Few-Shot Scene Classification of Optical Remote Sensing Images Leveraging Calibrated Pretext Tasks

Hong Ji, Zhi Gao, Yongjun Zhang, Yu Wan, Can Li, and Tiancan Mei

***Abstract*— Small data hold big artificial intelligence (AI) potential. As one of the promising small data AI approaches, few-shot learning has the goal to learn a model efficiently that can recognize novel classes with extremely limited training samples. Therefore, it is critical to accumulate useful prior knowledge obtained from large-scale base class dataset. To realize few-shot scene classification of optical remote sensing images, we start from a baseline model that trains all base classes using a standard cross-entropy loss leveraging two auxiliary objectives to capture intrinsical characteristics across the semantic classes. Specifically, rotation prediction learns to recognize the 2-D rotation of an input to guide the learning of class-transferable knowledge, and contrastive learning aims to pull together the positive pairs while pushing apart the negative pairs to promote intraclass consistency and interclass inconsistency. We jointly optimize two such pretext tasks and semantic class prediction task in an end-to-end manner. To further overcome the overfitting issue, we introduce a regularization technique, adversarial model perturbation, to calibrate the pretext tasks so as to enhance the generalization ability. Extensive experiments on public remote sensing benchmarks including Northwestern Polytechnical University (NWPU)-RESISC45, aerial image dataset (AID), and Wuhan University (WHU)-remote sensing (RS)-19 demonstrate that our method works effectively and achieves best performance that significantly outperforms many state-of-the-art approaches.**

***Index Terms*— Adversarial model perturbation (AMP), few-shot scene classification, multitask learning, optical remote sensing image, pretext task.**

I. Introduction

SCENE classification of optical remote sensing images has attracted remarkable attention and usually relies on powerful high-capacity models with trainable parameters ranging from millions to tens of millions, requiring a substantial amount of annotated training data. However, in practice, the lack of intensive annotations becomes the bottleneck to make precise and timely decisions. For instance, in the research of remote-sensing-based post-disaster assessment and rescue, it is fairly labor-intensive and time-consuming to collect a large amount of training data and may even risk the danger of life. Inspired by the ability of human to learn new concepts very quickly, recent advances develop a new mechanism to perform few-shot classification [8], [44], where a model could generalize to new-coming classes with extremely limited training samples. In this manner, the learned models can be applied to a variety of fields, such as land-use land-cover classification [63], disaster monitoring [31], and urban planning [65], meanwhile having the ability to recognize novel classes that were unseen in the training.

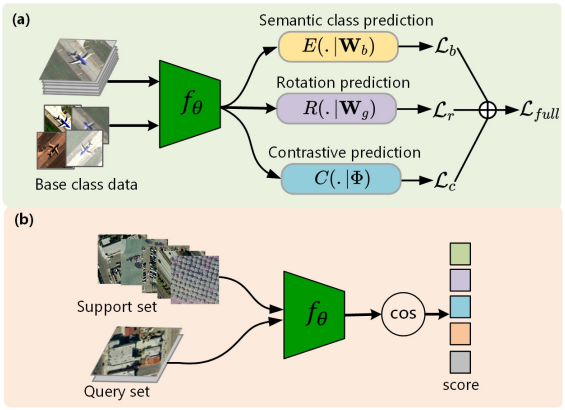


Fig. 1. Illustration of the proposed framework. (a) Training phase containing semantic class prediction $\mathcal{L}_b$, rotation prediction $\mathcal{L}_r$, and contrastive prediction $\mathcal{L}_c$. (b) Test phase that makes decision by a nearest-neighbor rule. The distances between query images and support images are computed by cosine similarity.

Over the past years, few-shot learning has received increasing attention for its big artificial intelligence (AI) potential. The research literature on this community shows great diversity, following the core idea of transferring knowledge or experience from seen tasks (sampled from base class data) to previously unseen tasks (sampled from novel class data). Formally, a task takes the form of K-way N-shot, which consists of K classes with each class N training samples and M testing samples. The goal is to classify the $K \times M$ testing samples into K classes after observing the $K \times N$ training samples (usually, $K = 5$, $N = 1$, $M = 15$). Among the available few-shot learning algorithms, an intuitive method, termed as baseline model, is training all base classes with a standard cross-entropy loss, followed by a naive fine-tuning step. However, this simple transfer learning method tends to overfit due to data scarcity. To overcome this limitation, meta-learning is proposed to train a model on a variety of learning tasks, and in such tasks the data are sampled from a large-scale base class data to mimic the real-world few-shot scenarios, such that it can solve new learning tasks using only a small number of training samples [8], [44], [47].

Manuscript received 3 April 2022; revised 16 May 2022; accepted 9 June 2022. Date of publication 17 June 2022; date of current version 28 June 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 42192580 and Grant 42192583, in part by the Hubei Province Natural Science Foundation under Grant 2021CFA088 and Grant 2020CFA003, in part by the Science and Technology Major Project under Grant 2021AAA010 and Grant 2021AAA010-3, and in part by the Chinese Association For Artificial Intelligence (CAAI)-Huawei MindSpore Open Fund. *(Corresponding author: Zhi Gao.)*

Hong Ji, Zhi Gao, Yongjun Zhang, Yu Wan, and Can Li are with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: jihong@whu.edu.cn; gaozhinus@gmail.com; zhangyj@whu.edu.cn; wanyu2017@whu.edu.cn; volcano.lee.4@gmail.com).

Tiancan Mei is with the School of Electronic Information, Wuhan University, Wuhan 430072, China (e-mail: mtc@whu.edu.cn).

Digital Object Identifier 10.1109/TGRS.2022.3184080

Although a variety of meta-learning-based approaches have designed sophisticated algorithms and network architectures, such methods still suffer from severe model overfitting [45] and the resulting performance is far from satisfactory.

Rather than focusing on the design of network architectures and classifiers, superior feature embedding has proved critical to improve few-shot learning performance. To learn a favorable representation, recent efforts introduce auxiliary objectives to promote feature robustness [28], [33], [37], [60]. These auxiliary objectives usually take the form of pretext task learning so as to build an embedding that can be used for other downstream tasks [1], [10], [14], [19]. Traditionally, the pretext tasks are predefined and the supervisory signals are generated from the data itself (self-supervision) leveraging their structure, such as grayscale image colorizing [62], image jigsaw puzzle [35], and image rotation [13]. Most recently, the robustness and simplicity of such pretext tasks have fueled wide applicability in other areas beyond self-/un-supervised representation learning [15], [22], [58], and their great benefits in few-shot learning have been verified with convincing results [12], [21], [22], [28], [33], [37]. In particular, rotation prediction [33], [37], [60] and contrastive prediction [21], [28], [37] have been studied independently to improve the few-shot learning capability. Despite the efforts reported in studying the role of single pretext task in few-shot learning, such paradigm has not been fully explored yet, ignoring the potential of ensemble effect of multiple pretext tasks.

In the task of remote sensing scene classification, few-shot learning has reported encouraging results [4], [23]–[25]. Most of these methods leverage meta-learning framework to learn a prototype representation to cluster the testing samples. Regarding the auxiliary techniques, only one available approach SCL-metric learning network (MLNet) [25] has incorporated a self-supervised contrastive prediction task to perform scene classification, following the meta-learning pipeline. In short, most of the available methods have underestimated the potential of transfer learning in few-shot task of optical remote sensing images, let alone the effectiveness of training different pretext tasks and their combinations.

To overcome the aforementioned challenges and shortcomings, we propose a framework to realize few-shot scene classification of optical remote sensing images leveraging two calibrated pretext tasks, rotation prediction and contrastive prediction. Specifically, the former learns to recognize the 2-D rotation of an input to guide the learning of class-transferable knowledge since class-agnostic supervision is adopted, and the latter aims to pull together the positive pairs (e.g., input and its transformation) while pushing apart the negative pairs under fully supervised setting to promote intraclass consistency and interclass inconsistency. During training, we jointly optimize the above two pretext tasks and semantic class prediction task in a multitask learning framework. To ease the dilemma of overfitting, we introduce adversarial model perturbation (AMP) [64], which has strong theoretical justifications for regularizing the network, into our model to calibrate the aforementioned pretext tasks to facilitate training. As shown in Fig. 1, we build a robust feature extractor by training a multitask model and then perform few-shot classification with cosine-based nearest-neighbor rule in the inference stage.

In summary, the main contributions of our work can be summarized as follows. 1) To our best knowledge, our work is the first attempt of few-shot remote sensing scene classification leveraging multiple pretext tasks. We shed new light on few-shot learning in remote sensing topics by devising auxiliary objectives and their synergies. 2) We justify the utilization of the AMP regularization technique in few-shot learning tasks, which facilitates network training and improves the resulting performance. 3) We conduct extensive experiments to demonstrate the effectiveness of our multitask learning framework and verify the contribution of each ingredient. We show that our framework achieves the best performance on public remote sensing datasets, including Northwestern Polytechnical University (NWPU)-RESISC45 [5], aerial image dataset (AID), [55], and Wuhan University (WHU)-remote sensing (RS)-19 [6].

The remainder of this article is organized as follows. Section II discusses related works. Section III is devoted to the details of few-shot classification via calibrated pretext task learning. Section IV presents our extensive experiments and analysis, and the conclusion is summarized in Section V.

II. Related Works

We here review some related works that are closely relevant to our work in the following topics.

A. Transfer Learning

Transfer learning has the goal to transfer knowledge learned from the source domain to the target domain [30], [46]. In few-shot learning, the source and target domains correspond to the seen and previously unseen tasks, respectively. Such two types of tasks are separately sampled from base and novel class datasets, whose semantic classes are relevant but disjoint. To solve this knowledge transfer problem, an intuitive baseline model is to train base class data using a standard cross-entropy loss, followed by fine-tuning a new linear classifier on each novel task [2]. In its variant baseline++, the linear classifier was replaced by a distance-based classifier. Besides, rather than the above two parametric fine-tuning strategies, a nearest-neighbor classifier [3] can be applied over the learned features to make predictions directly. Based on the baseline model, some researchers apply auxiliary techniques to improve the performance [33], [37]. In [33], self-supervision techniques augmented with Manifold Mixup [51] (S2M2) were adopted to enforce the model to learn representations that are robust to small changes in data distribution. Spatial contrastive learning (SCL) was proposed by [37] to promote class-independent discriminative patterns. In summary, the naive baseline model and its variants are prone to overfit due to data scarcity, which inspires the application of auxiliary techniques. However, most existing works overlooked the value of such techniques in remote sensing field and do not explore the usefulness of multitask learning, and our work tries to fill the gap.

