

Received 9 November 2025, accepted 24 November 2025, date of publication 26 November 2025,
date of current version 4 December 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3637551

RESEARCH ARTICLE

Rethinking Metric Learning: Enhancing Generalization to Unseen Classes

JINHEE PARK^{1,2,*}, HEE BIN YOO^{3,*}, BYOUNG-TAK ZHANG³, AND JUNSEOK KWON^{1,4}

¹School of Computer Science and Engineering, Chung-Ang University, Seoul 06974, South Korea

²Korea Electronics Technology Institute, Seongnam-si 13488, Republic of Korea

³School of Computer Science and Engineering, Seoul National University, Seoul 08826, Republic of Korea

⁴Graduate School of Artificial Intelligence, Chung-Ang University, Seoul 06974, South Korea

Corresponding author: Junseok Kwon (jskwon@cau.ac.kr)

This work was supported in part by Chung-Ang University Graduate Research Scholarship, in 2021; in part by the Institute of Information Communications Technology Planning Evaluation (IITP) grant (Artificial Intelligence Graduate School Program, Chung-Ang University) under RS-2021-II211341; and in part by the IITP RS-2022-II220953-PICA.

*Jinhee Park and Hee Bin Yoo contributed equally to this work.

ABSTRACT Deep Metric Learning (DML) is critical for constructing feature spaces where similar samples cluster closely, yet conventional cross-entropy (CE)-based methods often struggle to generalize to novel, unseen classes. The primary challenge arises from train-test scale misalignment, which disproportionately affects ambiguous (low-confidence) samples at test time and reduces separability. This paper presents a robust CE-based DML framework specifically designed to enhance generalization for unseen categories. We introduce Embedding Space Augmentation (ESA), a plug-in training component that i) deliberately lowers the confidence of hard samples to amplify corrective gradients and ii) guides updates along each class's principal eigen-direction, thereby reducing intra-class variance and restoring train-test norm alignment. We quantify alignment with scale-invariant diagnostics—the Relative Radius Ratio and mean-radius gaps—and observe that ESA maintains small gaps throughout training, whereas CE drifts over time. Comprehensive experiments on standard image retrieval benchmarks, including CUB200, CARS196, SOP, and In-Shop, validate the effectiveness of the proposed framework. Our method addresses the generalization gap, achieving superior retrieval performance and consistently yielding substantial Recall@1 gains over multiple baseline networks. The technique is modular and can be seamlessly integrated into existing CE-based DML methods, offering significant and consistent accuracy improvements.

INDEX TERMS Artificial intelligence, machine learning algorithms, computer vision.

I. INTRODUCTION

Leveraging the high representational capacity of deep neural networks (DNNs) [1], recent advances in metric learning—commonly referred to as deep metric learning—have led to significant performance gains. These advances have enabled a wide range of practical applications in computer vision, including image retrieval [2], [3], [4], clustering [5], [6], and classification tasks [7], [8].

The primary objective of deep metric learning (DML) is to learn a low-dimensional embedding space using DNNs, wherein semantically similar samples are projected close together, while dissimilar ones are mapped farther apart. To this end, a variety of pairwise loss-based approaches

have been proposed [2], [3], [9], [10], [11], leveraging relative distance constraints in the embedding space. Despite their empirical success, these methods often struggle to generalize effectively to unseen classes, limiting their applicability in open-set scenarios. Moreover, pairwise loss-based strategies—particularly those relying on triplet and quadruplet formulations—suffer from poor scalability, as the number of training pairs grows polynomially with respect to the size of the dataset, even when the dataset itself increases linearly.

To address the limitations inherent in pairwise loss-based DML, alternative approaches leveraging classification loss have been introduced [4], [12], [13], [14]. These methods employ ground-truth labels to train classifiers and construct embedding functions through softmax-based architectures. Notably, softmax regression-based DML [12] has shown

The associate editor coordinating the review of this manuscript and approving it for publication was Asadullah Shaikh¹.

comparable or superior effectiveness to pairwise-based methods in retrieval scenarios. Furthermore, Boudiaf et al. [15] theoretically establish that the cross-entropy (CE) loss can be viewed as an upper bound on pairwise losses, thereby providing a principled foundation for CE-based DML. In addition to strong empirical results, CE-based methods offer several advantages: they typically exhibit faster convergence, owing to the absence of explicit relative distance computations, and benefit from well-understood supervised learning dynamics.

Despite the empirical success of CE-based DML, conventional formulations primarily focus on maximizing inter-class separability in the embedding space. This often comes at the expense of intra-class compactness, resulting in suboptimal feature representations for samples belonging to the same class. While pairwise loss-based methods tend to cluster same-class samples into tight, point-like distributions, naïve CE-based approaches frequently project these samples along elongated directions or lines in the embedding space.

To explicitly reduce intra-class variance, various strategies have been proposed. These include incorporating intra-class distance penalties into the loss function [16], applying feature normalization techniques [12], [17], [18], introducing temperature scaling to modulate the loss landscape [19], [20], and enforcing angular margins between class centers [13], [21], [22]. Among them, temperature scaling has proven effective in reducing intra-class variance by amplifying output confidence scores during training [23].

Temperature scaling is widely used to control the sharpness of predicted probabilities: $p_i = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)}$, where smaller T yields *peaky* posteriors and larger T yields *smooth* ones for logits z . In metric-learning classifiers with dot-product logits $z \approx \|f\| \|w\| \cos \theta$, the *feature norm* $\|f\|$ effectively acts as a logit-scale (hence, an inverse-temperature) factor. **If the test-time scale distribution deviates from training (train–test scale misalignment)**, the model’s probabilities can become unintentionally smoother or peakier than intended, leading to confidence shifts and weaker margins, especially for ambiguous samples.

Figure 1 illustrates this effect in 1-D: (a) Under misalignment, smaller test-time norms shrink logits, flatten the softmax, and place ambiguous unseen samples near the decision boundary (nearly tied posteriors). (b) With our alignment mechanism, the effective logit scale is restored, logits regain the intended magnitude, posteriors sharpen on the correct side, and margins/separability improve, without any supervision from unseen classes.

While full *feature-normalization* can sidestep the above sensitivity, it also discards magnitude information that can serve as a useful proxy for confidence and calibration. Our goal is to retain this information *without* sacrificing generalization: we show that one can improve metric-learning performance by *aligning* the feature-norm scale between train and test (rather than removing it), thereby preserving confidence cues while mitigating scale-induced instability.

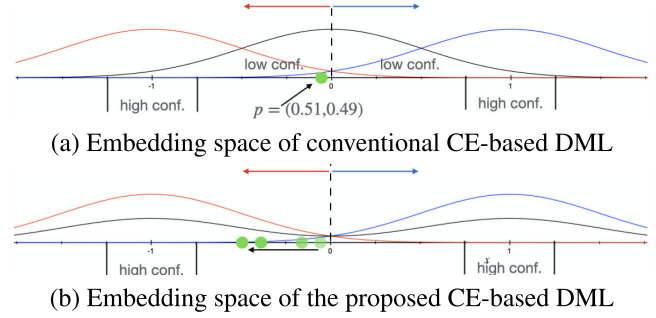


FIGURE 1. Basic idea with softmax sensitivity to logit scale. (a) *Scale misalignment at test time.* When test features have a *smaller feature norm* than the training distribution, the *logit scale shrinks* (logits ↓), making softmax posteriors overly flat and boundary-biased. As a result, ambiguous unseen samples (green) sit near the decision boundary (dashed line) with almost tied confidences (e.g., $p \approx (0.51, 0.49)$), even though class directions (red/blue) are clear. (b) *Scale alignment with our method.* By aligning the feature-norm scale (via eigen-guided updates on hard samples), logits are restored to the appropriate magnitude (logits ↑), which sharpens softmax and pushes samples toward high-confidence regions on the correct side. This increases margins and separability—without any supervision from unseen classes.

