Direct-Sum Approach to Integrate Losses Via Classifier Subspace

Takumi Kobayashi[1,2]
takumi.kobayashi@aist.go.jp

[1] National Institute of Advanced Industrial Science and Technology
Tsukuba, Japan

[2] University of Tsukuba
Tsukuba, Japan

Abstract

Deep models are successfully applied to various visual recognition tasks through end-to-end learning. A loss function is fundamental for the learning and various losses could be combined via arithmetic summation to improve performance. The simple summation, however, can bring about interference between the losses in back-propagation, deteriorating their synergy in the learning. In this paper, we propose a new approach to effectively integrate losses by mitigating the interference; we focus on classification and metric-based losses which are widely employed in discriminative supervised learning. The method leverages a *classifier subspace* to separate whole feature space into disjoint subspaces to which the two types of losses are respectively applied. Thereby, the losses are integrated in a *direct-sum* manner beyond a simple *arithmetic* summation to collaboratively work on learning feature representation without interference. In the experiments on few-shot image classification tasks which demand generalizable feature representation to unseen-class samples, the proposed method favorably improves performance by effectively combining the two types of losses.

1 Introduction

As deep models have been advanced in the last decade, end-to-end learning is arguably the most successful approach to build an image recognition model of high performance [15]. It optimizes lots of parameters of the deep model for a target task by utilizing training samples which empirically describe the characteristics of the task with annotation (labels) in a supervised setting. While optimizers [8] play a vital role in the training, a *loss* is also fundamental for the end-to-end learning to trigger back-propagation.

There are diverse types of loss functions useful for the back-propagation, i.e., being differentiable. In supervised classification, we can directly leverage a class label to build a loss function, such as a softmax cross-entropy loss which is widely applied to multi-class classification tasks with some theoretical analyses [1, 5], inspiring large-margin variants [10, 22]. Binary cross-entropy (BCE) intrinsically formulated for binary labels is also capable of measuring a multi-class classification loss, especially in a multi-label setting through decomposition into multiple binary classifications [2, 27]. While these losses are constructed

to evaluate discrepancy between a sample (feature) and its class label, it is possible to explore relationships among samples in a way of metric learning [19] by indirectly utilizing labels to annotate the relationships as intra-class and inter-class ones. Discriminative feature representation is demanded to render high similarities to intra-class samples while distancing intra-class samples to exhibit low similarity. The sample relationships are effectively exploited by contrastive [16], triplet [30] and quadruplet [7] losses. Neighborhood component analysis (NCA) [14] describes those relationships by means of probabilistic models and center loss [32] incorporates them into a form of class prototype (mean). Those losses are frequently employed in the literature of few-shot learning [20, 25] that poses a challenge to learn generalizable feature representation so as to discriminate novel classes with access to only a few training samples [12]. The above-mentioned softmax cross-entropy loss is also favorably applied to the few-shot learning [6, 11, 26].

For further improving feature representation, this work explores to *combine* those two types of losses, classification-based and metric-based losses mentioned above. A naïve approach is to simply sum up those losses by means of algebraic addition. It, however, might be less effective since the two types of losses work on a feature space in different updating ways and thus could be interfered to each other, though sharing the same goal [4]. Therefore, we propose a new approach to integrate those losses through feature space separation by means of *subspaces*, which thereby results in a *direct-sum* formulation of losses. The proposed method leverages a classifier subspace to enhance the synergy between the two losses while reducing the interference.

2 Method

Suppose we train a backbone model $\phi_{\boldsymbol{\theta}}$ in a supervised manner using n training samples $\{\mathcal{I}_i, y_i\}_{i=1}^n$, pairs of an image $\mathcal{I}$ and its ground-truth class label $y \in \{1, \cdots, C\}$; for simplicity, we regard n as a size of mini-batch. The model encodes an image $\mathcal{I}$ to a d-dimensional feature vector by $\boldsymbol{x} = \phi_{\boldsymbol{\theta}}(\mathcal{I}) \in \mathbb{R}^d$, which is fed into the following two types of losses based on class labels and sample metrics to learn parameters of the backbone model $\boldsymbol{\theta}$.

2.1 Classification loss ℓ_{cls}

The classification is usually addressed by using a linear classifier $\hat{\boldsymbol{y}} = \boldsymbol{W}^\top \boldsymbol{x} + \boldsymbol{b}$ with trainable parameters of classifier weights $\boldsymbol{W} \in \mathbb{R}^{d \times C}$ and biases $\boldsymbol{b} \in \mathbb{R}^C$. In a supervised scenario, a softmax cross-entropy (SCE) loss is commonly applied to train the classifier $\{\boldsymbol{W}, \boldsymbol{b}\}$ as well as the feature representation $\boldsymbol{x}$ via $\phi_{\boldsymbol{\theta}}$ by exploiting discriminative characteristics among C classes as

