NPT-Loss: Demystifying face recognition losses with Nearest Proxies Triplet

Anonymous ICCV submission

Paper ID 10441

Abstract

Face recognition (FR) using deep convolutional neural networks (DCNNs) has seen remarkable success in recent years. One key ingredient of DCNN-based FR is the appropriate design of a loss function that ensures discrimination between various identities. The state-of-the-art (SOTA) solutions utilise normalised Softmax loss with additive and/or multiplicative margins. Despite being popular and effective, these losses are justified only intuitively with little theoretical explanations. In this work, we utilise an alternative framework of proxy-based triplet losses that offer a more direct mechanism of achieving discrimination among the features of various identities. We show that under the LogSumExp (LSE) approximation, the SOTA Softmax losses become equivalent to a proxy-triplet loss that focuses on nearest-neighbour negative proxies only. This motivates us to propose a variant of the proxy-triplet loss entitled Nearest Proxies Triplet (NPT) loss. Unlike SOTA solutions, the proposed loss converges for a wider range of hyper-parameters and offers flexibility in proxy selection, which allows us to outperform SOTA techniques. We generalise many SOTA losses into a single framework and give theoretical justification that minimising the proposed loss ensures a minimum separability between all identities. We also show that the proposed loss has an implicit mechanism of hard-sample mining. We conduct extensive experiments using various DCNN architectures on a number of FR benchmarks including an independent evaluation on the IFRT challenge. The proposed loss consistently achieves SOTA performance in all experiments [1].

1. Introduction

Automated face recognition (FR) has a wide variety of applications including surveillance, access-control, healthcare, advertisement etc. Owing to its significance, it is a widely studied topic in computer vision literature. Recently, deep convolution neural network (DCNN) based solutions [30, 7, 15, 27, 28, 25, 22] have seen remarkable success in FR applications and these methods have replaced the classical FR techniques altogether. Generally, all state-of-art CNN based systems rely on the following procedure: in the **training** phase, a deep CNN is trained using a large scale datasets such as CasiaWeb [34] and/or MS-Celeb1M [10]. Some preprocessing such as face detection and alignment is carried out before training, and a suitable loss function, such as triplet loss [25], normalised Softmax [27], ArcFace [7] etc., is used to train the network. Once the training is complete, the loss layer is discarded and the output of the CNN (usually a 512- or a 2048-dimensional vector) is treated as the feature vector corresponding to a given input face image. In the **testing** phase, a pair of inputs is fed to the trained network and the cosine similarity of the resulting feature vectors is evaluated. If the score is greater than a given threshold than the image pair is recognised as belonging to the same identity.

From the above procedure, we note that an accurate CNN based FR system should satisfy the following conditions: **(C1)** all feature vectors belonging to the same identity should have a large cosine similarity, i.e., all features belonging to the same person must be clustered close together, in terms of angular distance[2], in the n-dimensional feature space. **(C2)** Feature vectors that belong to different identities should have a sufficient amount of angular separation in the feature space, to ensure discrimination between the various identities.

We can identify two different approaches in the FR literature that try to satisfy the above mentioned conditions: the first approach is to tackle the problem directly using metric losses, such as contrastive [3] and triplet loss [25]. For instance, in the case of contrastive loss, a pair of images is fed to the network and the loss function minimises the distance between the feature vectors if the pair belongs to the same identity, and maximises the distance, if the images in the pair belong to different persons. In the case of triplet loss, instead of a pair of images, a triplet of images, consisting of an anchor, a positive and a negative sample is

[1] All codes available at: https://github.com/cerebrai/npt-loss

[2] The ordering of nearest neighbours remains unaffected if we sort them using cosine similarity or with angular distance

fed to the network. The loss is designed to minimise the distance between the anchor-positive pair and maximise it for the anchor-negative pair. While these methods are direct and straightforward, they suffer from sampling issues. For instance, if a dataset has k classes with each class containing n samples, then we would have triplets in the order of $\mathcal{O}(n^3)$. Hence, these direct metric losses have to rely on sampling/mining strategies that try to extract the most informative pairs/triplets out of all possibilities. Firstly, it is difficult to make efficient mining strategies; secondly, even after sampling/mining the convergence is slow for these methods.

The second approach relies on proxies rather than the actual data pairs. One example of this approach is the CenterLoss [30], which evaluates the centres of features corresponding to each identity. The distance between the features and the centres is then minimized. Note that if the maximum distance between a set of features and its centres is ϵ, then by triangle inequality, the maximum distance between any two features, of the same set, would be less than 2ϵ. Hence by minimizing the distance between the features and a proxy centre vector, we are essentially minimizing the distances among feature vectors as well. Another example is the proxy-triplet loss [18, 27], in which the triplet formulation is used albeit with positive and negative samples replaced with weight vectors corresponding to sample identities. Once the feature vector and class weight vectors are normalised to lie on a hyper-sphere, the resulting loss tends to maximise the cosine similarity (and hence minimise the angular distances) among the features and their corresponding class vectors. Hence the class vectors play a similar role of proxy vectors as the feature-centres played in CenterLoss. The most popular approach in the category of proxy-based losses is the Normalised Softmax (NS) loss [27] and its variants such as SphereLoss [15], CosineLoss [28] and ArcLoss [7] etc. In the NS loss, the features and weight vectors are normalised to lie on a hyper-sphere with a large radius. To enforce a separation between clusters of different identities, variants of NS loss, such as SphereLoss, CosineLoss and ArcLoss etc., introduce the concept of a margin that enforces separation among the weight vectors of the various identities.

It has been observed [27, 28, 7] that the NS loss and its variants do not converge unless a sufficiently large radius of the hyper-sphere is employed. This observation (and also the effectiveness of the margin) is usually explained intuitively without much theoretical justifications. In our work, we discuss in detail (in Section 4.1.1) that the requirement of a large radius essentially means that the Softmax function tightly approximates the ArgMax function. Consequently, the derivatives of the loss function contain contributions from the target class and the nearest-neighbour negative class only, while the derivatives of all other negative classes approach to zero. Alternatively, for a hyper-sphere of a large radius, the logSumExp (LSE) function tightly approximates the max function. As a result, the NS loss and its variants become approximately equal to a proxy-triplet loss that only focuses on nearest-neighbour negative proxy. This motivates us to propose a novel variant of the proxy-triplet loss (entitled nearest-proxies-triplet or **NPT loss**) that directly creates a separation between a feature and its nearest-neighbour negative proxies/class-weight vectors.

