A Unified Multiple Proxy Deep Metric Learning Framework Embedded With Distribution Optimization for Fine-Grained Ship Classification in Remote Sensing Images

Jianwen Xu and Haitao Lang, *Member, IEEE*

***Abstract*—Improving ship classification performance in remote sensing imagery by deep metric learning (DML) is a newly emerging research topic and has good application prospects. From the perspective of the use of metric loss (classification loss and pairwise loss) and the way of proxy learning (a single proxy or multiple proxies), this study summarizes the existing DML methods into four representative frameworks and proposes a novel framework, namely, a unified multiple proxy deep metric learning framework embedded with distribution optimization (UMP+D). Specifically, the UMP+D not only unifies the combination of classification loss and pairwise loss into a single loss function containing only pairwise representation but also fuses it with multiple proxy learning. In addition, a distribution loss branch is embedded in the UMP+D to refine the distribution of samples in the feature embedding space to further tighten the intraclass samples and pull apart the interclass samples. Extensive experiments on two optical remote sensing datasets and one synthetic aperture radar dataset demonstrate that the proposed UMP+D framework outperforms the existing frameworks and achieves state-of-the-art performance.**

***Index Terms*—Deep metric learning (DML), deep neural networks (DNN), fine-grained ship classification, optical remote sensing (ORS), synthetic aperture radar (SAR).**

I. Introduction

As WE enter the era of Big Data, the amount of remote sensing images, including optical and synthetic aperture radar (SAR) images, available to us has grown substantially both in quality (e.g., higher resolution) and quantity (e.g., more data sources) [1], [2], [3]. The development of maritime ship monitoring technology has progressed from initial ship detection to later coarse-grained ship classification, and now to the present fine-grained ship classification. In light of this situation, supervised image classification algorithms utilizing deep learning, specifically deep neural networks (DNN) [4], [5], [6], demonstrate excellent prospects for fine-grained ship classification in remote sensing image and have emerged as the current research hotspot [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23].

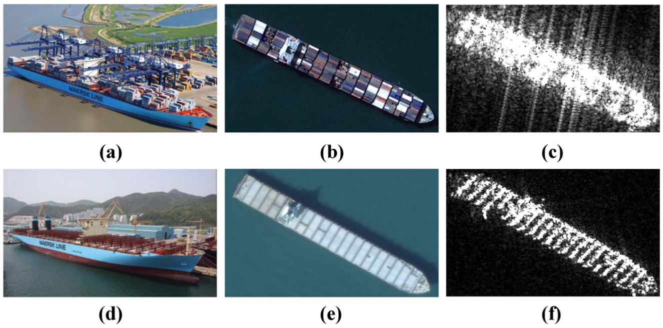


Fig. 1. Fully loaded (top row) and empty (bottom row) container ships exhibit distinct characteristics in natural images [(a) and (d) of left column], optical remote sensing images [(b) and (e) of middle column], and SAR images [(c) and (f) of right column].

One crucial aspect to consider is that the existing sophisticated supervised image classification algorithms based on deep learning are primarily intended for natural images and have been adapted for use in remote sensing image classification tasks. Despite this adaptation, it should be noted that due to differences in imaging sensors, mechanisms, geometry, etc. [24], [25], remote sensing images (both optical and SAR) exhibit distinct characteristics that differentiate them from natural images, as evidenced by Fig. 1. As is commonly known, natural images are usually taken by a camera at a relatively close distance, often with either a front or side view, and possess high resolution, allowing for an abundance of detail about the ship target [Ref. Fig. 1(a) and (d)]. In contrast, remote sensing images are usually acquired from a considerable distance, as indicated by their name, and present a top–down view [Ref. Fig. 1(b) and (e) and (c) and (f)]. Given the distinctions between natural images and remote sensing images, remote sensing images typically provide limited information about the superstructure of the ship target and have a reduced resolution compared with natural images.

Manuscript received 29 August 2023; revised 15 December 2023; accepted 9 February 2024. Date of publication 13 February 2024; date of current version 6 March 2024. This work was supported by the National Natural Science Foundation of China under Grant 62071030. *(Corresponding author: Haitao Lang.)*

Jianwen Xu is with the College of Mathematics and Physics, Beijing University of Chemical Technology, Beijing 100029, China.

Haitao Lang is with the College of Mathematics and Physics, Beijing University of Chemical Technology, Beijing 100029, China, and also with the Beijing Key Laboratory of Environmentally Harmful Chemicals Analysis, Beijing University of Chemical Technology, Beijing 100029, China (e-mail: haitaolang@hotmail.com).

Digital Object Identifier 10.1109/JSTARS.2024.3365472

This makes it more challenging and complex to extract discriminative ship representation features from remote sensing images. Consequently, achieving fine-grained ship classification in remote sensing images remains an an open area of research [10], [11].

The existing supervised learning methods based on DNNs mainly focus on three aspects in order to improve the performance of fine-grained ship classification in remote sensing images. These aspects include:

1) more effective network architecture [8], [12], [13], [21], [23], [26], [27];
2) more comprehensive feature fusion [14], [22], [28], [29]; and
3) more optimized objective function [9], [10], [11], [26], [27], [30].

This study endeavors to improve the performance by refining the objective function through deep metric learning (DML) [31], and leads to the development of a unified multiple proxy deep metric learning framework embedded with distribution optimization (UMP+D). Extensive experiments have confirmed that the proposed UMP+D significantly outperforms existing methods and achieves state-of-the-art (SOTA) performance in fine-grained ship classification in remote sensing images.

The major contributions of this study are three-fold.

First, according to the use of two types of elemental loss (i.e., classification loss and pairwise loss) and proxy-based classification learning (i.e., SP and MPs), which is a new taxonomic perspective, we categorized existing DML methods into four representative frameworks: Single proxy (SP) classification learning, multiple proxy (MP) classification learning, single proxy classification learning combined pairwise learning (SP&P), and multiple proxy classification learning combined pairwise learning (MP&P). We have presented the principle models of these frameworks and the canonical formulas of their corresponding metric loss functions. Our goal is to offer useful insights for future researchers in this field.

Second, we developed a unified multiple proxy DML framework embedded with distribution optimization (UMP+D). This framework integrates classification loss and pairwise loss into a single loss function containing only pairwise representation and fuses it with MPs learning. Furthermore, we have included a distribution loss (DL) branch to refine the distribution of samples in the feature embedding space, further tightening the intraclass samples and separating the interclass samples.

Third, we conducted extensive experiments comparing the proposed UMP+D framework with existing ones (SP, MP, SP&P, and MP&P). The results demonstrate that the UMP+D is superior to the others and achieves the highest level of SOTA performance.

II. Related Work

A. Fine-Grained Ship Classification

The fine-grained ship classification task aims to classify ships into specific categories, such as bulk carriers, container ships, oil tankers, or even different types of warships [10], [15], [16], [21], [32], [33]. To address the unique challenges posed by fine-grained ship classification in remote sensing images, researchers have conducted numerous studies across three main areas.

1) Network Architecture: Many DNNs and their variants, which have achieved notable success in related fields, are employed for ship classification in remote sensing images. Liu et al. [26] improved the inceptionV3 network by adding a fully connected layer to the original network architecture for obscured ship classification. Li et al. [8] designed a prototypical structure network to conduct feature extraction for the purpose of reducing the computational cost in the distance metric calculation. The network is made up of four convolutional blocks used to extract visual information from the original images automatically, and three fully connected layers paired with the rectified linear unit (ReLU) to create a nonlinear Softmax classifier. In addition, to learn a nonlinear distance metric automatically, this study also proposed an end-to-end relation network. Zhao and Lang [21] created a dual-branch network that utilizes ResNet-50 as the deep branch and ResNet-18 as the shallow branch to enhance the performance of the subdomain adaptation. To extract more discriminative deep transferable features, they also embedded the convolutional block attention module after the first and last convolutional layer of each branch. He et al. [27] proposed a group of bilinear convolutional neural network (GBCNN) to extract discriminative ship representations from the pairwise vertical–horizontal polarization and vertical–vertical polarization SAR images. Zhang and Zhang [13] proposed a Laplacian pyramid network with the squeeze-and-excitation (SE) attention mechanism. Chen et al. [23] proposed a push-and-pull network (P^2Net), where a dual-branch network architecture is adopted includes a "push-out stage" forces all the instances to be decorrelated and a "pull-in stage" groups them into each subclass, while an integration module is designed to aggregate the decorrelated images into their corresponding subclass together with a proxy-based module designed for acceleration.

2) Feature Fusion: Zhang et al. [28] proposed a multilevel enhanced feature representation that fuses two local levels feature to represent the "symbol" of a particular category of the ships, and one global level feature to reveal the senior semantic information. Zhang et al. [14] combined the traditional handcrafted histogram of oriented gradient features and modern abstract convolutional neural network features to improve ship classification accuracy. Xiong et al. [22] proposed an explainable attention network (EAN) to seek to increase attention to discriminative parts of objects and explore intrinsic relationships between multiple attention parts and predicted outcomes. It differs from other networks mainly in two modules: Causal multihead attention model (CMAM) and filter aggregation mechanism (FAM). CMAM is used to generate multiple causal attention maps. The multiple causal attention maps will be fused with the original features and, then, fed into the final group of convolution filters. The FAM in the final group of convolution filters will provide the explainable information for the convolutional filter training process.

3) Objective Function: The design of appropriate objective function that can enhance the discriminative power is one of the main challenges for fine-grained ship classification. He et al. [9]

introduced DML to improve the intraclass compactness and interclass separation for medium-resolution SAR ship classification for the first time. They adopted triplet loss and Cross Entropy Loss jointly to optimize a triplet network and proposed to use the fisher discrimination regularization term to explore the global information of learned feature embeddings. Xu and Lang [10] proposed an interclass distribution shift regularization term to improve the original Laplacian regularized metric learning framework to further improve the discriminative ability of ship feature representations. In their later work, a geometric transfer metric learning (GTML) method was proposed. GTML achieves discriminative information preservation, and geometric structure preservation (GSP) handles the domain shift simultaneously by integrating pairwise constraints, joint distribution adaptation, and manifold regularization into a unified optimization function, aiming to make full use of their complementarity to improve SAR ship classification performance [11]. Zeng et al. [30] proposed hybrid channel feature loss for dual-polarized SAR ship classification. The loss contains three terms: The first term is the Cross Entropy Loss for classification learning, the second term enforces the channels of the feature map to be class-aligned, and the third term ensures that the feature vectors within the same category are also diverse enough. In addition, Liu et al. [26] enhanced the objective function by adding a center loss to reduce the distance between classes. He et al. [27] constructed the multipolarization fusion loss to fully explore the dual polarization information.

B. *Deep Metric Learning*

A fundamental assumption of the classification task is that data/image belonging to the same class is somewhat similar, while data belonging to different classes are less similar. And this similarity can be measured in a space (such as a feature space, a transformation space, or an embedding space) in the form of some distance metric [34], [35].