B. Meta-Learning

Meta-learning aims to learn a representation that is broadly suitable for many tasks via mimicking practical few-shot scenarios, and the techniques can be further categorized into metric-based and gradient-based.

Metric-based learning methods learn a similarity metric that can be used to compare or cluster the query samples [29], [36], [44], [47], [52]. Therein, matching networks [52] proposed a nonparametric approach to solve few-shot problem, leveraging the long short-term memory (LSTM) [17] module over a learned embedding of support set to perform classification on query set with a cosine-similarity-based classifier. ProtoNet [44] adopted class mean classifier to address the overfitting problem of few-shot learning, where average class features of support set are used to classify the query samples using Euclidean distance. RelationNet [47] extended ProtoNet by formulating a convolutional neural network (CNN)-based module to compare the relationship between images such that it can learn a transferable deep metric. *Gradient-based* learning methods intend to optimize the model with a few training samples and a small number of training steps [8]. To this end, it learns a meta-learner capable of producing parameters for a task-specific network after observing the support set. The task-specific network is then evaluated on the query set, and the gradient is used to update the meta-learner. Model-agnostic meta-learning (MAML) [8] explicitly modeled such an optimization process using two learning loops during each iteration. Based on this, Reptile [34], Meta-stochastic gradient descent (SGD) [26], latent embedding optimization (LEO) [42], and meta-transfer learning (MTL) [45] further improved upon the adaptation ability. Furthermore, the memory-based module (e.g., LSTM-based meta-learner) is used to capture both short-term and long-term knowledge for training a meta-learner such that the knowledge can be generalized to unseen tasks [38].

Although meta-learning has achieved encouraging results, it is still confronted with the risk of overfitting and the performance is far from satisfactory. In this work, we carefully devise auxiliary objectives to enhance the representation, bypassing complex learning strategies in meta-learning.

C. Pretext-Task-Based Representation Learning

Pretext tasks were initially proposed for learning effective visual representations in an annotation-free manner. Typically, a pretext task is a predefined task for networks to learn an embedding where images that are semantically similar are close, while semantically different ones are far apart. Toward this end, an input X of a model is first transformed to X', whose outputs Y and Y' are supposed to be close (e.g., in Euclidean space). Examples of such pairs $\{X, X'\}$ include luminance and chrominance color channels of an image [48], patches [35], rotated copies [13], or Contrastive Learning of visual Representations (SimCLR)-type [1] transformations of an image, and different frames of a video [54], etc. Among them, image rotation prediction [13] and contrastive prediction [1] are two representative approaches, where the former uses spatial context structure of an image and the latter is based on context similarity [20].

In the context of few-shot learning, rotation prediction and self-supervised contrastive prediction have been individually studied to boost the performance [33], [37], [60]. Besides, strong supervision signals are also used to train other surrogate tasks, e.g., fully supervised contrastive prediction using class information [28], [37], or feature learning based on extra semantic annotations [59]. In our work, we use a self-supervised rotation prediction task and a fully supervised contrastive prediction task for remote sensing scene classification, and the AMP regularization technique is investigated to calibrate the tasks to facilitate training. Different from S2M2 [33] that relies on a two-stage training strategy, our model converges after one round training.

D. Few-Shot RS Scene Classification

In the few-shot remote sensing scene classification, we can roughly divide the available methods into transfer learning [18], [40] and meta-learning [9], [41]. In practice, meta-learning is currently the dominant strategy for few-shot scene classification of optical remote sensing image. Following the idea of metric-based learning paradigm, a Siamese-prototype network (SPNet) with prototype self-calibration and intercalibration was proposed by [4] to improve the performance of prototype-based classification. Scaled cosine similarity was adopted by [61] to measure the distance between query data and support data. Other distance metrics such as CNN relation module [23], [24], [32], [56] and Euclidean distance [27], [53] were also used to train various metric-based learning frameworks.

In SCL-MLNet [25], the contrastive prediction task has been applied to facilitate few-shot remote sensing scene classification. However, the effectiveness and potentials of pretext tasks have not been fully explored yet, and our work tries to shed new light on this topic.

III. Our Method

This section is dedicated to the details of our proposed method, including the subsections of notation introduction, full objective, rotation prediction task, contrastive prediction task, and network regularization.

A. Notation and Preliminary

In this section, we introduce the formulation of few-shot learning and briefly review the baseline model. In the few-shot classification, given a large amount of labeled data $\mathcal{D}_{\text{base}}$ with a set of classes $\mathcal{C}_{\text{base}}$, the goal is to train a model that could generalize to novel data $\mathcal{D}_{\text{novel}}$ with a set of novel classes $\mathcal{C}_{\text{novel}}$. Note that $\mathcal{C}_{\text{base}} \cap \mathcal{C}_{\text{novel}} = \varnothing$. In each task, we sample K classes from the dataset. Therein, the support data are denoted as $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{NK}$ (i.e., each class contains N examples) and the query data are $\mathcal{Q} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{MK}$. $\mathbf{x}_i$ and $y_i \in \{1, 2, \ldots, K\}$ represent the input data and corresponding category label, respectively. We term $\{\mathcal{S}, \mathcal{Q}\}$ as a $K-$way $N-$shot few-shot task. Notably, our method follows a transfer learning pipeline, where the samples are datapoints (instead of tasks as in meta-learning) in the training phase. In the inference phase, a set of tasks are sampled from $\mathcal{D}_{\text{novel}}$ to evaluate the model. Given support set $\mathcal{S}$ as training data, the final evaluation is done by testing the query set $\mathcal{Q}$.

For training, the baseline model trains a feature extractor f_θ (parameterized by the backbone network θ) and a linear classifier $E(.|\mathbf{W}_b)$ (parameterized by a weight matrix $\mathbf{W}_b \in \mathbb{R}^{d \times |\mathcal{C}_{\text{base}}|}$) via minimizing a standard cross-entropy

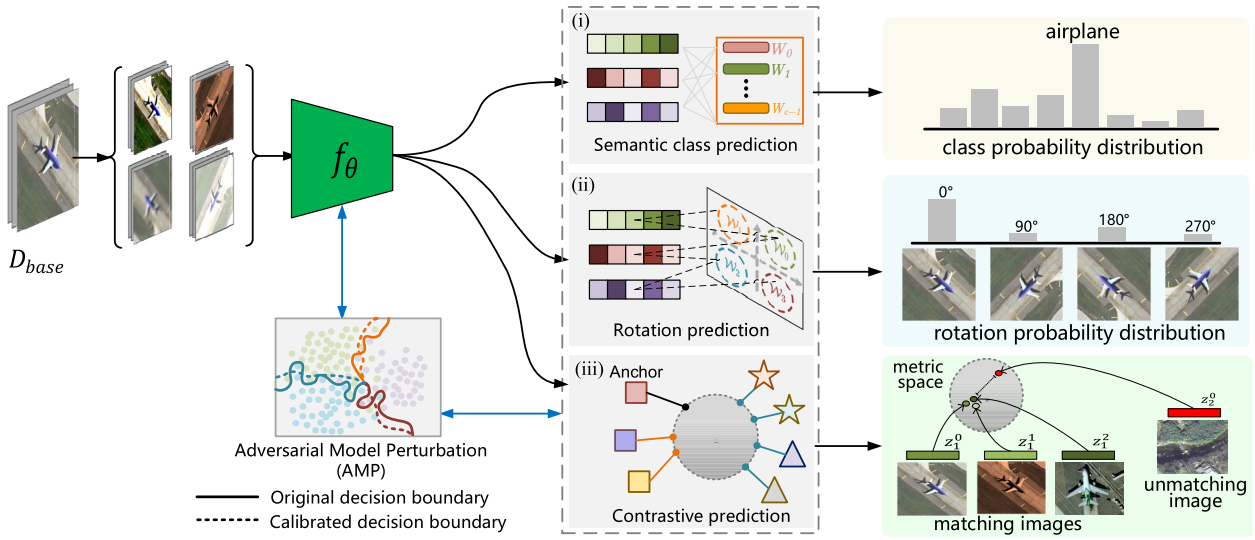


Fig. 2. Overview of the whole framework. It contains a shared feature extractor for three tasks, namely, (a) semantic class prediction to recognize the image category, (b) rotation prediction to identify the 2-D rotation degree, and (c) contrastive prediction to cluster the matching images and push apart unmatching images. Therein, AMP is applied on the parameter space to calibrate the pretext tasks.

loss. The classifier $E(.|\mathbf{W}_b)$ is constituted by a linear layer $\mathbf{W}_b^\top f_\theta(\mathbf{x}_i)$ and a Softmax operation ($\mathbf{x}_i \in \mathcal{D}_{\text{base}}$). For inference, a nearest-neighbor rule is used to perform classification for each novel task $\{\mathcal{S}, \mathcal{Q}\}$. Suppose $\mathcal{S}_k = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ denotes the support data of kth category, the average feature is computed as its prototype, i.e., $\boldsymbol{c}_k = (1/|\mathcal{S}_k|) \sum_{\mathbf{x}_i \in \mathcal{S}_k} f_\theta(\mathbf{x}_i)$. For a query $\mathbf{x} \in \mathcal{Q}$, the probability distribution is given by a Softmax operation over its distance with each class prototype

$$p_\theta(\hat{y} = k|\mathbf{x}) = \frac{\exp\left(\langle f_\theta(\mathbf{x}), \boldsymbol{c}_k\rangle\right)}{\sum_k \exp\left(\langle f_\theta(\mathbf{x}), \boldsymbol{c}_k\rangle\right)} \tag{1}$$

where $\langle \cdot, \cdot \rangle$ is the cosine similarity operator. The prediction of the query is performed by finding the nearest class prototype.