Addressing the aforementioned generalization challenge, we propose a CE-based DML framework that *explicitly models and corrects train–test scale misalignment*, especially for ambiguous (low-confidence) samples. By correcting these behaviors, the proposed method substantially enhances performance in metric learning tasks. To support our approach, we conduct a thorough analysis of how unseen class samples are positioned in embedding spaces trained exclusively on seen classes. Our analysis indicates that performance degradation stems from *relative norm shifts with respect to the training distribution* and the resulting *scale misalignment*. Nevertheless, we observe that such unseen samples often share latent similarities with a particular seen class (denoted as c_a), and their representations resemble those of hard samples within c_a that also exhibit low confidence scores. Motivated by this observation, our method encourages hard samples in c_a to attain higher confidence values during training. Consequently, ambiguous samples are *aligned* toward c_a along its principal direction, reducing intra-class variance and restoring train–test alignment. This mechanism alleviates misalignment and improves separability among unseen test classes without requiring unseen supervision. Furthermore, by promoting high-confidence representations of hard samples through embedding space augmentation, our approach simultaneously reduces intra-class variance for the training data and enhances class separability at test time.

Our contributions can be summarized as follows.

- We analyze unseen-class embedding behaviors through the lens of *train–test scale misalignment* and provide a revised perspective on CE-based metric learning for unseen generalization.
- We introduce an embedding-space augmentation technique that *aligns* ambiguous samples with semantically related seen-class directions, improving separability without unseen supervision.

TABLE 1. Comparative analysis of deep metric learning methods. The Cat (Category) column is abbreviated: **P** denotes Pairwise-based methods, and **C** denotes classification-based methods.

Cat	Method (Citation)	Core Mechanism	Primary Focus	Unseen Robustness Effort	Major Limitation
P	Contrastive Loss [9], [10]	Distance Margin Constraint	Intra-class Compactness	Hard Sample Selection	High Training Pair Count
	Triplet Loss [3], [24]	Anchor-P vs. N Distance	Intra-class Compactness	Hard Negative Mining	Sensitivity to Mining
	MS Loss [11]	General Pair Weighting	Effective Sample Selection	Adaptive Hard/Easy Handling	Limited Global Structure
C	SoftTriple Loss [14]	Multiple Proxies per Class	Inter-class Separation	Stable Class Prototypes	Increased # of parameters
	Proxy-NCA [4]	Sample to Proxy Attraction	Training Efficiency	Stable Class Proxy Centers	Weak Intra-class Compactness
	ArcFace/CosFace [13], [21]	Angular Margin Enforcement	Inter-class Separation	Feature/Weight Normalization	Limits Low-Conf. Handling
	Embedding Expansion [25]	Synthesize features w/ interpolation	Feature space enrichment	Enriching latent space	not target feature collapse
	Ours (ESA)	Eigenvector-guided Confidence Modulation	Unseen Class Generalization	Explicitly corrects train–test scale misalignment	Training Overhead (Eigenvector)

- The proposed method can be seamlessly integrated into existing CE-based DML frameworks, restoring train–test norm alignment and leading to substantial improvements in classification and retrieval performance.

II. RELATED WORK

A. PAIRWISE-BASED AND CLASSIFICATION-BASED DML

Pairwise-based DML methods rely on the relative distances between training samples by constructing positive and negative pairs, guided by the loss function. To improve the effectiveness of distance-based supervision, hard negative mining techniques have been employed—for instance, selecting the most challenging sample within each mini-batch [24]. Alternatively, semi-hard mining was introduced to mitigate the risk of convergence to poor local minima caused by overly difficult negatives [26]. Although more adaptive approaches have been proposed, where pairs are automatically selected based on their relative distances [11], these methods often involve additional hyperparameters and incur substantial computational overhead due to pairwise sampling.

To alleviate these issues, classification-based DML approaches such as Proxy-NCA [4] and Norm-Softmax [12] have been proposed. These methods aim to bring embedded features closer to the prototype (or proxy) of their corresponding class while pushing them away from other class proxies. The embedding space is constructed via a conventional classifier trained with the cross-entropy (CE) loss, leveraging softmax-based predictions. To further refine the embedding geometry, a temperature parameter was introduced into the loss function [20], [23], allowing the model to emphasize different difficulty levels—such as easy, boundary, and hard samples—thereby improving placement in the embedding space.

In a similar vein, our method also leverages the CE loss to learn discriminative embeddings. However, it departs from conventional approaches by explicitly considering the expected distributional behavior of unseen class samples. Our framework adaptively repositions features in the embedding space to reflect this behavior, thereby enhancing generalization and inter-class separability for unseen categories.

B. EMBEDDING AUGMENTATION

With the advent of embedding augmentation techniques, there has been significant progress in enhancing the representational capacity of embedding spaces. These methods augment training samples by interacting with existing feature representations or leveraging generative models to synthesize new data. For instance, the embedding expansion approach proposed in [25] draws inspiration from query expansion techniques, generating new features through combinations of existing ones to enrich the training process and improve overall network performance.

Embedding augmentation has also been applied to address class imbalance issues, particularly when datasets exhibit an unequal distribution of samples across classes [27]. In such scenarios, dominant classes with abundant samples tend to occupy disproportionately large regions of the embedding space, inflating their confidence scores. Conversely, underrepresented classes (*i.e.*, tail classes) often yield low confidence predictions due to insufficient coverage. To mitigate this, synthetic samples for tail classes are introduced, enabling the model to better calibrate confidence scores and effectively learn from long-tailed data distributions.

In a similar spirit, our method employs embedding augmentation to modulate the confidence scores of training samples. However, unlike previous approaches, our method is designed in a more principled and targeted manner for CE loss-based DML, with a specific focus on improving generalization to unseen classes.