There are a number of advantages of the proposed scheme over SOTA losses. Firstly, the proposed loss is a direct and simple mechanism to achieve conditions **C1** and **C2**, described above. Secondly, the proposed loss converges for small as well as large values of the radius of the normalising hyper-sphere. This leads to performance improvements over SOTA losses as we show in our experiments in Section 6.2.1 and 6.2.2. Also, the proposed loss has the flexibility of considering top-k nearest-neighbours which is lost in Softmax losses once the radius is large. This flexibility allows us to outperform SOTA losses in some cases, as we show in Section 6.1

The proposed scheme can be viewed in another interesting perspective. It is intuitively obvious that for a given input image, a large number of its hard-negative examples will be located in the nearest-neighbour identity (See Figure 1). We give theoretical evidence to support this intuition in Proposition 1 in Section 4 (and also provide empirical evidence in supplementary material). Moreover, by working with a loss that directly creates a separation between a feature and its nearest-neighbour negative classes, we are able to theoretically show that minimising the proposed loss does ensure a minimum separation between all classes (See Property 2, Section 4.3).

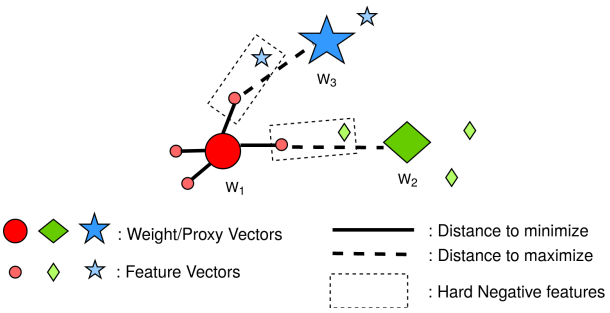

Figure 1. W_1, W_2 and W_3 represent class-weight vectors that can concisely represent an identity and work as proxies for the features of their respective identities. The dotted rectangles represent hard-negative pairs. The proposed loss maximises the distance of a feature to its nearest-neighbour negative proxy only, which is most likely to contain the hard negative.

2. Related Work

There is a long list of FR solutions in the computer vision literature; however, as far as FR using DCNN is concerned,

Facenet [25] can be considered as the pioneering work. In [25], it is proposed to use triplet loss with batch-mining for face recognition. Some subsequent works [9, 32] focused on efficient mining/sampling strategies to improve the performance of the trained network. Instead of directly using the feature triplets, in [30], it was suggested to approximate and update the feature centres of each class in each batch and then minimise the distance of the features from their respective centres. Another line of research focused solely on Softmax based loss functions and in [23] and [27], it was suggested that constraining the features and weight vectors to lie on a hyper-sphere is an effective strategy for improving the performance of Softmax loss for FR applications. In [15], [28], [7] and some other similar works, an additive/multiplicative margin was introduced inside the normalised Softmax loss function. The introduction of a margin significantly boosts the performance of these Softmax based loss functions and the additive margin introduced in CosFace [28] and ArcFace[7] outperformed all other solutions and has become a standard baseline for FR systems. In the last couple of years, a number of variants to [28, 7] has been suggested with marginal improvements in the accuracy. In [35], a mechanism is defined that adjusts the hyper-parameters of CosFace in an adaptive manner. In [29], an effort is made to combine the advantages of margin and feature mining and recently in [12], curriculum-learning [1] is incorporated with ArcFace to improve the performance. Another variant of ArcFace has been suggested in [6] that utilises multiple weight vectors for each identity. While this technique does not improve performance in itself; however, it does make the training more robust to noise in the training data.

It is also worth mentioning some recent developments in deep proxy based metric learning [18] that are intimately related with the modern FR solutions discussed above. In [18], proxy-based Neighbourhood Component Analysis (NCA) loss [24] and proxy-based triplet loss are discussed. Note that proxy-triplet loss also appeared independently in the FR literature in [27]. The normalised Softmax is closely related with the proxy-NCA formulation and recently, [20] has shown that the normalised Softmax loss is also a smoothed version of the proxy-triplet loss. In [8] an upper bound on proxy-triplet has been suggested and optimised and in [14], an effort is made to combine pair-based and proxy-based losses.

Our work is closely related to the recent developments in both DCNN based FR and deep proxy-based metric learning. We generalise proxy-triplet [18], Normalised Softmax [27], CosFace [28] and Proxy-NCA [18] in a single framework and show that these disparate losses are essentially special cases of our proposed formulation. Also, our formulation allows us to theoretically show that minimising the proposed loss guarantees a minimum separation among all identities in the n-dimensional feature space. No SOTA solution has been shown to exhibit this property, to the best of our knowledge. Empirical results show that our proposed loss consistently achieves SOTA performance, which confirms its effectiveness.

3. Problem Formulation

Let us suppose, we have a training dataset of N face images that belong to C different identities. Let us denote our CNN as a nonlinear function $f(\boldsymbol{x};\boldsymbol{w})$, where $\boldsymbol{x}$ is the input and $\boldsymbol{w}$ are the weights of the CNN. For each arbitrary input image $\boldsymbol{x}_i$ belonging to the ith identity, we have an output feature vector $\boldsymbol{z}_i = f(\boldsymbol{x}_i;\boldsymbol{w}) \in \mathcal{R}^n$. We define a triplet of features $(\boldsymbol{z_{i1}}, \boldsymbol{z_{i2}}, \boldsymbol{z_j})$, i.e., $\boldsymbol{z_{i1}}$ and $\boldsymbol{z_{i2}}$ belong to the same identity i, and $\boldsymbol{z_j}$ belongs to an identity $j \neq i$. Let $T = \{(\boldsymbol{z_{i1}}, \boldsymbol{z_{i2}}, \boldsymbol{z_j})\}$ be the set of all possible triplets from the training data. Let $d(\boldsymbol{z}_i, \boldsymbol{z}_j)$ be a distance metric defined on $\mathcal{R}^n$, then we can state the *condition of ideal ranking* as:

$$d(\boldsymbol{z}_{i1}, \boldsymbol{z}_{i2}) < d(\boldsymbol{z}_{i1}, \boldsymbol{z}_j) \qquad \forall(\boldsymbol{z_{i1}}, \boldsymbol{z_{i2}}, \boldsymbol{z_j}) \in T. \quad (1)$$