Algorithm 1: Pipeline of the Proposed Method

Input: Datasets $\mathcal{D}_{base}$, $\mathcal{D}_{val}$, and $\mathcal{D}_{novel}$
Output: Feature extractor f_θ

```
▷ Training
while θ not converged do
    for (x, y) ∈ D_base do
        Optimize θ by (11)
    end
    Select best θ using D_val
end
Output f_θ
▷ Inference
for {S, Q} ∈ D_novel do
    Perform classification by (1)
end
```

B. Full Objective

The overall framework of our proposed method is depicted in Fig. 2, which consists of three modules. The three tasks in these modules are jointly learned in an end-to-end manner. To this end, we propose a full objective as

$$\mathcal{L}_{\text{full}} = \mathcal{L}_b + \alpha\mathcal{L}_r + \beta\mathcal{L}_c \tag{2}$$

where $\mathcal{L}_b$, $\mathcal{L}_r$, and $\mathcal{L}_c$ are the losses of semantic class prediction learning, rotation prediction learning that relies on class-agnostic supervisory signal, and contrastive prediction learning that maps a representation vector to a low-dimensional vector and attracts the vectors of positive sample pairs in the embedding space, respectively. To calibrate the two pretext tasks, we apply the AMP technique in the parameter space, and the regularized objective is then written as (11). The hyperparameters α and β control the contribution of each pretext task. After the network converges, we obtain a feature extractor f_θ and resort to the nearest-neighbor rule for inference. Algorithm 1 summarizes the process of the training and inference stages.

C. Rotation Prediction Task

Given an image, the goal of rotation prediction is to tell which one of the several rotations this image undergoes, e.g., four rotations with {0°, 90°, 180°, and 270°}, and thus it can be formulated as a four-way classification task. Fig. 3 visualizes some images that are rotated by different rotation degrees. Formally, we define a set of rotation operators as $G = \{g_r\}_{r=1}^R$, where $\mathbf{x}^r = g_r(\mathbf{x})$ denotes the transformed image by a rotation degree and R is the number of rotations. In practice, the number and magnitude of recognized rotations are flexible, thereby forming different rotation recognition tasks. Given rotation classifier parameters $\mathbf{W}_g = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_R]$, the likelihood on input $\mathbf{x}_i$ is

$$p\left(\hat{y}_i^g = r|\mathbf{x}_i\right) = \frac{\exp\left(\mathbf{w}_r^\top f_\theta(g_r(\mathbf{x}_i))\right)}{\sum_{r=1}^R \exp\left(\mathbf{w}_r^\top f_\theta(g_r(\mathbf{x}_i))\right)}. \tag{3}$$

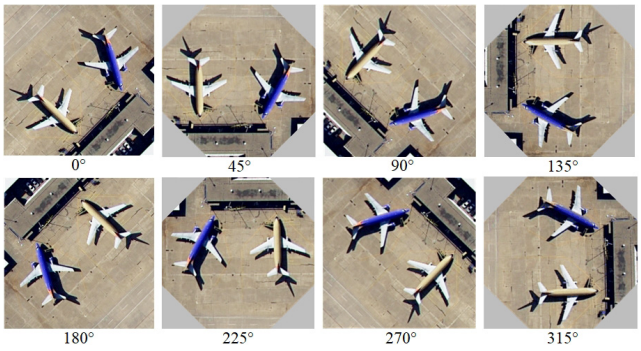


Fig. 3. Visualization of images rotated by different rotation degrees.

The loss function can then be formulated as a standard cross-entropy loss

$$\mathcal{L}_r = -\sum_{i=1}^{B}\sum_{r=1}^{R}\mathbb{I}\left(y_i^g = r\right)\log\left(p\left(\hat{y}_i^g = r|\mathbf{x}_i\right)\right). \tag{4}$$

Different from semantic class labels, here the supervisory signal is class-agnostic, which significantly promotes the information sharing across classes. More importantly, the insight behind rotation supervisory signal is that the neural networks should have recognized the classes and learned the object parts before it effectively performed rotation recognition task.

In [7], quantitative results are released to show a strong linear correlation between rotation prediction and semantic classification accuracies when training such two tasks simultaneously. Such observation hints that rotation supervisory signal may have a positive effect on semantic classification. Nevertheless, the authors do not see obvious gains in semantic classification accuracy using a multitask framework. In contrast, in our work, we observe that the rotation prediction task is helpful for the few-shot remote sensing classification accuracy. We conjecture this is due to task discrepancy: 1) whole classification denotes the task where training and testing classes are the same, which follows the common definition of generalization and 2) few-shot classification denotes the task where training and testing classes are disjoint. Because distributions of the training and testing sets in few-shot classification are much more far apart than that of whole classification, information sharing across classes is more critical in the few-shot setting. Such universal information usually contains some intrinsical characteristics. Therefore few-shot classification can benefit from the rotation prediction task.

Besides, in the training process, all the rotated copies of an image are fed into the network simultaneously. Because each transformed image has a semantic class label in addition to the rotation label, these rotated copies are expected to share certain rotation-irrelevant features. Consequently, the semantic classifier should be trained to recognize the category information of all transformed images and their originates. The semantic classification loss is then formulated as

$$\mathcal{L}_b = -\sum_{i=1}^{BR}\sum_{c=1}^{C}\mathbb{I}(y_i = c)\log(p(\hat{y}_i = c|\mathbf{x}_i)). \tag{5}$$

Given classifier parameters $\mathbf{W}_b = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_C]$, the likelihood is calculated as

$$p(\hat{y}_i = c|\mathbf{x}_i) = \frac{\exp\left(\mathbf{w}_c^\top f_\theta(g_r(\mathbf{x}_i))\right)}{\sum_{c=1}^{C}\exp\left(\mathbf{w}_c^\top f_\theta(g_r(\mathbf{x}_i))\right)}. \tag{6}$$

D. Contrastive Prediction Task

Contrastive learning aims to learn an embedding that can separate samples from two different distributions. Over the train set $\mathcal{D}_{\text{base}}$ that consists of a collection of samples $\{v_i = (\mathbf{x}_i, y_i)\}_{i=0}^{H}$, the goal is to contrast congruent and incongruent pairs, i.e., samples from the same distribution $\zeta \sim p(v_1, v_2)$ are considered as a positive pair, while samples $\mu \sim p(v_1)p(v_2)$ from the product of marginals are considered as a negative pair. Next we are going to introduce the details of recognizing these positive and negative pairs.

Let f_θ map the input $\mathbf{x}$ to a feature vector $\mathbf{r} \in \mathbb{R}^d$, which is then mapped into a lower dimensional embedding vector $\mathbf{z} \in \mathbb{R}^{d'}$, i.e., $\mathbf{z} = C(\mathbf{r}|\Phi)$. The projection network $C(.|\Phi)$ can be developed as either a multilayer perceptron (MLP) or just a single linear layer. Here, we instantiate it by a single fully connected layer with the dimension of 128. We normalize the output of this network, which makes it feasible to compute the cosine similarity using inner product.

For a minibatch with B input images, there are B pairs randomly augmented samples for training. To identify the positive counterpart $\mathbf{z}_p$ for each sample $\mathbf{z}_i$ ($p \in A(i)$, $A(i)$ is the set of indices of all positives in a training batch distinct from i), the contrastive loss holds the following form:

$$\mathcal{L}_c = \sum_{i=1}^{2B}\frac{-1}{|A(i)|}\sum_{p\in A(i)}\log\frac{\exp\left(\tau\langle\mathbf{z}_i, \mathbf{z}_p\rangle\right)}{\sum_{j=1}^{2B}\mathbb{I}(j \neq i)\exp\left(\tau\langle\mathbf{z}_i, \mathbf{z}_j\rangle\right)} \tag{7}$$

where τ is a scaling factor and set to 10 in this work.

It is clear that $\mathcal{L}_c$ is trained to obtain a high similarity value for positive pairs and low for negative pairs. We discuss such loss function in two cases, or consider the positive pairs in two cases, for each sample: 1) self-supervised setting that only the another augmented sample originating from the source sample is seen as a positive ($|A(i)| = 1$) and 2) fully supervised setting that all the samples belonging to the same semantic class as the origin are seen as positives ($|A(i)| \geq 1$). As a result, there would be an increasing number of positive pairs when semantic class information is accessible.

E. Network Regularization

The AMP aims to find a flat minima of empirical risk. Such a flat minima is a large connected region in weight space where the error remains approximately constant [16] and corresponds to a low-complexity (e.g., less fit parameters) network. It is commonly believed that intermediate model complexities strike a balance between underfitting and overfitting and thus benefit model generalization (e.g., less sensitive to quirks like noise of the training set) [39], [43]. In Fig. 2, we show the decision boundary before and after calibration during training. Obviously, the network tends to be high-complexity and overfitting (i.e., capture complicated statistical relationships in

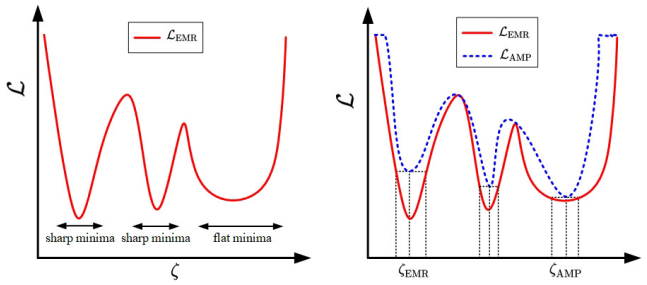


Fig. 4. Example that shows empirical loss curve (left) and the corresponding AMP loss curve (right). It can be observed that AMP loss can generate optimal parameters at the flatter minima.

the underlying data distribution) before calibration, while the calibrated decision boundary is smoother and a simpler network is preserved for testing. Subsequently, we will elaborate the details of applying the AMP regularization technique into our multitask learning model. Suppose $\boldsymbol{\zeta}$ denotes the whole training parameters of our model, which is defined in a weight space Θ. For each training sample $\{(\mathbf{x}, \mathbf{y})\} \in \mathcal{D}_{\text{base}}$, we use $\mathcal{L}_{\text{full}}(\mathbf{x}, \mathbf{y}; \boldsymbol{\zeta})$ to denote its full loss of our model with respect to the ground truth $\mathbf{y}$ (includes class label and those from pretext tasks). Based on the empirical risk minimization principle [50], we write an empirical loss

$$\mathcal{L}_{\text{EMR}}(\boldsymbol{\zeta}) := \frac{1}{|\mathcal{D}_{\text{base}}|} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{\text{base}}} \mathcal{L}_{\text{full}}(\mathbf{x}, \mathbf{y}; \boldsymbol{\zeta}). \tag{8}$$

To overcome overfitting, instead of minimizing the empirical loss $\mathcal{L}_{\text{EMR}}$, we minimize an AMP loss $\mathcal{L}_{\text{AMP}}$. As shown in Fig. 4, $\mathcal{L}_{\text{AMP}}$ essentially forces the network to a find flatter minima. Formally, for any positive ϵ and $\boldsymbol{\eta} \in \Theta$, suppose $\mathbf{B}(\boldsymbol{\eta}; \epsilon)$ be a norm ball in the high-dimensional space Θ with radius ϵ centered at $\boldsymbol{\eta}$

$$\mathbf{B}(\boldsymbol{\eta}; \epsilon) := \{\boldsymbol{\zeta} \in \Theta : \|\boldsymbol{\zeta} - \boldsymbol{\eta}\| \leq \epsilon\} \tag{9}$$

where the norm ball is defined over the L_2 norm.