Unlike the aforementioned methods that primarily focus on improving intra-class compactness or enhancing feature space stability, our framework directly addresses the unique challenge of generalization to unseen classes inherent in CE-based DML. While pairwise and classification methods attempt to handle hard samples through complex mining or margin strategies, they largely overlook the *train–test scale misalignment* that systematically affects ambiguous (low-confidence) samples at test time. Similarly, existing embedding augmentation techniques mainly target data imbalance or global feature enrichment, but they do not explicitly correct the *relative norm shifts with respect to the training distribution* that drive miscalibrated confidences and weaker margins on unseen classes. As summarized in Table 1, our proposed Embedding Space Augmentation (ESA) is

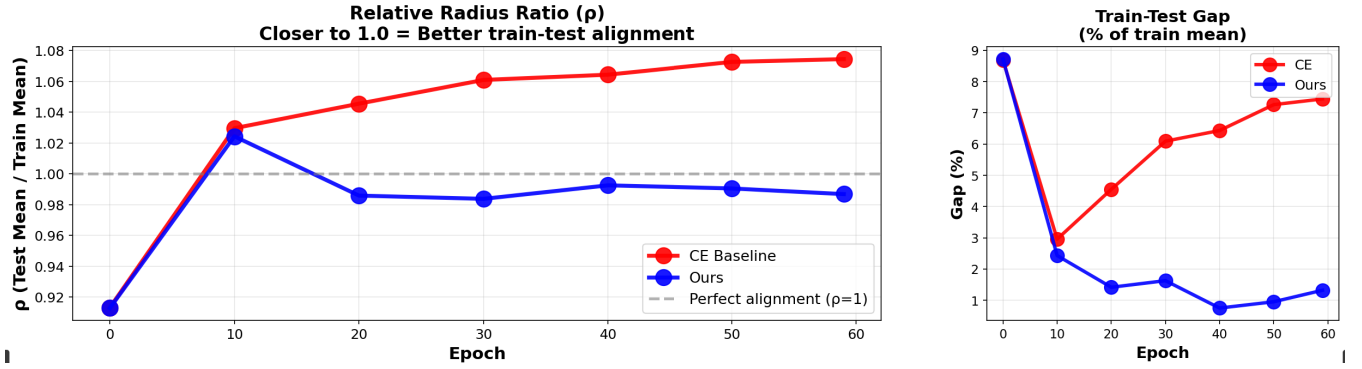


FIGURE 2. Train-test scale alignment diagnostics. (Left) *Relative Radius Ratio* $\rho = \mu_{\text{test}}/\mu_{\text{train}}$ (closer to 1 is better): the CE baseline drifts above 1 over training, while ours stays near $\rho \approx 1$. (Right) *Train-Test Gap* (as % of train mean): CE exhibits increasing gaps, whereas ours maintains small, flat gaps. Together, these indicate that our method preserves train-test scale alignment, mitigating softmax confidence flattening at test time.

distinct in its goal: it leverages the observed failure patterns of unseen classes to strategically modulate the confidence scores of seen-class hard samples, thereby *aligning* feature-norm scale between train and test and improving separability for novel categories without unseen supervision.

III. BEHAVIOR OF UNSEEN CLASS SAMPLES IN DML

A. SCALE-INVARIANT DIAGNOSTICS (DEFINITIONS)

We quantify train-test scale alignment with the following metrics:

$$\rho := \mu_{\text{test}}/\mu_{\text{train}}, \quad \Delta\mu := \frac{|\mu_{\text{train}} - \mu_{\text{test}}|}{\mu_{\text{train}}}. \quad (1)$$

Here μ denotes the mean radius of feature norm $\|f\|$.

B. WHAT THE CE BASELINE SHOWS (EVIDENCE)

The CE baseline exhibits a growing train-test scale gap (Figure 2): • *Relative Radius Ratio* ρ : CE drifts above 1 over training. • *Mean-radius gap* $\Delta\mu$: increases steadily, indicating compressed logits and flatter posteriors for unseen samples.

C. PROPOSED METHOD AT A GLANCE (WHAT CHANGES)

Our ESA aligns feature-norm scale by (i) lowering confidence for hard samples to amplify corrective gradients and (ii) guiding updates along the class principal direction: • $\rho \approx 1$ throughout training, • smaller, flatter $\Delta\mu$ trajectories. Having established the evidence for scale misalignment under CE and its correction under ESA, we now describe how ESA is implemented as a plug-in training component (Section IV).

IV. PROPOSED METHOD

In the proposed method, the embedding space is optimized not only for seen class samples—used during training—but also implicitly tailored to accommodate unseen class samples at test time. To achieve this, we analyze the behavior of conventional CE loss-based DML and derive a principled modification that intensifies the training focus on hard samples from seen classes. This enhanced supervision enables ambiguous unseen samples to be naturally drawn toward semantically similar seen classes during inference, thereby improving inter-class separability among unseen

categories. The proposed mechanism is realized through an embedding space augmentation strategy, which reshapes the feature space to support more robust generalization.

A. CONVENTIONAL CE LOSS IN DML

Let $\{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ denote a training dataset consisting of N samples, where x_i represents the i -th input image and $y_i \in \{1, \dots, C\}$ is its corresponding class label. For notational simplicity, we omit the subscript i in the following derivation. The standard cross-entropy (CE) loss function $\mathcal{L}_{CE}$ used for classification can be expressed as follows [15]:

$$\mathcal{L}_{CE}(x) = - \sum_{k=1}^C y_k \log \left(\frac{\exp(\mathbf{w}_k^\top f(x))}{\sum_{c=1}^C \exp(\mathbf{w}_c^\top f(x))} \right), \quad (2)$$

where $\mathbf{w}_k$ denotes the weight vector corresponding to the k -th class, and $f(x)$ is the embedding (feature representation) of input x , i.e., the output of the penultimate layer before the softmax layer. Minimizing the CE loss encourages the embedding function $f(\cdot)$ to map samples from the same class to similar regions in the embedding space—i.e., $f(x)$ and $f(x')$ are pulled closer together if x and x' share the same label. The classifier is trained using softmax regression applied to $f(x)$, yielding class probabilities of the form: $\left(\frac{\exp(\mathbf{w}_k \cdot f(x))}{\sum_{c=1}^C \exp(\mathbf{w}_c \cdot f(x))} \right)$ as shown in (2).

We can minimize $\mathcal{L}_{CE}(x)$ in (2) by maximizing $\mathbf{w}_k \cdot f(x)$, in particular, 1) by increasing $\|\mathbf{w}_k\|$, 2) by increasing $\|f(x)\|$, and 3) by reducing the angle between $\mathbf{w}_k$ and $f(x)$. Despite yielding accurate classification results, conventional CE loss-based methods often suffer from overconfidence issues [28], characterized by abnormally large norms of the embedding vectors $\|f(x)\|$ and class weights $\|\mathbf{w}\|$. These inflated magnitudes can hinder angular optimization, as observed in prior works [12], [17], [18], where the model tends to rely on increasing vector magnitudes rather than reducing the angle between $\mathbf{w}_k$ and $f(x)$. To address this, those studies introduce normalization of both $\|f(x)\|$ and $\|\mathbf{w}\|$ in the final layer before softmax, thereby constraining the softmax operation to focus on angular relationships.

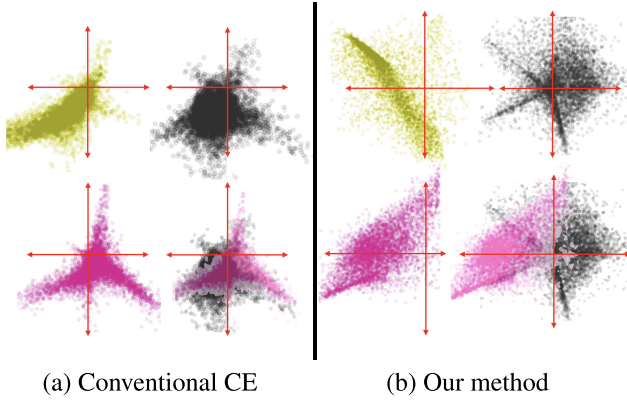


FIGURE 3. Separability of unseen class samples. In this experiment, we trained the embedding space using digits “0, 1, 2, 3, 4” from the MNIST dataset as seen classes, and evaluated generalization by testing on unseen classes “6, 7, 8”. The resulting embeddings of the unseen samples are visualized, with different colors representing different classes. (a) In the baseline scenario, unseen class samples are concentrated near the origin, leading to significant overlap between different classes (e.g., pink and black), thereby reducing class separability. (b) In contrast, our method places unseen class samples farther from the origin, resulting in improved separation among classes and enhanced discriminability in the embedding space.