While any arbitrary distance metric might be employed in (1), in this work, we define $d(\boldsymbol{z}_i, \boldsymbol{z}_j)$ to be the squared-euclidean distance, i.e., $d(\boldsymbol{z}_i, \boldsymbol{z}_j) = (\boldsymbol{z}_i - \boldsymbol{z}_j)^T(\boldsymbol{z}_i - \boldsymbol{z}_j)$. Note that the squared-euclidean distance is not a proper metric, in the sense, that it does not obey the triangle inequality; however, the ordering of the nearest-neighbours is not affected if we use either the euclidean or the squared-euclidean distance, i.e., if $d(\boldsymbol{z}_{i1}, \boldsymbol{z}_{i2}) < d(\boldsymbol{z}_{i1}, \boldsymbol{z}_j)$, then $d_e(\boldsymbol{z}_{i1}, \boldsymbol{z}_{i2}) < d_e(\boldsymbol{z}_{i1}, \boldsymbol{z}_j)$, or vice versa, where $d_e(\boldsymbol{z}_i, \boldsymbol{z}_j) = \sqrt{d(\boldsymbol{z}_i, \boldsymbol{z}_j)}$ is the euclidean distance between $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$. Since the squared-euclidean is easier to optimise and does not affect the ranking condition in (1), hence we opt for the squared distance, in our work.

It has been suggested in [27], that CNN based FR systems work better if the features are normalised to lie on a hyper-sphere. A theoretical justification is provided in [18], where it has been suggested that surrogate metric losses, such as triplet-loss etc, provide a tighter upper bound on the ideal ranking loss if features are normalised to lie on a hyper-sphere. Accordingly, in our work, we force all features $\boldsymbol{z}_i$ to lie on a hyper-sphere of radius r and $d(\boldsymbol{z}_i, \boldsymbol{z}_j) = 2r^2 - 2\boldsymbol{z}_i^T\boldsymbol{z}_j = 2r^2(1 - \hat{\boldsymbol{z}}_i^T\hat{\boldsymbol{z}}_j)$, where $\hat{\boldsymbol{z}}_i$ and $\hat{\boldsymbol{z}}_j$ are unit vectors in the direction of $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$, respectively, and $\hat{\boldsymbol{z}}_i^T\hat{\boldsymbol{z}}_j$ is the cosine similarity between $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$.

A direct method to achieve the condition in (1) is to minimise the standard triplet loss [25], defined as:

$$L_{\text{triplet}} = \max\{0, d(\boldsymbol{z}_{i1}, \boldsymbol{z}_{i2}) - d(\boldsymbol{z}_{i1}, \boldsymbol{z}_j) + \Delta\}, \quad (2)$$

where Δ is a positive margin. However, the standard triplet loss has two serious shortcomings: firstly, for a training set of N images, we will have possible triplets of the order

$\mathcal{O}(N^3)$. This significantly increases the total iterations required in the training process. Secondly, most of the triplets will satisfy the triplet constraint, i.e., $d(\boldsymbol{z}_{i1}, \boldsymbol{z}_{i2}) - d(\boldsymbol{z}_{i1}, \boldsymbol{z}_j)$ will be less than $-\Delta$ for many of the triplets and hence the loss will be zero. These triplets will slow down convergence of the training process. To overcome these shortcomings, it is suggested in [25] to search/mine for *hard-positive* samples, i.e., $\max_k\{d(\boldsymbol{z}_i, \boldsymbol{z}_k)\}$, where $\boldsymbol{z}_i$, $\boldsymbol{z}_k$ belong to the same identity, and to search for *hard-negative* samples, i.e., $\min_j\{d(\boldsymbol{z}_i, \boldsymbol{z}_j)\}$, where $\boldsymbol{z}_i$, $\boldsymbol{z}_j$ belong to the separate identities. However, such a mining process is infeasible if applied across the entire training data. Since the CNN is trained using a batch stochastic gradient descent algorithm, an alternative is to mine for hard-positives and -negatives across a single batch in each iteration. This, however, necessitates that a sufficiently large batch-size is chosen, which is computationally prohibitive and also can reduce the generalisation of the trained network.

An alternative to the standard triplet loss is the proxy triplet loss suggested in [27, 18]. To apply the proxy triplet loss, we take a set of C, n-dimensional weight vectors $\boldsymbol{W}_1, \boldsymbol{W}_2, \cdots, \boldsymbol{W}_C$, with $||\boldsymbol{W}_i|| = r$, for all i. The weight vector $\boldsymbol{W}_i$ acts as a proxy for all features that belong to the ith identity. Consequently, instead of evaluating distances between sample pairs, we evaluate distances between samples and proxies, i.e., for each training sample $\boldsymbol{x}_i$, with corresponding feature $\boldsymbol{z}_i$, the following loss is evaluated:

$$L_{\text{proxy-triplet}} = \sum_j \max\{0, d(\boldsymbol{z}_i, \boldsymbol{W}_i) - d(\boldsymbol{z}_i, \boldsymbol{W}_j) + \Delta\}. \tag{3}$$

Note that, as suggested in [18], for each input $\boldsymbol{x}_i$, the proxy-triplet loss considers **all** negative proxies and hence covers the entire dataset. We will show in our experiments, that the proxy-triplet loss does not work well for face recognition. We suggest an alternative loss that focuses on nearest-neighbouring negative proxies only and hence has an implicit mechanism to mine hard-negatives and outperforms the standard proxy-triplet loss and achieves SOTA results.