Then, an AMP loss is defined as the following form:

$$\mathcal{L}_{\text{AMP}}(\boldsymbol{\zeta}) := \max_{\Delta\in\mathbf{B}(\mathbf{0};\epsilon)} \frac{1}{|\mathcal{D}_{\text{base}}|} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{\text{base}}} \mathcal{L}_{\text{full}}(\mathbf{x}, \mathbf{y}; \boldsymbol{\zeta} + \Delta). \tag{10}$$

During training, the network proceeds by optimizing over a large number of batches which consist of B randomly sampled images and their augments corresponding to each pretext task. Thus, AMP loss $\mathcal{L}_{\text{AMP}}$ could be approximated by the loss that is computed over a random batch $\mathcal{B}$, namely

$$\mathcal{L}_{\text{AMP},\mathcal{B}}(\boldsymbol{\zeta}) = \max_{\Delta_{\mathcal{B}}\in\mathbf{B}(\mathbf{0};\epsilon)} \frac{1}{|\mathcal{B}|} \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{B}} \mathcal{L}_{\text{full}}(\mathbf{x}, \mathbf{y}; \boldsymbol{\zeta} + \Delta_{\mathcal{B}}). \tag{11}$$

To optimize the above objective, we first obtain a perturbation vector $\Delta_{\mathcal{B}}$ on parameters $\boldsymbol{\zeta}$ and then minimize $\mathcal{L}_{\text{AMP},\mathcal{B}}(\boldsymbol{\zeta})$. We describe our detailed steps in Algorithm 2. To be specific, each training step contains two loops where the inner loop aims to update $\Delta_{\mathcal{B}}$ along the direction of increasing empirical loss $\mathcal{L}_{\text{EMR}}$ so as to access $\mathcal{L}_{\text{AMP},\mathcal{B}}$ (line 6–13); the outer loop follows a common SGD algorithm to minimize $\mathcal{L}_{\text{AMP},\mathcal{B}}$ (line 14–15). Readers are referred to [64] for detailed theoretical justifications about AMP.

Algorithm 2: AMP Training

Input: Dataset $\mathcal{D}_{base}$; Batch size B; Loss functions $\mathcal{L}_b$, $\mathcal{L}_r$, and $\mathcal{L}_c$; Learning rate μ_1 and μ_2; Loss weights α and β; Rotation operator $\{g_r\}_{i=1}^{R}$ and SimCLR-type transformation operator h_s

Output: Model parameters $\boldsymbol{\zeta}$

1 Randomly initialize $\boldsymbol{\zeta}$; $k = 0$
2 **while** $\boldsymbol{\zeta}_k$ not converged **do**
3 $k = k + 1$; Sample $\mathcal{B} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{B}$ from $\mathcal{D}_{base}$
4 Augment $\mathcal{B}$ by rotation operator $\{g_r\}_{i=0}^{R-1}$ and SimCLR-type transformation operator h_s
5 Initialize perturbation: $\Delta_{\mathcal{B}} \leftarrow \mathbf{0}$
6 **for** $n \leftarrow 1$ to T **do**
7 Compute full loss: $\mathcal{L}_{full} = \mathcal{L}_b + \alpha\mathcal{L}_r + \beta\mathcal{L}_c$
8 Compute gradient: $\nabla\mathcal{L}_{\text{AMP},\mathcal{B}} \leftarrow \sum_{i=1}^{B} \nabla_{\boldsymbol{\zeta}}\mathcal{L}_{full}(\mathbf{x}_i, \mathbf{y}_i; \boldsymbol{\zeta}_k + \Delta_{\mathcal{B}})/B$
9 Update perturbation: $\Delta_{\mathcal{B}} \leftarrow \Delta_{\mathcal{B}} + \mu_1\nabla\mathcal{L}_{\text{AMP},\mathcal{B}}$
10 **if** $\|\Delta_{\mathcal{B}}\|_2 \geq \epsilon$ **then**
11 Normalize perturbation: $\Delta_{\mathcal{B}} \leftarrow \epsilon\Delta_{\mathcal{B}}/\|\Delta_{\mathcal{B}}\|_2$
12 **end**
13 **end**
14 Compute gradient: $\nabla\mathcal{L}_{\text{AMP},\mathcal{B}} \leftarrow \sum_{i=1}^{B} \nabla_{\boldsymbol{\zeta}}\mathcal{L}_{full}(\mathbf{x}_i, \mathbf{y}_i; \boldsymbol{\zeta}_k + \Delta_{\mathcal{B}})/B$
15 Update parameters: $\boldsymbol{\zeta}_{k+1} \leftarrow \boldsymbol{\zeta}_k - \mu_2\nabla\mathcal{L}_{\text{AMP},\mathcal{B}}$
16 **end**

TABLE I

FIVE-WAY CLASSIFICATION ACCURACY (%) ON NWPU-RESISC45 WITH 95% CONFIDENCE INTERVALS, FOR CHOOSING THE BEST Φ ARCHITECTURE. † AND ‡ DENOTE USING RESNET12 AND CONV-4-256 AS BACKBONE, RESPECTIVELY. THE **BEST** RESULTS OF EACH MODEL ARE HIGHLIGHTED

Projection Network	Dim. of Φ	NWPU-RESISC45	
		1-shot	5-shot
Φ (2 FC layers)†	640, 128	69.86±0.45	86.20±0.23
Φ **(Ours)**†	128	**70.72±0.46**	**85.82±0.24**
Φ (2 FC layers)‡	256, 128	61.53±0.48	79.77±0.32
Φ **(Ours)**‡	128	**63.72±0.47**	**80.64±0.32**

IV. EXPERIMENTS

In this section, we evaluate our framework on standard few-shot learning tasks. Below we describe the remote sensing datasets used in this work and implementation details, followed by analysis of the learned representations, the comparisons with the state-of-the-art methods, and an ablation study regarding the ingredients of our method.

A. Datasets

We conduct experiments on three benchmarks and set splits for training, validation, and testing as prior works [24].

NWPU-RESISC45 covers more than 100 countries and regions all over the world with different resolutions. In total, there are 45 classes with 700 samples of 256 × 256 images

per class. All the classes are split into 25, 10, and 10 classes for training, validation, and testing, respectively. *AID* was collected from Google Earth imagery and consists of 10 000 images divided into 30 land-use classes with 200–420 samples of 600 × 600 images per class. The base, validation, and novel class splits are 16, 7, and 7, respectively. *WHU-RS-19* has been widely used to evaluate many scene classification approaches, with spatial resolution up to 0.5 m per pixel. In contrast, it is a smaller dataset with 19 classes (1005 red green blue (RGB) images) of 600 × 600 pixels. These classes are grouped into nine, five, and five classes, respectively, for training, validation, and testing.

We also set *whole-classification splits* on the three datasets to further verify our conjecture in Section III-C. Specifically, all the classes of each dataset are used for training, validation, and testing. The splits are set as follows. For NWPU-RESISC45, each class is partitioned into 450, 50, and 200 samples. For AID, its class distribution is unbalanced. Thus, each class is partitioned into 60%, 10%, and 30% samples. For WHU-RS-19, since the number of samples for each class starts from 50 (too few), each of them is divided into two equal parts for training and testing, without validation.

B. Implementation Details

1) Network Architectures: We describe the details of model architectures, including feature extractor parameters θ, semantic classifier parameters $\mathbf{W}_b$, rotation classifier parameters $\mathbf{W}_g$, and contrastive prediction head parameters Φ. For θ, we implement a shallow backbone Conv-4-256 and a deeper backbone ResNet12 [14]. Concretely, **Conv-4-256** consists of four layers with 3 × 3 convolutions and finally produces a 256-D feature vector after a global average pooling layer. The number of filters starts from 32 and is doubled every layer. **ResNet12** is composed of four residual blocks, and each block contains three convolutional layers with 3 × 3 kernels. It eventually produces a 640-D feature vector after a global average pooling layer. The number of filters of those blocks is 64, 160, 320, and 640, respectively. Moreover, DropBlock [11] is applied on the feature maps of the last two blocks. For $\mathbf{W}_b$ and $\mathbf{W}_g$, they are both instantiated by a single fully connected layer, and the latter with a dimension of 4 (four rotations in our work). For Φ, we empirically find that a single fully connected layer is more effective in our case. As shown in Table I, we observe that the performance drops with the increase in the number of layers.

2) Training Setup: We implement the image rotations by 90°, 180°, and 270° (0° is the image itself). The contrastive learning uses SimCLR type [1] transformation to obtain matching pairs for each sample. Thus, at each iteration the network totally sees eight times more images than the batch size (32 in this work). All the models are trained with an SGD optimizer with momentum 0.9. The learning rate is initialized as 0.01 and the weight decay is 5*e*-4. The total training epochs are set to 120 for the above three datasets. Their corresponding decay steps of learning rate are [80,100]. The decay factor is set to 10. For AMP, ϵ is set to 0.3. At each iteration, we train the inner loop with one step with a learning rate of 0.01, except by 0.1 for the NWPU-RESISC45 dataset when adopting ResNet12. This is because NWPU-RESISC45 is a relatively large dataset, and thus larger learning rate should be applied to make greater perturbation regarding the deeper backbone. The loss weights α and β of (2) are both set to 1. We have also tried other choices like 0.5 for these two weights, but the resultant accuracies have no obvious differences with the current results. For evaluation, we randomly sample 2000 tasks for five-way classification, with query samples per task. The performance is evaluated in terms of the average accuracy with 95% confidence intervals. The entire model is implemented using the PyTorch framework.

