Deep Metric Learning based on Scalable Neighborhood Components for Remote Sensing Scene Characterization

Jian Kang, *Member, IEEE,* Ruben Fernandez-Beltran, *Member, IEEE,* Zhen Ye, Xiaohua Tong, *Senior Member, IEEE,* Pedram Ghamisi, *Senior Member, IEEE,* and Antonio Plaza, *Fellow, IEEE*

Abstract

With the development of convolutional neural networks (CNN), the semantic understanding of remote sensing scenes has been significantly improved based on their prominent feature encoding capabilities. Whereas many existing deep-learning models focus on designing different architectures, only few works in the remote sensing field have focused on investigating the performance of the learned feature embeddings and the associated metric space. In particular, two main loss functions have been exploited: the contrastive and the triplet loss. However, the straightforward application of these techniques to remote sensing images may not be optimal in order to capture their neighborhood structures in the metric space, due to the insufficient sampling of image pairs or triplets during the training stage, and to the inherent semantic complexity of remotely sensed data. To solve these problems, we propose a new deep metric learning approach, which overcomes the limitation on the class discrimination by means of two different components: 1) scalable neighborhood component analysis (SNCA), which aims at discovering the neighborhood structure in the metric space; and 2) the cross entropy loss, which aims at preserving the class discrimination capability based on the learned class prototypes. Moreover, in order to preserve feature consistency among all the mini-batches during training, a novel optimization mechanism based on momentum update is introduced for minimizing the proposed loss. An extensive experimental comparison (using several state-of-the-art models and two different benchmark datasets) has been conducted to validate the effectiveness of the proposed method from different perspectives, including: 1) classification; 2) clustering; and 3) image retrieval. The related codes of this paper will be made publicly available for reproducible research by the community.

Index Terms

Deep learning, metric learning, remote sensing scene characterization, dimensionality reduction.

I. Introduction

With the ongoing development of different Earth Observation missions and programmes, the semantic understanding of remote sensing (RS) image scenes plays a fundamental role in many important applications and societal needs [1], including preservation of natural resources [2], urban and regional planning [3], contingency management [4], land-cover analysis [5] and global Earth monitoring [6], among others. From a practical perspective, the RS scene recognition problem consists of predicting the semantic concept associated with a given aerial scene, based on its own visual content. In this way, scene-based recognition methods are expected to deal with high intra-class and low inter-class variabilities, since airborne and spaceborne optical data often comprise a wide variety of spatial structures that lead to a particularly challenging characterization for RS scenes [7].

In the literature, extensive research has been conducted and a wide variety of scene recognition methods have been presented within the RS field [8], [9]. From hand-crafted feature-based approaches [10], [11] to more elaborated unsupervised techniques [12], [13], the inherent complexity of the RS image domain often limits the performance of these traditional schemes when dealing with high-level semantic concepts [14]. More recently, deep-learning methods have shown a great potential to uncover highly discriminating features in aerial scenes [15], being the so-called deep metric learning approach one of the most prominent trends [16]–[18]. Specifically, deep metric learning aims at projecting semantically similar input data to nearby locations in the final feature space, which is highly appropriate to manage complex RS data [19]. Nonetheless, there are multiple factors, e.g. large-scale archives, sensor types or image acquisition conditions, that still make the semantic understanding of aerial scenes very challenging, thus motivating the development of new models to effectively learn discriminative CNN-based characterizations for unconstrained land cover scenes [9].

In order to address all these challenges, this paper proposes a new RS scene characterization approach, which provides a new perspective on the traditional deep embedding scheme typically used in land cover recognition tasks [16], [17]. The main objective of the proposed method consists of learning a low-dimensional metric space that can properly capture the semantic similarities among all the RS scenes based on the CNN-based feature embedding of the whole data collection. Moreover, the learned feature embedding in such metric space has to be effectively

generalized by means of out-of-sample RS scenes. To achieve this goal, we first investigate the scalable neighborhood component analysis (SNCA) [20] and further analyze the limitations of this recent method on the discrimination of RS scenes. Then, we develop an innovative deep metric learning approach that has been specifically designed to manage the particular semantic complexity of the RS image domain. Specifically, two main components are involved in this new design: 1) SNCA, which aims at discovering the neighborhood structure in the metric space; and 2) the Cross Entropy (CE) loss, which aims at preserving the class discrimination capability based on the learned class prototypes. In addition, a novel optimization mechanism (based on the momentum update for SNCA) is proposed to generate consistent features within each training epoch. In order to demonstrate the effectiveness of our contribution when characterizing RS scenes, we conduct a comprehensive experimental comparison, which reveals that our newly proposed RS scene characterization method provides competitive advantages with respect to different state-of-the-art models in three different RS applications (scene classification, clustering, and retrieval), over two benchmark datasets. The main contributions of this paper can be summarized as follows:

1) To the best of our knowledge, this work investigates for the first time in the literature the suitability of using the SNCA method for characterizing remotely sensed image scenes while also analyzing its main limitations in RS.
2) We propose a new deep metric learning model specifically designed to characterize RS scenes. Our new approach is able to learn a metric space based on CNN models that preserve the discrimination capability for the highly variant RS semantic concepts.
3) In order to improve the consistency of the feature embeddings generated on the whole dataset during training, we propose a novel optimization mechanism based on momentum update for minimizing the SNCA-based losses.
4) Based on three different RS applications, we demonstrate the superiority of our newly proposed method with respect to several state-of-the-art characterization methods over different datasets. The related codes will be released for reproducible research inside the RS community.

The rest of this paper is organized as follows. Section II reviews some related works and highlights their main limitations when effectively characterizing RS scenes. Section III presents the proposed deep metric learning model for RS. In Section IV, extensive experiments are

conducted on several publicly available benchmark datasets. Finally, Section V concludes the paper with some remarks and hints at plausible future research lines.

II. Related Work

A. RS Scene Characterization

Broadly speaking, three different trends can be identified when characterizing remotely sensed scenes: (1) low-level feature-based techniques; (2) unsupervised approaches; and (3) deep-learning methods. A recent work published in [21] reviews the evolution of feature extraction approaches from shallow to deep by comprehensively evaluating both supervised and unsupervised approaches. The former group of techniques is focused on extracting salient features from the input images using straightforward visual descriptors, such as color, texture, spectral-spatial information, or a combination of descriptors. From the simplest low-level feature-based approaches, which make use of color histograms [10], [22], to the most elaborated techniques, that consider texture features as well as gradient shape descriptors [11], [23], [24], all these methods exhibit limitations when dealing with high-level semantic concepts, due to the inherent complexity of the RS image domain [14], [25].

In order to enhance the visual characterization and generalization, unsupervised feature learning approaches have been proposed to classify airborne and space optical data. The rationale behind this kind of methods is based on encoding the low-level features of the input scene into a higher-level feature space by means of unsupervised learning protocols. For instance, sparse coding [12], [13], topic modeling [26], [27], manifold learning [28], [29] and auto-encoders [30], [31] are some of the most recent unsupervised paradigms that have been successfully applied to the RS field. Despite the fact that these and other methods are able to provide performance advantages with respect to traditional low-level feature-based techniques, the unsupervised perspective of the encoding procedure may eventually reduce the intra-class discrimination ability, since actual scene classes are not taken into account.

Recently, deep-learning methods have attracted the attention of the RS research community due to their great potential to uncover highly discriminating features in aerial scenes [15]. More specifically, these approaches aim at projecting the input data onto the corresponding semantic label space through a hierarchy of nonlinear mappings and layers, which generate a high-level data characterization useful to classify remotely sensed imagery [32]. For instance, Yao *et al.* proposed in [33] a stacked sparse auto-encoder that extracts deep features used to

effectively classify aerial images. Lu *et al.* also presented in [34] an unsupervised representation learning method based on deconvolution networks for RS scene classification. With the increasing popularity of CNNs, other authors advocate the use of more complex deep-learning architectures (e.g., AlexNet [35], VGGNet [36] and GoogleNet [37]) to characterize and classify RS scenes. It is the case of Hu *et al.* who present in [38] two different scenarios to make use of VGGNet: (i) one directly using the last fully connected layers as image descriptors; and (ii) another considering an encoding procedure over the last convolutional layer feature maps. Chaib *et al.* also presented in [39] an RS classification method that employs the VGGNet model as feature extractor mechanism. Specifically, the authors adopt a feature fusion strategy in which each layer is regarded as a separate feature descriptor. Zang *et al.* defined in [40] a deep ensemble framework based on gradient boosting, which effectively combines several CNN-based characterizations. Analogously, Li *et al.* proposed in [41] a multi-layer feature fusion framework, which takes advantage of multiple pre-trained CNN models for RS scene classification. Cheng *et al.* also developed in [42] an RS classification approach using a bag of convolutional features obtained by different off-the-shelf CNN models. For fine-grained land-use classification, Kang *et al.* exploited multiple CNN models and categorized different types of buildings based on street view images [43].

