EFM-Net: An Essential Feature Mining Network for Target Fine-Grained Classification in Optical Remote Sensing Images

Yonghao Yi, *Student Member, IEEE*, Yanan You, *Member, IEEE*, Chao Li, and Wenli Zhou

***Abstract*— Target fine-grained classification has been the research hotspot in remote sensing image interpretation, which has received general attention in application fields. One challenge of the fine-grained classification task is to learn the most discriminative feature using the deep convolutional neural network (DCNN). At present, many works of fine-grained image classification obtain target features by optimizing the feature extraction and enhancement, which are not accurate enough in remote sensing images. In this article, we propose an essential feature mining network (EFM-net) based on DCNN to address this issue. Its major motivation is to obtain the essential feature, which is fine enough to distinguish between similar instances. The proposed pipeline includes the Miner for purifying the essential feature and the Refiner for data augmentation. These two modules can work in a mutually reinforcing way and extract the essential feature of targets. We evaluate EFM-Net on two public fine-grained classification datasets in remote sensing, FGSC-23 and FGSCR-42, and our Aircraft-16. The results show that the proposed method consistently outperforms existing alternatives. We have released our source code in GitHub https://github.com/JACYI/EFM-Net-Pytorch.git.**

***Index Terms*— Attention mechanism, data augmentation, deep learning, essential feature extraction, fine-grained target classification.**

I. Introduction

TARGET fine-grained classification is an important task in remote sensing image interpretation. In the field of remote sensing classification, there are many important tasks, such as hyperspectral image classification [1], [2], image scene classification [3], and the challenging target fine-grained classification. Different from the target classification of large categories [4], target fine-grained classification aims to distinguish subordinate object categories, such as different species of ships [5]. The characteristic of this task is the high interclass similarity and high sample diversity. Existing works [6], [7], [8], [9] based on deep convolutional neural network, i.e., DCNN, optimize feature extraction ability and enhance features to improve the performance of models. The main challenge is to obtain highly discriminative features.

At present, a lot of works focus on extracting the characteristics of targets themselves in images (i.e., the intrinsic feature defined in our work) as perfectly as possible. Shamsolmoali et al. [10] propose an image pyramid network based on rotation equivariance convolution to extract more representative features. Zhang et al. [11] combine low-level details, texture features, and high-level semantics to extract sufficiently fine-grained intrinsic features from images for classification. Some feature enhancement methods attempt to obtain the attention feature, which is a more effective feature representation for targets and is derived from the intrinsic feature. Nie et al. [6] train a classifier by learning the difference in attention features of various classes of objects. In the work proposed by Han et al. [5], a dual-mask attention module is designed to optimize the pyramid features for highlighting the differences between ship features and suppressing clutter to enhance the features. After fusing the extracted multiscale features, channel attention is proposed in [12] to enhance the fused features. In addition, Liang et al. [8] use attention-guided data augmentation to help the network locate discriminative regions more accurately to improve classification accuracy.

In the existing works, especially the feature enhancement ones, the target fine-grained classification mainly relies on the attention feature from the intrinsic feature. Few works combine these two different-level features. However, these features are in a one-to-one correspondence in the position within the feature channel. In fact, bilinear pooling is performed to correlate these features and obtain a discriminative descriptor of the target [13]. Therefore, it is urgent to propose a fusion mechanism to aggregate the intrinsic feature and the attention feature and obtain the high-level feature, which is consistent within the class and separable between the classes. Fig. 1 shows target image patches containing discriminative features of different fine-grained categories. Accurately extracting these discriminative features that reflect the essential attributes of targets is the crux of the target fine-grained classification task. This essential feature represents the high-level semantics of target categories in feature space. It is the conspicuous characteristics of an object different from others. The recently proposed ConvNeXt [14] with a ResNet-like hierarchical structure [15] has stronger feature extraction capabilities, so it can be combined into a fine-grained classification model to

Manuscript received 12 October 2022; revised 25 February 2023; accepted 4 April 2023. Date of publication 10 April 2023; date of current version 19 April 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62101060; and in part by the Beijing Natural Science Foundation, China, under Grant 4214058. *(Corresponding author: Yanan You.)*

The authors are with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: yiyonghao@bupt.edu.cn; youyanan@bupt.edu.cn; chaoli1998@163.com; zwl@bupt.edu.cn).