The Euclidean, Mahalanobis, Matusita, Bhattacharyya, and Kullback–Leibler are fundamental distance (similarity) metrics used for data/image classification [34]. However, these predefined distance metrics have limited capabilities in data/image classification, because distance metrics do not have a good learning ability independent of the problem itself. To address this problem, metric learning presents a new way to learn a new distance metric by analyzing data instead of adopting a predefined one. By using similarity relationships between samples, this new distance metric provides a new data representation that has more meaningful and powerful discrimination. The main purpose of metric learning aims to learn a new metric to reduce the distances between samples of the same class and increase the distances between the samples of different classes. That is to say, metric learning aims to bring similar samples closer, and dissimilar samples farther.

In recent years, metric learning and deep learning have been brought together to introduce the concept of DML [31] and have significantly boosted the performance of scene classification of remote sensing images [36]. DML usually does not use a generalized Mahalanobis distance but learns an embedding space using a DNN. The sampling strategy, architecture of DNN, and metric loss function are three factors to be considered as a whole for DML. Among them, the metric loss function is widely recognized as the most important one. Generally, DML aims to obtain discriminative feature representations by specially designing the metric loss function to make a small intraclass distance and a large interclass distance in embedding space. Various metric loss functions exist for DML. Those loss functions can teach the DNN to separate different classes in the embedding space. In general, the metric loss functions are usually divided into two categories, i.e., *classification loss* and *pairwise loss*, which are traditionally thought to be distinct but have been shown to have a unified form by recent studies.

1) Classification Loss: Softmax Loss is one of the most basic classification losses and is composed of Softmax function and Cross Entropy Loss. Its full name is Softmax with Cross Entropy Loss. As its name suggests, the Softmax function breaks the whole (sum to unit one) into different elements with probability rather than selects a maximal value: Maximal element getting the largest portion of the distribution whereas other smaller elements get a relatively small value of it as well. This property of the Softmax function, which generates a probability distribution makes it suitable for probabilistic interpretation in classification tasks. Cross Entropy Loss requires Softmax or sigmoid activation function at the last layer of a DNN so the output values are ranging from zero to one. Minimizing Cross Entropy Loss separates classes for classification. It is theoretically justified in [37] that the Cross Entropy is an upper-bound on the metric learning losses so its minimization for classification also provides embedding features. To address the problem of Softmax Loss, namely, it is good at optimizing the interclass variance, but is unable to reduce the intraclass variation, Wang et al. [38] proposed AM-Softmax Loss, which imports the angular margin into the target logit of Softmax Loss with feature and weight normalized. After normalization, features with small norms will get a much bigger gradient compared with features that have big norms. By back-propagation, the network will pay more attention to the low-quality data/image, which usually has small norms. In addition, AM-Softmax optimizes the multiplicative margin in L2-Softmax [39] and A-Softmax [40] into the additive margin to improve the efficiency of optimization. ArcFace is another additive angular margin loss proposed by Deng et al. [41]. Unlike AM-Softmax, the sign before the parameter of additive angular margin penalty in ArcFace Loss function is positive ("+") instead of negative ("−"). Proxy-NCA Loss [42] is the neighborhood components analysis (NCA) Loss [43] but using a class proxy in order to accelerate computation and make it memory-efficient. It defines some proxies in the embedding space of a DNN and utilizes them in the original NCA loss. The proxies are representatives of classes in the embedding space and they can be defined in various ways. The most straightforward method is to define a proxy for each class and calculate the proxy of every class as the mean of embedded points of that class. In real-world applications, one class may contain several local clusters rather than a single one, a SP of each class may sometimes not represent the class well. To address this problem, Qian et al. [44] proposed softtriple loss to expand a SP per class to MPs per class.

2) Pairwise Loss: Various loss functions use pairs or triplets of samples to push the positive sample toward the anchor point and pull the negative sample away from it. Doing this iteratively for all pairs or triplets will make the intraclass variances smaller and the interclass variances larger for better discrimination of classes. Contrastive Loss [45] uses the anchor-positive and anchor-negative pairs of samples. The samples in an anchor-positive pair are similar, and the samples in an anchor-negative pair are dissimilar. Its loss function contains two terms. The first term minimizes the embedding distances of similar samples and the second term maximizes the embedding distances of dissimilar samples. It tries to make the distances of similar samples as small as possible to keep the intraclass distance as small as possible, and the distances of dissimilar samples at least greater than a margin to increase the interclass distance. Triplet loss [46] uses the triplets of anchor-positive-negative points. This loss makes the distances between the negative and anchor greater than the distances between the positive and anchor by at least a margin. The sampling and training strategy of Contrastive Loss and triplet loss cannot take full advantage of the training batches used during the minibatch stochastic gradient descent training of the DNN, since it first takes randomly sampled pairs or triplets to construct the training batches and compute the loss on the individual pairs or triplets within the batch. To make full use of the batch, one key idea is to enhance the minibatch optimization to use all $O(N^2)$ pairs in the batch, instead of $O(N)$ separate pairs. In view of this, lifted structure loss [47] proposes to lift the vector of pairwise distances $[O(N)]$ within the batch to the matrix of pairwise distances $[O(N^2)]$. It samples a few positive pairs at random, and then, actively adds their difficult neighbors to the training minibatch.

3) Difference Between Classification Loss and Pairwise Loss: The difference between the two kinds of losses stems from three aspects: In use of samples, calculation of the similarity matrix, and basic classification ability.

1) *Sampling strategy:* The sampling strategy is the way that samples are input in the network and the relationship between them. Classification loss utilizes the samples with the class-level labels one by one, and pushes each sample into its corresponding class. While for pairwise loss, samples are used in pairs (or triplets) with pairwise labels. To make the pairs or triplets, every sample is considered as the anchor point. Then, one of the similar/dissimilar samples to the anchor point is taken as the positive/negative point. If class labels are available, one can use them to find the positive point as one of the points in the same class as the anchor point and to find the negative point as one of the points in a different class from the anchor point's class.
2) *Similarity matrix:* There are obvious differences between the two elemental supervised representation learning approaches, i.e., classification learning and pairwise learning. As shown in Fig. 2, the similarity matrix of classification learning is the product of embedding vectors (a matrix with the size of $K \times D$) and a proxy matrix whose size is $D \times C$ matrix. The resulting matrix dimension is $K \times C$, where K is the size of the minibatch and C is the number of classes [48]. Based on this output, K samples in the minibatch can be classified into the corresponding classes one by one. In contrast, the similarity matrix of pairwise learning is the product of the embedding matrix and its transpose, leading to a $K \times K$ dimensional similarity matrix.

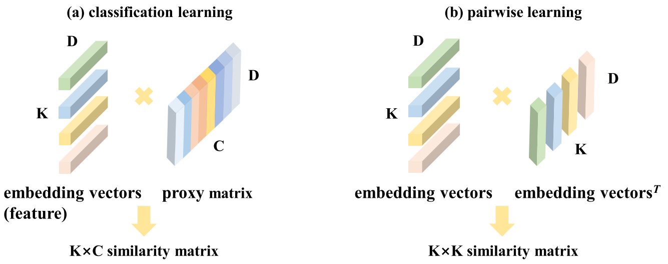


Fig. 2. Calculation of the similarity matrix of (a) classification learning and (b) pairwise learning. D is the dimension of the embedding, K is the size of the minibatch, and C is the number of classes [48].

3) *Classification ability:* In general, the DNN trained with classification loss focuses on the global discriminative information, the features extracted by the network generally stay on a separable feature that just distinguishes different classes. Whereas the DNN trained with pairwise loss focuses on the local discriminative information, the features are more discriminative than the classification learning [49]. In the field of remote sensing image classification, it is difficult to deal with extremely heterogeneous data/images by using classification learning or pairwise learning alone. Therefore, researchers propose the method of combining classification loss and pairwise loss [9].

4) Unity of Classification Loss and Pairwise Loss: Although classification learning and pairwise learning seem to be quite different, they actually have similar optimization modes in terms of loss functions. Qian et al. [44] demonstrated that Softmax Loss is equivalent to a smoothed triplet loss where each class has a SP. Sun et al. [48] found that the loss functions of classification loss and pairwise loss are actually designed to reduce the minimum optimization unit of $S_n - S_p$, where S_n is the similarity of the negative pair, S_p is the similarity of the positive pair. Based on this analysis, they unified the mathematical form of these two types of losses and proposed Circle Loss.

III. Methodology

In this section, we first categorize existing DML methods into four representative frameworks according to the use of metric loss (i.e., classification loss or/and pairwise loss), and the way of proxy classification learning (i.e., SP or MPs), which is a new taxonomic perspective. Next, we describe the proposed prototype framework which is designed to improve the existing framework by adding the DL into the objective function, followed by the UMP+D framework, which is the refinement of the prototype framework based on a unified perspective of classification loss and pairwise loss.

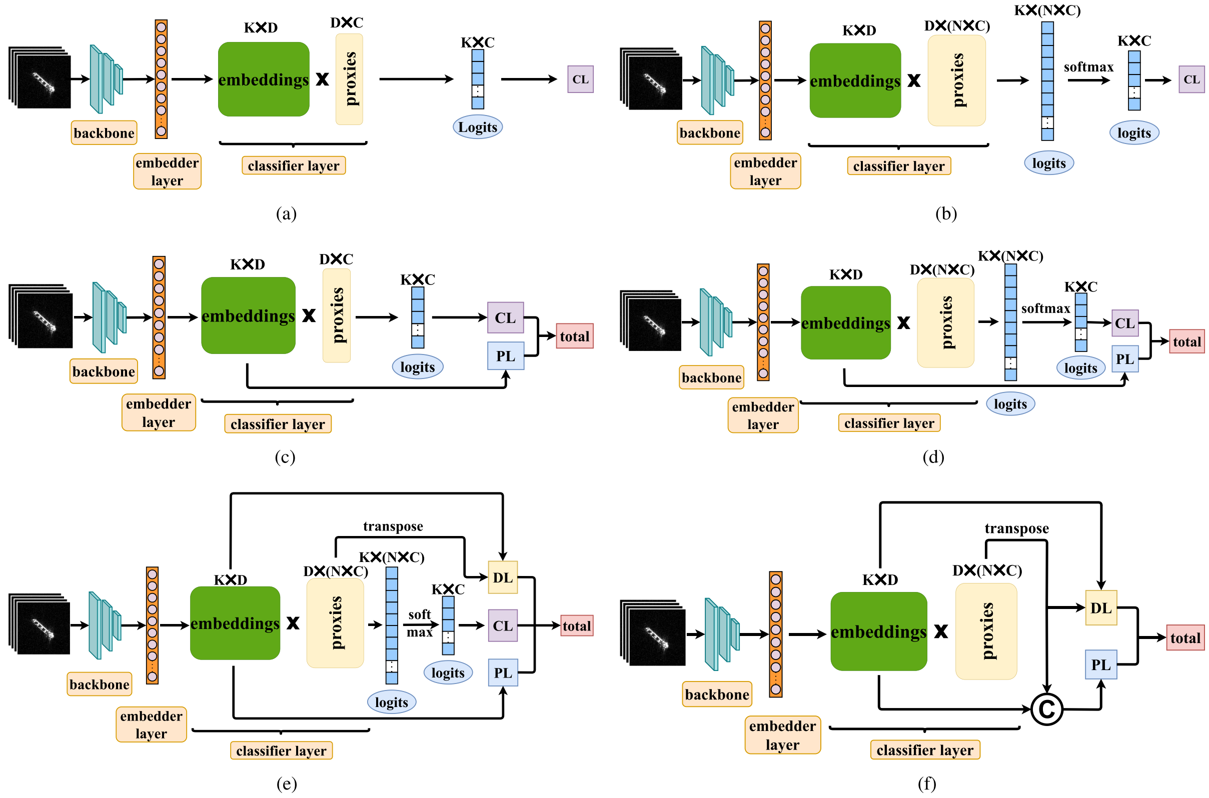


Fig. 3. Various DML frameworks are induced according to the use of metric loss (i.e., classification loss or/and pairwise loss), and the way of proxy learning (i.e., a SP or MPs). (a) SP classification learning framework. (b) MP classification learning framework. (c) SP classification learning combined pairwise learning framework (SP&P), (d) MP classification learning combined pairwise learning framework (MP&P). (e) Prototype of the proposed framework, i.e., MP classification learning combined pairwise learning framework embedded with distribution optimization (MP&P+D). (f) Proposed UMP+D framework, i.e., unified multiple proxy DML framework embedded with distribution optimization (UMP+D). Legend: K is the batch size, D is the embedding dimension, C is the number of classes, and N is the number of proxies in each class. ©represents concatenate operation. CL, PL, and DL denote classification loss, pairwise loss, and distribution loss, respectively. (a) SP. (b) MP. (c) SP&P. (d) MP&P. (e) Prototype framework. (f) UMP+D.