Despite the effectiveness achieved by these and other relevant methods in the literature [44], multiple research works highlight the benefits of using deep-learning embeddings to characterize aerial scenes [19]. In general, the so-called deep metric learning approach aims at projecting semantically similar input datasets to nearby locations in the final feature space, by means of non-isotropic metrics [45]. As a result, this is a highly appropriate scheme to simplify complex topological spaces (which are often found in RS data). The unprecedented availability of airborne and space optical data, together with the constant development of the acquisition technology, are substantially increasing the complexity of the RS data and consequently its visual interpretation [1]. In addition, the probability of encountering unseen target scenes increases with the data complexity, which also makes the embedding strategy appropriate for transferring the knowledge from the training samples to broader semantic domains [46].

Several works in the most recent RS literature exemplify these facts. For instance, Gong *et al.* adopted in [16] the Lifted Structured Feature Embedding approach [47], which defines a structured objective function based on lifted pairwise distances within each training batch. The authors introduced an additional diversity-promoting criteria to decrease the metric pa-

rameter factor redundancy for RS scene classification. Cheng *et al.* presented in [17] a simple but effective method to learn highly discriminative CNN-based features for aerial scenes. In particular, the authors imposed a metric learning regularization term on the CNN features by means of the contrastive embedding scheme [48], which intrinsically enforces the model to be more discriminative and to achieve competitive performance. Similarly, Yan *et al.* proposed in [18] a cross-domain extension that aims at reducing the feature distribution bias and spectral shift in aerial shots, considering a limited amount of target samples. Whether the model is created using network ensembles [40] or more elaborated semantic embeddings [16], [17], the special particularities of the RS domain still raised some important challenges when classifying aerial scenes [9]. Specifically, the huge within-class diversity and between-class similarity of RS scenes motivate the development of new operational processing chains to effectively learn discriminative CNN-based characterizations that can obtain better semantic generalization for unconstrained land cover scenes. Note that there are many factors (such as different sensing dates, instrument positions, lighting conditions and sensor types) that also affect remotely sensed data and hence their semantic understanding.

B. Deep Metric Learning

Deep metric learning methods aim at learning a low-dimensional metric space based on CNN models, where the feature embeddings of semantic-similar images should be close and those of dissimilar images should be separated. The metric space with such characteristics can be learned by applying proper loss functions. Most of the existing deep metric learning methods can be categorized based on two types of loss functions [17], [49]–[51]: 1) the contrastive loss [48]; and 2) the triplet loss [52]. Some useful notations, as well as the definitions of these two losses are given below. Let $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ define as a set of N RS images and $\mathcal{Y} = \{\mathbf{y}_1, \cdots, \mathbf{y}_N\}$ is the associated set of label vectors, where each label vector $\mathbf{y}_i$ is represented by the one-hot vector, i.e., $\mathbf{y}_i \in \{0, 1\}^C$, where C is the total number of classes. If the image is annotated by the class c, the c-th element of $\mathbf{y}_i$ is 1, and 0 otherwise. $\mathbf{v}_i \in \mathbb{R}^D$ denotes the feature of the i-th image $\mathbf{x}_i$ obtained by a complex nonlinear mapping $\mathcal{F}(\mathbf{x}_i; \theta)$ based on a CNN model, where the set θ represents its learnable parameters. D is the dimension of the feature and $\mathbf{f}_i$ is the normalized feature on the unit sphere (i.e., $\mathbf{f}_i = \mathbf{v}_i / \|\mathbf{v}_i\|_2$). To train the deep metric learning system, a set $\mathcal{T}$ with M images is extracted from $\mathcal{X}$. According to this notation, the two aforementioned loss functions can be defined as:

1) Contrastive Loss:

$$L_{\text{contrastive}} = \sum_{i,j} l_{ij} \|\mathbf{f}_i - \mathbf{f}_j\|_2^2 + (1 - l_{ij}) h(m - \|\mathbf{f}_i - \mathbf{f}_j\|_2)^2, \tag{1}$$

where $h(\cdot)$ represents the hinge loss function, i.e., $h(x) = \max(0, x)$, m is the predefined margin, and l_{ij} is the label indicator satisfying:

$$l_{ij} = \begin{cases} 1, \text{if } \mathbf{y}_i = \mathbf{y}_j, \\ 0, \text{otherwise.} \end{cases} \tag{2}$$

Given an image pair $(\mathbf{x}_i, \mathbf{x}_j)$, the first term minimizes (during the training) the Euclidean distance of the two feature embeddings if they share the same class, and the second term is minimized to separate their distance by a certain margin m if they belong to different classes.

2) Triplet Loss:

$$L_{triplet} = \sum_i h(\|\mathbf{f}_i^a - \mathbf{f}_i^p\|_2^2 - \|\mathbf{f}_i^a - \mathbf{f}_i^n\|_2^2 + m), \tag{3}$$

where $\mathbf{f}_i^a$, $\mathbf{f}_i^p$, and $\mathbf{f}_i^n$ are the feature embeddings of an anchor image $\mathbf{x}_i^a$, a positive image $\mathbf{x}_i^p$, and a negative image $\mathbf{x}_i^n$. Normally, the positive image shares the same class with the anchor image, and the class of the negative image is different from that of the anchor image. Given a triplet $(\mathbf{f}_i^a, \mathbf{f}_i^p, \mathbf{f}_i^n)$, the triplet loss is minimized to push the negative image away from the anchor image so that the distance is larger than the distance of the positive pair with a certain margin.

C. Current Limitations in RS Scene Characterization

Most existing deep-learning based methods for RS scene characterization focus on developing different CNN architectures for improving the classification performance based on the semantic labels predicted by the CNN models. However, only few works in the RS field have addressed the problem of how to analyze the performance of the learned feature embeddings and the associated metric space. One of such pioneer works is [17], which introduced a novel loss function composed of the contrastive loss and the CE loss for learning discriminative features from RS images. The contrastive loss was also exploited in [53] for encoding Synthetic Aperture Radar (SAR) scene images into low-dimensional features. In [54], an RS image retrieval method was proposed based on the learned metric space by utilizing the triplet loss. Normally, the optimization of CNN models with respect to the contrastive or triplet loss functions is conducted stochastically

with mini-batches. For the contrastive loss, negative and positive pairs are usually constructed for training the CNN models within each mini-batch. Nonetheless, this scheme has an important limitation when considering the inherent semantic complexity of the RS image domain. For example, we assume that each RS image can be seen once during one epoch of training and $\mathbf{x}_i$ exists in one mini-batch for the current training iteration. The positive and negative images with respect to $\mathbf{x}_i$ in this mini-batch can be only seen during the current iteration of training. However, CNN models cannot capture all the other positive and negative images with respect to $\mathbf{x}_i$ outside the current mini-batch during this training epoch, which may lead to insufficient learning due to the particularly high intra-class and low inter-class variability of RS images. For the triplet loss, one should build the whole set of possible triplets when training the CNN models, where the number of possible triplets is in the order of $\mathcal{O}(|\mathcal{X}|^3)$ [55]. When considering a large-scale dataset (which is often the case in RS problems), sufficiently training CNN models will inevitably lead to a practically unaffordable computational cost.

III. Proposed Deep Metric Learning for RS

Our newly proposed end-to-end deep metric learning model for RS scene characterization consists of three main parts. First, a backbone CNN architecture is considered in order to generate the corresponding feature embedding space for the input images. In this work, we make use of the ResNet [56] architecture due to its good performance to classify RS scenes [57]. Second, a new loss function, which contains a joint CE term and an SNCA term, is used to optimize the proposed model in order to address the within-class diversity and between-class similarity inherent to RS scenes. Third, a novel optimization mechanism based on momentum update is proposed. Our mechanism can preserve the feature consistency within each training epoch better than the memory-bank based mechanism in [20]. Figure 1 provides a graphical illustration of our newly proposed deep metric learning approach. In the following sections, we describe in more detail the newly defined loss function and the considered optimization algorithm.