Since softmax relies on the dot product, which is scale-sensitive, such normalization does not significantly affect classification accuracy and enables angular discrimination through a normalized CE loss.

However, as pointed out in [29], normalization may degrade generalization to unseen classes. In particular, fixing the feature/weight norms erases magnitude cues that affect the *logit scale*; To overcome this limitation, we propose a novel embedding augmentation strategy for CE loss-based DML, which avoids explicit normalization while still improving generalization by adaptively adjusting feature embeddings during training.

B. REVISIT ON THE CE LOSS IN DML

In this paper, we propose a method that provides explicit guidance for the directional movement of seen class samples within the embedding space. This guidance enables unseen class samples to be drawn toward regions occupied by seen class clusters, thereby increasing their confidence scores, as illustrated in Figure 1. Consequently, the unseen samples are repositioned in the embedding space in a manner that enhances inter-class separability, as shown in Figure 3. It is important to note that this guidance is grounded in a formal mathematical framework, as established in Theorem 2.

To achieve the aforementioned goal, we revisit the CE loss in (2) by redefining the softmax-based class probability, which is the likelihood that input x belongs to class k .

$$p(k|x) = \frac{\exp(\mathbf{w}_k \cdot f(x))}{\sum_{c=1}^C \exp(\mathbf{w}_c \cdot f(x))}, \quad (3)$$

where $f(x)$ and $\mathbf{w}$ denote the embedded feature vector and the class weight vector, respectively. Let $q(k|x)$ represent the ground-truth distribution modeled as a Dirac delta function, where $q(k|x) = 1$ if k corresponds to the ground-truth class,

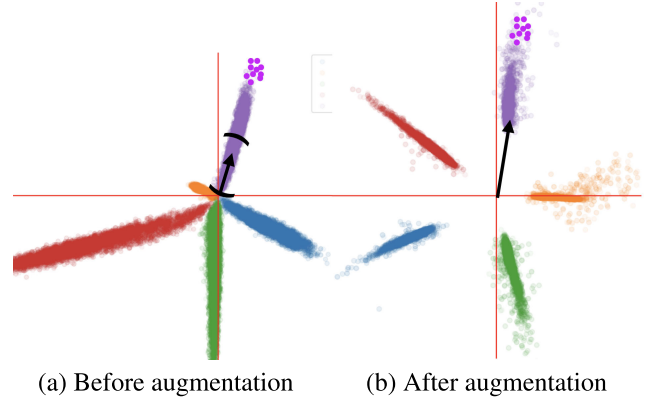


FIGURE 4. Embedding augmentation. (b) By augmenting virtual samples (e.g., violet circles) within the embedding space, our method effectively reduces the intra-class variance of seen class clusters and simultaneously shifts these clusters farther from the origin.

and $q(k|x) = 0$ otherwise. Then, the objective of DML is to minimize the CE loss $\mathcal{L}_{CE}(x)$ by reducing the divergence between the predicted class distribution $p(k|x)$ (obtained via softmax) and the target distribution $q(k|x)$:

$$\mathcal{L}_{CE}(x) = - \sum_{k=1}^C \log p(k|x)q(k|x), \quad (4)$$

where the derivatives of $\mathcal{L}_{CE}(x)$ with respect to the logit $z_k = \mathbf{w}_k \cdot f(x)$ and the embedded feature vector $f(x)$ are given as follows.

$$\frac{\partial \mathcal{L}_{CE}}{\partial z_k} = p(k|x) - q(k|x), \quad (5)$$

$$\frac{\partial \mathcal{L}_{CE}}{\partial f} = \sum_{k=1}^C \frac{\partial \mathcal{L}_{CE}}{\partial z_k} \frac{\partial z_k}{\partial f} = \sum_{k=1}^C \mathbf{w}_k (p(k|x) - q(k|x)). \quad (6)$$

In (6), the embedded feature $f(x)$ can be updated based on two factors, namely $\mathbf{w}_k$ and $p(k|x) - q(k|x)$.

- $\mathbf{w}_k$ makes the feature updating to be performed in the direction of the weight vector for the k -th class.
- $p(k|x) - q(k|x)$ is inversely proportional to the probability of x belonging to class k . Thus, $p(k|x) - q(k|x)$ has a large value, if x is less confident for the k -th class.

Building upon the findings above, we can induce a large gradient of $\frac{\partial \mathcal{L}_{CE}}{\partial f}$ by intentionally reducing the confidence score of $f(x)$ for the k -th class. This encourages a substantial update to $f(x)$ during training. As a result, the embedding $f(x)$ is guided to move in the direction of the corresponding class weight vector $\mathbf{w}_k$ and to larger radii (effective logit scale) in the feature space, as illustrated in Figure 4. By explicitly steering $f(x)$ toward $\mathbf{w}_k$, the proposed method relocates seen class samples to regions of higher confidence and lower intra-class variance. This reorganization of the embedding space leads to improved class separability at test time, as ambiguous unseen samples are naturally pushed toward semantically similar seen class regions, enhancing generalization performance.

Algorithm 1 Embedding Augmentation for the k -Th Class

Input $T = \{(x_i, y_i)\}$: set of training data, $\mathbf{w}_k^{(t)}$: weight vector at the iteration t of the softmax layer
Hyperparameters $\epsilon \in \mathbb{R}$: noise, ξ : weighting parameter
Output virtual samples f_{C+k}

```

while data  $(x_i, y_i) \in T$  do
  while label  $k \in C$  do
    if  $k = y_i$  then
       $f_k \leftarrow f(x_i)$ 
    end if
  end while
end while
 $e_k \leftarrow$  the first eigenvector for  $f_k$ 
// Obtain virtual samples on  $\text{span} \left\{ (1 - \xi) \mathbf{w}_k^{(t)} + \xi \cdot e_k \right\}$ 
 $f_{C+k} \leftarrow \mathcal{N} \left( (1 - \xi) \mathbf{w}_k^{(t)} + \xi \cdot e_k, \epsilon^2 \right)$ 

```

C. EMBEDDING SPACE AUGMENTATION

Our method intentionally reduces the confidence score of $f(x)$ for the k -th class by leveraging embedding space augmentation. Specifically, we introduce C virtual classes, thereby expanding the total number of classes to $2C$. For each original class $k \in 1, \dots, C$, we construct an augmented class $(C + k)$ whose weight vector is designed to be highly similar to that of class k . Now, consider a sample x that belongs to the k -th class. Due to the similarity between $\mathbf{w}_k$ and $\mathbf{w}_{C+k}$, the embedded feature $f(x)$ exhibits high similarity to both weight vectors. Consequently, in the softmax denominator of (3), both the k -th and the $(C + k)$ -th weight vectors contribute significantly, thereby increasing the denominator and reducing the predicted confidence score for class k . To determine which samples should undergo confidence reduction, we utilize statistics based on the norm of feature embeddings in the embedding space, allowing the method to adaptively target low-confidence regions.