A. Four Representative Frameworks Induced From Existing DML Methods

1) SP Classification Learning: As shown in Fig. 3(a), in the SP framework, each class has a proxy that is embedded in the classifier layer as weight parameters. The main difference between this framework and a typical DNN classification framework is that the SP's classifier layer only contains weight parameters and no bias parameters. The input images are first encoded as features through the backbone network, which serves as a feature extractor. These features are then projected by the embedder layer into a high-dimensional embedding space, where they become the embeddings. The embeddings are fed into the classifier layer, which outputs a classification score for each class (e.g., a similarity matrix or logits). The logits are used to calculate the classification loss. The SP framework's total loss function is composed of only the classification loss, as follows:

$$L_{\text{total}} = L_{\text{cls}}(S_{\text{cls}}, Y_{\text{cls}}) \tag{1}$$

where S_{cls} is the similarity matrix (i.e., logits), which is the output of the classifier layer, and Y_{cls} is the corresponding classification label matrix.

2) M P Classification Learning: The MP framework is induced from the work of Qian et al. [44]. As shown in Fig. 3(b), the classifier layer of the MP framework is extended to a MPs way, where each class has N proxies (with the same number for all classes). The resulting similarity matrix has dimensions $K \times (N \times C)$, and a "softmax" operation is applied to compute the soft maximum values of the similarities between the input sample and its proxies from the same class. The specific calculation is as follows:

$$s'_{x_i,p_y^n} = \sum_{n=1}^{N} \frac{\exp(\frac{1}{\gamma} s_{x_i,p_y^n})}{\sum_{n=1}^{N} \exp(\frac{1}{\gamma} s_{x_i,p_i^n})} s_{x_i,p_y^n} \tag{2}$$

where x_i is the ith sample in the minibatch, p_y^n is the nth proxy in the class y $(y \in \{1, \ldots C\})$, and γ is a parameter to control the smooth scale of the function. s_{x_i,p_y^n} is an element of similarity matrix S_{cls}, which is the output of the classifier layer and has the size of $K \times (N \times C)$, s'_{x_i,p_y^n} is the element of the final logits S' with the size of $K \times C$. The total loss of the MP framework can be written as

$$L_{\text{total}} = L_{\text{cls}}(S', Y_{\text{cls}}). \tag{3}$$

The literature [44] adopted AM-Softmax Loss [38] as the classification loss and extended it to a MPs form

$$L_{\text{total}} = -\log \frac{\exp(\lambda(s'_{x_i,p_y^n} - \delta))}{\exp(\lambda(s'_{x_i,p_y^n} - \delta)) + \sum_{y \neq y_i} \exp(\lambda s'_{x_i,p_y^n})} \tag{4}$$

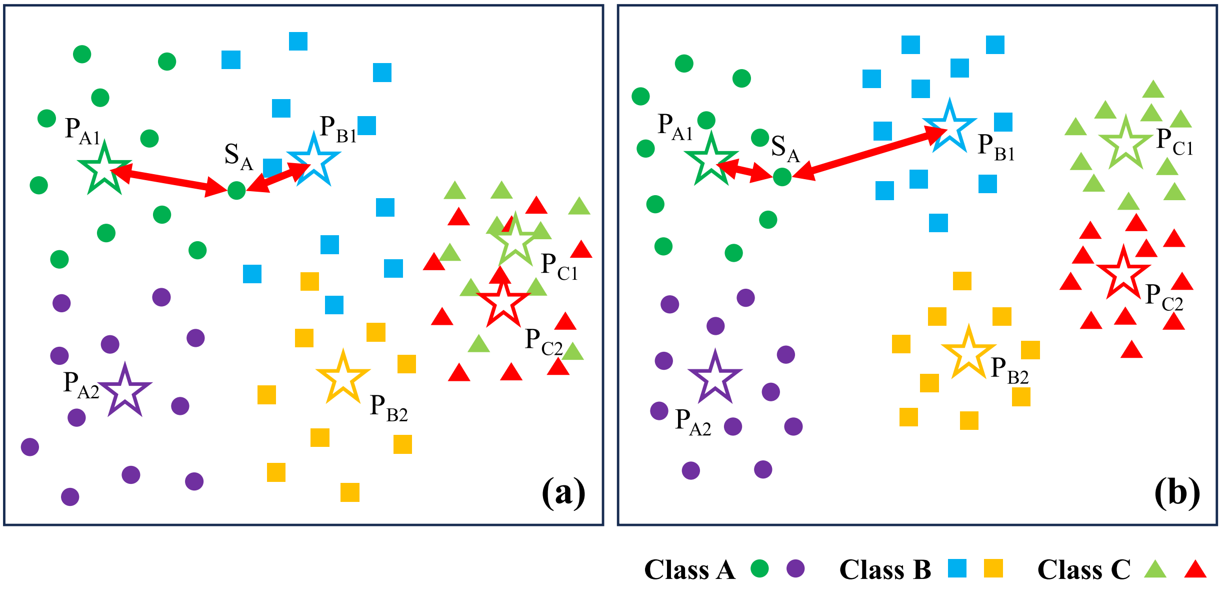


Fig. 4. Motivation of distribution optimization. (a) Problem not solved by the existing MP and MP&P frameworks. (b) Distribution optimization can improve the intraclass tightness and increase the interclass separateness. This diagram illustrates that there are three classes of data (circle, square, and triangle), which need to be classified. Each class contains two local clusters (e.g., fully loaded and empty container ship), which are depicted in different colors. The five-pointed star represents a proxy for each of local clusters.

where λ is a scale parameter and δ is a margin parameter.

3) SP Classification Learning Combined Pairwise Learning (SPand P): He et al. [9] proposed to combine the Cross Entropy Loss and the triplet loss to incorporate the advantages of the classification loss and pairwise loss. The core idea of this framework is shown in Fig. 3(c) and is termed as SP&P. The total loss of SP&P is also formulated by integrating classification loss (L_{cls}) and pairwise loss (L_{pair}) as follows:

$$L_{\text{total}} = \theta L_{\text{cls}}(S_{\text{cls}}, Y_{\text{cls}}) + (1-\theta)L_{\text{pair}}(S_{\text{pair}}, Y_{\text{pair}}) \quad (5)$$

where S_{pair} denotes the pairwise similarity matrix, and Y_{pair} is the corresponding pairwise label matrix. $0 < \theta < 1$ is the weight to tradeoff the relative importance of the two loss terms. The authors recommend the θ high enough, e.g., larger than 0.5, to lay more stress on the supervised classification information.

4) M P Classification Learning Combined Pairwise Learning (MP&P): The framework of MP&P integrates the advantages of MP and SP&P as shown in Fig. 3(d). The improvement of MP&P compared with SP&P is that it upgrades the SP to multiple proxies. The total loss of MP&P is formulated as

$$L_{\text{total}} = \theta L_{\text{cls}}(S', Y_{\text{cls}}) + (1-\theta)L_{\text{pair}}(S_{\text{pair}}, Y_{\text{pair}}). \quad (6)$$

B. Prototype Framework

When examining the characteristics of ship classification in remote sensing images, one may observe that a particular class of ships may consist of several localized clusters instead of a single one. This is illustrated in Fig. 1, where an empty container ship (bottom row) and a fully loaded container ship (top row) have distinct superstructure appearances, resulting in separate clusters in the feature/embedding space. To address this issue, a MPs framework is more suitable since it can capture the differences between localized clusters and help to reduce intraclass variance. On the other hand, as mentioned in Section II-B, classification learning (i.e., proxy learning) is effective for capturing global information, whereas pairwise learning focuses on local information. It is reasonable to use the complementary nature of these two types of learning methods to enhance ship classification performance. Therefore, the MP&P framework is particularly well-suited for fine-grained ship classification from a theoretical perspective.

We have studied both theoretically and experimentally the MP and MP&P frameworks and found that there is still room for improvement. Due to the heterogeneity of ship data in remote sensing images, the distribution of data in feature or embedded space is highly mixed, which greatly reduces the representation power of proxies. Fig. 4(a) illustrates this situation: The proxies P_{A1} and P_{B1} belong to class A and B, respectively, but the distance between the sample at the edge of class A and P_{A1} is even greater than the distance between the same sample and P_{B1}; on the other hand, two proxies P_{C1} and P_{C2} are almost indistinguishable, losing the benefit of MPs.

To address this problem, we propose to add a distribution metric loss to the existing MP&P framework as shown in Fig. 3(e). Distribution metrics are frequently used to measure the divergence between two domains and are commonly employed in domain adaptation approaches. In our study, we consider proxies and real samples as two domains and employ the local maximum mean discrepancy (LMMD) [50] to measure and further constrain the divergence between proxies and corresponding real samples in a given class under a supervised learning way. As illustrated in Fig. 4(b), we aim to improve the intraclass tightness and increase the interclass separation through distribution optimization by constraining both marginal and conditional distributions. Each proxy returns to a more optimized position, which better represents its corresponding local cluster.

To adapt the MP&P framework, we modify the LMMD to a supervised version and incorporate the newly developed DL into the MP&P framework. The DL is formulated as

$$L_{\text{dist}} = \frac{1}{C}\sum_{c=1}^{C}\left\| \sum_{x_i^r \in \mathcal{D}_r} \omega_i^{rc}\phi(x_i^r) - \sum_{x_j^p \in \mathcal{D}_p} \omega_j^{pc}\phi(x_j^p) \right\|_{\mathcal{H}}^2 \tag{7}$$

where the real samples and proxies are regarded as two domains: $\mathcal{D}_r$ and $\mathcal{D}_p$, x_i^r and x_j^p denote the sample in the domain $\mathcal{D}_r$ and $\mathcal{D}_p$, respectively. $\| \cdot \|_{\mathcal{H}}^2$ denotes the L2-norm of the vector in reproducing kernel Hilbert space (RKHS), $\phi(\cdot)$ denotes the feature projection to project the original samples to RKHS. w_i^{rc} (w_j^{pc}) is a binary weight. If x_i^r (x_j^p) belongs to class c, the value of w_i^{rc} (w_j^{pc}) is 1, otherwise, it is 0. The total loss function is

$$\begin{aligned} L_{\text{total}} = {} & L_{\text{cls}}(S_{\text{cls}}, Y_{\text{cls}}) + \alpha L_{\text{pair}}(S_{\text{pair}}, Y_{\text{pair}}) \\ & + \beta L_{\text{dist}}(\mathcal{D}_r, \mathcal{D}_p, y_r, y_p) \end{aligned} \tag{8}$$

where Y_{cls} and Y_{pair} denote the label matrix of the classification label and pairwise label, y_r and y_p are the labels of samples in the domain $\mathcal{D}_r$ and $\mathcal{D}_p$. α and β are the hyperparameters to balance the classification loss, pairwise loss, and DL.