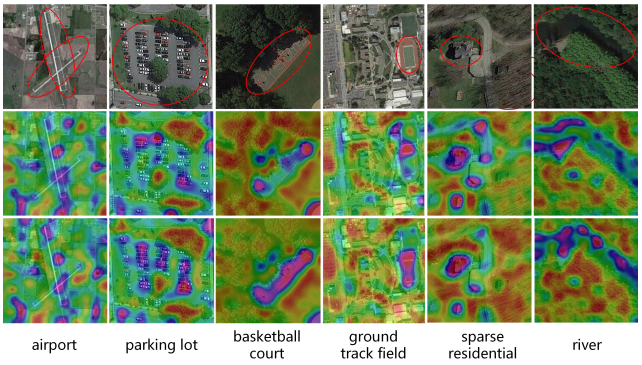


Fig. 5. Visualization of attention maps by *trans baseline* (top) and our full model (bottom). Each triplets denotes the input image and the corresponding attention maps. Red circles give the class-specific image regions.

3) Ablative Settings: To demonstrate the contribution of each ingredient, we carefully design several ablative settings: two baselines without pretext task learning but only classic learning, named as *trans**, three baselines of jointly training with different pretext tasks, named as *pre**, and three baselines of regularizing the above models with AMP, named as *AMP**. We note that all these models, as well as our method, use the nearest-neighbor rule during test phase. Table VII shows the results under such settings. The bullet names adopted in this table are explained as follows.

trans baseline (or *rotation*). These are naive transfer learning models. *trans baseline* trains with a standard cross-entropy loss. *trans rotation* trains with $\mathcal{L}_b$ of (5), which uses image rotation as augmentation but without self-supervision.

pre rotation (or *contrast* or [*rotation*; *contrast*]). These are used to evaluate the individual pretext task and their ensemble effect, without the AMP regularization technique.

AMP trans (or *rotation* or *contrast*). These are used to evaluate the AMP regularization technique, applied to different models.

C. Analysis of Learned Representations

1) Visualization of Representations: In Fig. 5, we visualize some attention maps generated by the *trans baseline* and our full model. Such attention maps are obtained based on the magnitude of activations at each spatial cell of the last convolutional layer and reflect where the network highlights (i.e., brighter areas in the Fig. 5) to recognize the images. While those classes have not been seen during training, the network can locate the right regions of them in the image.

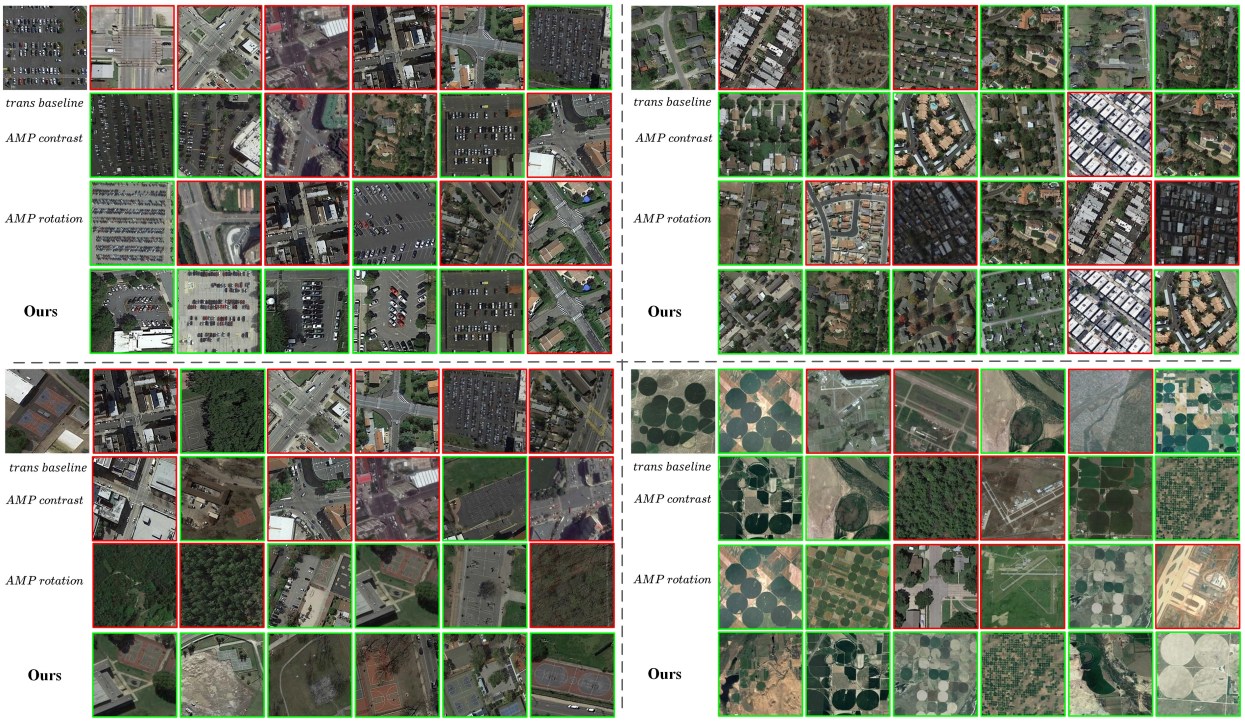


Fig. 6. Nearest-neighbor retrieval results. We show the six nearest neighbors of *trans baseline*, *AMP contrast*, *AMP rotation*, and our method. Queries are randomly selected from novel classes of NWPU-RESISC45. The semantic labels of the four queries are "parking lot," "medium residential," "basketball ground field," and "circular farmland," respectively. Semantically related and unrelated retrievals are separately marked with green and red boxes, respectively.

TABLE II

FIVE-WAY CLASSIFICATION ACCURACY (%) ON NWPU-RESISC45 AND AID DATASETS. THE **BEST** AND ***SECOND BEST*** RESULTS ARE HIGHLIGHTED. FEW-SHOT LEARNING (FSL) MEANS FEW-SHOT LEARNING. NOTE THAT THE STANDARD VARIANCE DEPENDS ON THE NUMBER OF TEST TASKS AND THE CLASS SPLITS. OUR SETTINGS FOLLOW SCL-MLNET [25] AND SPNET [4]

FSL method		Backbone	NWPU-RESISC45		AID	
			1-shot	5-shot	1-shot	5-shot
Transfer learning	*trans baseline* [2]	Conv-4-256	62.21±0.48	79.13±0.32	64.25±0.48	78.00±0.32
	trans baseline [2]	ResNet12	69.02±0.46	***85.62±0.25***	67.12±0.47	81.27±0.27
	S2M2 [33]	ResNet12	63.24±0.47	83.23±0.28	66.22±0.45	***82.87±0.29***
Gradient-based	MAML [8]	Conv-4-64	58.99±0.45	72.67±0.38	60.11±0.50	70.28±0.41
	Meta-SGD [26]	ResNet12	60.63±0.90	75.75±0.65	53.14±1.46	66.94±1.20
	LLSR [57]	Conv-4-64	51.43	72.90	-	-
Metric-based	MatchingNet [52]	Conv-4-64	60.21±0.77	71.66±0.45	63.31±0.46	73.35±0.35
	ProtoNet [44]	Conv-4-64	52.77±0.44	75.32±0.35	55.14±0.44	75.77±0.35
	RelationNet [47]	Conv-4-64	60.77±0.47	76.08±0.34	60.67±0.48	72.55±0.38
	MatchingNet [52]	ResNet12	61.57±0.49	76.02±0.34	64.30±0.46	74.49±0.35
	ProtoNet [44]	ResNet12	64.52±0.48	81.95±0.30	67.08±0.47	82.44±0.29
	RelationNet [47]	ResNet12	65.52±0.85	78.38±0.31	***68.56±0.49***	79.21±0.35
	RS-MetaNet [23]	ResNet50	52.78±0.09	71.49±0.81	53.37±0.56	72.59±0.73
	SCL-MLNet [25]	Conv-256	62.21±1.12	80.86±0.76	59.49±0.96	76.31±0.68
	DLA-MatchNet [24]	Conv-256	68.80±0.70	81.63±0.46	57.21±0.82	73.45±0.61
	SPNet [4]	ResNet18	67.84±0.87	83.94±0.50	-	-
	DLA-MatchNet [24]	ResNet12	71.56±0.30	83.77±0.64	-	-
	IDLN [56]	ResNet12	***75.25±0.75***	84.67±0.23	-	-
	Ours	Conv-4-256	64.79±0.49	81.40±0.30	66.73±0.49	81.71±0.29
	Ours	ResNet12	**76.70±0.44**	**89.87±0.21**	**72.67±0.43**	**87.33±0.23**

In particular, we can observe that the targets are more spatially scattered and indeed show great orientation variations compared with that of natural scene images. Under these circumstances, both the networks still work by putting focus on key regions. For example, for the "airport" class, they highlight the airstrip intersection areas. Besides, for other classes that are relatively compact like "basketball court," the networks can produce compact attention maps, even if the target is occluded by trees. Furthermore, comparing the two networks, we find that ours performs better, e.g., for the "ground track field" class, our model can accurately localize class-specific image region instead of missing certain information. For the "sparse residential" class, our model focuses more on the key region and do not highlight other irrelevant areas like road.