4. Proposed Loss

The proposed loss relies on the concept of *nearest-neighbour negative proxies* defined as follows:

Definition. *Let $\mathcal{W} = \{\boldsymbol{W}_1, \boldsymbol{W}_2, \cdots, \boldsymbol{W}_C\}$ be the set of weight vectors (i.e., proxies) corresponding to the C identities. Let $\boldsymbol{z}_i$ be an arbitrary feature of identity i with weight vector $\boldsymbol{W}_i$. We define a set $\mathcal{W}_n^{(i)} = \{\boldsymbol{W}_1^{(i)}, \boldsymbol{W}_2^{(i)}, \cdots, \boldsymbol{W}_k^{(i)}\} \subseteq \mathcal{W}$ to be the top-k nearest-neighbour negative proxies of $\boldsymbol{z}_i$, if $d(\boldsymbol{z}_i, \boldsymbol{W}_1^{(i)}) \leq d(\boldsymbol{z}_i, \boldsymbol{W}_2^{(i)}) \cdots \leq d(\boldsymbol{z}_i, \boldsymbol{W}_k^{(i)}) \leq d(\boldsymbol{z}_i, \boldsymbol{W}_k)$, for all $k \neq i$ and $\boldsymbol{W}_k \notin \mathcal{W}_n^{(i)}$.*

Using the above definition, we define the proposed loss as follows:

$$L_{\text{NPT}} = \sum_k \underbrace{max\{0, d(\boldsymbol{z}_i, \boldsymbol{W}_i) - d(\boldsymbol{z}_i, \boldsymbol{W}_k^{(i)}) + \Delta\}}_{L_k}. \tag{4}$$

Alternatively, we can write

$$L_{\text{NPT}} = 2r^2 \sum_k max\{0, \hat{\boldsymbol{z}}_i^T \hat{\boldsymbol{W}}_k^{(i)} - \hat{\boldsymbol{z}}_i^T \hat{\boldsymbol{W}}_i + \frac{\Delta}{2r^2}\}. \tag{5}$$

where $\hat{\boldsymbol{z}}_i$, $\hat{\boldsymbol{W}}_k^{(i)}$ and $\hat{\boldsymbol{W}}_i$ are unit vectors in the direction of $\boldsymbol{z}_i$, $\boldsymbol{W}_k^{(i)}$ and $\boldsymbol{W}_i$, respectively. Also, $\hat{\boldsymbol{z}}_i^T \hat{\boldsymbol{W}}_k^{(i)} = \cos\theta_k^{(i)}$ and $\hat{\boldsymbol{z}}_i^T \hat{\boldsymbol{W}}_i = \cos\theta_i$, where $\theta_k^{(i)}$ and θ_i are the angular separations between $\boldsymbol{z}_i$ and $\boldsymbol{W}_k^{(i)}$ and $\boldsymbol{z}_i$ and $\boldsymbol{W}_i$, respectively. Note that instead of using all negative proxies, the proposed loss only uses the top-k nearest-neighbour negative proxies. We motivate and justify the proposed loss using the following

1. We show that SOTA Softmax based losses are a special case of the proposed loss under LogSumExp (LSE) approximation.

2. Since the samples corresponding to the nearest-neighbour negative proxy are most likely to be hard-negatives (see Figure 1), the proposed loss is a proxy-triplet loss with an implicit mechanism of sample mining. This intuitive idea is further confirmed theoretically in Section 4.2

4.1. Relation with SOTA Softmax losses

CosFace[28], ArcFace[7] and their variants [35, 12] are the SOTA losses that rely on Normalised Softmax[27] formulation with additive margins to achieve SOTA accuracy in FR as well as in metric learning [19] tasks. The CosFace loss is described as[3]

$$L_{\text{CF}} = -\log \underbrace{\frac{exp(R^2(\cos\theta_i - m))}{exp(R^2(\cos\theta_i - m)) + \sum_{j\neq i} exp(R^2 \cos\theta_j)}}_{p_i}, \tag{6}$$

where $\cos\theta_l = \hat{\boldsymbol{z}}_i^T \hat{\boldsymbol{W}}_l, \forall l$. We can re-write it as

$$L_{\text{CF}} = -R^2(\cos\theta_i - m) + \underbrace{\log\left(exp(R^2(\cos\theta_i - m)) + \sum_{j\neq i} exp(R^2\cos\theta_j)\right)}_{\text{LSE}}. \tag{7}$$

[3] A similar discussion, with slight adjustments for ArcFace, is presented in supplementary material

The LSE term in (7) is bounded by

$$\max\{\cos\theta_i - m, \cos\theta_j\} < \frac{1}{R^2} LSE \leq \max\{\cos\theta_i - m, \cos\theta_j\} + \frac{\log(C)}{R}, \tag{8}$$

for $j = 1, 2, \cdots, C$, $j \neq i$ and C is the total number of identities. Hence, when R^2 is large, LSE $\approx R^2 \max\{\cos\theta_i - m, \cos\theta^{(i)}_{\text{min}}\}$, where $\cos\theta^{(i)}_{\text{min}} = \hat{\boldsymbol{z}}_i^T \hat{\boldsymbol{W}}_1^{(i)}$, where $\hat{\boldsymbol{W}}_1^{(i)}$ is the nearest-neighbour negative proxy of $\boldsymbol{z}_i$. Under this approximation, $L_{\text{CF}} \approx 0$ if $\cos\theta^{(i)}_{\text{min}} \leq \cos\theta_i - m$; otherwise, $L_{\text{CF}} \approx R^2(\cos\theta^{(i)}_{\text{min}} - \cos\theta_i + m)$, hence

$$L_{CF} \approx R^2 \max\{0, \cos\theta^{(i)}_{\text{min}} - \cos\theta_i + m\}. \tag{9}$$

Note that (9) is a special case of (5), with $k = 1$ and $R^2 = 2r^2$ and $m = \frac{\Delta}{2r^2}$. If we put $m = 0$, in (6), it becomes the expression for Normalised Softmax or NormFace [27]. Accordingly, NormFace is a special case of our proposed formulation with $k = 1$ and $R^2 = 2r^2$ and $\Delta = 0$. Another interesting relation of the proposed scheme can be established with the Proxy-NCA [18] loss. The proxy-NCA is defined as

$$L_{\text{P-NCA}} = -\log \frac{\exp(-d(\boldsymbol{z}_i, P_i))}{\sum_{j \neq i} \exp(-d(\boldsymbol{z}_i, P_j))}. \tag{10}$$

Under LSE approximation, Proxy-NCA becomes

$$L_{\text{P-NCA}} \approx d(\boldsymbol{z}_i, P_i) - d(\boldsymbol{z}_i, P_1^{(i)}), \tag{11}$$

where $P_1^{(i)}$ is the nearest-neighbour negative proxy of $\boldsymbol{z}_i$. Note that (11) is a special case of (4) with $P_i = \boldsymbol{W}_i$, $P_j = \boldsymbol{W}_j$, and $\Delta > \max\{d(\boldsymbol{z}_i, P_1^{(i)}) - d(\boldsymbol{z}_i, P_i)\}$. For such a large Δ, the max condition is never satisfied and (11) and (4) become equivalent. Note that the distances in Proxy-NCA are evaluated on a hyper-sphere, similar to other SOTA losses and $\max\{d(\boldsymbol{z}_i, P_1^{(i)}) - d(\boldsymbol{z}_i, P_i)\} < 4r^2$, where r is the radius of the hyper-sphere. This also shows that under the LSE approximation, Proxy-NCA is special case of CosFace with $m > 2$.