A. Loss Function

The neighborhood component analysis (NCA) [58] is a supervised dimensionality reduction method to learn a metric space through a linear projection of the input data such that the *leave-one-out* KNN score is stochastically maximized in the metric space. The SNCA [20], built upon the NCA, aims to find a metric space that can preserve well the neighborhood structure based

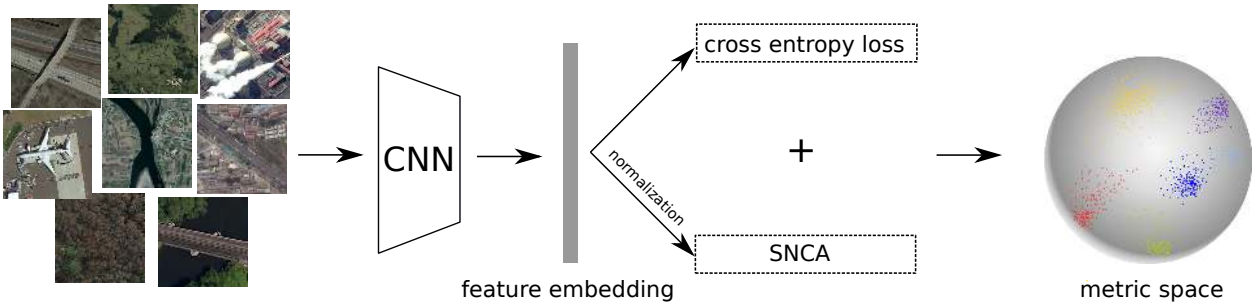


Fig. 1. Graphical illustration of the proposed end-to-end deep metric learning model, which is optimized using our newly defined loss function. With the proposed approach, we aim to encode RS images into the learned metric space through a CNN model, where the intra-class feature embeddings are grouped together, and the inter-class feature embeddings are separated.

on *deep models* with *scalable datasets*. Given a pair of images $(\mathbf{x}_i, \mathbf{x}_j)$ from the training set $\mathcal{T}$, their similarity s_{ij} in the metric space can be modeled with the cosine similarity:

$$s_{ij} = \mathbf{f}_i^T \mathbf{f}_j. \tag{4}$$

This means that the image $\mathbf{x}_i$ selects the image $\mathbf{x}_j$ as its neighbor in the metric space with a probability p_{ij} as:

$$p_{ij} = \frac{\exp(s_{ij}/\sigma)}{\sum_{k \neq i} \exp(s_{ik}/\sigma)}, \quad p_{ii} = 0, \tag{5}$$

where σ is the temperature parameter controlling the concentration level of the sample distribution [59]. When $i = j, \quad p_{ii} = 0$ indicates that each image cannot select itself as its own neighbor in the metric space. When $i \neq j$, p_{ij} indicates the probability that the image $\mathbf{x}_j$ can be chosen as a neighbor of the image $\mathbf{x}_i$ in the metric space and inherited the class label from $\mathbf{x}_i$. The higher the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$, the higher the opportunity that $\mathbf{x}_j$ can be selected as a neighbor of $\mathbf{x}_i$ in the metric space and inherited the class label from $\mathbf{x}_i$ as compared to the other images $\mathbf{x}_k$. This probability is often termed as *leave-one-out* distribution on $\mathcal{T}$. Based on this, the probability that $\mathbf{x}_i$ can be correctly classified is:

$$p_i = \sum_{j \in \Omega_i} p_{ij}, \tag{6}$$

where $\Omega_i = \{j | \mathbf{y}_i = \mathbf{y}_j\}$ is the index set of training images sharing the same class with $\mathbf{x}_i$. Intuitively, the image $\mathbf{x}_i$ can be correctly classified at a higher chance if more images $\mathbf{x}_j$ sharing

the same class with $\mathbf{x}_i$ are located as its neighbors in the metric space. Then, the objective of SNCA is to minimize the expected negative log likelihood over $\mathcal{T}$ with the definition:

$$L_{\text{SNCA}} = -\frac{1}{|\mathcal{T}|}\sum_i \log(p_i). \tag{7}$$

The gradient of L_{SNCA} with respect to $\mathbf{f}_i$ is given by:

$$\frac{\partial L_{\text{SNCA}}}{\partial \mathbf{f}_i} = \frac{1}{\sigma}\sum_k p_{ik}\mathbf{f}_i - \frac{1}{\sigma}\sum_{k\in\Omega_i} \tilde{p}_{ik}\mathbf{f}_k, \tag{8}$$

where $\tilde{p}_{ik} = p_{ik}/\sum_{j\in\Omega_i} p_{ij}$ is the normalized distribution of the ground-truth class. Based on the gradient in (8), an optimal solution of (7) will be reached when the probability p_{ik} of negative images (i.e. $k \notin \Omega_i$) equals 0. In other words, the similarities between $\mathbf{x}_i$ and some of positive images ($k \in \Omega_i$) can also be very low in the metric space, as long as there exist other positive images which are the neighbors of $\mathbf{x}_i$. On the one hand, this characteristic can be beneficial to discover the inherent locality structure among the images in the metric space, especially if there are intra-class variations in the dataset. On the other hand, there is one limitation of SNCA for K nearest neighbours (KNN) classification. Since some of the positive images ($k \in \Omega_i$) do not need to be close to $\mathbf{x}_i$, their feature embeddings may be closer to those of other negative images in the metric space. As illustrated in Figure 2(a), the classes A and B are separated, and their intra-class variation can also be discovered, which is represented by the groups of light and dark points. However, given the presence of some out-of-sample images sharing similar features with some images from both classes, they cannot be correctly categorized by exploiting the KNN classifier. One way to solve this problem is to separate the images from the two classes farther away from each other, which is illustrated in Figure 2(b). With the same feature embeddings as in Figure 2(a), the out-of-sample images are well recognized by the KNN classifier. To achieve this goal, we introduce the CE loss for learning the class-wise prototype to align the images with respect to their associated classes.

The CE loss aims to measure the distance between the distribution of model outputs and the real distribution. In terms of classification, the CE loss is defined as:

$$L_{\text{CE}} = -\frac{1}{|\mathcal{T}|}\sum_i\sum_c y_i^c \log(p_i^c), \tag{9}$$

where p_i^c denotes the probability that $\mathbf{x}_i$ is classified into the class c, formulated as:

$$p_i^c = \frac{\exp(\mathbf{w}_c^T\mathbf{v}_i)}{\sum_j \exp(\mathbf{w}_j^T\mathbf{v}_i)}, \tag{10}$$

(a) (b)

Fig. 2. Graphical illustration of the main limitation of SNCA. (a) In the metric space produced by SNCA, it will be challenging for the KNN classifier to distinguish the out-of-sample images located near the border between the two classes. (b) By introducing CE loss, these two classes are further separated in the metric space, and the same out-of-sample images in (a) can be accurately categorized by the KNN classifier.

where $\mathbf{w}_c$ are the learned parameters from class c. Minimizing the CE loss (9) consists of aligning all the images within the same class with the same vector $\mathbf{w}_c$. In that case, images from different classes are separated.

At this point, by taking advantage of the two losses, we propose a new joint loss function for learning a low-dimensional metric space, which can preserve the neighborhood structure among the images and also distinguish the images from different classes. The proposed joint function, termed as *SNCA-CE*, is defined as:

$$L = L_{\text{CE}} + \lambda L_{\text{SNCA}}, \tag{11}$$

where λ denotes a penalty parameter to control the balance between these two terms.

B. Optimization via Memory Bank

By applying the chain rule, we can obtain the gradient of the joint loss function with respect to $\mathbf{f}_i$:

$$\begin{aligned}\frac{\partial L_i}{\partial \mathbf{f}_i} = &- y_i^c(1-p_i^c)\|\mathbf{v}_i\|_2\mathbf{w}_c + \frac{\lambda}{\sigma}\sum_k p_{ik}\mathbf{f}_k \\ &- \frac{\lambda}{\sigma}\sum_{k\in\Omega_i}\tilde{p}_{ik}\mathbf{f}_k.\end{aligned} \tag{12}$$

From (12), we can infer that the feature embeddings of the entire dataset are needed for calculating the gradient. Following [20], we exploit a memory bank to store the normalized features, i.e., $\mathcal{B} = \{\mathbf{f}_i, \cdots, \mathbf{f}_M\}$ and we assume that these are up-to-date with regards to the

CNN parameters θ trained at the t-th iteration, i.e., $\mathbf{f}_i^{(t)} \approx \mathcal{F}(\mathbf{x}_i;\theta^{(t)})/\|\mathbf{v}_i\|_2$. At the $(t+1)$-th iteration, the gradient of the joint loss function with respect to $\mathbf{f}_i$ is:

$$\begin{aligned}\frac{\partial L_i}{\partial \mathbf{f}_i} =& -y_i^c(1-p_i^c)\|\mathbf{v}_i^{(t)}\|_2\mathbf{w}_c + \frac{\lambda}{\sigma}\sum_k p_{ik}\mathbf{f}_k^{(t)} \\ & - \frac{\lambda}{\sigma}\sum_{k\in\Omega_i}\tilde{p}_{ik}\mathbf{f}_k^{(t)}.\end{aligned} \tag{13}$$

Then, θ can be learned by using the back-propagation technique, and $\mathcal{B}$ can be updated by:

$$\mathbf{f}_i^{(t+1)} \leftarrow m\mathbf{f}_i^{(t)} + (1-m)\mathbf{f}_i, \tag{14}$$

where m is a parameter used for proximal regularization of $\mathbf{f}_i$ based on its historical versions. We term this optimization strategy as *SNCA-CE(MB)*. The associated optimization scheme is described in Algorithm 1.