2) Nearest-Neighbor Retrieval: We conduct nearest-neighbor retrieval to evaluate the networks' ability of

TABLE III

FIVE-WAY CLASSIFICATION ACCURACY (%) ON THE WHU-RS-19 DATASET. THE **BEST** AND ***SECOND BEST*** RESULTS ARE HIGHLIGHTED

FSL method	Backbone	WHU-RS-19	
		1-shot	5-shot
trans baseline [2]	Conv-4-256	71.42±0.33	86.12±0.16
trans baseline [2]	ResNet12	75.57±0.36	88.65±0.18
S2M2 [33]	ResNet12	69.00±0.41	82.14±0.21
MAML [8]	Conv-4-64	59.92±0.35	82.30±0.23
Meta-SGD [26]	Conv-4-64	51.54±2.31	61.74±2.02
LLSR [57]	Conv-4-64	57.10	70.65
MatchingNet [47]	Conv-4-64	73.52±0.35	86.04±0.20
ProtoNet [44]	Conv-4-64	59.52±0.37	85.58±0.33
RelationNet [47]	Conv-4-64	66.97±0.35	79.62±0.21
MatchingNet [52]	ResNet12	76.14±0.35	84.00±0.20
ProtoNet [44]	ResNet12	77.00±0.36	91.70±0.15
RelationNet [47]	ResNet12	77.76±0.34	86.84±0.15
DLA-MatchNet [24]	Conv-256	68.27±1.83	79.89±0.33
SPNet [4]	ResNet18	***81.06±0.60***	88.04±0.28
DLA-MatchNet [24]	ResNet12	70.21±0.32	81.86±0.52
IDLN [56]	ResNet12	73.89±0.88	83.12±0.56
Ours	Conv-4-256	80.30±0.29	***92.49±0.13***
Ours	ResNet12	**86.29±0.30**	**92.96±0.12**

capturing object relationships. We randomly select five classes with 15 samples per class and compute the cosine distances between these samples and an arbitrary query that belongs to the selected five classes. As shown in Fig. 6, we give the six nearest neighbors that are arranged from left to right in terms of increasing distance.

It can be observed that our method tends to capture more fine-grained similarity. For example, for the first query sample that belongs to "parking lot" class, it is easy to confuse this class with "intersection" class, which is marked with red box. This is likely because the model retrieves similar background rather than the foreground (i.e., vehicles). Obviously, the top five retrievals of *trans baseline* appertain under "intersection" class, for *AMP contrast* and *AMP rotation* three of six fall into this class. However, for our model only the last retrieval is "intersection" example. In addition, for the second query sample that belongs to "medium residential" class, its most similar class is "dense residential." In this case, our model also gives the best results. Besides, for other queries, our method can successfully find the semantically similar samples, which demonstrates our model's discriminative ability.

D. Comparison to the State-of-the-Art

Tables II and III report the overall comparisons to relevant works. For transfer learning, we evaluate the performance of baseline and S2M2. For meta-learning, we follow previous works [4] to compare with the gradient-based and metric-based learning methods, i.e., MAML, Meta-SGD, MatchingNet, ProtoNet, and RelationNet. These methods are implemented by open-source code [2] except for Meta-SGD which is borrowed from [4]. We implement them by Conv-4-64, which is used in the original articles. This architecture is similar to our Conv-4-256, but has only 64 filters for every layer. We also apply ResNet12 to the metric-based learning methods for a fair comparison. Furthermore, we compare with those methods for few-shot remote sensing classification, including lifelong learning for scene recognition (LLSR), discriminative learning of adaptive (DLA)-MatchNet, SPNet, and iterative distribution learning network (IDLN) [56].

TABLE IV

FIVE-WAY CLASSIFICATION ACCURACY (%) ON NWPU-RESISC45, FOR CHOOSING THE MOST PROPER ROTATION RECOGNITION TASK

#	Rotations	NWPU-RESISC45	
		1-shot	5-shot
8	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	71.14±0.45	86.88±0.24
2	0°, 180°	70.54±0.47	86.44±0.25
2	90°, 270°	69.77±0.46	86.30±0.24
4	0°, 90°, 180°, 270°	**71.58±0.32**	**87.07±0.26**

NWPU-RESISC45. As seen in Table II, our model achieves new state-of-the-arts for both one-shot and five-shot tasks, respectively. It is clear that our model outperforms meta-learning algorithms and those methods in remote sensing field by large margins, which reveals the great potential of transfer learning. Besides, compared with another transfer learning method S2M2, our one-shot result surpasses it by around 13.5% when ResNet12 is adopted. This confirms the superiority of the multitask learning framework. Regarding the network architecture, those models that use deeper backbones (i.e., ResNet12) obviously outperform Conv-4-64 and Conv-4-256 models.

AID. In Table II, we also show the results on AID. "Ours + ResNet12" achieves the best performance on both one-shot and five-shot tasks. Moreover, an interesting observation is that the performance of DLA-MatchNet undergoes a sharp decline from NWPU-RESISC45 to this smaller dataset AID, while our method consistently performs well on these two datasets. This validates the good generalization ability of our framework.

WHU-RS-19. In Table III, we show the results on the smaller dataset WHU-RS-19. From the table, we again confirm that our method outperforms others. Besides, we also observe that the gaps between Conv-4-256 and ResNet12 are much smaller than that on NWPU-RESISC45 and AID, e.g., "Ours + Conv-4-256" achieves one-shot result of 80.30%, which is 6% lower than the result of "Ours + ResNet12." Especially in the five-shot task, using ResNet12 only brings around 0.5% gains over Conv-4-256. Differently, the margins raise to about 12.0% and 8.5% on NWPU-RESISC45. It is likely that the larger dataset is more sensitive to the depth of networks.

E. Ablation Study

From Tables IV–VII, we show the ablative studies on the above three datasets. We first confirm which are the best rotation and contrastive prediction tasks for our framework, followed by detailed ablative analysis on them and the AMP technique, and eventually the efficiency of our method.

1) Performance of Different Rotation Recognition Tasks: We analyze how the number of discrete rotations in the prediction task affects the quality of learned features. To this end, we define extra three rotation recognition tasks to train *pre rotation*: (a) one with all the eight rotations shown in Fig. 3(b), one with only 0° and 180°, and Fig. 3(c) one with only 90° and 270°. As shown in Table IV, we observe

TABLE V

FIVE-WAY CLASSIFICATION ACCURACY (%) ON TWO CLASSIFICATION TASKS. FEW-SHOT. AND WHOLE-CLS. DENOTE THE FEW-SHOT CLASSIFICATION AND WHOLE-CLASSIFICATION TASKS, RESPECTIVELY

Task	Settings	NWPU-RESISC45		AID		WHU-RS-19	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
Few-shot.	*trans baseline*	69.02±0.46	85.62±0.25	67.12±0.47	81.27±0.27	75.57±0.36	88.65±0.18
	trans rotation	70.84±0.46	86.97±0.23	68.08±0.46	82.23±0.33	75.98±0.31	89.15±0.16
	pre rotation	71.58±0.45	87.07±0.26	67.13±0.44	83.41±0.26	81.70±0.32	92.09±0.15
Whole-cls.	*trans baseline*	90.12±0.31	96.88±0.13	87.12±0.38	95.50±0.17	86.24±0.38	93.87±0.19
	tran rotation	94.15±0.24	98.14±0.10	87.34±0.38	95.77±0.17	86.30±0.37	93.69±0.19
	pre rotation	94.16±0.24	98.15±0.10	87.48±0.38	95.96±0.17	86.45±0.38	93.96±0.18

TABLE VI

FIVE-WAY CLASSIFICATION ACCURACY (%) USING ABLATIVE CONTRASTIVE LOSSES ON THE NWPU-RESISC45 DATASET. SELF-SUPERVISED (SS) AND FULLY SUPERVISED (FS). MEAN SELF-SUPERVISED AND FULLY SUPERVISED LOSSES, RESPECTIVELY. RESNET12 IS USED. THE **BEST** AND ***SECOND BEST*** RESULTS ARE HIGHLIGHTED

Loss	Settings	NWPU-RESISC45	
		1-shot	5-shot
-	*trans baseline*	69.02±0.46	85.62±0.25
SS.	*pre contrast*	64.22±0.47	82.96±0.30
	pre [*contrast*; *rotation*]	67.80±0.45	85.52±0.28
	AMP contrast	67.72±0.44	85.45±0.28
	Ours	69.97±0.44	86.84±0.26
FS.	*pre contrast*	70.72±0.46	85.82±0.24
	pre [*contrast*; *rotation*]	73.50±0.44	87.90±0.23
	AMP contrast	***74.27±0.44***	***88.35±0.23***
	Ours	**76.70±0.44**	**89.87±0.21**

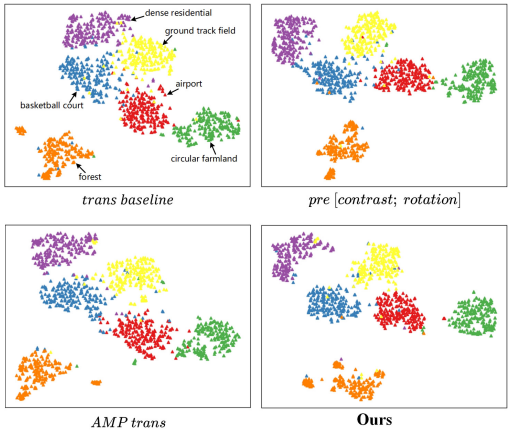


Fig. 7. t-SNE visualization of the image features learned with (bottom) and without (top) AMP regularization technique. The features are from six NWPU-RESISC45 testing classes with 200 randomly selected images per class.

that the four rotations' setting outperforms the other three settings. We believe that this is because the two rotations' cases contain too few classes for recognition, which cannot provide enough supervisory information for the model. As for the eight rotations' setting, the geometric transformations in this case are not distinguishable enough, and thus the performance is inferior to the four rotations' setting. Another observation is that, for the two rotations' settings, the model trained with 0° and 180° obtains better performance than that trained with 90° and 270°. It is possibly because in the latter case the model cannot see what the image really looks like (i.e., 0° rotation) during training. Thus, we use the four rotations' setting in this work.