$$\text{SCE:}\ \ \ell_{cls}(\boldsymbol{x}, y; \boldsymbol{W}, \boldsymbol{b}) = -\log \frac{\exp(\boldsymbol{w}_y^\top \boldsymbol{x} + b_y)}{\sum_{c=1}^C \exp(\boldsymbol{w}_c^\top \boldsymbol{x} + b_c)}, \tag{1}$$

where $\boldsymbol{w}_c$ is the c-th column vector of a classifier weight matrix $\boldsymbol{W} \in \mathbb{R}^{d \times C}$ and b_c is the c-th component of bases $\boldsymbol{b} \in \mathbb{R}^C$. The other popular classification loss is a binary cross-entropy (BCE) which decomposes multi-class classification into C binary classification tasks by

$$\text{BCE:}\ \ \ell_{cls}(\boldsymbol{x}, y; \boldsymbol{W}, \boldsymbol{b}) = -\log \frac{\exp(\boldsymbol{w}_y^\top \boldsymbol{x} + b_y)}{1 + \exp(\boldsymbol{w}_y^\top \boldsymbol{x} + b_y)} - \sum_{c \neq y} \log \frac{1}{1 + \exp(\boldsymbol{w}_c^\top \boldsymbol{x} + b_c)}. \tag{2}$$

The BCE aggregates class-wise losses, thus being well applied to multi-label classification [27], while the SCE pays attention to comparison among class categories. These classification losses are contributive to enhancing class-discriminative feature representation $\boldsymbol{x}$ by directly utilizing class label y.

2.2 Metric-based loss ℓ_{metric}

We demand feature representation $\boldsymbol{x}$ to exhibit good metric that intra-class distances are small while inter-class samples are separated. A metric-based loss [19] works on (pair-wise) sample relationships without a classifier, while the classification losses (Sec. 2.1) rather directly connect class labels with samples through the classifier. In the few-shot learning, the following two metric losses are frequently applied to learn the feature representation $\boldsymbol{x}$ generalizable for unseen classes.

Following the approach of center loss [32], each class can be represented by a prototype, the mean of intra-class samples. A prototypical loss [25] is formulated by evaluating distances to respective class prototypes in a softmax fashion of

$$\text{Proto:}\quad \ell_{metric}(\{\boldsymbol{x}_i,y_i\}_{i=1}^n) = -\mathop{\mathrm{E}}_i \log \frac{\exp(-\|\boldsymbol{x}_i-\boldsymbol{\mu}_{y_i\backslash i}\|_2^2)}{\sum_{c=1}^C \exp(-\|\boldsymbol{x}_i-\boldsymbol{\mu}_{c\backslash i}\|_2^2)},\ \text{where}\ \boldsymbol{\mu}_{c\backslash i} = \mathop{\mathrm{E}}_{j\neq i|y_j=c} \boldsymbol{x}_j, \quad (3)$$

and we apply a leave-one-out scheme to compute prototypes in order to maximally utilize available samples for the loss [20]. In the Proto loss, by minimizing distance to the target class prototype while maximizing the others, sample metrics are effectively improved in a similar way to discriminant analysis [13] that maximizes the ratio of between-class variance to within-class variance.

A pair-wise relationships among samples is more directly exploited by neighborhood component analysis (NCA) [14] which inspires an NCA loss [20, 28] as

$$\text{NCA:}\quad \ell_{metric}(\{\boldsymbol{x}_i,y_i\}_{i=1}^n) = -\mathop{\mathrm{E}}_i \log \frac{\sum_{j\neq i|y_j=y_i} \exp(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2^2)}{\sum_{j\neq i} \exp(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2^2)}. \quad (4)$$

Instead of using class prototypes (3), it aggregates softmax probabilities of intra-class samples for directly reducing pair-wise distance within a class while enlarging inter-class distance; similarly to (3), the NCA loss is computed in a leave-one-out manner across samples. These Proto and NCA losses perform metric learning in different ways akin to GMM [23] and KDE [29], respectively.

2.3 Comparison of ℓ_{cls} and ℓ_{metric}

As analyzed in [4], the classification loss (Sec. 2.1) and metric-based loss (Sec. 2.2) share the same goal to maximize mutual information between labels y and features $\boldsymbol{x}$. From a viewpoint of training via back-propagation, however, those two losses provide different types of gradients as follows.

Derivatives of the losses w.r.t. the feature $\boldsymbol{x}_i$ are given by

$$\frac{\partial \ell_{cls}}{\partial \boldsymbol{x}_i} = \boldsymbol{W}\frac{\partial \ell_{cls}}{\partial (\boldsymbol{W}^\top \boldsymbol{x}_i)} \in \mathtt{span}(\boldsymbol{W}),\quad \frac{\partial \ell_{metric}}{\partial \boldsymbol{x}_i} = 2\sum_{j\neq i} \frac{\partial \ell_{metric}}{\partial (\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2)}(\boldsymbol{x}_i-\boldsymbol{x}_j) \in \mathtt{span}(\boldsymbol{X}), \quad (5)$$

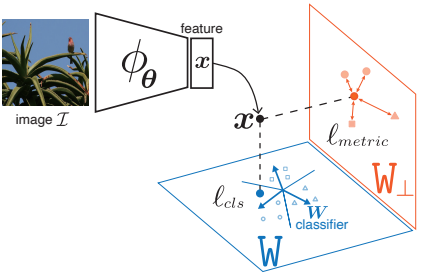


Figure 1: Direct sum of two losses.