Digital Object Identifier 10.1109/TGRS.2023.3265669

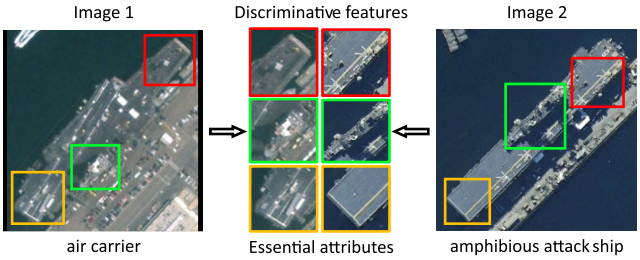

Fig. 1. Illustration of target image patches containing discriminative features of different fine-grained categories. The aircraft carrier contains at least three kinds of discriminative features: ship island, the white line at the stern, and the wireless line at the bow. The amphibious attack ship also has a similar pattern. If the network can observe these discriminative features, which reflect the essential attributes of targets, it could obtain the essential feature and classify them correctly. Images from FGSC-23 [16].

help extract the essential feature of the target. Furthermore, data augmentation is effective in guaranteeing the robustness of the network that suffers from the few sample categories. For datasets of fine-grained image classification, such as FGSC-23 [16] and FGSCR-42 [17], it is necessary to introduce data augmentation in the backbone.

In this article, following the way of feature enhancement, we propose an essential feature mining network based on DCNN, i.e., EFM-Net, for target fine-grained classification in remote sensing images. EFM-Net consists of two parts, the Miner for acquiring the essential feature and the Refiner for data augmentation. We combine the novel ConvNeXt into Miner to extract the intrinsic feature of the target itself due to its more concise and efficient block structure than ResNet. A multiple hierarchies attention module (MHAM) is constructed in Miner to filter and enhance the intrinsic feature to obtain attention feature. For feature fusion, we propose a low computational time-complexity feature bilinear polymerization pooling (BPP) in Miner, which associates the intrinsic feature with the attention feature and stacks higher level semantics to form the essential feature vector. The feature mapping network (FMN) is utilized to card high-dimensional features before the classification layer to prevent feature dimension entanglement. The Refiner is an auxiliary network used for adaptive data augmentation in the training phase, which can reduce the impact of the few sample categories on feature extraction. Our method can be trained in a mutually reinforcing manner by the Miner and Refiner, finally extracting the essential feature of the target stably. Extensive experiments based on two public datasets FGSC-23 and FGSCR-42, and one of our Aircraft-16 aircraft datasets demonstrate the proposed pipeline's excellent classification and generalization capabilities.

The main contribution of this article can be summarized as follows.

1) We propose an EFM-net for fine-grained targets' classification in remote sensing images, including the Miner for extracting the essential feature and the Refiner for data augmentation.
2) In Miner, ConvNeXt is applied to build a new fine-grained feature extraction framework for the first time. A low computational time-complexity BPP method aggregates and stacks the features to obtain the separable essential feature.
3) In the training stage, the proposed Refiner crops and masks the input images according to the information in the Miner to generate augmented samples. In this way, the Refiner promotes the stability of Miner's feature extraction on remote sensing fine-grained classification datasets.
4) We established a remote sensing image target fine-grained classification dataset Aircraft-16, and the proposed pipeline achieved the most competitive results on this dataset and the public remote sensing image target fine-grained classification datasets FGSC-23 and FGSCR-42.

The remainder of this article is organized as follows. We describe related works in Section II and introduce our proposed EFM-Net in Section III. Evaluations on two public datasets and a dataset of our own are presented in Section IV, followed by conclusions in Section V.

II. Related Works

In this section, two research directions related to our work are elaborated, including feature extraction and feature enhancement methods.

A. Feature Extraction

DCNNs, such as VGG [18], Inception [19], ResNet [15], and DenseNet [20], have powerful capabilities to understand complex semantic information and process high-resolution remote sensing images. To further optimize the extraction capabilities and the representation of DCNNs, many works [21], [22] modify the block structure of the ResNet [15]. Recently, Liu et al. [14] draw on the design of the Swin Transformer [23] and propose the ConvNeXt network with a concise and efficient block structure. These DCNNs have excellent performance in coarser classification tasks.

However, due to the gap between fine-grained classification and coarser classification tasks in remote sensing images, the benefits of CNNs for fine-grained analysis remain limited. Since the emergence of multilevel features constructed by the feature pyramid network (FPN) [24], it has become a milestone in CNN-based classification methods. FPN consists of a top-down semantic information transfer structure, which enables each layer of features to incorporate high-level semantic information. Many works apply multiscale features to fine-grained classification models. Zhang et al. [11] use the Laplacian operator to extract high-frequency features of multiscale features to obtain enough detailed features for classification. In [10], an image pyramid network based on rotation equivariant convolution is proposed to improve the detection ability of small objects. Shi et al. [25] use pyramids to build multiscale features for fusion and classification.

Data augmentation methods can ensure the stability of feature extraction in the backbone network. The datasets of fine-grained images are generally long-tailed distributions [16], [17], which is due to the difficulty of obtaining image samples for some categories. The unbalanced data

distribution affects the stability of feature extraction of different categories. In general classification methods, data augmentation methods are used to reduce this impact. In [26], translation, rotation, and local random noise are proposed to augment unbalanced datasets. Chen et al. [27] establish a data augmentation pool based on the current popular data augmentation methods to enhance the stability and effectiveness of training.