C. UMP+D Framework

The prototype framework can enhance classification performance by integrating the DL to the MP&P framework, but it also has significant drawbacks: not only does it increase the complexity of the framework [as shown in Fig. 3(e)], but it also requires balancing three loss functions [ref. Equation (8)], which makes training unstable and challenging to implement in practical scenarios.

To refine the prototype framework, we adopt the idea of combining classification loss and pairwise loss in [48] and propose UMP+D framework as shown in Fig. 3(f). The proposed UMP+D framework simplifies the prototype framework by only including two components: The pairwise loss computation branch and the DL computation branch. The DL computation branch is identical to that in the original prototype framework [Fig. 3(e)]. The pairwise loss computation branch in the UMP+D framework combines the classification learning branch with the pairwise learning branch of the prototype framework. Although the similarity matrix for classification loss and pairwise loss is calculated differently, there is no significant difference in the objective function to be optimized. As long as the similarity matrix used for the metric loss includes both similarities between real samples and similarities between samples and proxies, the final result will be similar to optimizing both pairwise loss and classification loss.

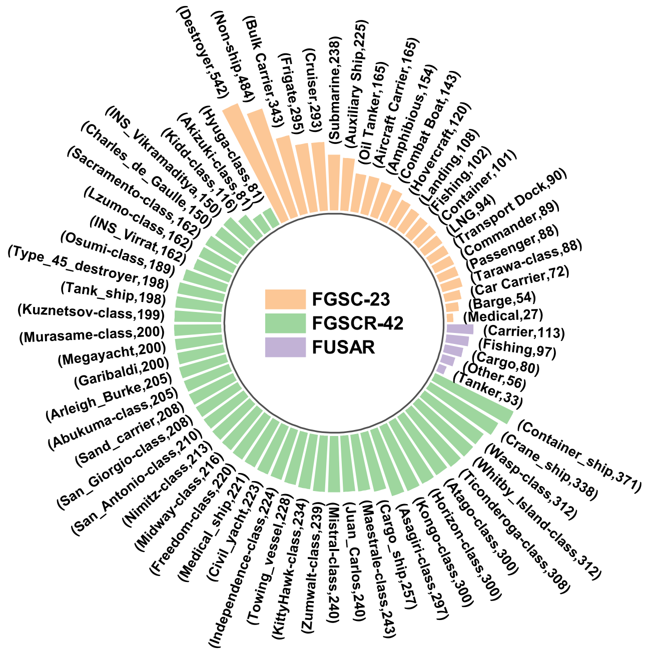


Fig. 5. Datasets used in our experiments. FGSC-23 contain of 23 fine-grained ship classes (orange bar). FGSCR-42 is composed of 42 fine-grained ship classes (green bar). FUSAR is a SAR dataset including five ship classes (purple bar).

Based on this unified understanding, we extract the proxies embedded within the last fully-connected layer (classification head) and concatenate them with the embeddings generated by the embedder. We, then, consider the concatenated embedding matrix as our new set of embeddings and use it to compute any desired similarity metric, which is obtained by taking its transpose to produce a similarity matrix for loss calculation purposes. Such a similarity matrix contains both global information and local information of the current minibatch. Feeding such a similarity matrix into a pairwise loss for training is equivalent to training with both pairwise loss and classification loss as used in frameworks of SP&P, MP&P, and the prototype. By combining the benefits of both learning methods, we can achieve our desired outcome without having to include two distinct loss terms within the overall loss function. This approach enables us to capitalize on the unique advantages of each method while minimizing the complexity of the training process. The total loss function in the UMP+D framework is

$$L_{\text{total}} = L_{\text{pair}}(S_{\text{pair}}, Y_{\text{pair}}) + \mu L_{\text{dist}}(\mathcal{D}_r, \mathcal{D}_p, y_r, y_p) \tag{9}$$

where μ is a hyperparameter to balance the metric loss and DL. By minimizing the optimization objective, the parameters of the networks are obtained, which will be used for inference.

IV. Data

We assessed the efficacy of our proposed framework using two optical remote sensing datasets and a SAR dataset. The classes of ships included in each dataset, along with the corresponding numbers of ships per class, are presented in Fig. 5. A detailed description of each dataset is also provided below.

A. FGSC-23

FGSC-23 is an optical remote sensing dataset specifically designed for ship classification. It was collected by Zhang et al. [28] using Google Earth public images and the GF-1 satellite. Some of representative sample images from FGSC-23 can be found in Fig. 6.

The dataset consists of a total of 22 fine-grained classes of ships, with a resolution range of 0.4 to 2.0 m and a separate

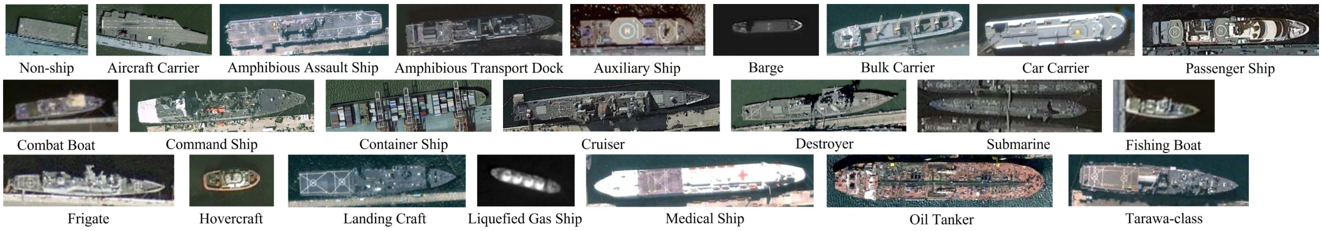


Fig. 6. Representative sample of each class in FGSC-23.

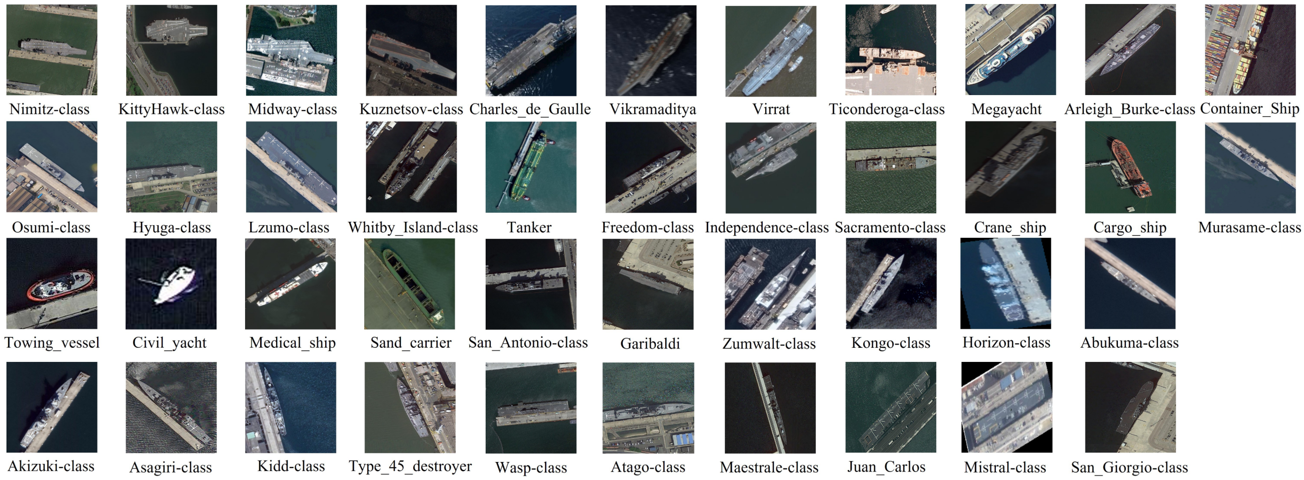


Fig. 7. Representative sample of each class in FGSCR-42.

"nonship" class that contains negative samples resembling ships. All ship classes are labeled by human judgment. FGSC-23 categorizes ships into fine-grained classes, such as classifying a coarse cargo ship class into various fine-grained classes, such as container ship, bulk carrier, car carrier, oil tanker, and liquefied gas ship. The high degree of intraclass similarity among the fine-grained classes makes the ship classification task more difficult. In addition, this dataset exhibits an imbalanced distribution, with certain ship classes, such as medical (27) and barge (54) being scarcer compared to others, such as destroyer (542) and bulk carrier (343). In our experiments, we utilized the division ratio recommended by the dataset provider, which allocated 80% of the data for training purposes and 20% for testing.

B. FGSCR-42

Another optical remote sensing fine-grained ship classification dataset is FGSCR-42, which was collected and released by Di et al. [51].

The dataset consists of 42 fine-grained ship classes, with a total of 9320 ship samples. The dataset comprises images sourced from three primary sources. The first source comprises several public remote sensing datasets such as DOTA, HRSC2016, and NWPU VHR-10. The second part of the images are remote sensing images collected by data providers from over 40 ports worldwide. To enhance the classifier's generalization capabilities and balance the sample numbers across various classes, the data provider employs data augmentation techniques to obtain some additional images, which form the third part of the dataset. Some of the representative sample images from FGSCR-42 can be found in Fig. 7. We also followed the division ratio recommended by the dataset provider, which allocated 50% of the data for training purposes and 50% for testing in our experiments.

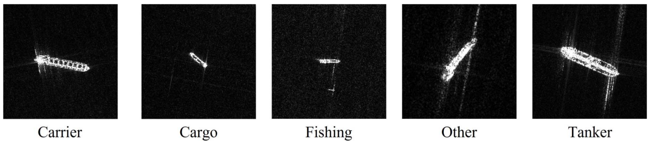


Fig. 8. Representative sample of each class in FUSAR-Ship.

C. Fusar-Ship

FUSAR-Ship is a fine-grained ship classification dataset for SAR images, collected by Hou et al. [52] from 126 GF-3 images with an azimuth resolution of 1.124 m and a slant range resolution ranging from 1.700 to 1.754 m. Some of representative sample images are illustrated in Fig. 8.

Due to the abundance of nonship classes and noisy ship samples in the original FUSAR dataset, we conducted a data cleaning process to ensure that our experiment focused on fine-grained ship classification. As a result, the dataset now consists of a total of 379 ship samples across five classes. We utilized 60% of this dataset for training purposes and reserved the remaining 40% for testing.