Remark 1. *: Since margin has been understood only intuitively in the literature, it led to the mistaken notion in [28] that CosFace will not converge for large values of m. As we have shown above, a large margin effectively means the* max *(or the hinge) condition is never satisfied and we get an algorithm similar to proxy-NCA that does converge and its performance is only slightly inferior to CosFace. It is intuitively obvious that algorithms with the hinge condition would be more robust as compared to solutions without hinge; however, a detailed analysis of the significance of hinge from an optimisation perspective is beyond the scope of this work and shall be dealt with in our future works.*

4.1.1 Analysis of derivatives

It is interesting to compare the derivatives of the proposed loss with those of Softmax based losses. Let us define a logit as $\gamma_l = \hat{\boldsymbol{z}}_i^T \hat{\boldsymbol{W}}_i$. The derivatives of the proposed loss $w.r.t.$ to the logits are given as

$$\frac{\partial L_{\text{NPT}}}{\partial \gamma_l} = \begin{cases} -2r^2 & l = i \\ 2r^2 & l \in \text{top-k} \\ 0 & e.w. \end{cases} \tag{12}$$

Whereas, the derivatives of CosFace are given as

$$\frac{\partial L_{\text{CF}}}{\partial \gamma_l} = \begin{cases} -R^2(1 - p_i) & l = i \\ R^2 p_j & l = j \end{cases}, \tag{13}$$

where p_i is defined in (6) and $p_j = \frac{\exp(R^2 \cos\theta_j)}{\sum_l \exp(R^2 \cos\theta_l)}$. While both the derivatives might appear quite distinct, they are intimately related for large values of R. Note that Softmax is essentially a soft version of the argmax function and for large values of R the approximation approaches equality. Let us consider an example: suppose we have 5 classes with the following values of logits $[0.4, 0.5, 0.2, 0.3, 0.01]$. The corresponding output of the Softmax function for $R = 1$ is $[0.22, 0.24, 0.18, 0.2, 0.15]$. Now consider the output of the Softmax for $R = 64.0$, i.e., $[1.65e^{-3}, 0.998, 4.5e^{-9}, 2.7e^{-6}, 2.4e^{-14}] \approx [0, 1, 0, 0, 0]$. Consequently, for large values of R, all probability in the derivatives in equation (13) is concentrated only in the maximum logit. In other words if $\gamma_i > \gamma_1^{(i)}$, where $\gamma_1^{(i)}$ is the logit corresponding to nearest-neighbour negative proxy, then $p_i \approx 1$, $p_j \approx 0$ and hence all derivatives become zero. On the other hand, if $\gamma_1^{(i)}$ is greater than the target logit γ_i, then $p_i \approx 0$, and probability corresponding to $\gamma_1^{(i)}$ approaches 1. The derivative of the target class in this case is $-R^2$ and that of the nearest-neighbour non-target is R^2 and all other derivatives are zero. Note that, only the nearest-neighbour negative class is playing any role in the derivatives, similar to the proposed loss with $k = 1$.

4.2. Implicit sample mining in the proposed loss

In this section, we show that the proposed loss is a proxy-triplet loss with an implicit mechanism of sample mining using the following proposition:

Proposition 1. *Let $\boldsymbol{z}_i$ be a given feature vector belonging to identity i. Let $\boldsymbol{z}_j$ and $\boldsymbol{z}_k$ be random and independent feature vectors drawn from identities j and k, respectively, with $\mathbb{E}[\boldsymbol{z}_j] = \boldsymbol{m}_j$ and $\mathbb{E}[\boldsymbol{z}_k] = \boldsymbol{m}_k$. Let $\tilde{\boldsymbol{m}}_j = r\frac{\boldsymbol{m}_j}{||\boldsymbol{m}_j||}$ and $\tilde{\boldsymbol{m}}_k = r\frac{\boldsymbol{m}_k}{||\boldsymbol{m}_k||}$. Let $\boldsymbol{W}_i$, $\boldsymbol{W}_j$, $\boldsymbol{W}_k$ be the weight vectors corresponding to $\boldsymbol{z}_i$, $\boldsymbol{z}_j$, and $\boldsymbol{z}_k$, respectively. If the following assumptions are true:*

A1: $||\boldsymbol{m}_j|| = \beta$, *for all* $j = 1, 2, \cdots, C$.

A2: $\boldsymbol{W}_j = \tilde{\boldsymbol{m}}_j$, *for all* $j = 1, 2, \cdots, C$.

Then, $\mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{z}_j)|\boldsymbol{z}_i] < \mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{z}_k)|\boldsymbol{z}_i]$, *if* $d(\boldsymbol{z}_i, \boldsymbol{W}_j) < d(\boldsymbol{z}_i, \boldsymbol{W}_k)$.

Proof: In supplementary material

Remark 2. From the above result, we note that, on average, the distance between $\boldsymbol{z}_i$ and a sample from its nearest-neighbour negative proxies is less than the distance between $\boldsymbol{z}_i$ and a sample from any other proxy. Hence, if we mine for hard-negative samples, on average, they will be found in the identity corresponding to the nearest-neighbour negative proxies. The proposed loss in (4) is enforcing a separation between $\boldsymbol{z}_i$ and nearest proxies, and hence, on average is enforcing a separation between $\boldsymbol{z}_i$ and its corresponding hard-negatives. Consequently, the proposed loss is acting as a proxy-triplet loss similar to (3); however, with implicit hard-negative mining.