The proposed augmentation strategy imposes implicit regularization on the norms of both the weight vectors and the feature embeddings in the embedding space, as follows:

Theorem 1: The weight vector w_k and the corresponding augmented vector w_{C+k} are optimized in an adversarial manner.¹ Therefore, our learning process restricts the size of $\|w_k\|$, $\|f(x_k)\|$, and $\|w_{C+k}\|$.

Proof: Please refer to Appendix. $\square$

The proposed augmentation method also plays a crucial role in preventing the increase of intra-class variance. However, it requires careful tuning via the hyperparameter β , as described in Algorithm 2. If we naively define the total loss $\mathcal{L}_{CE}$ as the sum of the CE losses for both the original and augmented samples, i.e., $\mathcal{L}_{CE} = \mathcal{L}_{CE}(x_k) + \mathcal{L}_{CE}(x_{C+k})$ with a fixed 1:1 ratio, the embedding function $f(x)$ may not be sufficiently optimized. To balance between effectively learning $f(x)$ and minimizing intra-class

Algorithm 2 Overall Training Procedure

Input f_k : feature, f_{C+k} : virtual features, w_k : weight vector, w_{C+k} : virtual weight vector
Hyperparameters κ : threshold, β : balancing parameter
Output $\mathcal{L}_{ours}$: CE loss value

```

mean = empirical average of  $\|f_k\|$ 
std = empirical standard deviation of  $\|f_k\|$ 
while  $f \in f_k$  do
  if  $\|f\| \leq \text{mean} + \kappa * \text{std}$  then
     $F_L \leftarrow f$ 
  else
     $F_H \leftarrow f$ 
  end if
end while
 $F_A \leftarrow f_{C+k}$ 

 $\mathcal{L}_H \leftarrow \mathcal{L}_{CE}$  in (4) of  $F_H$  and  $w_k$ 
// Adjust the confidence using virtual samples
 $\mathcal{L}_L \leftarrow \mathcal{L}_{CE}$  in (4) of  $F_L$ ,  $w_k$ , and  $w_{C+k}$ 
 $\mathcal{L}_A \leftarrow \mathcal{L}_{CE}$  in (4) of  $F_A$  and  $w_{C+k}$ 
 $\mathcal{L}_{ours} \leftarrow \mathcal{L}_H + \mathcal{L}_L + \beta \cdot \mathcal{L}_A$ 

```

variance, we introduce a trade-off mechanism. In practice, we down-weight the contribution of the augmented data by a factor of 10 using the hyperparameter β , as implemented in Algorithm 2.

To guide the direction of the weight vectors for the C augmented classes, we propose a mathematically grounded approach, as described in Theorem 2. Specifically, the direction of the augmented weight vector $\mathbf{w}_k^{(t+1)}$ at iteration $t + 1$ is determined by the position of the feature embedding $f(x)$ at iteration t . The principal eigenvector of the covariance matrix of $f(x)$ is used to define the direction of $\mathbf{w}_k^{(t+1)}$. To ensure that the augmented weight vectors align with the directions of existing class weights, our method constructs augmented samples that lie along the dominant eigenvector of each seen class cluster. For every seen class, we compute its principal eigenvector and generate augmented samples along this direction in the embedding space. This augmentation allows features located near the origin to be displaced toward the direction of the corresponding weight vector, as illustrated in Figure 4. Consequently, the variance along the axis defined by the weight vector is reduced, which in turn minimizes intra-class variance and enhances the compactness of class-specific clusters.

D. THEORETICAL GROUND

We establish the theoretical foundation of the proposed embedding space augmentation method in Theorem 2.

Theorem 2: $\mathbf{w}_k^{(t+1)}$ is estimated by $(1 - \xi) \mathbf{w}_k^{(t)} + \xi \mathbf{e}_k^{(t)}$, where $\mathbf{e}_k^{(t)}$ is the the first eigenvector of the embedded feature $f(x)$ for the k -th class and ξ is a hyperparameter.

¹Here, we ignore the balancing parameter β . i.e., $\beta = 1$.

TABLE 2. Comparison on the CUB200, CARS196, SOP, and In-shop datasets in terms of Recall@k. The superscript associated with each model indicates the embedding size. Model backbones are denoted as BN for BN-Inception with batch normalization [30] and R50 for ResNet-50 [31], respectively. The best performance for each metric is highlighted in bold.

R@k	Net	CUB200				Cars196				SOP				In-shop			
		1	2	4	8	1	2	4	8	1	10	100	1000	1	10	20	30
ProxyNCA ⁶⁴	BN	65.1	71.6	85.0	90.7	73.2	82.4	86.4	88.7	73.7	-	-	-	-	-	-	-
Margin ¹²⁸	R50	63.9	75.3	84.4	94.8	86.9	92.7	95.6	98.7	72.7	86.2	93.8	98.0	-	-	-	-
MS ⁵¹²	BN	65.7	77.0	86.3	91.2	84.1	90.4	94.0	96.5	78.2	90.5	96.0	98.7	89.7	97.9	98.5	98.8
SoftTriple ⁵¹²	BN	65.4	76.4	84.5	90.4	84.5	90.7	94.5	96.9	78.3	90.3	95.9	-	-	-	-	-
ProxyAnchor ⁵¹²	BN	68.4	79.2	86.8	91.6	86.1	91.7	95.0	97.3	80.3	91.4	96.4	98.7	91.5	98.1	98.8	-
NormSoftmax ⁵¹²	R50	61.3	73.9	83.5	90.0	84.2	90.4	94.4	96.9	78.2	90.6	96.2	-	88.6	97.5	98.4	98.8
NormSoftmax ²⁰⁴⁸	R50	65.3	76.7	85.4	91.8	89.3	94.1	96.4	98.0	79.5	91.5	96.7	-	89.4	97.8	98.7	99.0
Ours + Softmax ²⁰⁴⁸	R50	65.1	77.9	85.7	91.7	88.4	92.6	95.0	97.5	79.6	91.5	97.4	98.8	91.7	98.1	98.7	98.6
Ours + Arcface ²⁰⁴⁸	R50	70.7	80.2	87.5	92.2	90.0	94.7	96.5	98.7	81.0	91.6	97.5	98.8	91.8	98.2	99.0	99.5
Ours + Cosface ²⁰⁴⁸	R50	68.9	78.7	86.8	91.1	89.0	94.4	96.3	97.4	79.4	91.2	97.4	98.7	90.8	97.8	98.8	99.0
Ours + Sphereface ²⁰⁴⁸	R50	68.8	78.0	85.8	91.1	89.2	93.0	96.0	97.3	79.0	91.1	96.5	98.0	90.6	97.5	98.8	99.1