Table 1: Discriminant score [13] ($0 \le \sigma_B/\sigma_T \le 1$) of test samples on mini-ImageNet. It is measured in two subspaces by projecting $\boldsymbol{x}$ via $\mathtt{W}^\top\boldsymbol{x}$ and $\mathtt{W}_\perp^\top\boldsymbol{x}$. Higher score indicates better discriminativity.

Feature space	SCE	sum (6) SCE+NCA	ours (7) SCE⊕NCA
Classifier subspace $\mathtt{W}^\top\boldsymbol{x}$	0.28	0.32	0.31
Complementary $\mathtt{W}_\perp^\top\boldsymbol{x}$	0.16	0.12	0.30

where $\mathtt{span}$ indicates a subspace spanned by column vectors of a matrix $\boldsymbol{W}$ and $\boldsymbol{X}=[\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n]\in\mathbb{R}^{d\times n}$, and for ℓ_{metric} we show the derivative of the NCA loss (4) while the similar form of loss gradient is produced by the Proto loss (3) as shown in the supplementary material. The classification loss restricts gradient-based updates within the classifier subspace $\mathtt{span}(\boldsymbol{W})$, effectively enhancing discriminativity of feature representation as distinctive characteristics among classes are encoded by the weight $\boldsymbol{W}$. On the other hand, in the metric-based loss, the updating direction lies in the sample subspace, paying broad attention to rather whole feature space for effectively improving all feature components.

Thus, combining these two mechanism could boost discriminative learning. To that end, it is a straightforward approach to sum up those two losses by

$$\ell_+ = \mathop{\mathrm{E}}_i[\ell_{cls}(\boldsymbol{x}_i,y_i;\boldsymbol{W},\boldsymbol{b})]+\ell_{metric}(\{\boldsymbol{x}_i,y_i\}_{i=1}^n). \tag{6}$$

In back-propagation, however, it results in a simple addition of the two different updating formulas (5) which could be interfered due to the subspace overlapping, $\mathtt{span}(\boldsymbol{W})\cap\mathtt{span}(\boldsymbol{X})\neq\emptyset$. Therefore, we can conjecture that the simple approach to sum up the two losses would be less effective for enhancing feature representation; it is also empirically shown in Sec. 3.

2.4 Direct sum of losses

We propose an approach to effectively integrate those two losses by leveraging the *classifier subspace* to reduce the interference; an overview of the method is depicted in Fig. 1. Here, we assume that the feature dimensionality is greater than the number of classes, $d>C$, which is frequently found in classification tasks.

Let an orthonormal basis matrix of the classifier subspace $\mathtt{span}(\boldsymbol{W})$ be denoted by $\mathtt{W}\in\mathbb{R}^{d\times C}$ such that $\mathtt{W}^\top\mathtt{W}=\mathtt{I}$ and $\boldsymbol{W}=\mathtt{W}\mathtt{W}^\top\boldsymbol{W}$. A linear classifier is thereby written in $\boldsymbol{W}^\top\boldsymbol{x}=\boldsymbol{W}^\top(\mathtt{W}\mathtt{W}^\top\boldsymbol{x})$ where $\mathtt{W}\mathtt{W}^\top\boldsymbol{x}$ is a projection from $\boldsymbol{x}$ to the classifier subspace. Thus, as shown in the gradient (5), the classification loss ℓ_{cls} elaborates the classifier subspace feature $\mathtt{W}\mathtt{W}^\top\boldsymbol{x}$ without paying attention to its complementary subspace; we denote bases of the complementary subspace as $\mathtt{W}_\perp\in\mathbb{R}^{d\times(d-C)}$ such that $\mathtt{W}_\perp^\top\mathtt{W}_\perp=\mathtt{I}$ and $\mathtt{W}^\top\mathtt{W}_\perp=0$. Preliminary experimental results in Table 1 demonstrate that the features learnt by ℓ_{cls} of SCE (1) exhibits less discriminativity on the complementary subspace $\mathtt{W}_\perp$. Even by combining ℓ_{cls} with ℓ_{metric} of NCA (4) in the simple summation (6), the complementary feature representation is not improved but rather degraded a bit due to the interference by the overlap of gradient subspaces as discussed in Sec. 2.3. Therefore, it implies that there is a room in the complementary subspace to further improve feature representation.