Algorithm 1 SNCA-CE(MB)

Require: $\mathbf{x}_i$, and $\mathbf{y}_i$

1: Initialize θ and $\mathcal{B}$ (randomly), along with σ, λ, D and m.
2: **for** $t = 0$ to maxEpoch **do**
3: Sample a mini-batch.
4: Obtain $\mathbf{f}_i^{(t)}$ and $\mathbf{v}_i^{(t)}$ based on CNN with $\theta^{(t)}$.
5: Calculate s_{ij} with reference to $\mathcal{B}$.
6: Calculate the gradients based on (13).
7: Back-propagate the gradients.
8: Update $\mathcal{B}$ via (14).
9: **end for**

Ensure: θ, $\mathcal{B}$

C. Optimization via Momentum Update

In the SNCA-CE(MB) optimization scheme, the features in $\mathcal{B}$ are assumed to be up-to-date during training. However, this assumption cannot be easily satisfied, especially for scalable datasets. Suppose the image $\mathbf{x}_i$ is observed in the first iteration of one training epoch and the associated feature $\mathbf{f}_i^{(1)}$ –generated by the CNN with the parameters $\theta^{(1)}$– is stored in $\mathcal{B}$. Due to the training mechanism, this image cannot be observed again within the current epoch. Therefore,

for the t-th iteration, the feature $\mathbf{f}_j^{(t)}$ associated to image $\mathbf{x}_j$ –generated by the CNN with $\theta^{(t)}$– would not be consistent with $\mathbf{f}_i^{(1)}$, which is generated by a historical state of CNN. Since the optimization of SNCA-CE requires a look-up of the whole set of stored feature embeddings in $\mathcal{B}$ for each iteration, such inconsistency may lead to a sub-optimal training of the CNN.

To solve this issue, we propose a novel optimization mechanism based on momentum update [60] for the proposed SNCA-CE, termed as *SNCA-CE(MU)*. Instead of updating the feature embeddings stored in $\mathcal{B}$, the SNCA-CE(MU) progressively updates the state of the CNN in order to preserve the consistency of the features among all the mini-batches of each training epoch. To achieve this, an auxiliary CNN with parameters θ_{aux} is adopted, and θ_{aux} is updated by:

$$\theta_{\text{aux}}^{(t+1)} \leftarrow m\theta_{\text{aux}}^{(t)} + (1-m)\theta^{(t)}, \tag{15}$$

where $m \in [0, 1)$ is a momentum coefficient. It is worth noting that only the CNN with θ is updated by means of back-propagation. The auxiliary CNN with parameters θ_{aux} can evolve more smoothly than the CNN with θ. To this end, the features in $\mathcal{B}$ (encoded by the auxiliary CNN) are updated by:

$$\hat{\mathbf{f}}_i^{(t+1)} \leftarrow \hat{\mathbf{f}}_i^{(t)}, \tag{16}$$

where $\hat{\mathbf{f}}_i$ denotes the features generated by the auxiliary CNN. In other words, the features in $\mathcal{B}$ are replaced by the features encoded by the auxiliary CNN after each training epoch. The associated optimization scheme is described in Algorithm 2.

IV. Experiments

A. Dataset Description

In this section, we use two challenging RS image datasets to validate the effectiveness of the proposed methods. In the following, we provide a detailed description of the considered datasets:

1) **Aerial Image Dataset (AID)** [61]: This dataset is an important image collection, which has been specially designed for aerial scene classification and retrieval. In particular, it is made up of 10 000 RGB images belonging to the following 30 RS scene classes: airport, bare land, baseball field, beach, bridge, center, church, commercial, dense residential, desert, farmland, forest, industrial, meadow, medium residential, mountain, park, parking, playground, pond, port, railway station, resort, river, school, sparse residential, square, stadium, storage tanks and viaduct. Figure 3(a) illustrates some example scenes from this

Algorithm 2 SNCA-CE(MU)

Require: $\mathbf{x}_i$, and $\mathbf{y}_i$

Initialize θ, θ_{aux} and $\mathcal{B}$ (randomly), along with σ, λ, D and m.
for $t = 0$ to maxEpoch **do**
  Sample a mini-batch.
  Obtain $\mathbf{f}_i^{(t)}$ and $\mathbf{v}_i^{(t)}$ based on CNN with $\theta^{(t)}$.
  Obtain $\hat{\mathbf{f}}_i^{(t)}$ and $\hat{\mathbf{v}}_i^{(t)}$ based on the auxiliary CNN with $\theta_{\text{aux}}^{(t)}$.
  Calculate s_{ij} based on $\mathbf{f}_i^{(t)}$ and $\mathcal{B}$.
  Calculate the gradients based on (13).
  Back-propagate the gradients of θ.
  Update the parameters θ_{aux} of the auxiliary CNN via (15).
  Update $\mathcal{B}$ via (16).
end for

Ensure: θ

dataset. All the images are RGB acquisitions with a size of 600×600 pixels. In addition, the number of images per class ranges from 220 to 420, and the spatial resolution also varies from 8 to 0.5 meters. The AID dataset is publicly available[1].

2) **NWPU-RESISC45** [9]: This is a large-scale RS dataset, which contains 31 500 images uniformly distributed in 45 scene types: airplane, airport, baseball diamond, basketball court, beach, bridge, chaparral, church, circular farmland, cloud, commercial area, dense residential, desert, forest, freeway, golf course, ground track field, harbor, industrial area, intersection, island, lake, meadow, medium residential, mobile home park, mountain, overpass, palace, parking lot, railway, railway station, rectangular farmland, river, roundabout, runway, sea ice, ship, snow-berg, sparse residential, stadium, storage tank, tennis court, terrace, thermal power station and wetland. Figure 3(b) shows some sample scenes from this dataset. All these aerial images are RGB shots with size of 256×256 pixels and spatial resolution ranging from 30 to 0.2 meters. This dataset is also publicly available[2].

[1]AID dataset : http://goo.gl/WrJhu6

[2]NWPU-RESISC45 dataset: http://goo.gl/7YmQpK

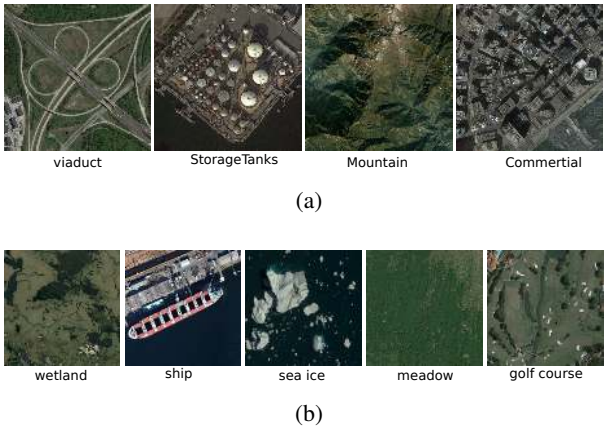


Fig. 3. Sample images from the two considered benchmark datasets: (a) AID; and (b) NWPU-RESISC45.

B. Experimental Setup

In order to extensively evaluate the effectiveness of the proposed method, we carry out several experiments from different perspectives, including: 1) image classification based on the KNN classifier; 2) clustering; and 3) image retrieval.

1) Classification: Given an out-of-sample image $\mathbf{x}^*$, its feature embedding $\mathbf{f}^*$ is obtained by applying $\mathcal{F}(\cdot)$ with the learned parameter set θ. Based on the Euclidean distance between $\mathbf{f}^*$ and the other stored embeddings in $\mathcal{B}$, we can obtain the closest K nearest neighbors, and the predicted class $\mathbf{y}^*$ can be determined based on their classes via majority voting. To evaluate classification performance, we adopt the overall accuracy and class-wise F1 score as metrics.