Remark 3. The above results rely on the assumptions **A1** and **A2**. A discussion and some empirical evidence regarding the validity of these assumptions is provided in supplementary material.

4.3. Properties of the Proposed Loss

In addition to the motivations described above, in this section we discuss some properties of the proposed loss that further support its suitability for face recognition tasks.

Property 1. If $L_{\text{NPT}} < \Delta$ for a given $\boldsymbol{z}_i$, then $d(\boldsymbol{z}_i, \boldsymbol{W}_i) < d(\boldsymbol{z}_i, \boldsymbol{W}_j)$ for all $j = 1, 2, \cdots, C, j \neq i$.

Proof: Noting that $L_{\text{NPT}} = \sum_k L_k$, where $L_k > 0$ is defined in (4). If $L_{\text{NPT}} < \Delta$, then $L_k < \Delta, \forall k$, then $d(\boldsymbol{z}_i, \boldsymbol{W}_i) - d(\boldsymbol{z}_i, \boldsymbol{W}_1^{(i)}) < 0$, hence $d(\boldsymbol{z}_i, \boldsymbol{W}_i) < d(\boldsymbol{z}_i, \boldsymbol{W}_1^{(i)}) < d(\boldsymbol{z}_i, \boldsymbol{W}_j)$, where the last inequality follows from the fact that $\boldsymbol{W}_1^{(i)}$ is the nearest-neighbour negative proxy of $\boldsymbol{z}_i$.

Remark 4. Note that if the above property is true for all training samples $\{\boldsymbol{z}_i\}$, then we get ideal classification.

Property 2. If $\mathbb{E}[L_{\text{NPT}}] < \epsilon$, where ϵ is a small positive number and assumptions **A1** and **A2** are true, then $d(\boldsymbol{W}_i, \boldsymbol{W}_j) > \Delta - \epsilon$ for all $i, j = 1, 2, \cdots, C, i \neq j$.

Proof: If $\mathbb{E}[L_{\text{NPT}}] < \epsilon$, then $\mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{W}_n^{(i)})] - \mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{W}_i)] > \Delta - \epsilon$. Noting that $\mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{W}_i)] = 2r^2 - 2\gamma r^2$ and $\mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{W}_n^{(i)})] = 2r^2 - 2\gamma \boldsymbol{W}_i^T \boldsymbol{W}_n^{(i)}$, we can write

$$\begin{aligned}\mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{W}_n^{(i)})] - \mathbb{E}[d(\boldsymbol{z}_i, \boldsymbol{W}_i)] &= \gamma(2r^2 - 2\boldsymbol{W}_i^T \boldsymbol{W}_n^{(i)}) \\ &= \gamma d(\boldsymbol{W}_i, \boldsymbol{W}_n^{(i)}) > \Delta - \epsilon \end{aligned} \tag{14}$$

Note that $\gamma = \frac{\beta}{r}$, where $\beta = ||\mathbb{E}[\boldsymbol{z}_i]||$. Since $\mathbb{E}[d(\boldsymbol{z}_i, \mathbb{E}[\boldsymbol{z}_i])] = r^2 - \beta^2 > 0$, hence $\beta < r$ and $0 < \gamma < 1$. Consequently, $d(\boldsymbol{W}_i, \boldsymbol{W}_n^{(i)}) > \frac{\Delta - \epsilon}{\gamma} > \Delta - \epsilon$. Also, by definition of $\boldsymbol{W}_n^{(i)}$, $d(\boldsymbol{W}_i, \boldsymbol{W}_n^{(i)}) \leq d(\boldsymbol{W}_i, \boldsymbol{W}_j)$ for $j \neq i$, hence we can write $d(\boldsymbol{W}_i, \boldsymbol{W}_j) > \Delta - \epsilon$, which is the required proof.

Remark 5. Note that at the end of the training process, $\mathbb{E}[L_{\text{proposed}}] \approx 0$ and hence $d(\boldsymbol{W}_i, \boldsymbol{W}_j)$ is guaranteed to be greater that Δ. Therefore, minimizing the proposed loss ensures that all classes are at-least separated by a margin Δ. Ideally, we would like Δ to be large; however, if we make it too large then the hinge condition in (4) is never satisfied and the performance is slightly degraded. Our empirical results show that setting $\Delta = r^2$ gives optimal performance.

5. Experiments

5.1. Datasets

For training, we work with the CASIA (i.e., small protocol) [34] and the MS1M (i.e., large protocol) [10] datasets. Both datasets are available from the insightFace repository [4] and we use the ArcFace version of MS1M, i.e., MS1Mv2. CASIA contains around 10K identities and 0.5M images; whereas, MS1Mv2 contains about 85K identities and around 5.8M images. For testing, we use the standard benchmark LFW[11], CPF-FP [26] for FR under pose variations, CALFW [37] and AgeDB[17] for FR under age variations, and large scale datasets such as MegaFace [13], IJBB [31], IJBC [16] for FR under realistic unconstrained environments. For an independent evaluation, we submitted one of our trained network in the IFRT [5] challenge and we give the corresponding results.

5.2. Implementation Details

We perform experiments using two different architectures; a lightweight MobileFaceNet [2] with 256-dimensional feature vector and a ResNet-50 architecture with 512-dimensional feature vector. For CASIA, we employ a batch size of 64 and a base learning rate of 0.1 that is divided by 10, at 30th and 45th epoch. The training is finished after 65 epochs. For MS1M, we use a batch size of 512, and a base learning rate of 0.1 that is divided by 10, at 10th and 18th epoch. The training is finished after 21 epochs. The momentum is set to 0.9 for all experiments. For ResNet-50, weight decay is the same for both datasets., i.e., $1e^{-4}$; whereas, for MobileFaceNet, we apply a layer-wise weight decay of $5e^{-4}$ for loss layer, $1e^{-4}$ for fc layer and $1e^{-5}$ for the backbone network.