Table 2: Datasets for few-shot learning. The number of classes and samples are shown respectively for training/validation/test set which are disjoint in terms of class categories.

	mini-ImageNet [28]	tiered-ImageNet [24]	CUB200 few-shot [6]	Cifar100 few-shot [3]
Classes	64 / 16 / 20	351 / 97 / 160	100 / 50 / 50	64 / 16 / 20
Samples	38400 / 9600 / 12000	448695 / 124261 / 206209	5885 / 2950 / 2953	38400 / 9600 / 12000

We exploit the complementary subspace $\mathtt{W}_\perp$ ignored in ℓ_{cls} by using ℓ_{metric}. The proposed method is formulated by *decomposing* feature $\boldsymbol{x}$ into the classifier subspace $\mathtt{W}$ and its complementary one $\mathtt{W}_\perp$ to which the two types of losses are respectively applied as

$$\ell_\oplus = \mathop{\mathrm{E}}_i[\ell_{cls}(\boldsymbol{x}_i, y_i; \boldsymbol{W}, \boldsymbol{b})] + \ell_{metric}(\{\mathtt{W}_\perp \mathtt{W}_\perp^\top \boldsymbol{x}_i, y_i\}_{i=1}^n), \tag{7}$$

where note that $\ell_{cls}(\boldsymbol{x}_i, y_i) = \ell_{cls}(\mathtt{W}\mathtt{W}^\top \boldsymbol{x}_i, y_i)$. Since this is the sum of two losses measured on disjoint subspaces as shown in Fig. 1, our approach (7) is regarded as *direct sum* of those losses in terms of the classifier subspace $\mathtt{W}$ and its complementary subspace $\mathtt{W}_\perp$. By splitting the feature space, we can reduce interference between the two losses and effectively harness the updating mechanisms (5) embedded in those losses to enhance feature representation. It is noteworthy that the disentanglement is applied only to a loss without touching a backbone architecture, making the proposed method applicable to various networks. The preliminary result in Table 1 shows that our direct-sum approach improves discriminativity on both the subspace representations $\mathtt{W}$ and $\mathtt{W}_\perp$.

We compute the projection to the complementary subspace without explicitly extracting the basis $\mathtt{W}_\perp$ as

$$\mathtt{W}_\perp \mathtt{W}_\perp^\top \boldsymbol{x} = \boldsymbol{x} - \mathtt{W}\mathtt{W}^\top \boldsymbol{x} \approx \boldsymbol{x} - \boldsymbol{W}(\boldsymbol{W}^\top \boldsymbol{W} + \varepsilon \mathtt{I})^{-1} \boldsymbol{W}^\top \boldsymbol{x}, \tag{8}$$

where a small fraction ε is introduced to avoid rank reduction of $\boldsymbol{W}$; we set $\varepsilon = 0.001$. The approximated form of projection is justified in the supplementary material.

3 Result

We apply the proposed direct-sum loss (Sec. 2.4) to train a model $\phi_{\boldsymbol{\theta}}$ on few-shot image classification tasks by utilizing two types of losses (Sec. 2.1&2.2).

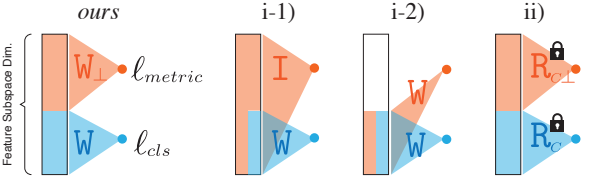
3.1 Experimental setting

Datasets: We evaluate image classification performance on few-shot scenarios using mini-ImageNet [28], tiered-ImageNet [24], CUB200 [6] and Cifar100 [3], of which details are shown in Table 2. In the few-shot learning, a dataset is split into training/validation/test sets which are disjoint in terms of class categories; a model learned on a training set is deployed to classify test samples drawn from novel classes which are unseen in the training.

Training: We construct a backbone model $\phi_{\boldsymbol{\theta}}$ by ResNet-12 [20, 21], a popular variant of ResNet [17] in the literature, which produces $d = 640$-dimensional feature $\boldsymbol{x}$. In training, the model is equipped with a classification head parameterized by $\{\boldsymbol{W}, \boldsymbol{b}\}$ (Sec. 2.1) to distinguish training class categories so that both classification and metric-based losses join in optimizing the parameters $\boldsymbol{\theta}$. The learning is performed in a standard supervised way using training samples annotated by class labels; we apply SGD optimizer with momentum of 0.9,

Table 3: Ablation study regarding subspace-based loss integration. The left table shows classification accuracies (%) by 5-way 1-shot evaluation on mini-ImageNet; the top two columns are the results of individual losses, SCE and NCA, without loss integration. The right chart depicts approaches to leverage various subspaces for the integration.