2) Clustering: With the provided set of out-of-sample images, we can generate their feature embeddings based on $\mathcal{F}(\cdot)$. Their quality can be assessed by applying a clustering task, such as K-means clustering. If the intra-class features are close and the inter-class features are separated in the metric space, they can be well clustered, and the clustered labels can accurately match the ground-truth semantic labels. For the evaluation of clustering performance, the first measure that we use is the Normalized Mutual Information (NMI) [62], defined as:

$$\mathrm{NMI} = \frac{2 \times I(\mathbf{Y}; \mathbf{C})}{H(\mathbf{Y}) + H(\mathbf{C})}, \tag{17}$$

where $\mathbf{Y}$ represents the ground-truth class labels, and $\mathbf{C}$ denotes the cluster labels based on the clustering method. $I(\cdot;\cdot)$ and $H(\cdot)$ represent the mutual information and entropy function, respectively. This metric measures the agreement between the ground-truth labels and the assigned

labels based on the clustering method. We also calculate the unsupervised clustering accuracy as our second metric, formulated by:

$$\mathrm{ACC} = \max_{\mathcal{M}} \frac{\sum_{i=1}^{N} \delta(l_i = \mathcal{M}(c_i))}{N}, \tag{18}$$

where l_i denotes the ground-truth class, c_i is the assigned cluster of image $\mathbf{x}_i$, and $\delta(\cdot)$ represents the Dirac delta function. $\mathcal{M}$ is a function than finds the best mapping between the cluster assigned labels and the ground-truth labels.

3) Image Retrieval: Image retrieval aims to find the most semantically similar images in the archive based on their distances with regards to the query images. Such distance is measured by evaluating the similarity of the feature embdeddings between the query images and the full set of images in the archive in the given metric space. Given the query image, more relevant images can be retrieved based on the feature embeddings generated by a more effective metric learning method. To evaluate the performance in terms of image retrieval, we adopt the Precision-Recall (PR) curve to substantiate the precision and recall metrics with respect to a variable number of retrieved images.

For these tasks, we randomly select 70% of the benchmark data for training, 10% for validation, and 20% for testing. The clustering task is conducted on the feature embeddings of the test sets generated by the learned CNN model. For image retrieval, the test set is served for querying, and the training set is the archive. The proposed method is implemented in PyTorch [63]. The backbone CNN architecture is selected as ResNet18 [56] for all the considered methods. It is worth noting that other CNN architectures, such as ResNet50, can also be applied with the proposed loss and optimization mechanism. For the sake of simplicity, we utilize ResNet18 in this paper. The images are all resized to 256×256 pixels, and three data augmentation methods are adopted during training: 1) *RandomGrayscale*, 2) *ColorJitter*, and 3) *RandomHorizontalFlip*. The parameters D, σ, λ and m are set to 128, 0.1, 1.0 and 0.5, respectively. The Stochastic Gradient Descent (SGD) optimizer is adopted for training. The initial learning rate is set to 0.01, and it is decayed by 0.5 every 30 epochs. The batch size is 256 and we totally train the CNN model for 100 epochs. To validate the effectiveness of the proposed method, we compare it to several state-of-the-art methods based on deep metric learning, including: 1) D-CNN [17], 2) deep metric learning based on triplet loss [52], [54] –simply termed as Triplet hereinafter– and 3) SNCA(MB) [20]. It is worth noting that the original SNCA algorithm is optimized with memory bank, i.e., SNCA(MB). In order to validate the effectiveness of the proposed optimization mechanism, we

also consider our new SNCA(MU) and compare its performance with the original SNCA [20]. For the triplet loss, the margin parameter is selected as 0.2 and the parameters in D-CNN are set to the same values as in the original paper. All the experiments are conducted on an NVIDIA Tesla P100 graphics processing unit (GPU).

C. Experimental Results

1) Classification: Figure 4 plots the curves of classification accuracy versus the number of training epochs obtained for different learning methods, using the KNN classifier (with $K = 10$) as a baseline, and the NWPU-RESISC45 dataset. As Figure 4 shows, in order to achieve an accuracy of 90%, SNCA(MU), SNCA-CE(MB), and SNCA-CE(MU) require less than 20 epochs, while the other tested methods require more than 20 epochs. As the learning curves converge, SNCA(MU), SNCA-CE(MB), and SNCA-CE(MU) reach an accuracy of about 94%, which is around 2% higher than that achieved by the other methods. Among them, the performances of SNCA-CE(MB) and SNCA-CE(MU) are slightly better than that of SNCA(MU), and SNCA-CE(MU) achieves the fastest learning speed. By comparing SNCA-CE[3] with SNCA, the introduction of the CE loss can not only increase the learning speed, but also improve the classification obtained by the KNN classifier.

By comparing the MB and MU optimization mechanisms, we conclude that updating the state of the CNN model can lead to better results than updating the memory bank. We report the overall accuracy of all the methods on the considered test sets in Table I, using various values of K. Consistently with the validation, SNCA-CE(MB) and SNCA-CE(MU) achieve the best classification performance on the two benchmark datasets. Compared with SNCA-CE(MB), the classification accuracy of SNCA-CE(MU) is slightly higher on the NWPU-RESISC45 dataset, while it is slightly lower on the AID dataset. Since the MU optimization mechanism aims at preserving feature consistency among all the mini-batches through each training epoch, its advantage over MB is more obvious in a large dataset such as NWPU-RESISC45. For the AID dataset, there are not many mini-batches within one training epoch, e.g., around 28 when the batch size is 256. The obtained feature embeddings in $\mathcal{B}$ may not vary severely within each training epoch. Thus, the associated performance is comparable with that of the MU mechanism.

[3]For simplicity, SNCA-CE refers to both SNCA-CE(MB) and SNCA-CE(MU), and SNCA refers to both SNCA(MB) and SNCA(MU).

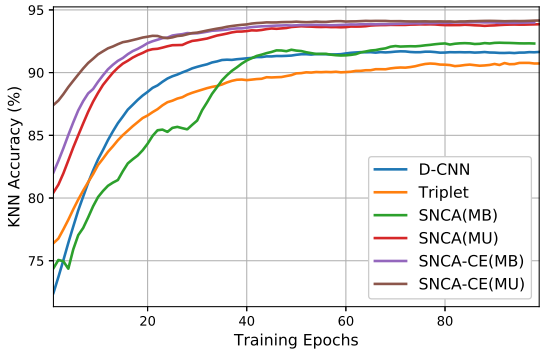


Fig. 4. Learning curves of different methods on the validation set with respect to the number of training epochs (NWPU-RESISC45 dataset). The KNN classification accuracy (%) with $K = 10$ at each epoch is reported.

In turn, SNCA-CE can obtain more accurate performance, with more than 1% improvement compared with SNCA and more than 2% compared to the other two methods. With the adoption of momentum update, SNCA(MU) achieves an accuracy improvement of around 0.5% with regards to SNCA(MB).

Moreover, Table II and Table III show the class-wise F1 scores achieved by the different learning methods (based on the KNN classifier) in the test sets of the AID and NWPU-RESISC45 datasets, respectively, using $K = 10$. For the AID dataset, the F1 score of SNCA-CE(MB) on *Resort* class achieves more than 5% performance gain than the other methods. For the NWPU-RESISC45 dataset, we can see that the performances of most classes obtained by SNCA-CE are the best ones when compared with the others.

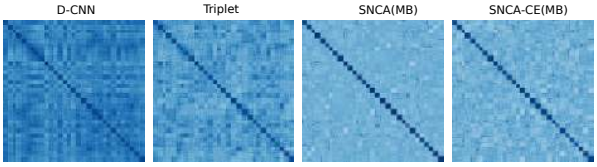
In addition, Figure 5(a) and Figure 5(b) illustrate the similarities of the feature embeddings generated by D-CNN, Triplet, SNCA(MB) and SNCA-CE(MB) on the test sets of the AID and NWPU-RESISC45 datasets, respectively. The similarity is measured by applying the *cosine distance*, i.e. $\mathbf{f}_i^* \mathbf{f}_j^*$. As shown by the obtained similarity matrices, higher color contrast between the diagonal blocks and the background demonstrates higher dissimilarity between the images from one class and those from the others in metric space. In terms of cosine distance, both SNCA(MB) and SNCA-CE(MB) achieve better performances than D-CNN and Triplets when distinguishing between different classes in metric space.

2) Clustering: Table IV displays the NMI scores obtained after applying K-means clustering (with different learning methods) to the feature embeddings of the considered test sets. It can be observed that SNCA-CE achieves the best matching between the ground-truth labels and the

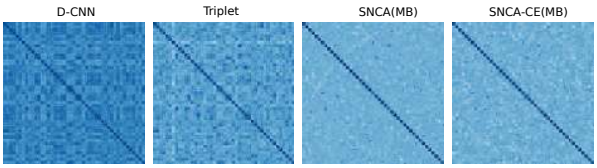
TABLE I

KNN CLASSIFICATION ACCURACIES (%) OBTAINED BY USING DIFFERENT LEARNING METHODS, FOR $K = 1, 5, 10$.