V. Experiments

A. Experimental Settings

1) Backbone Network: Given that the optical datasets FGSC-23 and FGSCR-42 offer adequate training samples, we opted to employ ConvNext [53], which is a DNN, as our backbone network for ship feature extraction from optical images. ConvNext achieved first place in the ImageNet [54] classification challenge in 2022, showcasing its exceptional performance in image recognition tasks. Since the FUSAR-ship dataset has a relatively small size, we decided to utilize ResNet18 [55] as the backbone to extract ship features from SAR images in accordance with recent research findings [56]. Both ConvNext and ResNet18 were initially trained (pretrained) on the ImageNet dataset and subsequently fine-tuned using training data specific to each dataset.

2) Sampling Strategy of MiniBatch: It is worth noting that all datasets utilized in our experiments were imbalanced, as specified in Section IV. Training a model with an imbalanced dataset can potentially lead to a model biasing toward classes with more samples. To mitigate this risk, we opted for the qk sampling approach instead of traditional random sampling for minibatch selection. The qk sampling method is commonly employed in DML and entails selecting a minibatch of size $q \times k$, where q denotes the number of classes in the minibatch, and k denotes the number of samples per class.

In our experiments, we set k to 8 and q to 16 for the FGSC-23, k to 4 and q to 32 for the FGSCR-42, and k to 16 and q to 5 for the FUSAR, empirically.

3) Hard Mining Strategy: For the hard mining strategy, we adopted a simple but effective method proposed by Wang et al. [57]. Specifically, a negative pair is compared with the hardest positive pair (with the lowest similarity), whereas a positive pair is sampled by comparing to a negative one having the largest similarity. Formally, assume x_m is an anchor, a negative pair $\{x_u, x_v\}$ is selected if its pairwise similarity s_{uv} satisfies the condition

$$s_{uv}^{-} > \min_{y_t = y_u} s_{ut} - \epsilon \tag{10}$$

where ϵ is a given margin. If $\{x_u, x_v\}$ is a positive pair, the condition is

$$s_{uv}^{+} < \max_{y_t \neq y_u} s_{ut} + \epsilon. \tag{11}$$

4) Hyperparameters Setting: Some of hyperparameters were preset empirically for the proposed UMP+D framework: For FGSC-23 and FGSCR-42 datasets, the μ in our loss function was set to 0.01, training epochs was set to 120, and each epoch samples 100 minibatches with qk sampler. The optimizer adopted was the AdamW with learning rate 1e–4 and weight decay 5e–4. For FUSAR dataset, the training epochs and minibatches per epoch were set to 25 and 5, respectively.

5) Experimental Platform: All of our experiments were run on a deep learning workstation with two Nvidia GeForce RTX 2080Ti GPUs, Intel(R) Core(TM) i9-9980XE @ 3.00 GHz CPU, 64 GB RAM, and Ubuntu 20.04 operation system with the PyTorch deep learning framework [58] and Pytorch-metric-learning library [59].

B. Experimental Contents

To gain a comprehensive understanding of the proposed framework, we organized three separate groups of experiments.

1) Hyperparameters Selection: Given the variations in experimental datasets FGSC-23, FGSCR-42, and FUSAR-ship, the values of hyperparameters N and D can significantly affect the classification outcomes across different datasets. To identify the optimal settings for these parameters, we conducted an experiment utilizing a traversal search method within a 2-D space, which will serve as a foundation for subsequent experiments.

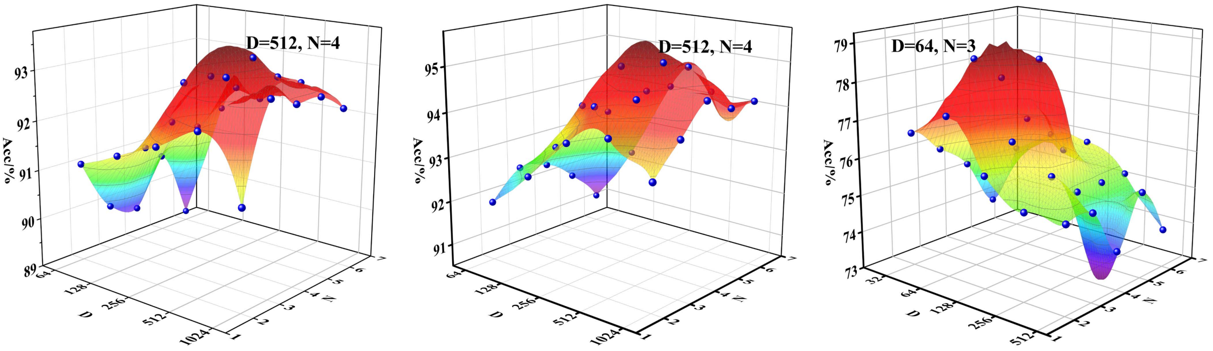
2) Comparison Between Various Frameworks: This group of experiments assessed the performance of the proposed UMP+D framework, in addition to the prototype framework, by comparing them with four existing frameworks, namely, SP, MP, SP&P, and MP&P. To ensure statistically significant results, we employed three classification losses (Cross Entropy, AM-Softmax [38], and ArcFace [41]) and three pairwise losses (Contrastive [45], Multisimilarity [57], and Circle [48]) individually or combined for comparative analysis across different frameworks.

3) Comparing With SOTA Methods: In this set of experiments, we compared our UMP+D approach with some SOTA approaches. These approaches cover the three main following aspects that are crucial for improving the performance of fine-grained ship classification:

1) more effective network structures, ResNet 50 [55], ConvNext [53], and P^{2}Net [23];
2) more comprehensive feature fusion, AMEFRN [28] and EAN [22]; and
3) more refined objective functions, Combination Loss [9] and DSL Loss [60].

Here, we will provide a brief overview of these approaches.

1) ResNet 50 [55]*:* It is a variant of the ResNet (residual network) architecture, which was designed to solve the problem of vanishing gradients in DNNs. ResNet 50 has achieved SOTA performance on various image recognition tasks, such as the imagenet large scale visual recognition challenge.
2) ConvNext [53]*:* The architecture of ConvNext was proposed by facebook AI research (FAIR) and is based on the concept of 1) "next-generation" convolutional layers, which are designed to be more efficient than traditional convolutional layers; 2) "residual connections," which helps to reduce the number of parameters and computational complexity required for training; 3) "preactivation residual blocks," which helps to improve the training speed and stability of the model; and 4) "SE modules," which are used to extract important features from the input data and provide additional information to the convolutional layers for improved accuracy.
3) P^{2}Net [23]*:* This dual-branch network separates or decorrelates images of distinct subclasses using two separate



(a) (b) (c)

Fig. 9. Hyperparameters selection for UMP+D on three datasets. (a) FGSC-23. (b) FGSCR-42. (c) FUSAR.

branches. An integration module combines these decorrelated images into their respective subclasses, while a proxy-based module accelerates the process. The network includes a "push-out stage" that encourages all instances to be decorrelated, followed by a "pull-in stage" that groups them into each subclass.

4) AMEFRN [28]*:* This approach utilizes feature fusion to represent the symbol of a particular category of ships using two local level features, while also revealing senior semantic information through one global level feature.
5) EAN [22]*:* EAN is also a feature fusion based method. This approach relies on two key components: a CMAM that generates multiple causal attention maps, and a FAM that integrates these maps with the original features to provide interpretable information during convolutional filter training.
6) Combination Loss [9]*:* To address the problem of SAR ship classification, He et al. [9] combined the use of triplet loss and Cross Entropy Loss in optimizing a triplet network. Furthermore, they proposed incorporating the Fisher discrimination regularization term to improve the model's ability to extract and utilize global information from the learned feature embeddings.
7) DSL Loss [60]*:* DSL loss is designed to maintain consistency between the optimization objective and the original data distribution.

C. Evaluation Metric

To gain a statistically meaningful understanding, the experiment is repeat ten times to eliminate the possible data bias caused by random sampling of data in each individual experiment. The overall performance, i.e., average accuracy over ten runs, is reported to evaluate the performance of various methods. In each individual experiment, we define the classification accuracy as

$$\text{acc.} = \frac{|\{x : x \in D_{\text{test}} \wedge f(x) = y(x)\}|}{|\{x : x \in D_{\text{test}}\}|} \tag{12}$$

where D_{test} denotes the test dataset and x is the instance in D_{test}. $f(x)$ denotes the predicted label of x and $y(x)$ denotes the ground truth label of x.

VI. Results

A. Hyperparameters Selection

The number of proxies for each class (N) and the embedding dimension (D) are two important hyperparameters in the MPs learning framework. The number of proxies (N) determines the granularity of the representation learned by the network, whereas the embedding dimension (D) directly affects the discriminative power of the features. Determining optimal values for these parameters is crucial for achieving good performance on a given dataset, as different datasets may require different combinations of these hyperparameters. This experiment aims to find the optimal combination of hyperparameters for each dataset, which can then be used as a baseline for subsequent experiments.

Fig. 9 presents the hyperparameters selection results for the proposed UMP+D framework with Circle Loss. It is evident that for two optical datasets, FGSC-23 and FGSCR-42, the optimal combination of embedding dimension (D) and number of proxies (N) is 512 and 4, respectively. However, for the FUSAR dataset, the optimal combination is 64 and 3. This result is consistent with the theoretical analysis and makes sense. As we know, increasing the number of proxies (N) results in more granularity in the learned representations, but it may also lead to overfitting. On the other hand, reducing the number of proxies (N) results in simpler representations, but it may underfit the data. In contrast, due to the inherent imaging properties of SAR images, which have much less information content than optical images, it is difficult to distinguish subclasses as accurately as in optical images. Therefore, the optimized N value for SAR images $(N = 3)$ is smaller than that for optical images $(N = 4)$. Similarly, a higher value of embedding dimension (D) leads to more complex and abstract representations, which can capture more information from the input data. However, in cases where there is insufficient information, SAR images require only a relatively small D value, such as 64, whereas optical images require a larger value, such as 512.

B. Comparison Between Various Frameworks

In this experiment, we employed six frameworks detailed in the aforementioned article, including the existing four: SP

TABLE I
PERFORMANCE OF SP ON THREE DATASETS WITH DIFFERENT CLASSIFICATION LOSSES

	FGSC-23	FGSCR-42	FUSAR
Cross Entropy	89.44%	91.70%	73.38%
AM-Softmax	89.81%	92.19%	72.73%
ArcFace	89.81%	92.19%	74.03%
Average	89.69%	92.03%	73.38%

[Fig. 3(a)], MP [Fig. 3(b)], SP&P [Fig. 3(c)], MP&P [Fig. 3(d)], along with the transitional prototype framework [Fig. 3(e)] and our proposed UMP+D framework (Fig. 3(f), a refined version of the prototype framework), under identical experimental conditions (data division, training strategy, hyperparameter configuration, etc.). This allowed us to conduct fine-grained ship classification on each of the three datasets and compare the outcomes. The experimental results are listed in Tables I, II, III, IV, V, and VI.

To establish a common benchmark, we utilized the performance of the SP framework as our baseline (Table I). We then compared the performance of other frameworks against this baseline. In these tables, the value with a underline indicates the highest performance achieved by a specific loss function (or a combination of loss functions) on a particular dataset for the corresponding framework. In Tables II–VI, red font represents an improvement over the baseline with a gain (+), while green font indicates that the result is worse than the baseline with a loss (-).