Proof: Let $e \in \mathbb{R}^d$ be the estimation of $\mathbf{w}_k^{(t+1)}$. Then, we can find the best e as

$$\hat{e} = \arg \max_e \sum_{x \in \mathcal{X}} \frac{e}{\|e\|} \cdot \frac{f(x)}{\|f(x)\|}, \quad (7)$$

where the exact solution $\hat{e}$ cannot be obtained in closed form. As an alternative, we approximate $\hat{e}$ by finding a vector e that captures the maximum variance of the sample features. This approximation is equivalent, in terms of the second moment, to identifying the vector e that maximizes the inner product defined in (7). Specifically, we seek the vector e that describes the direction of maximum variance in the feature set F , formulated as follows:

$$\hat{e} = \arg \max_e \underbrace{\frac{1}{n} \sum_{i=1}^n (F_i \cdot e - \mu)^2}_{\text{variance of } F}, \text{ s.t. } \|e\|_2 = 1, \quad (8)$$

where $F \in \mathbb{R}^{n \times d}$ denotes the matrix composed of n feature vectors $f(x) \in \mathbb{R}^d$ as rows, $F_i \in \mathbb{R}^{1 \times d}$ represents the i -th row of F , and μ denotes the mean vector of F . To solve the optimization problem in (8), we apply the method of Lagrange multipliers, framing it as a constrained convex optimization problem. Note that we impose the constraint $\|e\|_2 = 1$ in (8), although this constraint can be relaxed or weighted via a hyperparameter ξ if needed. The optimal solution to (8) is obtained by setting the derivative of the objective function with respect to e_a to zero and solving for e accordingly:

$$\sum_{j=1}^d e_j \underbrace{\left(\frac{1}{n} \sum_{i=1}^n F_{ia} F_{ij} \right)}_{\text{covariance matrix of } F} - \lambda e_a = 0, \quad (9)$$

where e_a denotes the a -th component of the vector e , and F_{ij} represents the entry in the i -th row and j -th column of the matrix F . Under this formulation, the optimal direction e satisfies the eigenvalue equation $\Sigma e = \lambda e$ for all a , where Σ is the covariance matrix of F . The eigenvector corresponding to the largest eigenvalue—i.e., the first principal component—captures the direction of maximum variance. Given the estimated e , we update the

weight vector $\mathbf{w}_k^{(t)}$ using a convex combination with a mixing coefficient ξ , as follows: $\mathbf{w}_k^{(t+1)} = (1 - \xi)\mathbf{w}_k^{(t)} + \xi e_k^{(t)}$, which yields the updated weight vector $\mathbf{w}_k^{(t+1)}$ for the next iteration. $\square$

Algorithms 1 and 2 outline the procedures for the proposed embedding space augmentation and the overall training process, respectively.

V. EXPERIMENTS

To evaluate our DML approach, we conducted experiments on standard benchmark datasets: Caltech-UCSD Birds (CUB) [32], CARS196 (CAR) [33], Stanford Online Products (SOP) [2], and In-Shop Clothes Retrieval (In-Shop) [34]. For a fair comparison, we followed the experimental protocols proposed in [2], which are widely adopted in the metric learning community. experimental settings in [2], which are widely used for metric learning.

We implemented the proposed method using ResNet50 [31] as the backbone architecture, pre-trained on the ImageNet dataset [35]. Feature embeddings $f(x)$ were extracted from the final convolutional layer, followed by global average pooling. A softmax-based classifier was then applied to the resulting features without additional normalization. We tested several softmax variants, including standard softmax, CosFace [21], ArcFace [13], and SphereFace [36]. Notably, our method is compatible with any softmax-based formulation and does not require feature or weight normalization. When combined with angular-margin-based losses (e.g., CosFace, ArcFace, and SphereFace), our method consistently yielded superior performance. For data augmentation, we applied random cropping and horizontal flipping during training, and center cropping during testing. To ensure balanced training, we maintained an equal number of samples per class in each batch (batch size = 75). Specifically, we used 25 samples per class for the CUB and CAR datasets, 64 for SOP, and 4 for In-Shop, following standard practices.

Training was performed using stochastic gradient descent (SGD) with a momentum of 0.9, weight decay of $1e - 4$, and a step-wise learning rate schedule with a decay factor (gamma) of 0.1 at epoch 15. The initial learning rate was set to $1e - 3$. We fixed the embedding dimension to 2048 across all experiments. The hyperparameters κ , β in Algorithm 2, and

TABLE 3. Performance evaluation of the proposed method with different baseline networks in terms of Recall@1. We adopted Softmax, ArcFace [13], CosFace [21], and SphereFace [36] as baseline methods. Our approach was integrated into each baseline network and consistently yielded substantial improvements in accuracy. The performance gain over the baseline is indicated in parentheses (·).

method	CUB	CARS	SOP	Inshop
Softmax	62.4	84.7	78.5	90.5
Ours + Soft	65.1 (+2.7)	88.4 (+3.7)	79.6 (+1.1)	91.7 (+1.2)
Arcface	66.4	86.6	80.1	91.0
Ours + Arc	70.7 (+4.3)	90.0 (+3.4)	81.0 (+0.9)	91.8 (+0.8)
Cosface	64.2	85.2	78.8	90.5
Ours + Cos	68.9 (+4.7)	89.0 (+4.8)	79.4 (+0.6)	90.8 (+0.3)
Sphereface	64.1	84.9	78.6	90.4
Ours + Sphere	68.8 (+4.7)	89.2 (+4.3)	79.0 (+0.4)	90.6 (+0.2)

TABLE 4. Performance evaluation of the proposed method under varying values of κ in terms of Recall@1. The parameter κ , used as a threshold for identifying low-confidence samples in Algorithm 2, was evaluated on the CUB and CARS196 datasets.

κ	-1	-0.5	0	0.5	1
CUB	60.8	62.6	65.1	63.0	61.2
CAR	86.6	87.2	88.4	87.2	86.3

ξ in Algorithm 1 were all set to 0.1. All experiments were conducted on a single NVIDIA GeForce RTX 3090 GPU.

A. COMPARISON WITH STATE-OF-THE-ART METHOD

In this experiment, k nearest images were retrieved from the test dataset using cosine similarity. The proposed method was evaluated against several state-of-the-art deep metric learning (DML) approaches, including ProxyNCA [4], Margin Loss [26], Multi-Similarity [11], SoftTriple [14], ProxyAnchor [37], NormSoftmax, and the classification-based method proposed in [12], using Recall@ k as the evaluation metric.

As shown in Table 2, our method consistently outperforms existing approaches across multiple benchmark datasets. Notably, it achieves superior retrieval performance compared to both pairwise-based and classification-based DML methods. Although our model employs a larger embedding dimension than some competing methods, it delivers significant accuracy gains—exceeding other methods by more than 3%—without compromising computational efficiency. While the use of singular value decomposition (SVD) introduces additional overhead during training, the inference time of our method remains comparable to that of existing approaches. This demonstrates that our method achieves a favorable trade-off between accuracy and efficiency. Importantly, these results highlight the effectiveness of our approach in image retrieval tasks, where it significantly outperforms prior state-of-the-art methods without sacrificing runtime performance.

B. PROXY-ANCHOR + OURS: 3-SEED EVALUATION

To validate plug-in generality on a strong proxy-based objective, we insert **Ours** into **Proxy-Anchor** (PA) [37] without changing the backbone, optimizer, schedule, or data pipeline. We run 3 random seeds and report mean $\pm$ std of Recall@1 on the standard test splits.

We validate plug-in generality by inserting our method into **Proxy-Anchor** (PA) [37] under an identical training

TABLE 5. Proxy-anchor vs. PA+ours on CUB200 and CARS196 (ResNet-50, $d=512$, 3 seeds).