	AID			**NWPU-RESISC45**		
	1	5	10	1	5	10
D-CNN	93.10	93.70	93.75	91.21	91.62	91.48
Triplet	92.85	93.10	93.25	90.83	91.46	91.43
SNCA(MB)	94.55	94.50	94.60	92.13	92.21	92.14
SNCA(MU)	94.55	94.75	94.75	92.57	92.59	92.68
SNCA-CE(MB)	**95.75**	**95.55**	**95.45**	93.84	93.84	93.79
SNCA-CE(MU)	95.15	95.40	95.15	**93.89**	**93.87**	**93.97**



(a)



(b)

Fig. 5. Similarity matrices of the feature embeddings in the metric space obtained by different learning methods. The similarity is measured by the cosine distance. (a) AID and (b) NWPU-RESISC45.

pseudo-labels assigned by K-means clustering, which results in more than 5% performance gain with regards to the D-CNN. Table V reports the associated ACC scores obtained after using different learning methods. Consistent with the NMI results, the K-means clustering based on features generated by SNCA-CE can make the best label assignment unsupervisedly. In order to obtain further insight on the feature embeddings in the metric space, we exploit the t-distributed stochastic neighbour embedding (t-SNE) to visualize their projections in a 2D space. Figure 6 shows the t-SNE scatter plots of the feature embeddings obtained for the AID test set using: (a) D-CNN; (b) Triplet; and (c) SNCA-CE(MB). As illustrated in Figure 6, the intra-class features are more compact and inter-class features are more isolated in the proposed method. As a result,

TABLE II

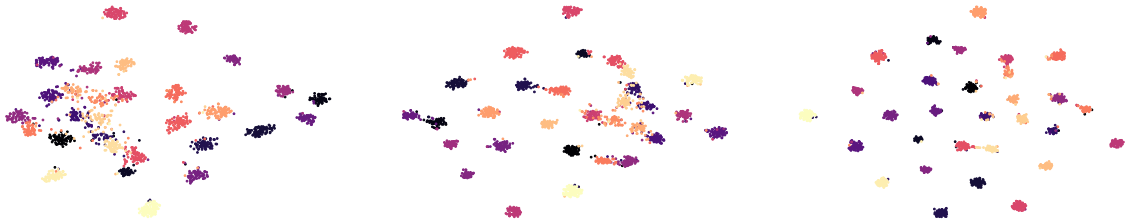
CLASS-WISE F1 SCORES OBTAINED BY THE KNN CLASSIFIER WITH DIFFERENT LEARNING METHODS ON THE AID TEST SET, FOR $K = 10$.

	D-CNN	**Triplet**	**SNCA(MB)**	**SNCA(MU)**	**SNCA-CE(MB)**	**SNCA-CE(MU)**
Airport	94.52	95.17	**97.18**	95.83	94.52	94.52
Bare Land	**95.93**	94.49	88.89	92.06	95.16	94.31
BaseballField	95.56	92.47	96.55	**97.73**	96.55	**97.73**
Beach	98.16	97.50	96.30	98.11	99.37	**100.00**
Bridge	95.77	97.18	**99.30**	98.61	**99.30**	97.90
Center	88.46	85.71	88.00	87.38	**89.11**	88.24
Church	88.66	88.17	94.85	**94.95**	87.38	93.07
Commercial	95.10	93.71	**96.50**	92.86	95.04	95.10
Dense Residential	93.33	94.55	98.18	96.34	**98.80**	96.93
Desert	96.61	93.33	91.67	95.73	**97.48**	**97.48**
Farmland	97.96	97.99	97.30	98.67	98.63	**99.32**
Forest	**100.00**	**100.00**	98.00	**100.00**	**100.00**	**100.00**
Industrial	**93.75**	92.31	91.61	92.81	92.50	93.51
Meadow	98.21	99.10	94.02	98.25	**99.12**	98.25
Medium Residential	94.83	92.04	**97.39**	94.12	**97.39**	94.21
Mountain	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Park	82.99	83.33	85.92	87.14	**89.05**	**89.05**
Parking	**99.35**	**99.35**	98.09	**99.35**	98.72	98.72
Playground	92.81	92.00	96.73	97.33	95.42	**97.37**
Pond	97.01	97.04	95.91	97.04	96.55	**97.08**
Port	93.42	92.31	96.10	95.48	96.15	**97.44**
Railway Station	93.20	93.07	**95.41**	91.89	95.24	93.46
Resort	71.70	70.37	71.84	74.55	**81.48**	75.00
River	96.34	96.93	96.97	97.56	**99.39**	98.78
School	80.67	75.21	**84.75**	80.34	80.36	82.05
Sparse Residential	98.33	98.33	**99.16**	97.48	98.33	98.31
Square	83.33	85.27	89.23	89.39	**90.77**	85.48
Stadium	92.31	93.58	94.92	**97.35**	95.65	97.39
Storage Tanks	95.83	96.50	**97.26**	95.71	96.45	95.04
Viaduct	98.25	98.25	98.81	**99.41**	**99.41**	98.82

TABLE III

CLASS-WISE F1 SCORES OBTAINED BY THE KNN CLASSIFIER WITH DIFFERENT LEARNING METHODS ON THE NWPU-RESISC45 TEST SET, FOR $K = 10$.

	D-CNN	**Triplet**	**SNCA(MB)**	**SNCA(MU)**	**SNCA-CE(MB)**	**SNCA-CE(MU)**
Airplane	96.82	96.86	98.22	98.57	**98.93**	98.23
Airport	91.84	92.15	88.32	92.14	**95.44**	95.41
Baseball diamond	95.00	94.58	97.12	**98.21**	96.45	96.09
Basketball court	92.59	92.94	96.77	96.80	**97.86**	97.16
Beach	94.62	96.77	96.35	96.75	97.16	**98.55**
Bridge	94.58	95.68	95.71	95.68	94.89	**96.73**
Chaparral	97.90	98.94	98.59	98.59	98.94	**99.29**
Church	72.46	71.33	74.26	76.47	**78.57**	76.12
Circular farmland	98.21	98.19	98.22	**99.64**	**99.64**	**99.64**
Cloud	**97.20**	96.55	94.85	96.50	**97.20**	96.55
Commercial area	85.22	81.12	87.32	85.11	**89.45**	88.81
Dense residential	88.00	87.63	87.59	88.81	90.97	**91.58**
Desert	91.51	93.43	92.68	94.37	94.16	**95.37**
Forest	94.48	93.52	93.52	95.80	**96.50**	96.14
Freeway	84.53	87.46	88.06	87.97	89.45	**91.58**
Golf course	95.68	96.38	97.51	95.68	**98.56**	98.23
Ground track field	96.17	96.73	96.84	97.16	98.23	**98.93**
Harbor	98.22	98.56	**98.92**	98.56	98.58	**98.92**
Industrial area	85.02	85.51	85.71	86.11	**87.77**	87.41
Intersection	88.36	92.68	91.17	94.66	94.08	**95.47**
Island	95.41	94.37	94.58	92.14	95.41	**95.77**
Lake	90.78	92.25	88.81	88.44	91.53	**92.73**
Meadow	91.45	90.39	91.76	92.09	**94.93**	94.24
Medium residential	86.11	83.33	83.92	84.10	**86.43**	86.33
Mobile home park	93.57	92.25	**96.14**	95.00	95.77	96.11
Mountain	88.05	91.29	90.34	92.86	92.68	**93.29**
Overpass	93.62	92.58	94.58	91.17	92.53	**93.91**
Palace	72.66	67.18	71.59	73.19	**75.18**	73.19
Parking lot	94.44	95.71	95.74	96.80	96.03	**97.84**
Railway	85.31	81.94	90.03	90.66	91.61	**92.36**
Railway station	86.93	83.87	84.59	**91.43**	88.17	90.00
Rectangular farmland	90.32	89.21	90.65	87.77	91.10	**91.58**
River	88.89	90.04	88.57	90.11	92.36	**93.48**
Roundabout	95.24	95.00	95.07	94.77	96.11	**96.14**



(a) (b) (c)

Fig. 6. 2D projection of the feature embeddings on the AID test set using t-SNE: (a) D-CNN; (b) Triplet; and (c) SNCA-CE(MB).

TABLE IV

NMI SCORES OF THE FEATURE EMBEDDINGS OF THE TEST SETS PRODUCED BY DIFFERENT LEARNING METHODS.

	AID	**NWPU-RESISC45**
D-CNN	88.83	85.30
Triplet	89.87	88.14
SNCA(MB)	92.96	90.20
SNCA(MU)	93.02	90.60
SNCA-CE(MB)	**93.98**	92.01
SNCA-CE(MU)	93.75	**92.28**

clustering methods can more easily discover the inherent structure of the feature embeddings in the metric space produced by the proposed method, resulting in an NMI score that is higher than the one obtained by the other learning methods.