6. Results and Discussion

6.1. Ablation Study

We note from the expression of the proposed loss in (5) that the radius of the hyper-sphere is acting as a multiplication factor and can be lumped in the learning rate. Therefore, we set it to 1 for all experiments. The other hyper-parameters in the proposed loss are Δ and k. First, we set $k = 1$ and evaluate the performance of our models trained on CASIA for different values of Δ on a subset of MegaFace with 10K distractors. The rank-1 accuracy for MobileFaceNet and ResNet-50 are tabulated in Table 1. In both cases, we get the best accuracy when $\Delta = r^2$. The accuracy falls rather sharply when Δ is made smaller, which is obvious from **Property 2** in Section 4.3, i.e., Δ controls the minimum separation between identities and a small Δ will lead to performance degradation. Interestingly, for large values of Δ, performance decreases but only slightly. As has been remarked earlier, a large Δ means that the hinge (or $\max$) condition is never invoked and it leads to performance degradation since the derivatives never become zero even when the network is close to the optimum. Note, however, a large Δ does not cause any significant loss in performance and we observe similar results for all $\Delta \geq 2r^2$.

We then set $\Delta = r^2$ and tabulate the rank-1 accuracy of our models in Table 2 for various numbers of top-k nearest-neighbour proxies. Note that considering more than one nearest-neighbour is giving superior performance for MobileFaceNet; however, the performance is best for $k = 1$ for ResNet-50. Even for MobileFaceNet, considering more than 6 nearest proxies leads to a performance loss. In both cases, we observe a significant degradation in performance, when all proxies are considered. In all subsequent experiments, we set $k = 6$ for MobileFaceNet and $k = 1$ for ResNet-50.

$\Delta/2r^2$	MobileFaceNet	ResNet-50
0.2	90.48	91.5
0.3	93.1	96.35
0.5	93.3	96.61
1	92.83	96.51
2	92.83	96.32

Table 1. Rank-1 accuracy evaluated on MegaFace (10K distractors) for different values of Δ.

6.2. Comparison with SOTA

6.2.1 Small Protocol

SOTA FR solutions rarely report performance on small protocol; therefore, we trained our MobileFaceNet and ResNet-50 models on CASIA using a number of SOTA techniques. The performance of ResNet-50 on LFW, AgeDB, CFP-FP and CALFW is tabulated in Table 3. The performance of MobileFaceNet on MegaFace dataset with 1M distractors is tabulated in Table 4. We note that in all instances, the proposed loss outperforms the SOTA solutions. In Figure 2, we plot the ROC curve of the proposed solution, evaluated on MegaFace, and compare it with ArcFace and CosFace. We note that the proposed scheme is consistently better than other losses for all values of false acceptance rate (FAR).

Top-k	MobileFaceNet	ResNet-50
1	93.30	97.40
2	93.79	97.19
6	94.50	97.15
10	94.46	95.83
All	49.53	66.93

Table 2. Rank-1 accuracy evaluated on MegaFace (10K distractors) for different values of top-k nearest-proxies.

Method	LFW	AgeDB	CFP-FP	CALFW
Norm-Softmax[27]	97.55	87.14	87.15	88.46
Proxy-Triplet[27, 18]	97.48	84.15	90.9	85.31
ArcFace[7]	99.3	94.23	95.3	93.34
Curricular-Face[12]	99.36	94.18	95.61	93.34
Proposed (NPT loss)	**99.40**	**95.38**	**96.81**	**93.46**

Table 3. Performance comparison on LFW, AgeDB, CFP-FP and CALFW for ResNet-50 trained on CASIA.

Method	Id	Veri
Normalised-Softmax[27]	67.20	73.37
Proxy-Triplet[27, 18]	21.67	29.30
ArcFace[7]	83.91	87.65
CosFace[28]	84.02	88.25
Proposed (NPT loss)	**84.7**	**89.24**

Table 4. Performance comparison on MegaFace for MobileFaceNet trained on CASIA. Id refers to rank-1 accuracy and Veri. is face verification performance at $1e^{-6}$ FAR.

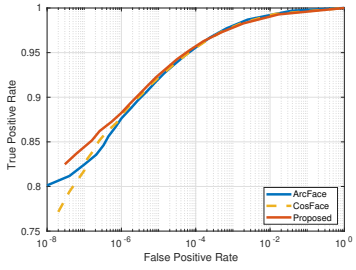

Figure 2. ROC curves evaluated on MegaFace for MobileFaceNet models trained on CASIA.

6.2.2 Large Protocol

In this section, we compare the performance of the proposed loss trained on MS1Mv2. We retrain ArcFace and Curricular-Face using our experimental settings; whereas, the performance of other SOTA schemes has been taken from the literature. In Table 5, we list the rank-1 and verification accuracy of a number of SOTA schemes evaluated on MegaFace. The verification has been performed for an FAR of $1e^{-6}$. We note that the proposed loss outperforms all SOTA solutions in both rank-1 as well as verification accuracy. In Figure 3, we plot the ROC curve of the proposed solution, evaluated on MegaFace, and compare it against ArcFace and Curricular-Face. We note that all three solutions have comparable performances and the proposed scheme shows improvement over other schemes at low FARs. In Table 6, we compare the performance of SOTA schemes on IJB-B and IJB-C datasets. While the proposed loss outperforms all other SOTA losses, ArcFace appears to have a slight edge at FAR=$1e^{-4}$; however, as depicted in Table 8, the proposed loss outperforms ArcFace at FAR=$1e^{-6}$ on both IJB-B and IJB-C datasets.

Method	Id	Veri
AdaCos[35]	97.41	-
PS2Grad[36]	97.25	-
MV-Arc-SoftMax[29]	97.14	97.57
ArcFace[7]	97.58	98.15
Curricular-Face[12]	96.98	98.41
Proposed (NPT loss)	**97.80**	**98.55**

Table 5. Performance comparison on MegaFace. Id refers to rank-1 accuracy and Veri. is face verification performance at $1e^{-6}$ FAR.

Method	IJB-B	IJB-C
ResNet50+DCN[33]	84.1	88.0
CosFace[28]	-	91.82
Crystal Loss[21]	-	92.29
AdaCos[35]	-	92.4
PS2Grad[36]	-	92.3
ArcFace[7]	**93.15**	**94.79**
Curricular-Face[12]	92.40	94.33
Proposed(NPT loss)	92.85	94.58

Table 6. Performance comparison on IJB-B and IJB-C datasets. The performance is evaluated @ FAR=$1e^{-4}$

6.2.3 Results on IFRT

IFRT is a fair benchmark for face recognition that has been recently introduced by the same authors that created ArcFace [7]. It consists of over **162K** non-celebrity images of around **24K** identities. It contains images of various sex, age and race groups. In Table 7, we show the results of our proposed algorithm, evaluated independently by the IRFT team, on their benchmark. Unfortunately, the leader-board of the challenge has not yet been published and so we can only compare our proposed algorithm with the baseline ArcFace results available from [5]. Note that for the case of Caucasian face images, the proposed method has the same accuracy as that of ArcFace. For all other cases, the proposed scheme outperforms Arcface and achieves a significantly better overall accuracy.