	projection in ℓ_{cls}	projection in ℓ_{metric}	Acc. (%) 1 shot
SCE	$\mathtt{W}$	-	63.85
NCA	-	$\mathtt{I}$	63.95
ours	$\mathtt{W}$	$\mathtt{W}_{\perp}$	65.16
i-1)	$\mathtt{W}$	$\mathtt{I}$	64.20
i-2)	$\mathtt{W}$	$\mathtt{W}$	63.93
ii-1)	$\mathtt{R}_{C}$	$\mathtt{R}_{C\perp}$	64.25
ii-2)	$\mathtt{I}_{C}$	$\mathtt{I}_{C\perp}$	63.93



weight decay of 0.0005, initial learning rate of 0.1 which is dropped with a decay rate of 0.1 at the 80th and 120th epochs over 160 training epochs including 10-epoch warming up. A mini-batch size n is set to 512 samples on all the datasets except for CUB200 where 128 samples constitute a mini-batch. As the metric-based loss (Sec. 2.2) is built on pair-wise relationships among mini-batch samples, class distribution in the mini-batch would affect performance. Thus, we construct a mini-batch by N-way K-shot samples to control the class distribution under the same budget of mini-batch size, $n = NK$; we draw N class categories each of which contains K training samples. We report the best performance across various configurations of N and K for fair comparison of all the methods; the details are shown in the supplementary material.

Few-shot classification: After training, the classification head is detached and the backbone model $\phi_{\boldsymbol{\theta}}$ is transferred to classification of novel-class samples in a few-shot setting, referred to as an N-way K-shot scenario; for evaluation, we repeatedly draw 10,000 sets of $N = 5$-way $K \in \{1,5\}$-shot with 15 test query samples. We extract feature vectors by applying the *frozen* backbone $\phi_{\boldsymbol{\theta}}$ to the $N \cdot K$ samples and then construct a non-parametric classifier of nearest class centroid [20, 25, 31] to categorize test query samples into N classes.

3.2 Ablation study

Our method (7) contains the key processes that (*i*) two losses are separated by disjoint *subspaces* and (*ii*) the subspaces are built based on the trainable classifier weight $\boldsymbol{W}$, i.e., *classifier subspace*. We analyze the method from these aspects in an ablation manner by using SCE classification loss (1) and NCA metric loss (4); the performance comparison on mini-ImageNet is shown in Table 3 including the results of individual losses, SCE and NCA.

***i*) Subspace separation:** From a perspective to feed different feature spaces to the respective losses in (7), we can compute ℓ_{metric} on various subspaces of different overlapness with the classifier subspace $\mathtt{W}$ which is an intrinsic subspace of ℓ_{cls} as described in Sec. 2.4. The most straightforward approach is to use whole feature space, $\mathtt{I}\boldsymbol{x}$, using an identity projection matrix $\mathtt{I} \in \mathbb{R}^{d\times d}$; note that $\mathtt{I} = \mathtt{W}\mathtt{W}^{\top} + \mathtt{W}_{\perp}\mathtt{W}_{\perp}^{\top}$. This leads to a simple summation approach (6) which integrates two losses by directly computing ℓ_{metric} in an input feature space. It is also conceivable to formulate the metric loss *within* the classifier subspace as $\ell_{metric}(\{\mathtt{W}\mathtt{W}^{\top}\boldsymbol{x}_i\}_{i=1}^{n})$, which is contrary to our approach (7) computing ℓ_{metric} in the complementary subspace $\mathtt{W}_{\perp}$.

As shown in Table 3i, the simple summation (Table 3i-1) is slightly superior to the

loss only defined in the classifier subspace (Table 3i-2); the summation approach provides marginal improvement over the individual losses. As the classifier-subspace loss (i-2) causes heavy overlap between ℓ_{cls} and ℓ_{metric}, two loss gradients (5) are interfered in the classifier subspace, failing to improve performance. On the other hand, the proposed method that separates the two losses in (7) outperforms those variant approaches; specifically, it is superior to the simple summation (i-1). As discussed in Sec. 2.3, the two losses ℓ_{cls} and ℓ_{metric} provide distinct updating approaches toward discriminative feature representation [4]. In our method, separating those two updating mechanisms in terms of subspace increases their synergy without interference by successfully enhancing the complementary feature representation (Table 1).

***ii*) Classifier subspace:** We then manipulate the prior structure of the classifier subspace $\mathtt{W}$ which is gradually trained through the end-to-end learning via ℓ_{cls}. To that end, fixing classifier [18] inspires us to *fix* the classifier subspace bases to a predefined orthonormal matrix as follows. We can employ a random orthonormal matrix $\mathtt{R}_C \in \mathbb{R}^{d\times C}$ for the bases. While $\mathtt{R}_C$ is a dense matrix composed of non-zero values, a naive approach is to use a sparse matrix $\mathtt{I}_C \in \mathbb{R}^{d\times C}$, the first C-column sub-matrix of an identity matrix $\mathtt{I} \in \mathbb{R}^{d\times d}$; $\mathtt{I}_C^\top \boldsymbol{x}$ picks up the first C feature elements of $\boldsymbol{x}$. Though the two matrixes are coincident by rotation, they work differently in back-propagation. That is, $\mathtt{I}_C$ completely excludes the latter $d-C$ feature components in $\boldsymbol{x}$ while $\mathtt{R}_C$ conveys back-propagation to all d feature components. In the fixed subspace, trainable classifier weights are re-parameterized by $\boldsymbol{W} = \mathtt{R}_C\boldsymbol{\omega}$ or $\mathtt{I}_C\boldsymbol{\omega}$ where $\boldsymbol{\omega} \in \mathbb{R}^{C\times C}$ are trainable parameters; e.g., for the fixed random subspace, the direct-sum loss (7) is given by $\mathrm{E}_i\left[\ell_{cls}(\mathtt{R}_C^\top \boldsymbol{x}_i, y_i; \boldsymbol{\omega}, \boldsymbol{b})\right] + \ell_{metric}(\{\mathtt{R}_{C\perp}\mathtt{R}_{C\perp}^\top \boldsymbol{x}_i, y_i\}_{i=1}^n)$.