3) Image Retrieval: Figure 7 shows the PR curves describing the obtained image retrieval results from a given test set used for querying, where Figure 7(a) and Figure 7(b) respectively provide the results for the AID, and NWPU-RESISC45 datasets. In order to facilitate the comparison, a zoomed-in subplot is also highlighted. It can be seen that both SNCA and SNCA-CE exhibit superior performance with regards to Triplet and D-CNN as the number of retrieved images increases. As shown in the zoomed-in subplots, the introduction of the CE loss can further improve the precision and recall performances based on SNCA. For the SNCA-based methods (SNCA and SNCA-CE), the similarities of the images within one mini-batch during training are compared with all the other images in the dataset, so that the CNN model can be sufficiently optimized. As a comparison, for the contrastive loss utilized in D-CNN, the negative and positive

TABLE V

ACC SCORES OF THE FEATURE EMBEDDINGS OF THE TEST SETS PRODUCED BY DIFFERENT LEARNING METHODS.

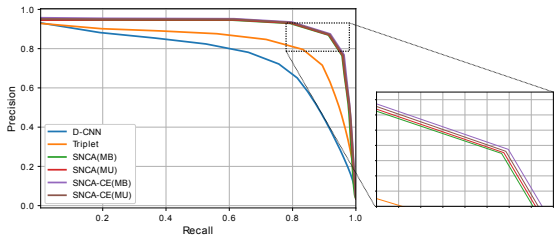
	AID	**NWPU-RESISC45**
D-CNN	84.50	87.44
Triplet	92.50	88.33
SNCA(MB)	94.65	92.13
SNCA(MU)	94.80	91.22
SNCA-CE(MB)	**95.65**	93.71
SNCA-CE(MU)	95.25	**93.83**

image pairs are just sampled within each mini-batch. For the other images outside this mini-batch, the corresponding negative and positive image pairs cannot be constructed, leading to insufficient training of the CNN model.

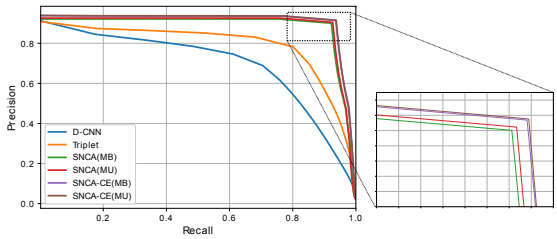
This is actually similar with respect to triplet loss. To make the CNN model capture the similarity and dissimilarity of all the images, one should make a triplet set with about $\mathcal{O}(|\mathcal{T}|^3)$ triplets, which is impossible for a scalable dataset. Such limitation of the trained CNN model based on contrastive and triplet losses may lead to the fact that that some images cannot be well separated with regards to other images with different classes, or that these images cannot be effectively grouped together with their relevant ones. This phenomenon can be observed in Figure 6, where some clusters shown in (a) and (b) are entangled with others. Additionally, this also leads to the important phenomenon that the image retrieval performance that can be achieved using both SNCA and SNCA-CE is superior to that of the methods based on the contrastive and triplet losses. With respect to SNCA, by introducing the CE loss, SNCA-CE can further improve the image retrieval performance, owing to its enhanced class distinction capability.

Figure 8 gives some retrieval examples with D-CNN, Triplet and the proposed method. Given two images from the two test sets, we present their top-5 nearest neighbors in the archive. As shown in Figure 8(a), *Park* and *School* are confused with *Resort* in the Triplet retrieval from the AID dataset. The *freeway* in NWPU-RESISC45 is confused with *overpass* by D-CNN, shown in Figure 8(b).

4) Parameter Sensitivity Analysis of SNCA-CE: There are three main parameters in the proposed methods, i.e., D, σ and λ, where D determines the dimensionality of the feature



(a)



(b)

Fig. 7. PR curves describing the image retrieval results obtained by different learning methods. The test sets are served for querying, and the training sets are used as the archives. (a) AID and (b) NWPU-RESISC45.

embeddings in the metric space, σ controls the compactness of the sample distribution, and λ balances the contributions of two loss terms, i.e., SNCA and CE. Table VI demonstrates the effectiveness of the KNN classification based on SNCA-CE(MB) with respect to different values of D, assuming that $K = 10$. As Table VI shows, the classification performance is robust to different values of D on both datasets. This is greatly beneficial for embedding large-scale datasets, since features with small dimensionality can also achieve high-quality classification performance. Based on the KNN classification results with $K = 10$, we also report the effectiveness of SNCA-CE(MB) in terms of σ in Table VII. Within a range of values from 0.05 to 0.2, the classification results are stable. This suggests that the proposed method is relatively insensitive to the choice of σ (in the range from 0.05 to 0.2) for the two considered datasets. Figure 9 displays a sensitivity analysis of λ in eq. (11). It can be seen that the KNN classification performs worst on the both datasets when λ is near zero, i.e., $\lambda = 0.1$. This indicates that the optimization of SNCA term can indeed improve the metric learning performance. When λ is larger than 0.1, the proposed method shows its insensitivity with respect to the setting of λ.

(a)

(b)

Fig. 8. Top-5 nearest neighbors retrieved with respect to the query images using different learning methods. (a) AID and (b) NWPU-RESISC45.

TABLE VI

SENSITIVITY ANALYSIS OF PARAMETER D.

	AID	**NWPU-RESISC45**
$D = 32$	95.15	94.02
$D = 64$	95.60	94.13
$D = 128$	95.45	93.79

V. CONCLUSIONS

In this paper, we introduce a new deep metric learning approach for RS images which improves scene discrimination by means of two different components: 1) SNCA, which aims at constructing the neighborhood structure in the metric space; and 2) the CE loss, which aims at preserving the class discrimination capability. Moreover, we propose a novel optimization mechanism based

TABLE VII

SENSITIVITY ANALYSIS OF PARAMETER σ.

	AID	**NWPU-RESISC45**
$\sigma = 0.05$	95.05	93.83
$\sigma = 0.1$	95.45	93.79
$\sigma = 0.15$	94.80	93.63
$\sigma = 0.2$	94.90	93.48

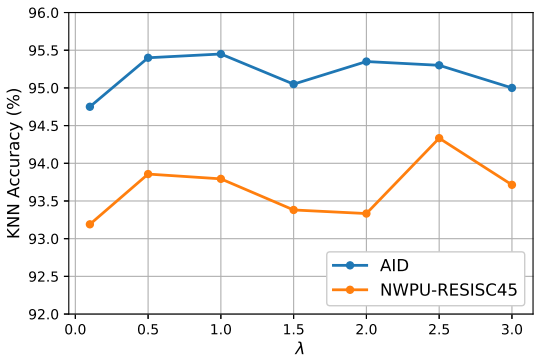


Fig. 9. Sensitivity analysis of the λ parameter.

on momentum update for SNCA and SNCA-CE. This mechanism is intended to preserve the consistency among all the stored features during training, which represents a highly innovative contribution to characterize RS scenes.

The conducted experiments validate the effectiveness of the proposed method from different perspectives, including RS scene classification, clustering, and retrieval. When compared to the state-of-the-art models, the newly defined SNCA-CE loss is able to group semantically-similar RS images better than other existing approaches, due to the effective use of an offline memory bank. Besides, SNCA-CE can further improve the class discrimination ability based on its learnable category prototypes. The proposed MU optimization mechanism also makes the features generated in each mini-batch more consistent within one training epoch than those generated via the MB mechanism. Such characteristic can be greatly beneficial when processing scalable datasets.

In addition to characterizing RS scenes, our newly proposed deep metric learning framework

also exhibits the potential to be used in other tasks, such as dimensionality reduction of RS hyperspectral images and fine-grained land-use or land-cover classification. As a possible future work, one can extensively analyze the influence of different backbone networks (e.g., VGG16, ResNet18, ResNet50, and ResNet101) on the performance of the proposed approach. Additionally, we will explore the adaptation of our method to the aforementioned problems, and also further evaluate its capacity to perform scene classification with limited supervision.

REFERENCES

[1] Y. Ma, H. Wu, L. Wang, B. Huang, R. Ranjan, A. Zomaya, and W. Jie, "Remote sensing big data computing: Challenges and opportunities," *Future Generation Computer Systems*, vol. 51, pp. 47–60, 2015.

[2] C. Corbane, S. Lang, K. Pipkins, S. Alleaume, M. Deshayes, V. E. G. Millán, T. Strasser, J. V. Borre, S. Toon, and F. Michael, "Remote sensing for mapping natural habitats and their conservation status–new opportunities and challenges," *International Journal of Applied Earth Observation and Geoinformation*, vol. 37, pp. 7–16, 2015.

[3] Q. Weng, D. Quattrochi, and P. E. Gamba, *Urban remote sensing*. CRC press, 2018.

[4] J. Li, Z. He, J. Plaza, S. Li, J. Chen, H. Wu, Y. Wang, and Y. Liu, "Social media: New perspectives to improve remote sensing for emergency response," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1900–1912, 2017.