Algo.	African	Caucasian	S-Asian	E-Asian	All
NPT-loss	**73.67**	**83.24**	**80.25**	**31.29**	**64.50**
ArcFace	71.97	**83.24**	79.66	22.94	56.20

Table 7. Comparison on the IFRT challenge. The results show True Acceptance rate (TAR), with False Acceptance Rate (FAR) less than 1^{-6} using ResNet-50 and MS1M-v2

Method	IJB-B	IJB-C
NPT loss (Proposed)[33]	**44.72**	**82.22**
ArcFace[28]	36.79	81.95

Table 8. Performance comparison of proposed loss with ArcFace on IJB-B and IJB-C datasets @ FAR=$1e^{-6}$

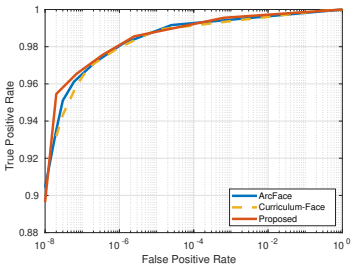

Figure 3. ROC curves evaluated on MegaFace for models trained on MS1Mv2.

7. Conclusion

In this work, we have proposed a novel loss formulation for face recognition tasks that generalises many state-of-the-art schemes. The proposed loss directly creates a separation between a feature vector and its top-k nearest-neighbour negative class weight vectors. We have shown that the proposed loss is equivalent to a triplet loss with proxies and an implicit mechanism of hard-negative mining. We have given theoretical evidence that minimising the proposed loss guarantees a separation between all classes/identities in the n-dimensional feature space. We have performed comprehensive set of experiments on a number of state-of-the-art face recognition benchmarks to confirm the efficacy of our solution. The proposed solution has consistently achieved state-of-the-art performance in all of our experiments.

References

[1] 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. 3

[2] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Chinese Conference on Biometric Recognition*, pages 428–438. Springer, 2018. 6

[3] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005. 1

[4] Jiankang Deng and Jia Guo. Insightface: 2d and 3d face analysis project. https://github.com/deepinsight/insightface, 2018. 6

[5] Jiankang Deng and Jia Guo. Insightface recognition test (ifrt). https://github.com/deepinsight/insightface/tree/master/challenges/IFRT, 2020. 6, 8

[6] Jiankang Deng, Jia Guo, Tongliang Liu, Mingming Gong, and Stefanos Zafeiriou. Sub-center arcface: Boosting face recognition by large-scale noisy web faces. In *Proceedings of the IEEE Conference on European Conference on Computer Vision*, 2020. 3

[7] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 1, 2, 3, 4, 7, 8

[8] Thanh-Toan Do, Toan Tran, Ian Reid, Vijay Kumar, Tuan Hoang, and Gustavo Carneiro. A theoretically sound upper bound on the triplet loss for improving the efficiency of deep distance metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10404–10413, 2019. 3

[9] Yueqi Duan, Lei Chen, Jiwen Lu, and Jie Zhou. Deep embedding learning with discriminative sampling policy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4964–4973, 2019. 3

[10] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: Challenge of recognizing one million celebrities in the real world. *Electronic imaging*, 2016(11):1–6, 2016. 1, 6

[11] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. 2008. 6

[12] 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. 3, 4, 7, 8

[13] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4873–4882, 2016. 6

[14] Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Proxy anchor loss for deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3238–3247, 2020. 3

[15] 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. 1, 2, 3

[16] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 International Conference on Biometrics (ICB)*, pages 158–165. IEEE, 2018. 6

[17] 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. 6

[18] Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 360–368, 2017. 2, 3, 4, 5, 7

[19] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *European Conference on Computer Vision*, pages 681–699. Springer, 2020. 4

[20] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Hao Li, and Rong Jin. Softtriple loss: Deep metric learning without triplet sampling. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6450–6458, 2019. 3

[21] Rajeev Ranjan, Ankan Bansal, Hongyu Xu, Swami Sankaranarayanan, Jun-Cheng Chen, Carlos D Castillo, and Rama Chellappa. Crystal loss and quality pooling for unconstrained face verification and recognition. *arXiv preprint arXiv:1804.01159*, 2018. 8

[22] Rajeev Ranjan, Ankan Bansal, Jingxiao Zheng, Hongyu Xu, Joshua Gleason, Boyu Lu, Anirudh Nanduri, Jun-Cheng Chen, Carlos D Castillo, and Rama Chellappa. A fast and accurate system for face detection, identification, and verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 1(2):82–96, 2019. 1

[23] Rajeev Ranjan, Carlos D Castillo, and Rama Chellappa. L2-constrained softmax loss for discriminative face verification. *arXiv preprint arXiv:1703.09507*, 2017. 3

[24] Sam Roweis, Geoffrey Hinton, and Ruslan Salakhutdinov. Neighbourhood component analysis. *Adv. Neural Inf. Process. Syst.(NIPS)*, 17:513–520, 2004. 3

[25] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 1, 3, 4

[26] 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. IEEE, 2016. 6

[27] 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. 1, 2, 3, 4, 5, 7

[28] 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. 1, 2, 3, 4, 5, 7, 8

[29] 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

[30] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer, 2016. 1, 2, 3

[31] 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. 6

[32] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2840–2848, 2017. 3

[33] Weidi Xie, Li Shen, and Andrew Zisserman. Comparator networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 782–797, 2018. 8

[34] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014. 1, 6

[35] 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 Conference on Computer Vision and Pattern Recognition*, pages 10823–10832, 2019. 3, 4, 8

[36] 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 Conference on Computer Vision and Pattern Recognition*, pages 9906–9914, 2019. 8

[37] Tianyue Zheng, Weihong Deng, and Jiani Hu. Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. *arXiv preprint arXiv:1708.08197*, 2017. 6