2) Rotation Recognition on Different Classification Tasks: In Table V, we explore how the rotation prediction task affects the whole-classification and few-shot classification tasks on remote sensing data. Clearly, we can see that rotation prediction task can facilitate few-shot classification but has no obvious gains on whole-classification accuracy. As discussed in Section III-C, this is caused by the discrepancy underlying such two tasks. Specifically, when train classes are congruent to test classes, the rotation prediction task has little impact on accuracy. Conversely, few-shot learning has the goal to transfer knowledge from base classes to novel classes, and rotation prediction task can promote knowledge transferring across classes because it uses class-agnostic supervision. **Performance of different contrastive losses**. Table VI shows the results obtained with fully supervised and self-supervised contrastive losses. We observe a great drop in one-shot and five-shot results when adopting the self-supervised setting. One potential reason is that the self-supervised loss constructs positive pairs with input transformation, while different instances belonging to the same class are viewed as negative pairs. This is contradictory to the idea of semantic classification, i.e., those instances belonging to the same class should be similar in the embedding space. Moreover, the benefit of self-supervised contrastive learning is that it can merely use the augmented samples and works in an unsupervised way. However, in our few-shot classification task the model operates in a fully supervised setting rather than an unsupervised setting. Therefore, it is very reasonable that fully supervised contrastive learning outperforms self-supervised contrastive learning. In this article, we use fully supervised loss to perform contrastive prediction.

3) Effect of Pretext Tasks: As shown in Table VII, we first study the effect of the two types of pretext tasks. We report the results using Conv-4-256 and ResNet12, which are shown on the top rows of each block. Overall, the three models incorporating the pretext task learning clearly obtain better results. This confirms the necessity of pretext task learning. For the multitask learning model, i.e., *pre* [*contrast; rotation*], it achieves better performance than individual pretext task learning except for the results on NWPU-RESISC45 with Conv-4-256, i.e., 63.21% is slightly lower than 63.72%. One possible reason is that for a relatively large dataset NWPU-RESISC45, it is difficult for a shallow backbone Conv-4-256 to optimize three tasks simultaneously. In contrast, for

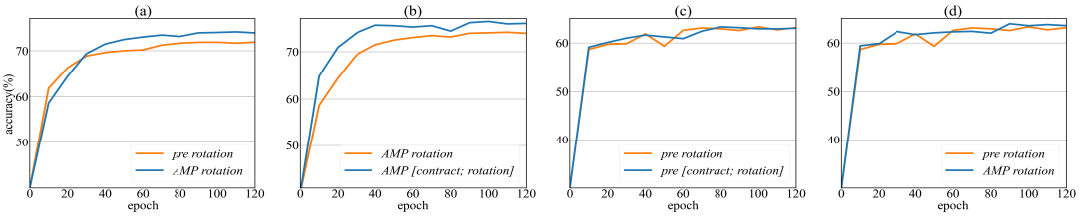


Fig. 8. Five-way one-shot classification test accuracy plots on the NWPU-RESISC45 dataset, using [(a), (b)] ResNet12 and [(c), (d)] Conv-4-256.

TABLE VII

FIVE-WAY CLASSIFICATION ACCURACY (%) USING ABLATIVE MODELS, ON THREE DATASETS. THE TOP BLOCK AND BOTTOM BLOCK USE CONV-4-64 AND RESNET12 AS BACKBONE, RESPECTIVELY. FOR EACH BLOCK, **BEST** AND ***SECOND BEST*** RESULTS ARE HIGHLIGHTED

Settings	NWPU-RESISC45		AID		WHU-RS-19	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
trans baseline	62.21±0.46	79.13±0.31	64.25±0.48	78.00±0.32	71.42±0.33	86.12±0.16
trans rotation	63.30±0.48	80.17±0.33	64.54±0.46	78.41±0.33	70.35±0.31	84.69±0.18
pre contrast	63.72±0.48	80.64±0.32	64.94±0.48	79.08±0.32	77.88±0.33	90.23±0.15
pre rotation	63.09±0.48	80.59±0.32	64.94±0.49	79.47±0.30	75.16±0.32	90.69±0.15
pre [*contrast; rotation*]	63.21±0.49	80.04±0.32	66.42±0.47	***81.49±0.30***	80.03±0.31	***92.28±0.14***
AMP trans	63.08±0.48	80.04±0.32	65.47±0.49	79.80±0.30	76.57±0.31	90.65±0.15
AMP contrast	63.94±0.47	80.73±0.32	**67.05±0.47**	81.47±0.30	***80.23±0.32***	91.72±0.13
AMP rotation	***64.22±0.48***	***80.84±0.32***	65.67±0.49	79.72±0.31	76.89±0.32	90.71±0.15
Ours	**64.98±0.48**	**81.81±0.32**	***66.73±0.49***	**81.71±0.29**	**80.30±0.29**	**92.49±0.13**
trans-baseline	69.02±0.46	85.62±0.25	67.12±0.47	81.27±0.27	75.57±0.36	88.65±0.18
trans rotation	70.84±0.46	86.97±0.23	68.08±0.46	82.23±0.33	75.98±0.31	89.15±0.16
pre contrast	70.72±0.46	85.82±0.24	68.19±0.47	83.51±0.26	78.19±0.38	88.84±0.16
pre rotation	71.58±0.45	87.07±0.26	67.13±0.44	83.41±0.26	81.70±0.32	92.09±0.15
pre [*contrast; rotation*]	73.50±0.44	87.90±0.23	***69.51±0.44***	***85.74±0.25***	**88.00±0.29**	**94.17±0.12**
AMP trans	71.68±0.45	86.56±0.25	68.19±0.47	83.51±0.26	77.62±0.35	90.08±0.17
AMP contrast	74.27±0.44	88.35±0.23	68.98±0.46	84.53±0.27	81.75±0.37	90.67±0.14
AMP rotation	***74.40±0.44***	***88.49±0.23***	68.74±0.46	84.88±0.25	78.03±0.44	88.49±0.23
Ours	**76.70±0.44**	**89.87±0.21**	**72.67±0.43**	**87.38±0.23**	***86.29±0.30***	***92.96±0.12***

a medium dataset AID and a much smaller dataset WHU-RS-19, multitask learning is superior. Comparing the results of the ResNet12 and Conv-4-256 models, we conclude that a deeper backbone can better handle the three tasks simultaneously.

4) Effect of AMP: We hereby explore the impact of the AMP technique by adding it to different models. The results are shown in the bottom rows of each block of Table VII. Clearly, it can improve the classification accuracy, both over the baseline model and *pre** models. This verifies that the calibrated pretext tasks can indeed boost the resulting performance. However, we find that *AMP rotation* performs worse than *pre rotation* on WHU-RS-19 when we use ResNet12, i.e., 78.03% versus 81.70%. It is potentially because there are too many parameters to optimize for such a small dataset, and the regularization technique fails to facilitate the rotation prediction task learning to improve the final accuracy. This also results in inferior performance of the corresponding full model compared with *pre* [*contrast; rotation*]. In contrast, when we adopt Conv-4-256, our full model has a slight advantage over the *pre* [*contrast; rotation*]. In addition, as shown in Fig. 7, t-Stochastic Neighbor Embedding (t-SNE) [49] visualization of image features affirms the effectiveness of the AMP technique in alleviating overfitting (e.g., reducing intraclass and increasing interclass variances) and forming more defined decision boundaries, and this aligns with our conceptually visualization in Fig. 2.

5) Efficiency: In Fig. 8, we show the test accuracy with respect to training epochs on the NWPU-RESISC45 dataset. We can see that all the models converge to a good performance after 100 epochs of training. Particularly, Fig. 8(a) shows that *AMP rotation* obviously outperforms *pre rotation*, and Fig. 8(b) shows that our full model obtains higher accuracies than *AMP rotation*. Such observations again confirm the superiority of the AMP regularization technique and pretext tasks. Besides, compared with Fig. 8(a) and (b), the curves of Fig. 8(c) and (d) demonstrate that each individual ingredient of our framework does not have impressive advantages over the baseline when we train a shallow network Conv-4-256. Furthermore, we conduct inference time experiments to investigate the computational efficiency of our model by calculating the inference time required for a single five-way, one-shot task, averaged over 2000 tasks. Using the ResNet12 (47.43 M parameters) and Conv-4-256 (0.43 M parameters) backbones, the results of 15 queries are both around 0.001 s per task. Since we use the nearest-neighbor rule for inference and avoid training a classifier as prior works [2], the inference speed can readily meet real-time requirements.

V. CONCLUSION

In this work, we have presented a novel framework leveraging calibrated pretext tasks for tackling the few-shot remote sensing classification problem. The two types of pretext tasks, i.e., rotation prediction and contrastive prediction, proved to be impressively effective for robust feature learning. The comparison between few-shot classification and whole-classification results verifies that rotation prediction task can

learn class-transferable knowledge which is useful for few-shot learning. The comparison between fully supervised and self-supervised contrastive losses shows that the semantically similar features are supposed to be close in embedding space explicitly during training, when contrastive loss is used as an auxiliary objective. To alleviate model overfitting, the AMP regularization technique is introduced to minimize an AMP loss instead of empirical loss. Unifying the two pretext tasks and the AMP technique in a multitask learning framework, our method achieves the best results on three representative remote sensing benchmarks, including NWPU-RESISC45, AID, and WHU-RS-19. The design of our framework is independent of any specific model or architecture and can be further generalized to other few-shot learning models in future work.

ACKNOWLEDGMENT

The numerical calculations in this article had been supported by the super-computing system in the Supercomputing Center of Wuhan University.

REFERENCES

[1] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[2] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. Wang, and J.-B. Huang, "A closer look at few-shot classification," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–16.

[3] Y. Chen, Z. Liu, H. Xu, T. Darrell, and X. Wang, "Meta-baseline: Exploring simple meta-learning for few-shot learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9062–9071.

[4] G. Cheng *et al.*, "SPNet: Siamese-prototype network for few-shot remote sensing image scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–11, 2022.

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

[6] D. Dai and W. Yang, "Satellite image classification via two-layer sparse coding with biased image representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 1, pp. 173–176, Jan. 2011.

[7] W. Deng, S. Gould, and L. Zheng, "What does rotation prediction tell us about classifier accuracy under varying testing environments?" in *Proc. 38th Int. Conf. Mach. Learn.*, 2021, pp. 2579–2589.

[8] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 1126–1135.

[9] K. Fu, T. Zhang, Y. Zhang, Z. Wang, and X. Sun, "Few-shot SAR target classification via metalearning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2021.

[10] Z. Gao, H. Ji, T. Mei, B. Ramesh, and X. Liu, "Eovnet: Earth-observation image-based vehicle detection network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 9, pp. 3552–3561, Sep. 2019.