[5] R. Fernandez-Beltran, A. Plaza, J. Plaza, and F. Pla, "Hyperspectral unmixing based on dual-depth sparse probabilistic latent semantic analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 11, pp. 6344–6360, 2018.

[6] N. Joshi, M. Baumann, A. Ehammer, R. Fensholt, K. Grogan, P. Hostert, M. R. Jepsen, T. Kuemmerle, P. Meyfroidt, E. T. Mitchard *et al.*, "A review of the application of optical and radar remote sensing data fusion to land use mapping and monitoring," *Remote Sensing*, vol. 8, no. 1, p. 70, 2016.

[7] D. Bratasanu, I. Nedelcu, and M. Datcu, "Bridging the semantic gap for satellite image annotation and automatic mapping applications," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, no. 1, p. 193, 2011.

[8] C. Gómez, J. C. White, and M. A. Wulder, "Optical remotely sensed time series data for land cover classification: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 116, pp. 55–72, 2016.

[9] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.

[10] Y. Yang and S. Newsam, "Spatial pyramid co-occurrence for image classification," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1465–1472.

[11] E. Aptoula, "Remote sensing image retrieval with global morphological texture descriptors," *IEEE transactions on geoscience and remote sensing*, vol. 52, no. 5, pp. 3023–3034, 2014.

[12] A. M. Cheriyadat, "Unsupervised feature learning for aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 1, pp. 439–451, 2014.

[13] M. L. Mekhalfi, F. Melgani, Y. Bazi, and N. Alajlan, "Land-use classification with compressive sensing multifeature fusion," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 10, pp. 2155–2159, 2015.

[14] J. A. Benediktsson, J. Chanussot, and W. M. Moon, "Very high-resolution remote sensing: Challenges and opportunities [point of view]," *Proceedings of the IEEE*, vol. 100, no. 6, pp. 1907–1910, 2012.

[15] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.

[16] Z. Gong, P. Zhong, Y. Yu, and W. Hu, "Diversity-promoting deep structural metric learning for remote sensing scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 1, pp. 371–390, 2018.

[17] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative cnns," *IEEE transactions on geoscience and remote sensing*, vol. 56, no. 5, pp. 2811–2821, 2018.

[18] L. Yan, R. Zhu, N. Mo, and Y. Liu, "Cross-domain distance metric learning framework with limited target samples for scene classification of aerial images," *IEEE Transactions on Geoscience and Remote Sensing*, 2019.

[19] O. A. Penatti, K. Nogueira, and J. A. dos Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?" in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 44–51.

[20] Z. Wu, A. A. Efros, and S. X. Yu, "Improving generalization via scalable neighborhood component analysis," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 685–701.

[21] B. Rasti, D. Hong, R. Hang, P. Ghamisi, X. Kang, J. Chanussot, and J. A. Benediktsson, "Feature extraction for hyperspectral imagery: The evolution from shallow to deep," 2020.

[22] G. Thoonen, Z. Mahmood, S. Peeters, and P. Scheunders, "Multisource classification of color and hyperspectral images using color attribute profiles and composite decision fusion," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 510–521, 2012.

[23] Z. Li and L. Itti, "Saliency and gist features for target detection in satellite images," *IEEE Transactions on Image Processing*, vol. 20, no. 7, pp. 2017–2029, 2011.

[24] W. Zhang, X. Sun, K. Fu, C. Wang, and H. Wang, "Object detection in high-resolution remote sensing images using rotation invariant parts based model," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 1, pp. 74–78, 2014.

[25] R. Fernandez-Beltran, P. Latorre-Carmona, and F. Pla, "Single-frame super-resolution in remote sensing: a practical overview," *International journal of remote sensing*, vol. 38, no. 1, pp. 314–354, 2017.

[26] R. Fernandez-Beltran, J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza, and F. Pla, "Remote sensing image fusion using hierarchical multimodal probabilistic latent semantic analysis," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2018.

[27] R. Fernandez-Beltran and F. Pla, "Sparse multi-modal probabilistic latent semantic analysis for single-image super-resolution," *Signal Processing*, vol. 152, pp. 227–237, 2018.

[28] L. Zhang, Q. Zhang, B. Du, X. Huang, Y. Y. Tang, and D. Tao, "Simultaneous spectral-spatial feature selection and extraction for hyperspectral images," *IEEE Transactions on Cybernetics*, vol. 48, no. 1, pp. 16–28, 2016.

[29] L. Zhang, L. Zhang, B. Du, J. You, and D. Tao, "Hyperspectral image unsupervised classification by robust manifold matrix factorization," *Information Sciences*, vol. 485, pp. 154–169, 2019.

[30] E. Li, P. Du, A. Samat, Y. Meng, and M. Che, "Mid-level feature representation via sparse autoencoder for remotely sensed scene classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 3, pp. 1068–1081, 2017.

[31] F. Lv, M. Han, and T. Qiu, "Remote sensing image classification based on ensemble extreme learning machine with stacked autoencoder," *IEEE Access*, vol. 5, pp. 9021–9031, 2017.

[32] G. Cheng, Z. Li, J. Han, X. Yao, and L. Guo, "Exploring hierarchical convolutional features for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 11, pp. 6712–6722, 2018.

[33] X. Yao, J. Han, G. Cheng, X. Qian, and L. Guo, "Semantic annotation of high-resolution satellite images via weakly supervised learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 6, pp. 3660–3671, 2016.

[34] X. Lu, X. Zheng, and Y. Yuan, "Remote sensing scene classification by unsupervised representation learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 9, pp. 5148–5157, 2017.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[38] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sensing*, vol. 7, no. 11, pp. 14 680–14 707, 2015.

[39] S. Chaib, H. Liu, Y. Gu, and H. Yao, "Deep feature fusion for vhr remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens*, vol. 55, no. 8, pp. 4775–4784, 2017.

[40] F. Zhang, B. Du, and L. Zhang, "Scene classification via a gradient boosting random convolutional network framework," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 3, pp. 1793–1802, 2016.

[41] E. Li, J. Xia, P. Du, C. Lin, and A. Samat, "Integrating multilayer features of convolutional neural networks for remote sensing scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 10, pp. 5653–5665, 2017.

[42] G. Cheng, Z. Li, X. Yao, L. Guo, and Z. Wei, "Remote sensing image scene classification using bag of convolutional features," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1735–1739, 2017.

[43] J. Kang, M. Körner, Y. Wang, H. Taubenböck, and X. X. Zhu, "Building instance classification using street view images," *ISPRS journal of photogrammetry and remote sensing*, vol. 145, pp. 44–59, 2018.

[44] K. Nogueira, O. A. Penatti, and J. A. dos Santos, "Towards better exploiting convolutional neural networks for remote sensing scene classification," *Pattern Recognition*, vol. 61, pp. 539–556, 2017.

[45] J. Hu, J. Lu, and Y.-P. Tan, "Deep transfer metric learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 325–333.

[46] A. Li, Z. Lu, L. Wang, T. Xiang, and J.-R. Wen, "Zero-shot scene classification for high spatial resolution remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 4157–4167, 2017.

[47] H. Oh Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4004–4012.

[48] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1735–1742.

[49] Y. Tian, C. Chen, and M. Shah, "Cross-view image matching for geo-localization in urban environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3608–3616.

[50] G. Cheng, P. Zhou, and J. Han, "Duplex metric learning for image set classification," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 281–292, 2017.

[51] J. Han, G. Cheng, Z. Li, and D. Zhang, "A unified metric learning-based framework for co-saliency detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 10, pp. 2473–2483, 2017.

[52] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[53] J. Wang, P. Virtue, and S. X. Yu, "Successive embedding and classification loss for aerial image classification," *arXiv preprint arXiv:1712.01511*, 2017.

[54] R. Cao, Q. Zhang, J. Zhu, Q. Li, Q. Li, B. Liu, and G. Qiu, "Enhancing remote sensing image retrieval with triplet deep metric learning network," *arXiv preprint arXiv:1902.05818*, 2019.

[55] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh, "No fuss distance metric learning using proxies," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 360–368.

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[57] S. Song, H. Yu, Z. Miao, Q. Zhang, Y. Lin, and S. Wang, "Domain adaptation for convolutional neural networks-based remote sensing scene classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 8, pp. 1324–1328, 2019.

[58] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R. Salakhutdinov, "Neighbourhood components analysis," in *Advances in neural information processing systems*, 2005, pp. 513–520.

[59] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[60] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," *arXiv preprint arXiv:1911.05722*, 2019.

[61] G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "Aid: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3965–3981, 2017.

[62] H. Schütze, C. D. Manning, and P. Raghavan, "Introduction to information retrieval," in *Proceedings of the international communication of association for computing machinery conference*, 2008, p. 260.

[63] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.