Upon observing the performance of the SP framework on three datasets (Table I), we observed that for a given dataset, the performance of the three classification loss functions was similar. For instance, on the FGSC-23 dataset, the Cross Entropy, AM-Softmax, and ArcFace classification loss functions all achieved scores of 89.44%, 89.81%, and 89.81%, respectively, which were very close to each other. When comparing the performance of SP framework on various datasets. We found that the average accuracy for classification on the FUSAR dataset was only 73.38% (even though it only contained five categories), which was significantly lower than the performance on the two optical data sets (FGSC-23: 89.69%, FGSCR-42: 92.03%). This demonstrates once again that the information content of SAR images is still the main bottleneck limiting its classification performance improvement. Further in-depth exploration of fine-grained ship classification based on SAR images remains to be done.

Comparing MP and SP frameworks (Table I versus II), we found that the MP framework improved classification accuracy across all datasets when using the same classification loss function. Especially on the FUSAR dataset, the average accuracy increased from 73.38% to 75.11%. Considering the difficulty of SAR image classification as previously mentioned, this 1.73% gain is not an insignificant improvement.

On the basis of the SP framework, the SP&P framework is obtained by combining classification loss and pairwise loss. The results in Table III present some complexity and also contain some regularity. From our experiments, we can find that combining any classification loss function (Cross Entropy, AM-Softmax, or ArcFace) with the Contrastive Loss (a kind of pairwise loss) leads to a decline in classification accuracy. On the contrary, combining any classification loss with the other two kinds of pairwise losses, namely, Circle Loss and Multisimilarity, leads to performance improvement. A similar yet more complex scenario occurs in the results of the MP&P framework (Table IV). Again, the combination of the Contrastive Loss and Cross Entropy leads to a decline in classification accuracy. However, its performance with respect to the other two classification losses (AM-Softmax and ArcFace) varies across datasets (some increase whereas others decrease).

Table V presents the performance achieved by the prototype framework. The prototype is a framework built on the basis of the MP&P, with the addition of DL as shown in Fig. 3(e). By comparing the average classification accuracy in the last row of Tables IV and V, we can see that the performance of prototype framework has been further improved by adding DL. Instead of using the combination of classification loss and pairwise loss, the proposed UMP+D unified the two kinds of losses into an individual pairwise loss mode. This refinement not only reduces the complexity of the framework, freeing up resources for other tasks (such as exploring optimal combinations of classification loss and pairwise loss), but also directly improves classification performance. The effectiveness of UMP+D framework can be observed by comparing the average classification accuracy in Table VI to that in Table V. The experimental results show that the gains in accuracy of the prototype framework are 1.91%, 1.64% and 2.38% on three different datasets, respectively, compared with the baseline method (SP). The gains in accuracy of the UMP+D framework are 2.46%, 2.10%, and 3.46% on the same three databases, respectively.

In order to further analyze how UMP+D achieves performance improvement, we plot the confusion matrices for the best classification results of each framework on the FUSAR dataset in Fig. 10. By examining the confusion matrices in Fig. 10(a)–(f), it can be observed that three classes, namely, "fishing," "other," and especially "cargo," appear to be difficult for all frameworks to handle. On the other hand, two classes, "carrier" and "tanker," seem relatively easy. When comparing the performance of different frameworks on the *HARD* classes, it can be seen that UMP+D excels at handling these types of problems. For the most *HARD* class, namely, the "cargo" class, while none of the frameworks surpasses 60.0% accuracy, UMP+D achieves 59.4% (the highest accuracy), beating MP, SP&P, and prototype framework by 3.2% with an accuracy of 56.2%, and SP and MP&P even lower at 53.1% up to 6.3%. Moreover, UMP+D achieves leading results in each individual ship class, resulting in the highest overall classification accuracy of 78.57%, which is 1.30% higher than the second-ranked prototype framework and 2.60% higher than the third-ranked MP&P framework.

C. *Comparing With SOTA Methods*

In this experiment, we compared our proposed UMP+D with several SOTA approaches. As mentioned in Sections II-A and

TABLE II
PERFORMANCE OF MP ON THREE DATASETS WITH DIFFERENT CLASSIFICATION LOSSES

	FGSC23	FGSCR-42	FUSAR
Cross Entropy	89.93% (+0.49%)	91.85% (+0.15%)	74.03% (+0.65%)
AM-Softmax	90.29% (+0.48%)	92.32% (+0.13%)	75.97% (+3.24%)
ArcFace	90.29% (+0.48%)	92.41% (+0.22%)	75.32% (+1.29%)
Average	90.17% (+0.48%)	92.19% (+0.16%)	75.11% (+1.73%)

Red and green represent gain (+) and loss (–), respectively.

TABLE III
PERFORMANCE OF SP&P ON THREE DATASETS WITH DIFFERENT COMBINATION OF CLASSIFICATION AND PAIRWISE LOSSES

		FGSC23	FGSCR-42	FUSAR
Cross Entropy	Contrastive	89.08% (–0.36%)	91.28% (–0.42%)	70.78% (–2.60%)
	Multisimilarity	89.81% (+0.37%)	92.26% (+0.56%)	74.03% (+0.65%)
	Circle	90.53% (+1.09%)	92.45% (+0.75%)	74.68% (+1.30%)
AM-Softmax	Contrastive	89.32% (–0.49%)	91.89% (–0.30%)	70.78% (–1.95%)
	Multisimilarity	90.17% (+0.36%)	92.45% (+0.26%)	72.73% (±0.00%)
	Circle	90.29% (+0.48%)	92.67% (+0.48%)	73.38% (+0.65%)
ArcFace	Contrastive	89.68% (–0.13%)	91.26% (–0.93%)	72.73% (–1.30%)
	Multisimilarity	90.41% (+0.60%)	92.84% (+0.65%)	74.03% (±0.00%)
	Circle	90.78% (+0.97%)	92.84% (+0.65%)	74.68% (+0.65%)
Average		89.93% (+0.24%)	92.19%(+0.16%)	73.01% (–0.37%)

Red and green represent gain (+) and loss (–), respectively.

TABLE IV
PERFORMANCE OF MP&P ON THREE DATASETS WITH DIFFERENT COMBINATION OF CLASSIFICATION AND PAIRWISE LOSSES

		FGSC23	FGSCR-42	FUSAR
Cross Entropy	Contrastive	89.32% (–0.12%)	91.41% (–0.29%)	72.08% (–1.30%)
	Multisimilarity	90.53% (+1.09%)	92.32% (+0.62%)	74.68% (+1.30%)
	Circle	90.66% (+1.22%)	92.93% (+1.23%)	74.68% (+1.30%)
AM-Softmax	Contrastive	89.68% (–0.13%)	92.26% (+0.07%)	73.38% (+0.65%)
	Multisimilarity	90.78% (+0.97%)	93.25% (+1.06%)	75.97% (+3.24%)
	Circle	90.90% (+1.09%)	93.62% (+1.43%)	74.68% (+1.95%)
ArcFace	Contrastive	90.17% (+0.36%)	92.26% (+0.07%)	73.38% (–0.65%)
	Multisimilarity	90.78% (+0.97%)	93.25% (+1.06%)	75.32% (+1.29%)
	Circle	91.02% (+1.21%)	93.62% (+1.43%)	75.97% (+1.94%)
Average		90.43% (+0.74%)	92.77% (+0.74%)	74.42% (+1.04%)

Red and green represent gain (+) and loss (–), respectively.

TABLE V
PERFORMANCE OF THE PROTOTYPE FRAMEWORK ON THREE DATASETS WITH DIFFERENT COMBINATION OF CLASSIFICATION AND PAIRWISE LOSSES

		FGSC23	FGSCR-42	FUSAR
Cross Entropy	Contrastive	89.93% (+0.49%)	91.93% (+0.23%)	74.03% (+0.65%)
	Multisimilarity	91.50% (+2.06%)	93.87% (+2.17%)	75.32% (+1.94%)
	Circle	92.11% (+2.67%)	93.92% (+2.22%)	75.97% (+2.59%)
AM-Softmax	Contrastive	90.41% (+0.60%)	92.80% (+0.61%)	74.03% (+1.30%)
	Multisimilarity	91.38% (+1.57%)	94.09% (+1.90%)	75.97% (+3.24%)
	Circle	92.23% (+2.42%)	94.55% (+2.36%)	76.62% (+3.89%)
ArcFace	Contrastive	91.50% (+1.69%)	93.14% (+0.95%)	75.32% (+1.29%)
	Multisimilarity	92.72% (+2.91%)	94.48% (+2.29%)	77.27% (+3.24%)
	Circle	92.60% (+2.79%)	94.27% (+2.08%)	77.27% (+3.24%)
Average		91.60% (+1.91%)	93.67% (+1.64%)	75.76% (+2.38%)

Red and green represent gain (+) and loss (–), respectively.

TABLE VI
PERFORMANCE OF UMP+D ON THREE DATASETS WITH DIFFERENT PAIRWISE LOSSES

	FGSC-23	FGSCR-42	FUSAR
Contrastive	90.29%	92.26%	73.38%
Multisimilarity	92.84%	94.96%	78.57%
Circle	93.33%	95.18%	78.57%
Average	92.15% (+2.46%)	94.13% (+2.10%)	76.84% (+3.46%)

Red and green represent gain (+) and loss (–), respectively.