As shown in Table 3ii, the random subspace $\mathtt{R}_C$ works slightly better than the sparse one $\mathtt{I}_C$. This comparison clarifies that separating *feature components* is less effective and dense subspace separation is important for direct-sum integration. Note that in the subspace separation, two loss gradients are disjoint in terms of subspaces, $\frac{\partial \ell_{cls}}{\partial \boldsymbol{x}}^\top \frac{\partial \ell_{metric}}{\partial \boldsymbol{x}} = 0$, but overlapped at each feature element, $\frac{\partial \ell_{cls}}{\partial x_j} \cdot \frac{\partial \ell_{metric}}{\partial x_j} \neq 0$. The fixed random subspace (ii-1) is inferior to our approach which adaptively optimizes a classifier subspace during training. These experimental results highlight efficacy of our adaptive subspace separation for integrating two types of losses in an end-to-end framework.

3.3 Performance comparison

As described in Sec. 2, we have two options of SCE and BCE for a classification loss (Sec. 2.1) while considering Proto and NCA losses as a metrics-based loss (Sec. 2.2). Table 4 thoroughly compares performances of those losses on four datasets (Table 2); a simple summation (6) of two losses (Table 3i-1) is denoted by '+' while our direct-sum approach (7) is indicated by '⊕', such as in SCE+NCA and SCE⊕NCA.

In the simple summation of ℓ_{cls} and ℓ_{metric}, NCA compensates the classification losses; SCE+NCA and BCE+NCA work relatively better than those with Proto loss. The NCA loss (4) is directly built upon pair-wise relationships across samples to enhance discriminativity in a complementary way to the classification losses, while the Proto loss (3) utilizes class-wise representation via class prototypes similarly to classification losses using $\boldsymbol{W}$; such a similarity regarding loss formulations could further induce interference. The simple summation approach, however, provides less evident improvement over the individual losses. On the other hand, our direct-sum method improves performance by successfully exploiting the synergy of two losses. While NCA works well as is the case with the simple summation,

Table 4: Classification accuracies (%) on 5-way {1, 5}-shot evaluation; in each cell, left/right number shows accuracy in 1/5-shot setting, respectively.

Method	mini-ImageNet	tiered-ImageNet	CUB200 few-shot	Cifar100 few-shot
Classification loss				
SCE	63.85, 79.68	71.67, 86.37	73.37, 88.53	70.88, 84.84
BCE	64.00, 80.20	72.14, 86.14	75.80, 89.57	70.15, 83.91
Metric loss				
NCA [20]	63.95, 79.47	70.40, 84.95	75.78, 87.74	71.49, 84.49
Proto [25]	61.71, 77.56	70.40, 85.39	76.33, 88.53	70.10, 83.97
Sum loss (6)				
SCE+NCA	64.20, 80.09	70.08, 84.87	78.47, 90.66	70.85, 85.39
SCE+Proto	62.84, 80.82	70.95, 85.66	76.85, 89.77	68.83, 85.13
BCE+NCA	64.64, 81.06	70.84, 85.48	78.95, 90.26	72.03, 85.17
BCE+Proto	63.43, 80.74	71.36, 85.95	78.66, 90.36	69.78, 83.28
Direct-Sum loss (Ours) (7)				
SCE⊕NCA	65.16, 81.89	72.20, 86.50	78.91, **90.73**	71.98, **85.69**
SCE⊕Proto	64.16, 81.19	**72.27, 86.75**	78.30, 90.10	71.68, 84.10
BCE⊕NCA	**65.61, 81.98**	71.61, 86.69	**79.89**, 90.49	**72.06**, 85.33
BCE⊕Proto	64.27, 81.44	72.14, 86.52	78.67, 89.99	70.74, 83.95

the proposed method also boosts performance of the Proto loss by means of the subspace separation.

We further evaluate those methods in a *transductive* few-shot classification [11] making use of 15 test query samples for inference, while we have so far applied an inductive classifier of nearest class centroid built only on the 5-way $\{1,5\}$-shot training samples. To that end, we replace the nearest mean classifier with LaplacianShot [33] to report performance results in Table 5. Similarly to the inductive classification results in Table 4, the proposed direct-sum method produces favorable performance improvement.

