant face recognition. The European Conference on Computer Vision (ECCV) (2018)
40. Shi, Y., Jain, A.K.: Probabilistic face embeddings. The IEEE International Conference on Computer Vision (ICCV) (2019)
41. Liu, H., Zhu, X., Lei, Z., Li, S.Z.: Adaptiveface: Adaptive margin and sampling for face recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
42. Guo, Y., Zhang, L., Hu, Y., He, X., Gao, J.: Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. The European Conference on Computer Vision (ECCV) (2016) 87–102
43. Liu, Z., Miao, Z., Zhan, X., Wang, J., Gong, B., Yu, S.X.: Large-scale long-tailed recognition in an open world. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
44. Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker verification using adapted gaussian mixture models. Digital signal processing **10** (2000) 19–41
45. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research **16** (2002) 321–357