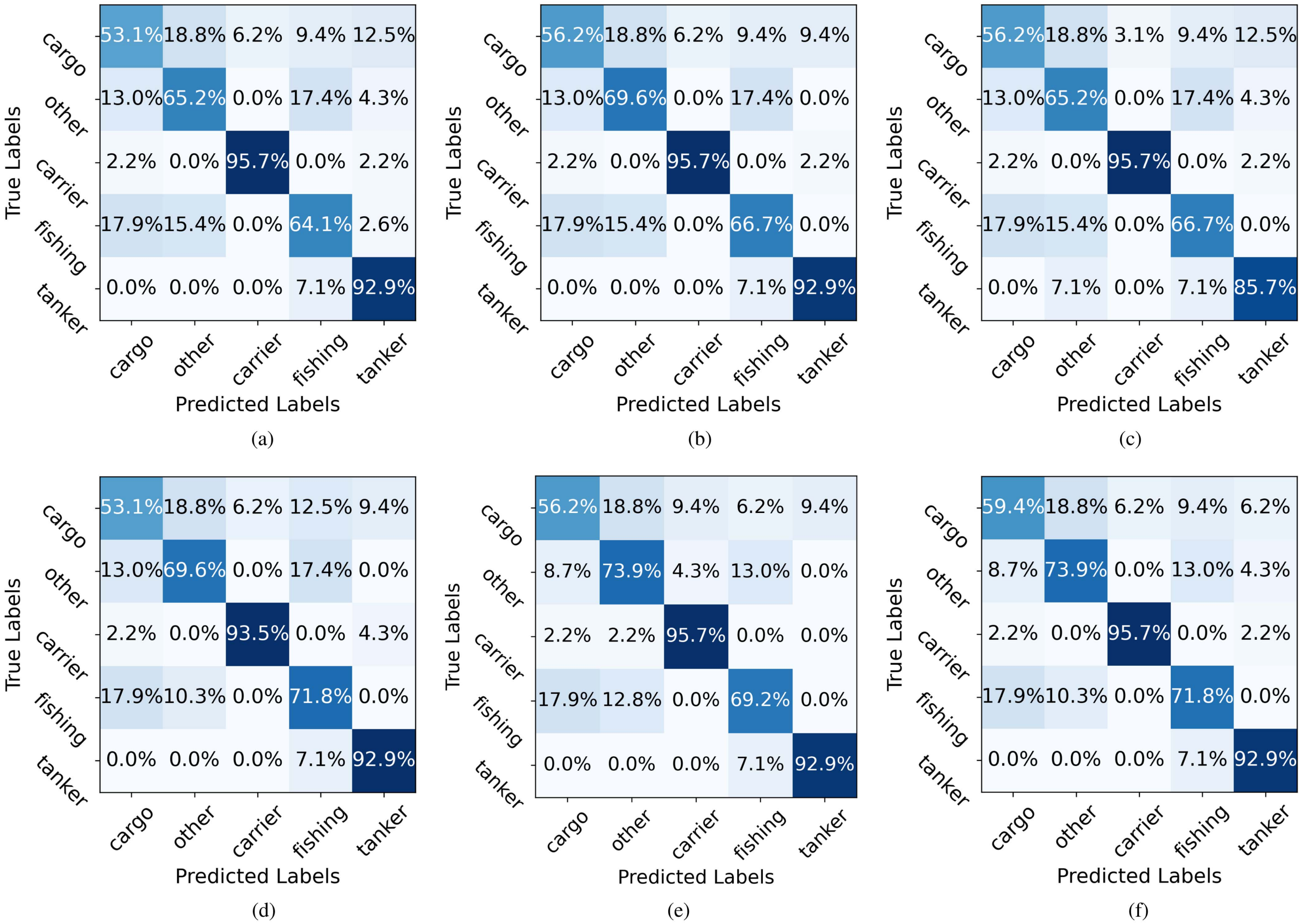


Fig. 10. Confusion matrices for the best classification results of each framework on the FUSAR dataset. (a) SP: 74.03%, ArcFace, (ref. Table I). (b) MP: 75.97%, AM-Softmax, (ref. Table I). (c) SP&P: 74.68%, ArcFace+Circle, (ref. Table III). (d) MP&P: 75.97%, ArcFace+Circle, (ref. Table IV). (e) Prototype framework, 77.27%, ArcFace+Circle, (ref. Table V). (f) UMP+D.78.57%, Circle, (ref. Table VI). (a) SP. (b) MP. (c) SP&P. (d) MP&P. (e) Prototype framework. (f) UMP+D.

V-B, these approaches represent the most recent advancements in three main areas of research. By comparing them to our proposed UMP+D, we can provide an objective evaluation of its performance.

As shown in the Table VII, UMP+D achieved leading results on all three datasets except for the FGSC-23 dataset where it was slightly lower than AMEFRN (93.33% versus 93.58%). On the other hand, UMP+D outperformed the comparison approaches by a significant margin in the other two datasets, with an edge of 1.12 percentage points over EAN in the FGSCR-42 dataset and nearly 2 percentage points over DSL in the FUSAR dataset. Upon further examination of the experimental results, it can be observed that ConvNext and P^{2}Net have made significant improvements in terms of network structure compared with ResNet50. Both ConvNext and P^{2}Net outperform ResNet50 by 6–7 percentage points on both optical datasets. By utilizing ConvNext as the backbone network in UMP+D, we achieve a leading performance. Furthermore, UMP+D further improves its performance by 2 percentage points compared with ConvNext by incorporating multiagent learning and unified classification

TABLE VII
COMPARING WITH SOTA METHODS

Dataset	Method	Overall accuracy
FGSC-23	ResNet50 [55]	83.03%
	ConvNext [53]	91.02%
	P^{2}Net [23]	89.07%
	AMEFRN [28]	93.58%★
	EAN [22]	91.78%★
	Combination Loss [9]	90.29%
	DSL Loss [60]	90.78%
	UMP+D	93.33%
FGSCR-42	ResNet50 [55]	87.24%
	ConvNext [53]	93.62%
	P^{2}Net [23]	93.29%
	AMEFRN [28]	——★★
	EAN [22]	94.06%★
	Combination Loss [9]	92.26%
	DSL Loss [60]	93.14%
	UMP+D	95.18%
FUSAR	ResNet18★★★ [55]	73.38%
	Combination Loss [9]	75.97%
	DSL Loss [60]	76.62%
	UMP+D★★★★	78.57%

★ Classification accuracy is quoted from original literature.
★★ Classification accuracy of AMEFRN [28] on the FGSCR-42 dataset is not available.
★★★ FUSAR does not have enough data to train more complex networks, such as ConvNext, P^{2}Net, AMEFRN, and EAN.
★★★★ For aforementioned reason, here UMP+D uses ResNet18 as the backbone network.

and pairwise loss functions. AMEFRN achieved comparable or better performance than UMP+D, mainly due to its more indepth feature fusion compared with UMP+D. While the idea of MP learning and loss function optimization is not contradictory with feature fusion, it is possible to further enhance the performance of UMP+D by incorporating these ideas into the feature fusion-based approaches.

Due to the limited amount of data in the FUSAR dataset, complex models such as AMEFRN, EAN, and P^{2}Net suffer from overfitting. Therefore, we only present results using ResNet 18 in Table VII. On the other hand, several loss function optimization based methods, such as Combination Loss [9], DSL Loss [60], and our proposed UMP+D, achieve good results. This demonstrates that optimizing the objective function can improve model performance and generalization ability without increasing network size or complexity when dealing with small datasets. This is why objective function optimization has become one of the three mainstream research directions.

VII. DISCUSSION

A. Rethinking Various Frameworks

Analyzing the experimental results described in Section VI-B, we believe that the performance of various frameworks has a direct relationship with proxy and loss function.

The performance of the MP framework is better than that of the SP framework because it performs more detailed demarcation for each class, further exploring the differences within classes and learning these differences through setting multiple subclass centers. This leads to improved feature representations in the final classification.

As for SP&P and MP&P frameworks, there are two points that should be kept in mind: 1) Combining classification loss with pairwise loss directly does not guarantee that it will always bring benefits. Researchers must consider, which combinations are truly effective for specific tasks. 2) While MP&P is still more effective than SP&P when using a combination of classification loss and pairwise loss, the complexity also increases.

The proposed UMP+D framework proves to be superior to existing frameworks mainly due to three following points:

1) The use of MP learning to obtain fine demarcation for sub-classes;
2) the unification of classification loss and pairwise loss into a single mathematical paradigm to improve training efficiency; and
3) the further optimization of MP positions through distribution constraint to enhance representation capabilities.

B. Ablation Study

We investigated and discussed the effects of two major innovations, namely, DL and unified pairwise loss, on UMP+D through ablation study. As shown in Fig. 3(f), these two innovations make up the DL and PL branches of the UMP+D framework, respectively.

1) Impact of DL Branch: In order to study the effect of DL on UMP+D framework, we removed DL branch from the framework, as shown in Fig. 3(f). In practice, this is equivalent to removing L_{dist} from the total objective function (9) but retaining all other terms. For the sake of expression, we refer to this trimmed UMP+D framework as UMP. The experimental results are listed in Table VIII. The results of UMP and UMP+D both used Circle Loss. The experiment results demonstrate that when DL is removed from the UMP+D framework, the performance of UMP on all three datasets significantly degrades. This result confirms that DL is an essential component of the UMP+D framework.

In addition to the positive effect of DL on classification as reflected in the quantitative classification accuracy, visualizing the distribution of samples in embedding space can help us see more clearly how DL reorganizes sample distribution to improve classification performance through t-SNE [61]. Fig. 11 illustrates the distribution of test samples in the embedding space of the FGSCR-42 dataset. By comparing the difference in sample distributions between Fig. 11(a) and (b), we can see the role that DL plays in separating samples from different classes and compacting samples within a class, especially when there are multiple possible subclasses within a particular class. Overall, the embedding space of UMP+D with DL [Fig. 11(a)] maintains a basic isolation between interclass samples while preventing overconcentration of intraclass samples, as shown in that of UMP [Fig. 11(b)]. This results in the MPs not being

TABLE VIII
RESULTS OF ABLATION STUDY ON THREE DATASETS

	FGSC-23	FGSCR-42	FUSAR
UMP+D	93.33%	95.18%	78.57%
UMP	92.48% (-0.85%)	94.01% (-1.17%)	76.62% (-1.95%)
Prototype	92.60% (-0.73%)	94.27% (-0.91%)	77.27% (-1.50%)

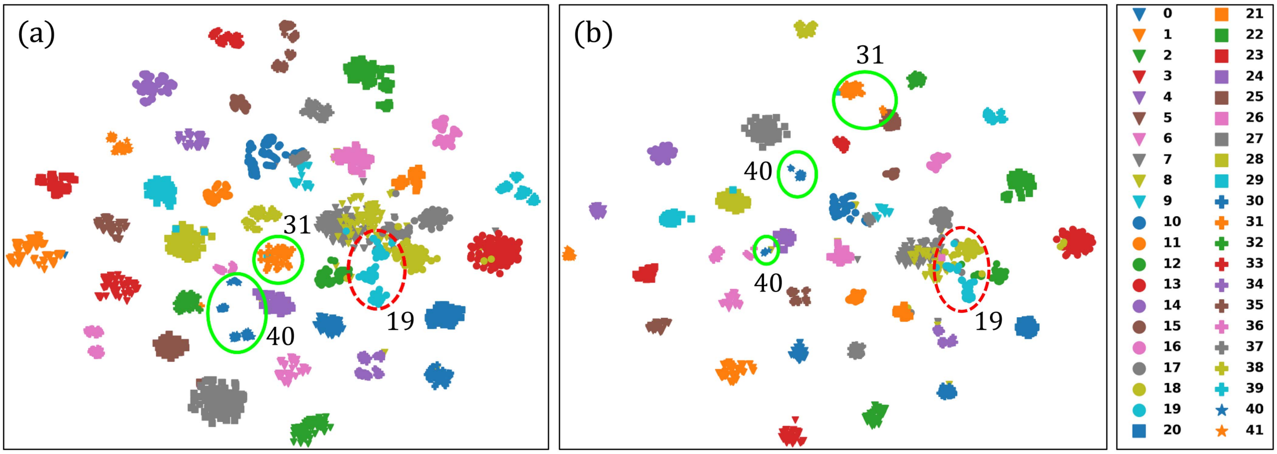


Fig. 11. Visualization of sample distribution in the feature space on the FGSCR-42 dataset utilizing t-SNE [61]. (a) UMP+D. (b) UMP.

mixed together due to the high clustering of samples within each class, thus, losing the discriminative capability between MPs within a class. Upon closer observation of specific classes, we can see that DL does effectively adjust and optimize the proxy positions. For example, in Fig. 11(b), class 19 is mixed with other classes as marked in the red dashed line. After DL optimization in Fig. 11(a), class 19 is clearly separated from other classes. For class 40, which contains multiple local clusters due to intraclass variation, it is not only separated by a large gap (intraclass dissimilarity) but also blocked by other classes (interclass similarity). However, after DL adjustment, class 40 no longer mixes with other classes and the four subclasses show clear boundaries.