3.4 Spectral sum of losses

The direct-sum approach in Sec. 2.4 is built upon the assumption that the feature dimensionality is greater than the number of classes, $d > C$, so as to exploit the complementary subspace of $d - C$ dimensions; in the above experiments, $d = 640$ of ResNet-12 satisfies the requirement. To cope with the case of $d \leq C$, we can extend the direct-sum method by introducing *spectral* decomposition of a classifier as follows.

The classifier weight $\boldsymbol{W} \in \mathbb{R}^{d\times C}$ can be decomposed via singular value decomposition as

$$\boldsymbol{W} = \tilde{\mathtt{W}}\mathtt{diag}(\boldsymbol{\lambda})\mathtt{V}^\top, \tag{9}$$

where $\tilde{\mathtt{W}} \in \mathbb{R}^{d\times d}$ indicates the subspace basis matrix and $\boldsymbol{\lambda} \in \mathbb{R}^d$ are singular values. In a case of $d > C$ (Sec. 2.4) where $\boldsymbol{W}$ has C singular values, we can pad $d - C$ zeros to form $\boldsymbol{\lambda} = [\lambda_1, \cdots, \lambda_C, 0, \cdots, 0] \in \mathbb{R}^d$ for general formulation. Then, the singular values are converted into spectral weights of

$$\tilde{\boldsymbol{\lambda}} = \frac{\boldsymbol{\lambda}}{\max_j \lambda_j} \in [0,1]^d, \tag{10}$$

which indicate normalized significance of the bases $\tilde{\mathtt{W}}$; i.e., $\tilde{\lambda}_j$ indicates how much the j-th

Table 5: Transductive classification performance by applying LaplacianShot [33] classifier to 5-way {1, 5}-shot evaluation.

Method	mini-ImageNet	tiered-ImageNet	CUB200 few-shot	Cifar100 few-shot
Classification loss				
SCE	71.25, 82.34	79.35, 87.38	83.97, 91.44	78.66, 86.12
BCE	70.87, 82.26	78.85, 87.61	84.56, 91.78	77.37, 84.68
Metric loss				
NCA [20]	71.13, 81.74	78.11, 86.43	82.68, 89.62	79.10, 85.49
Proto [25]	68.98, 80.30	79.11, 86.58	85.41, 90.47	77.47, 84.85
Sum loss (6)				
SCE+NCA	71.60, 82.61	78.15, 86.49	87.60, 92.81	79.71, 86.70
SCE+Proto	72.60, 84.12	79.47, 87.76	86.70, 91.76	79.39, **87.50**
BCE+NCA	72.69, 83.60	78.55, 86.68	87.40, 92.39	79.90, 86.20
BCE+Proto	72.39, 84.18	79.75, 87.84	87.95, 92.66	77.51, 84.04
Direct-Sum loss (Ours) (7)				
SCE⊕NCA	**73.39**, 84.26	79.87, 87.98	87.58, 92.24	**80.62**, 86.71
SCE⊕Proto	72.93, 84.07	79.96, **88.04**	87.65, 92.76	78.42, 84.88
BCE⊕NCA	72.95, **84.30**	79.44, 87.11	**88.11**, 92.85	80.32, 87.18
BCE⊕Proto	73.02, 84.11	**80.16**, 87.96	87.85, **92.95**	77.60, 85.08

Table 6: Classification performances on iNaturalist-2017 dataset [9]. As in Table 4, classification accuracies are measured on 5-way {1, 5}-shot evaluation by an inductive classifier.

p	0.5	0.4	0.3	0.2	0.1
SCE$\tilde{\oplus}_p$NCA	83.13, 92.57	83.49, 92.79	**83.73**, 92.77	83.51, **92.99**	83.50, 92.75
SCE$\tilde{\oplus}_p$Proto	83.30, 92.77	83.13, 92.71	**83.70**, 92.68	83.58, 92.77	83.42, **93.00**

Method	SCE	NCA	Proto	SCE+NCA	SCE+Proto
Acc. (%)	81.76, 92.73	82.42, 92.22	82.95, 92.08	82.43, 92.29	82.02, 91.87

column of $\tilde{\mathtt{W}}$ contributes to the classifier $\boldsymbol{W}$. By using the spectral weights, we formulate the *spectral-sum* method to integrate ℓ_{cls} and ℓ_{metric} in

$$\ell_{\tilde{\oplus}_p} = \mathop{\mathrm{E}}_{i}[\ell_{cls}(\boldsymbol{x}_i, y_i; \boldsymbol{W}, \boldsymbol{b})] + \ell_{metric}(\{\tilde{\mathtt{W}}\,\mathsf{diag}(\mathbf{1} - \tilde{\boldsymbol{\lambda}}^p)\tilde{\mathtt{W}}^\top \boldsymbol{x}_i\}_{i=1}^n), \tag{11}$$