Dataset	Method	R@1	R@2	R@4	R@8
CUB200	Proxy-Anchor	69.73 $\pm$ 0.21	79.47 $\pm$ 0.32	86.70 $\pm$ 0.10	91.30 $\pm$ 0.10
	PA + Ours	70.23 $\pm$ 0.06	80.00 $\pm$ 0.10	86.87 $\pm$ 0.06	91.37 $\pm$ 0.06
	Δ (abs.)	+0.50	+0.53	+0.17	+0.07
CARS196	Proxy-Anchor	87.43 $\pm$ 0.31	92.43 $\pm$ 0.25	95.30 $\pm$ 0.20	97.27 $\pm$ 0.06
	PA + Ours	88.17 $\pm$ 0.21	92.97 $\pm$ 0.15	95.70 $\pm$ 0.10	97.33 $\pm$ 0.06
	Δ (abs.)	+0.73	+0.53	+0.40	+0.07

recipe. Unless noted otherwise, we use ResNet-50 with embedding size $d=512$, batch size 120, learning rate $1e-4$, warm-up 5 epochs, BN-freeze enabled, and the official splits. For CUB200 we set `-lr-decay-step` 5, and for CARS196 `-lr-decay-step` 10. All other hyperparameters (optimizer, augmentations, schedule) are kept the same as PA. Our variant *only* (i) disables feature normalization and (ii) enables our components; no inference-time parameters are added.

Table 5 reports **Recall@K** ($K=1,2,4,8$) as mean $\pm$ std over 3 seeds. Across both datasets, **PA+Ours** outperforms PA consistently: on **CUB200**, R@1/2/4/8 improve by **+0.50/+0.53/+0.17/+0.07** points, and on **CARS196**, by **+0.73/+0.53/+0.40/+0.07**. The gains are stable (small stds) and appear across all K .

While the absolute improvements over a strong proxy-based baseline are *smaller* than the gains observed when augmenting a plain softmax (CE) objective, the trend is **uniform and reproducible** across datasets, K values, and random seeds. This indicates that our method acts as a *drop-in, training-only* component that consistently strengthens proxy objectives without changing inference, complementing the scale-alignment evidence shown earlier.

Across all four benchmarks, **PA+Ours** consistently outperforms PA with statistically significant gains. Improvements are robust across seeds and datasets, confirming that Ours is an *effective drop-in training component* for proxy objectives.

C. ABLATION STUDY ON THE PROPOSED METHOD

Accurately learning hard samples with low confidence is critical for softmax-based methods, particularly those that do not apply weight or feature normalization. Under this setting, the proposed method can be seamlessly integrated into softmax-based approaches, yielding significant improvements in prediction accuracy. As shown in Table 3, our method can be easily combined with a variety of baseline networks and consistently enhances their performance across multiple benchmark datasets. The substantial accuracy gains demonstrate the effectiveness of our embedding space augmentation algorithm in addressing low-confidence samples and generalizing to unseen classes during test time. In particular, our method achieves notably higher accuracy on the CUB and CARS196 datasets, where the number of samples per class is significantly larger than in SOP and In-Shop datasets (*e.g.*, SOP and In-Shop typically contain only four samples per class on average). This abundance of class-specific samples allows our method to estimate a more

TABLE 6. Training/inference cost comparison on CUB200. Overhead is relative to CE.

Metric	CE	Ours	Overhead
Parameters	102,400	102,400	0%
Forward FLOPs	12.3M	12.7M	2.6%
Backward FLOPs	24.7M	25.3M	2.6%
Inference FLOPs	102K	103K	0.5%

stable eigenvector representation, thereby inducing greater improvements in retrieval performance.

Table 4 illustrates how the performance varies with different values of the hyperparameter κ in Algorithm 2. Notably, when the average feature norm is used as the threshold, our method achieves the best performance. Overall, the proposed method exhibits low sensitivity to the choice of κ indicating its robustness with respect to this hyperparameter.

D. COMPUTATIONAL COMPLEXITY

In Table 6, both methods use the same $C \times D$ classifier/embedding dimension, so **parameter counts are identical**. Ours introduce a small FLOPs increase during training (forward/backward +2.6%) due to the *augmentation features/ extra denominator term for gated low-norm samples*, while **inference cost is essentially unchanged** (+0.5% FLOPs) since Ours add no test-time parameters or branches.

VI. CONCLUSION

We revisited CE-based deep metric learning through the lens of train–test scale misalignment, where test-time feature norms shift relative to training and destabilize softmax confidence—especially for ambiguous samples. We introduced a training-only, plug-in strategy that gates low-norm seen samples, briefly reduces their confidence to amplify corrective gradients, and guides updates along each class’s principal direction. This restores the effective logit scale and reduces intra-class variance without any unseen supervision or test-time changes. Across standard image-retrieval benchmarks, the method yields consistent gains in Recall@K and tighter scale alignment, indicating improved generalization to unseen classes.

This work targets standard fine-grained DML where class balance and inter-class overlap are controlled; heavy long-tailed or strongly overlapping settings are out of scope. While full normalization (purely angular objectives) can reduce scale drift, it may discard useful magnitude cues. Our approach preserves and *stabilizes* norms via low-norm gating and class-conditional guidance. Extending our method to (i) severe imbalance and (ii) overlap/multi-modal classes is promising and left for future work.

APPENDIX

PROOF OF THEOREM 1

Theorem 1: The weight vector w_k and the corresponding augmented vector w_{C+k} are optimized in an adversarial manner.² Therefore, our learning process restricts the size of $\|w_k\|$, $\|f(x_k)\|$, and $\|w_{C+k}\|$.

²Here, we ignore the balancing parameter β . i.e., $\beta = 1$.

Proof: In Algorithm 1, our method generates approximately the same number of augmented samples as real samples exhibiting low confidence values. From an optimization perspective, and without loss of generality, we focus on the loss formulation for samples belonging to the k -th class. Let $\mathcal{L}$ be $\mathcal{L}_{CE}(x_k) + \mathcal{L}_{CE}(x_{C+k})$. Then, $\mathcal{L}$ can be derived as

$$\begin{aligned}\mathcal{L} &= -\log P(k|x_k) - \log P(k|x_{C+k}) \\ &= -\log \frac{\exp(w_k \cdot f(x_k))}{\sum_{c=1}^{2C} \exp(w_c \cdot f(x_k))} \\ &\quad - \log \frac{\exp(w_{C+k} \cdot f(x_{C+k}))}{\sum_{c=1}^{2C} \exp(w_c \cdot f(x_{C+k}))}.\end{aligned}$$

In Algorithm 1, we assume that w_k and $f(x_k)$ are approximately equivalent to w_{C+k} and $f(x_{C+k})$, respectively, where the latter correspond to the augmented counterparts. By removing the constant or non-informative terms, the loss $\mathcal{L}$ can be simplified and expressed as:

$$\begin{aligned}\mathcal{L} &= -\log \frac{\exp(w_k \cdot f(x_k))}{\exp(w_k \cdot f(x_k)) + \exp(w_{C+k} \cdot f(x_k))} \\ &\quad - \log \frac{\exp(w_{C+k} \cdot f(x_{C+k}))}{\exp(w_k \cdot f(x_{C+k})) + \exp(w_{C+k} \cdot f(x_{C+k}))}.\end{aligned}$$