[11] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 10727–10737.

[12] S. Gidaris, A. Bursuc, N. Komodakis, P. P. Perez, and M. Cord, "Boosting few-shot visual learning with self-supervision," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8059–8068.

[13] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–14.

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

[15] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song, "Using self-supervised learning can improve model robustness and uncertainty," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 15663–15674.

[16] S. Hochreiter and J. Schmidhuber, "Flat minima," *Neural Comput.*, vol. 9, no. 1, pp. 1–42, 1997.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[18] Z. Huang, Z. Pan, and B. Lei, "Transfer learning with deep convolutional neural network for SAR target classification with limited labeled data," *Remote Sens.*, vol. 9, no. 9, p. 907, Sep. 2017.

[19] H. Ji, Z. Gao, T. Mei, and Y. Li, "Improved faster R-CNN with multiscale feature fusion and homography augmentation for vehicle detection in remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 11, pp. 1761–1765, Nov. 2019.

[20] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 4037–4058, Nov. 2020.

[21] P. Khosla *et al.*, "Supervised contrastive learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 18661–18673.

[22] H. Lee, S. J. Hwang, and J. Shin, "Self-supervised label augmentation via input transformations," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 5714–5724.

[23] H. Li *et al.*, "RS-MetaNet: Deep metametric learning for few-shot remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6983–6994, Aug. 2021.

[24] L. Li, J. Han, X. Yao, G. Cheng, and L. Guo, "DLA-MatchNet for few-shot remote sensing image scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7844–7853, Sep. 2021.

[25] X. Li, D. Shi, X. Diao, and H. Xu, "SCL-MLNet: Boosting few-shot remote sensing scene classification via self-supervised contrastive learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, 2022.

[26] Z. Li, F. Zhou, F. Chen, and H. Li, "Meta-SGD: Learning to learn quickly for few-shot learning," 2017, *arXiv:1707.09835*.

[27] B. Liu, X. Yu, A. Yu, P. Zhang, G. Wan, and R. Wang, "Deep few-shot learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2290–2304, Apr. 2019.

[28] C. Liu *et al.*, "Learning a few-shot embedding model with contrastive learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 8635–8643.

[29] L. Liu, T. Zhou, G. Long, J. Jiang, L. Yao, and C. Zhang, "Prototype propagation networks (PPN) for weakly-supervised few-shot learning on category graph," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 3015–3022.

[30] Y. Liu, Y. Su, A.-A. Liu, B. Schiele, and Q. Sun, "Mnemonics training: Multi-class incremental learning without forgetting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12245–12254.

[31] Z. Lv, W. Shi, X. Zhang, and J. A. Benediktsson, "Landslide inventory mapping from bitemporal high-resolution remote sensing images using change detection and multiscale segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 5, pp. 1520–1532, Mar. 2018.

[32] X. Ma, S. Ji, J. Wang, X. Liu, and H. Wang, "Classification of hyperspectral image based on task-specific learning network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8646–8656, Oct. 2021.

[33] P. Mangla, M. Singh, A. Sinha, N. Kumari, V. N. Balasubramanian, and B. Krishnamurthy, "Charting the right manifold: Manifold mixup for few-shot learning," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 2218–2227.

[34] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," 2018, *arXiv:1803.02999*.

[35] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 69–84.

[36] B. N. Oreshkin, P. Rodriguez, and A. Lacoste, "TADAM: Task dependent adaptive metric for improved few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 719–729.

[37] Y. Ouali, C. Hudelot, and M. Tami, "Spatial contrastive learning for few-shot classification," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Cham, Switzerland: Springer, 2021, pp. 671–686.

[38] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–11.

[39] J. W. Rocks and P. Mehta, "Memorizing without overfitting: Bias, variance, and interpolation in overparameterized models," *Phys. Rev. Res.*, vol. 4, no. 1, Mar. 2022, Art. no. 013201.

[40] M. Rostami, S. Kolouri, E. Eaton, and K. Kim, "Deep transfer learning for few-shot SAR image classification," *Remote Sens.*, vol. 11, no. 11, p. 1374, Jun. 2019.

[41] M. Ruswurm, S. Wang, M. Korner, and D. Lobell, "Meta-learning for few-shot land cover classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 200–201.

[42] A. A. Rusu *et al.*, "Meta-learning with latent embedding optimization," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–13.

[43] C. Schaffer, "Overfitting avoidance as bias," *Mach. Learn.*, vol. 10, no. 2, pp. 153–178, Feb. 1993.

[44] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 4080–4090.
[45] Q. Sun, Y. Liu, T.-S. Chua, and B. Schiele, "Meta-transfer learning for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 403–412.
[46] Q. Sun, B. Schiele, and M. Fritz, "A domain based approach to social relation recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3481–3490.
[47] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.
[48] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 776–794.
[49] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.
[50] V. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer, 1999.
[51] V. Verma *et al.*, "Manifold mixup: Better representations by interpolating hidden states," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 6438–6447.
[52] O. Vinyals *et al.*, "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3630–3638.
[53] L. Wang, X. Bai, C. Gong, and F. Zhou, "Hybrid inference network for few-shot SAR automatic target recognition," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9257–9269, Nov. 2021.
[54] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2794–2802.
[55] G. S. Xia *et al.*, "AID: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965–3981, Apr. 2017.
[56] Q. Zeng, J. Geng, W. Jiang, K. Huang, and Z. Wang, "IDLN: Iterative distribution learning network for few-shot remote sensing image scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.
[57] M. Zhai, H. Liu, and F. Sun, "Lifelong learning for scene recognition in remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 9, pp. 1472–1476, Sep. 2019.
[58] X. Zhai, A. Oliver, A. Kolesnikov, and L. Beyer, "S4L: Self-supervised semi-supervised learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1476–1485.
[59] H. Zhang, P. Koniusz, S. Jian, H. Li, and P. H. S. Torr, "Rethinking class relations: Absolute-relative supervised and unsupervised few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9432–9441.
[60] M. Zhang, J. Zhang, Z. Lu, T. Xiang, M. Ding, and S. Huang, "IEPT: Instance-level and episode-level pretext tasks for few-shot learning," in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–12.
[61] P. Zhang, Y. Bai, D. Wang, B. Bai, and Y. Li, "Few-shot classification of aerial scene images via meta-learning," *Remote Sens.*, vol. 13, no. 1, p. 108, Dec. 2020.
[62] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 649–666.
[63] B. Zhao, Y. Zhong, G.-S. Xia, and L. Zhang, "Dirichlet-derived multiple topic scene classification model for high spatial resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2108–2123, Apr. 2015.
[64] Y. Zheng, R. Zhang, and Y. Mao, "Regularizing neural networks via adversarial model perturbation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8156–8165.
[65] Y. Zhong *et al.*, "Open-source data-driven urban land-use mapping integrating point-line-polygon semantic objects: A case study of Chinese cities," *Remote Sens. Environ.*, vol. 247, Sep. 2020, Art. no. 111838.

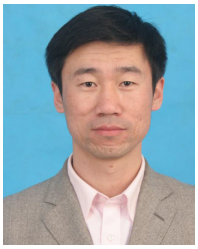
Hong Ji received the B.E. and M.S. degrees from the School of Electronic Information, Wuhan University, Wuhan, China, in 2017 and 2020, respectively, where she is currently pursuing the Ph.D. degree with the School of Remote Sensing and Information Engineering.

Her research interests include computer vision, artificial intelligence, few-shot learning, and their applications.

Zhi Gao received the B.E. and Ph.D. degrees from Wuhan University, Wuhan, China, in 2002 and 2007, respectively.

In 2008, he joined the Interactive and Digital Media Institute, National University of Singapore (NUS), Singapore, as a Research Fellow (A) and the Project Manager. In 2014, he joined the Temasek Laboratories in NUS (TL@NUS) as a Research Scientist (A) and the Principal Investigator. He is currently working as a Full Professor with the School of Remote Sensing and Information Engineering, Wuhan University. He has published more than 90 academic articles, which have been published in *International Journal of Computer Vision* (IJCV), IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (PAMI), *International Society for Photogrammetry and Remote Sensing* (ISPRS) *Journal of Photogrammetry and Remote Sensing* (JPRS), IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (ITS), and other top journals. His research interests include computer vision, machine learning, and remote sensing and their applications. In particular, he has strong interests in vision for intelligent systems and intelligent-system-based vision.

Dr. Gao received the prestigious "National Plan for Young Talents" Award and the Hubei Province Funds for Distinguished Young Scientists. In addition, he is also a "Chutian Scholar" Distinguished Professor in Hubei. He serves as an Associate Editor for *Unmanned Systems* journal.



Yongjun Zhang received the B.S. degree in geodesy, the M.S. degree in geodesy and surveying engineering, and the Ph.D. degree in geodesy and photography from Wuhan University, Wuhan, China, in 1997, 2000, and 2002, respectively.

He is currently the Dean of the School of Remote Sensing and Information Engineering, Wuhan University. He has published more than 180 research articles and one book. He holds 30 Chinese patents and 32 copyright registered computer software. His research interests include aerospace and low-attitude photogrammetry, image matching, combined block adjustment with multi-source data sets, object information extraction and modeling with artificial intelligence, integration of light detection and ranging (LiDAR) point clouds and images, and 3-D city model reconstruction.

Dr. Zhang is the Co-Editor-in-Chief of *The Photogrammetric Record*.

Yu Wan received the B.E. degree from the School of Computer Science, Wuhan University, Wuhan, China, in 2021, where he is currently pursuing the M.S. degree with the School of Remote Sensing and Information Engineering.

His research interests include computer vision, artificial intelligence, few-shot learning, and their applications.

Can Li received the B.E. degree from the School of Geography and Information Engineering, China University of Geosciences, Wuhan, China, in 2021. He is currently pursuing the M.S. degree with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan.

His research interests include computer vision, artificial intelligence, few-shot learning, and their applications.

Tiancan Mei received the Ph.D. degree from Wuhan University, Wuhan, China, in 2005, where he is currently an Associate Professor with the School of Electronic Information.

His research interests include remote sensing image understanding, statistical pattern classification, machine vision, and machine learning.