where $\hat{\boldsymbol{\lambda}}^p = [\tilde{\lambda}_1^p, \cdots, \tilde{\lambda}_d^p]^\top$ with a hyper-parameter $0 \le p \le 1$ of power exponent. The $\mathbf{1} - \tilde{\boldsymbol{\lambda}}^p$ works as *weighting* for constructing a feature (sub)space *complementary* to the classifier $\boldsymbol{W}$; it reduces overlap with $\boldsymbol{W}$ as much as possible, as shown in the supplementary material. It should be noted that the direct-sum method (7) for $d > C$ is reconstructed by $p = 0$ to produce the *binary* weights of $1 - \tilde{\lambda}^0 \in \{0, 1\}$ which pick up basis vectors of the complementary classifier subspace of $\lambda_j = 0$, $C < \forall j \le d$. Thus, the spectral-sum loss (11) is regarded as a natural extension of the direct sum (7) via soft weighting based on the singular values (10).

We applied the method to few-shot image classification on iNaturalist-2017 dataset [9] which provides $C = 908$ classes on the training set, greater than $d = 640$ feature dimensionality. As shown in Table 6, we apply the method (11) with various exponent p and see that the lower p, especially $p = 0.3$, produces favorable performance, outperforming the others.

4 Conclusion

We have proposed an approach to integrate two types of losses, classification and metric-based losses. As a simple summation of the two losses brings about interference in parameter updating, the proposed method divides a feature space into subspaces based on a classifier weight so that the two losses work on respective subspaces in a separated manner. The *direct-sum* loss enhances feature representation in the complementary classifier space ignored in the classification loss by exploiting the metric loss without interference. In the experiments on few-shot image classification which demands generalizable feature representation, the proposed method favorably improves performance by effectively combining the two losses.

References

[1] Atish Agarwala, Samuel Stern Schoenholz, Jeffrey Pennington, and Yann Dauphin. Temperature check: theory and practice for training models with softmax-cross-entropy losses. *Transactions on Machine Learning Research*, 2023.

[2] Emanuel Ben-Baruch, Tal Ridnik, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. Asymmetric loss for multi-label classification. In *ICCV*, pages 82–91, 2021.

[3] Luca Bertinetto, Jo ao Henriques, Philip H.S. Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. In *ICLR*, 2019.

[4] Malik Boudiaf, Jérôme Rony, Imtiaz Masud Ziko, Eric Granger, Marco Pedersoli, Pablo Piantanida, and Ismail Ben Ayed. A unifying mutual information view of metric learning: cross-entropy vs. pairwise losses. In *ECCV*, 2020.

[5] Sebastian Bruch, Xuanhui Wang, Michael Bendersky, and Marc Najork. An analysis of the softmax cross entropy loss for learning-to-rank with binary relevance. In *ICTIR*, pages 75–78, 2019.

[6] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019.

[7] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *CVPR*, pages 1320–1329, 2017.

[8] Dami Choi, Christopher J. Shallue, Zachary Nado, Jaehoon Lee, Chris J. Maddison, and George E. Dahl. On empirical comparisons of optimizers for deep learning. *arXiv,* 1910.05446, 2019.

[9] Bharath Hariharan Davis Wertheimer. Few-shot learning with localization in realistic settings. In *CVPR*, 2019.

[10] Jiankang Deng, Jia Guo, Xue Niannan, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019.

[11] Guneet S. Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. In *ICLR*, 2020.

[12] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE TPAMI*, 28(4):594–611, 2006.

[13] Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, Boston, 1990.

[14] Jacob Goldberger, Sam Roweis, Geoff Hinton, and Ruslan Salakhutdinov. Neighbourhood components analysis. In *NeurIPS*, 2005.

[15] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[16] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[18] Elad Hoffer, Itay Hubara, and Daniel Soudry. Fix your classifier: The marginal value of training the last weight layer. In *ICLR*, pages 5822–5830, 2018.

[19] Mahmut Kaya and Hasan Sakir Bilge. Deep metric learning: A survey. *Symmetry*, 11 (9):1066, 2019.

[20] Steinar Laenen and Luca Bertinetto. On episodes, prototypical networks, and few-shot learning. In *NeurIPS*, 2021.

[21] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *CVPR*, 2019.

[22] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. In *ICML*, pages 507–516, 2016.

[23] Geoffrey J. McLachlan and Kaye E. Basford. *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, NY, 1988.

[24] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018.

[25] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[26] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: A good embedding is all you need? In *ECCV*, 2020.

[27] Grigorios Tsoumakas and Ioannis Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3):1–13, 2007.

[28] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, 2016.

[29] M.P. Wand and M.C. Jones. *Kernel Smoothing*. Chapman and Hall, 1995.

[30] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jinbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014.

[31] Yan Wang, Wei-Lun Chao, Kilian Q. Weinberger, and Laurens van der Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. *arXiv,* 1911.04623, 2019.

[32] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, pages 499–515, 2016.

[33] Imtiaz Masud Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Laplacian regularized few-shot learning. In *ICML*, 2020.