From a minimization perspective, the expression above is equivalent to the following formulation:

$$\begin{aligned}\mathcal{L} &= \frac{\exp(w_k \cdot f(x_k)) + \exp(w_{C+k} \cdot f(x_k))}{\exp(w_k \cdot f(x_k))} \\ &\quad + \frac{\exp(w_k \cdot f(x_{C+k})) + \exp(w_{C+k} \cdot f(x_{C+k}))}{\exp(w_{C+k} \cdot f(x_{C+k}))}.\end{aligned}$$

By removing constant terms and the exponential function (due to its convexity), the objective can be approximated by the dot product of two norms and a cosine similarity term. Since the feature norm $\|f(x)\|$ is excluded in this formulation, it cannot be trivially increased to reduce the loss. Although the cosine similarity term $\cos \theta$ plays a role in learning meaningful directions for $f(x)$, its influence on the overall scale of the objective is limited and can therefore be treated as negligible in this context. Accordingly, the optimization problem can be reformulated as:

$$\min_{w_k, w_{C+k}} \frac{\|w_{C+k}\|}{\|w_k\|} + \frac{\|w_k\|}{\|w_{C+k}\|},$$

which indicates that w_k and w_{C+k} are optimized in an adversarial manner with respect to each other. $\square$

REFERENCES

- [1] S. Ashraf and Z. Aslam, “Data leakage analysis in wireless networks using super-vised and unsupervised testing,” *Innovación y Softw.*, vol. 4, no. 2, pp. 52–62, Sep. 2023.
- [2] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, “Deep metric learning via lifted structured feature embedding,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4004–4012.
- [3] F. Schroff, D. Kalenichenko, J. Philbin, F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” in *Proc. CVPR*, Jun. 2015, pp. 815–823.
- [4] Y. Movshovitz-Attias, A. Toshev, T. Leung, S. Ioffe, and S. Singh, “No fuss distance metric learning using proxies,” in *Proc. ICCV*, 2017, pp. 360–368.

- [5] E. P. Xing, M. I. Jordan, S. Russell, and A. Y. Ng, "Distance metric learning with application to clustering with side-information," in *Proc. NIPS*, vol. 15, 2002, pp. 521–528.
- [6] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, 2016, pp. 31–35.
- [7] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," 2017, *arXiv:1703.05175*.
- [8] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.
- [9] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, 2006, pp. 1735–1742.
- [10] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, 2005, pp. 539–546.
- [11] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott, "Multi-similarity loss with general pair weighting for deep metric learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5017–5025.
- [12] A. Zhai and H.-Y. Wu, "Classification is a strong baseline for deep metric learning," 2018, *arXiv:1811.12649*.
- [13] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 5962–5979, Oct. 2022.
- [14] Q. Qian, L. Shang, B. Sun, J. Hu, T. Tacoma, H. Li, and R. Jin, "SoftTriple loss: Deep metric learning without triplet sampling," in *Proc. ICCV*, 2019, pp. 6449–6457.
- [15] M. Boudiaf, J. Rony, I. M. Ziko, É. Granger, M. Pedersoli, P. Piantanida, and I. B. Ayed, "A unifying mutual information view of metric learning: Cross-entropy vs. pairwise losses," in *Proc. ECCV*, 2020, pp. 548–564.
- [16] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proc. ECCV*, 2016, pp. 499–515.
- [17] R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax loss for discriminative face verification," 2017, *arXiv:1703.09507*.
- [18] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, "Normface: L2 hypersphere embedding for face verification," in *Proc. ACM Multimedia*, 2017, pp. 1041–1049.
- [19] E. W. Teh, T. DeVries, and G. W. Taylor, "ProxyNCA++: Revisiting and revitalizing proxy neighborhood component analysis," in *Proc. ECCV*, 2020, pp. 448–464.
- [20] X. Zhang, F. Xinnan Yu, S. Karaman, W. Zhang, and S.-F. Chang, "Heated-up softmax embedding," 2018, *arXiv:1809.04157*.
- [21] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "CosFace: Large margin cosine loss for deep face recognition," in *Proc. CVPR*, Jun. 2018, pp. 5265–5274.
- [22] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Process. Lett.*, vol. 25, no. 7, pp. 926–930, Jul. 2018.
- [23] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.
- [24] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*.
- [25] B. Ko and G. Gu, "Embedding expansion: Augmentation in embedding space for deep metric learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7253–7262.
- [26] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krähenbühl, "Sampling matters in deep embedding learning," in *Proc. ICCV*, 2017, pp. 2840–2848.
- [27] J. Liu, Y. Sun, C. Han, Z. Dou, and W. Li, "Deep representation learning on long-tailed data: A learnable embedding augmentation perspective," in *Proc. CVPR*, Jun. 2020, pp. 2970–2979.
- [28] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. ICML*, 2017, pp. 1321–1330.
- [29] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "MagFace: A universal representation for face recognition and quality assessment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14220–14229.
- [30] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.
- [31] M. Shafiq and Z. Gu, "Deep residual learning for image recognition: A survey," *Appl. Sci.*, vol. 12, no. 18, p. 8972, Sep. 2022.
- [32] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200–2011 dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2010-001, 2010.
- [33] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. ICCV Workshops*, 2013, pp. 554–561.
- [34] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "DeepFashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1096–1104.
- [35] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. CVPR*, Jun. 2009, pp. 248–255.
- [36] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6738–6746.
- [37] S. Kim, D. Kim, M. Cho, and S. Kwak, "Proxy anchor loss for deep metric learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3235–3244.



JINHEE PARK received the B.S. and M.S. degrees in computer science from Chung-Ang University (CAU), Seoul, South Korea, in 2019 and 2021, respectively, where she is currently pursuing the Ph.D. degree in computer science. Her research interests include metric learning and confidence calibration.



HEE BIN YOO received the B.S. degree in software engineering from Chung-Ang University (CAU), Seoul, South Korea, in 2021. He is currently pursuing the Ph.D. degree with the Interdisciplinary Program in Artificial Intelligence, Seoul National University (SNU), Seoul. His research interests include image-domain representation learning, feature learning theory, neural collapse, and local elasticity.



BYOUNG-TAK ZHANG received the B.S. and M.S. degrees in computer science and engineering from Seoul National University (SNU), in 1986 and 1988, respectively, and the Ph.D. degree in computer science from the University of Bonn, Germany, in 1992. He is currently a Professor with the Department of Computer Science and Engineering, SNU. He is also affiliated with the Brain Science and Cognitive Science Programs and the Director of the Institute for Cognitive Science. His research interests include biointelligence models of learning, evolution, and development and their application to artificial intelligence and cognitive science.



JUNSEOK KWON received the B.Sc. degree, the M.Sc. degree in object tracking (supervised by Prof. Kyoung Mu Lee), and the Ph.D. degree in electrical engineering and computer science from Seoul National University, South Korea, in 2006, 2008, and 2013, respectively. He is currently a Professor with the School of Computer Science and Engineering, Chung-Ang University, Seoul, South Korea. He is working in the field of object tracking to capture the dynamics of cities. His research interests include visual tracking, visual surveillance, and Monte Carlo sampling method.

...