2) Impact of PL Branch: When the unified classification loss and pairwise loss are not used, but a direct combination of the two losses is adopted, UMP+D returns to the prototype framework. We relisted the best performance achieved by the prototype framework on the three datasets (i.e., the second row from the bottom in Table V) when using a combination of ArcFace Loss and Circle Loss, and compared it with that of UMP+D in Table VIII. This result again demonstrates that the unified classification loss and pairwise loss are more effective than directly combining them.

VIII. Conclusion

To improve ship classification performance in remote sensing imagery, the existing supervised deep learning methods are mainly studied from three following aspects:

1) more effective network architecture;
2) more comprehensive feature fusion; and
3) more refined metric loss function.

This study focuses on the third aspect, that is, improving the intraclass compactness and interclass separation of fine-grained ship samples through DML, so as to improve final ship classification performance. Through the indepth investigation of related work, this study summarized the existing DML methods into four representative frameworks from the perspective of the use of two kinds of elemental loss (i.e., classification loss and pairwise loss) and the way of classification learning (i.e., SP and MP). Inspired by the existing work, this study proposes the UMP+D framework, which has three novel characteristics:

1) Unifying the combination of classification loss and pairwise loss into a single loss function containing only pairwise representation.
2) Fusing pairwise representation with MP learning.
3) Embedding the DL to refine the distribution of samples in the feature embedding space to further tighten the intraclass samples and pull apart the interclass samples.

Extensive experiments demonstrate that the proposed UMP+D framework outperforms the existing ones and achieves SOTA performance.

The proposed UMP+D aims to refine the metric loss function, which gives it certain independent properties. This property allows it to be integrated with two other aspects of current research efforts. That is, we can further improve the performance of the proposed framework by designing a more effective network architecture, or by fusing more appropriate features. In future work, we will follow the above ideas and continue to improve the ship classification performance of remote sensing imagery through further adaptive expansion based on this framework.

Acknowledgment

The authors would like to thank the contributors of FGSC-23, FGSCR-42, and FUSAR-ship datasets for their data support for this study. Meanwhile, the authors would also like to thank the Associate Editor and anonymous reviewers for their valuable and constructive comments that significantly improved the study.

References

[1] X. Zhang, Y. Zhou, and J. Luo, "Deep learning for processing and analysis of remote sensing Big Data: A technical review," *Big Earth Data*, vol. 6, pp. 527–560, 2021.

[2] C. Persello et al., "Deep learning and earth observation to support the sustainable development goals: Current approaches, open challenges, and future opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 172–200, Jun. 2022.

[3] L. Zhang and L. Zhang, "Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 270–294, Jun. 2022.

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

[5] X. X. Zhu et al., "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[6] X. X. Zhu et al., "Deep learning meets SAR: Concepts, models, pitfalls, and perspectives," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 4, pp. 143–172, Dec. 2021.

[7] Y. Dong, H. Zhang, C. Wang, and Y. Wang, "Fine-grained ship classification based on deep residual learning for high-resolution SAR images," *Remote Sens. Lett.*, vol. 10, no. 11, pp. 1095–1104, 2019.

[8] Y. Li, X. Li, Q. Sun, and Q. Dong, "SAR image classification using CNN embeddings and metric learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4002305.

[9] J. He, Y. Wang, and H. Liu, "Ship classification in medium-resolution SAR images via densely connected triplet CNNs integrating fisher discrimination regularized metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3022–3039, Apr. 2021.

[10] Y. Xu and H. Lang, "Distribution shift metric learning for fine-grained ship classification in SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2276–2285, 2020.

[11] Y. Xu and H. Lang, "Ship classification in SAR images with geometric transfer metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6799–6813, Aug. 2021.

[12] J. A. Raj, S. M. Idicula, and B. Paul, "One-shot learning-based SAR ship classification using new hybrid siamese network," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4017205.

[13] T. Zhang and X. Zhang, "Squeeze-and-excitation Laplacian pyramid network with dual-polarization feature fusion for ship classification in SAR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4019905.

[14] T. Zhang et al., "HOG-shipCLSNet: A novel deep learning network with hog feature fusion for SAR ship classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5210322.

[15] H. Lang et al., "Semi-supervised heterogeneous domain adaptation via dynamic joint correlation alignment network for ship classification in SAR imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4508105.

[16] H. Lang et al., "Multi-source heterogeneous transfer learning via feature augmentation for ship classification in SAR imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5228814.

[17] H. Zheng, Z. Hu, J. Liu, Y. Huang, and M. Zheng, "Metaboost: A novel heterogeneous DCNNs ensemble network with two-stage filtration for SAR ship classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4509005.

[18] Y. Li, X. Lai, M. Wang, and X. Zhang, "C-SASO: A clustering-based size-adaptive safer oversampling technique for imbalanced SAR ship classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5231112.

[19] T. Zhang and X. Zhang, "A polarization fusion network with geometric feature embedding for SAR ship classification," *Pattern Recognit.*, vol. 123, 2022, Art. no. 108365.

[20] J. Ai, Y. Mao, Q. Luo, L. Jia, and M. d. Xing, "SAR target classification using the multikernel-size feature fusion-based convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5214313.

[21] S. Zhao and H. Lang, "Improving deep subdomain adaptation by dual-branch network embedding attention module for SAR ship classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 8038–8048, 2022.

[22] W. Xiong, Z. Xiong, and Y. Cui, "An explainable attention network for fine-grained ship classification using remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5620314.

[23] J. Chen, K. Chen, H. Chen, W. Li, Z. Zou, and Z. Shi, "Contrastive learning for fine-grained ship classification in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4707916.

[24] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 1, pp. 6–43, Mar. 2013.

[25] U. Kanjir, H. Greidanus, and K. Oštir, "Vessel detection and classification from spaceborne optical images: A literature survey," *Remote Sens. Environ.*, vol. 207, pp. 1–26, 2018.

[26] K. Liu, S. Yu, and S. Liu, "An improved inceptionv3 network for obscured ship classification in remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4738–4747, 2020.

[27] J. He et al., "Group bilinear CNNs for dual-polarized SAR ship classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4508405.

[28] X. Zhang, Y. Lv, L. Yao, W. Xiong, and C. Fu, "A new benchmark and an attribute-guided multilevel feature representation network for fine-grained ship classification in optical remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1271–1285, 2020.

[29] T. Zhang and X. Zhang, "Injection of traditional hand-crafted features into modern CNN-based models for SAR ship classification: What, why, where, and how," *Remote Sens.*, vol. 13, no. 11, 2021, Art. no. 2091.

[30] L. Zeng et al., "Dual-polarized SAR ship grained classification based on CNN with hybrid channel feature loss," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4011905.

[31] M. Kaya and H. Ş. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, 2019, Art. no. 1066.

[32] H. Lang, S. Wu, and Y. Xu, "Ship classification in SAR images improved by AIS knowledge transfer," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 3, pp. 439–443, Mar. 2018.

[33] Y. Xu, H. Lang, L. Niu, and C. Ge, "Discriminative adaptation regularization framework-based transfer learning for ship classification in SAR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 11, pp. 1786–1790, Nov. 2019.

[34] J. Lu, J. Hu, and J. Zhou, "Deep metric learning for visual understanding: An overview of recent advances," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 76–84, Nov. 2017.

[35] B. Ghojogh, A. Ghodsi, F. Karray, and M. Crowley, "Spectral, probabilistic, and deep metric learning: Tutorial and survey," 2022, *arXiv:2201.09267*.

[36] 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 Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, May 2018.

[37] M. Boudiaf et al., "A unifying mutual information view of metric learning: Cross-entropy vs pairwise losses," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 548–564.

[38] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Process. Lett.*, vol. 25, no. 7, pp. 926–930, Jul. 2018.

[39] R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax loss for discriminative face verification," 2017, *arXiv:1703.09507*.

[40] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," 2016, *arXiv:1612.02295*.

[41] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4690–4699.

[42] Y. M.-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh, "No fuss distance metric learning using proxies," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 360–368.

[43] J. Goldberger, S. T. Roweis, G. E. Hinton, and R. Salakhutdinov, "Neighbourhood components analysis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 1–8.

[44] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin, "SoftTriple loss: Deep metric learning without triplet sampling," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6450–6458.

[45] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 1735–1742.

[46] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[47] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4004–4012.

[48] Y. Sun et al., "Circle loss: A unified perspective of pair similarity optimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6398–6407.

[49] K. Kobs, M. Steininger, A. Dulny, and A. Hotho, "Do different deep metric learning losses lead to similar learned features?," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10644–10654.

[50] Y. Zhu et al., "Deep subdomain adaptation network for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 4, pp. 1713–1722, Apr. 2021.

[51] Y. Di, Z. Jiang, and H. Zhang, "A public dataset for fine-grained ship classification in optical remote sensing images," *Remote Sens.*, vol. 13, no. 4, 2021, Art. no. 747.

[52] X. Hou, W. Ao, Q. Song, J. Lai, H. Wang, and F. Xu, "FUSAR-ship: Building a high-resolution SAR-AIS matchup dataset of Gaofen-3 for ship detection and recognition," *Sci. China Inf. Sci.*, vol. 63, no. 4, pp. 1–19, 2020.

[53] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020 s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 11976–11986.

[54] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[56] H. Lang, R. Wang, S. Zheng, S. Wu, and J. Li, "Ship classification in SAR imagery by shallow CNN pre-trained on task-specific dataset with feature refinement," *Remote. Sens.*, vol. 14, 2022, Art. no. 5986. [Online]. Available: https://api.semanticscholar.org/CorpusID:255571732

[57] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott, "Multi-similarity loss with general pair weighting for deep metric learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5022–5030.

[58] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

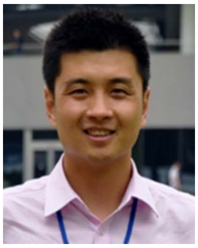
[59] K. Musgrave, S. Belongie, and S.-N. Lim, "A metric learning reality check," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 681–699.

[60] L. Fan, H. Zhao, H. Zhao, P. Liu, and H. Hu, "Distribution structure learning loss (DSLL) based on deep metric learning for image retrieval," *Entropy*, vol. 21, no. 11, 2019, Art. no. 1121.

[61] L. v. d. Maaten and G. E. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

Jianwen Xu received the B.E. degree in electronic science and technology from Beijing University of Chemical Technology (BUCT), Beijing, China, in 2020. He is currently working toward the M.S. degree in physics with the College of Mathematics and Physics, BUCT.

His research interestfocuses on algorithm for fine-grained ship classification in optical/SAR remote sensing imagery.



Haitao Lang (Member, IEEE) received the B.E. and M.S. degrees in optical engineering from Ocean University of China, Qingdao, China, in 2000 and 2003, respectively, and received the Ph.D. degree in optical engineering from Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai, China, in 2006.

He is currently a Professor with the College of Mathematics and Physics, Beijing University of Chemical Technology, Beijing, China. He has authored or coauthored prolifically in some prestigious refereed journals and conference proceedings such as IEEE Transactions on Image Processing, IEEE Transactions on Geoscience and Remote Sensing, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Geoscience and Remote Sensing Letters, CVPR, ACCV, IGARSS, SPIE Remote Sensing, etc. His research interests include machine learning, pattern recognition, and optical engineering with a focus on developing advanced image analysis and interpretation techniques for maritime remote sensing applications.