B. Attention Feature Enhancement

The attention mechanism is a powerful feature enhancement method that is widely used not only in scene classification [28] and object detection [5] but also plays an important role in fine-grained image classification. Through applying the attention mechanism, many works obtain global and local attention features for precise classification. In both point-guided pixel-level localization [8], [29] and anchor-guided region-level localization [30], attention feature maps are used to find these local areas accurately. The self-attention mechanism is used in [31] to enhance the global self-attention feature. More importantly, the attention mechanism can screen and strengthen the feature and improve the ability of feature representation. In [6], spatial attention and channel attention are used to filter out features at different levels, and the classifier is trained by learning the difference between these two attention feature maps. Zhuang et al. [32] obtain clues to distinguish different categories by comparing pairs of images. In [33], the feature maps of different granularities are fused and classified to improve the classification ability with multiscale objects. Zheng et al. [34] proposed a trilinear feature module to extract fine attention maps. Xiong et al. [7] distinguish the object categories in remote sensing images through multihead attention maps.

The attention mechanism provides global or local attention features, and it is also important to utilize these features for classification through dimensionality reduction. At present, feature dimensionality reduction methods in the field of fine-grained classification are mainly divided into fully connected layers, global average pooling, and bilinear pooling. Krizhevsky et al. [35] use the fully connected layer of high-dimensional features to map to low-dimensional and then import the softmax layer for classification. Global average pooling is proposed in [36], which forces the mapping between features and categories, and integrates spatial information. Bilinear pooling has been used since [13], and the position and semantics of the two feature maps are related by matrix multiplication of two feature maps to obtain higher level correlation features. Since the bilinear pooling can be trained end-to-end, it is widely used in classification task [37], [38], [39], [40]. Bilinear pooling is used in [41] to fuse the multimodal features of images and texts. In [40], bilinear pooling was used to aggregate the output features of different CNNs, resulting in compact and discriminative bilinear features for subsequent classification. Liang et al. [37] leverage the bilinear pooling to fuse information from different dimensions globally and locally. Li et al. [38] proposed a joint pooled method to reduce the number of feature dimensions, reduce computational complexity, and prevent overfitting.

It is an ingenious improvement of the feature dimensionality reduction method, which enables our model to enhance features by using the association between them. The way of correlating features before classification will be described in detail in Section III.

III. Proposed Method

A. Architecture of the EFM-Net

In this section, as a novel feature enhancement way for the target fine-grained classification task, we propose the EFM-net, which has two indispensable components: the essential feature mining module (Miner) and the data augmentation module (Refiner), as shown in Fig. 2. According to the feature flow in Miner, the ConvNext backbone extracts the intrinsic features from training samples in the intrinsic feature extraction stage. The MHAM obtains the attention feature by enhancing the intrinsic features. The features obtained above are imported into BPP and FMN, finally generating the essential features, which are the kernel parts of Miner for feature enhancement. In the training phase, the Refiner can adaptively augment the training samples, and it plays an important role in further mining the target features in images. The Miner and the Refiner can reinforce each other in an unsupervised manner for extracting accurate features.

B. Essential Feature Mining Module (Miner)

1) ConvNeXt for Feature Extraction: In Miner, a backbone network is integrated with ConvNeXt [14] to extract intrinsic features from remote sensing images containing objects of interest. The ConvNeXt network consists of several stages, and we denote the output features of the lth stage as $\mathbf{F}_l^P$.

2) Multiple Hierarchies Attention Module (MHAM): As shown in Fig. 3, MHAM is a dual-path structure constructed with several attention blocks. The right-to-left path conducts high level while applying upsampling to unify the size of two adjacent features and fuse them. The left-to-right path conducts detailed texture and position while using the feature vectors for transmission. This structure balances the number of features and computation complexity of the module. We preprocess the intrinsic features to the same dimension

$$\widetilde{\mathbf{F}}_l^P = \mathcal{F}^{\text{conv}}(\mathbf{F}_l^P; w_l^{\text{conv}}, b_l^{\text{conv}}) + \text{upsampling}(\mathbf{F}_{l+1}^P) \quad (1)$$

where $\mathbf{F}_l$ is the feature map of the lth channel of the feature map, upsampling$(\cdot)$ represents the double interpolation, and $\mathcal{F}^{\text{conv}}$ is the 1×1 convolutional layer with weight $w_l^{conv} \in \mathbb{R}^{C\times\kappa}$ and bias b_l^{conv}. For feature fusion, we unify the channel of the feature in the attention block with κ. Next, get the normalized channel weight vector $\widetilde{\mathbf{v}}$ of this layer according the squeeze and excitation network [42]

$$\widetilde{\mathbf{v}}_i = \mathcal{F}^{\text{SE}}\left(\widetilde{\mathbf{F}}_l^P; w^{\text{SE}}\right) \quad (2)$$

where $\mathcal{F}^{\text{SE}}$ represents the squeeze and excitation network with weight w^{SE}. Finally, we take the weighted summation of two

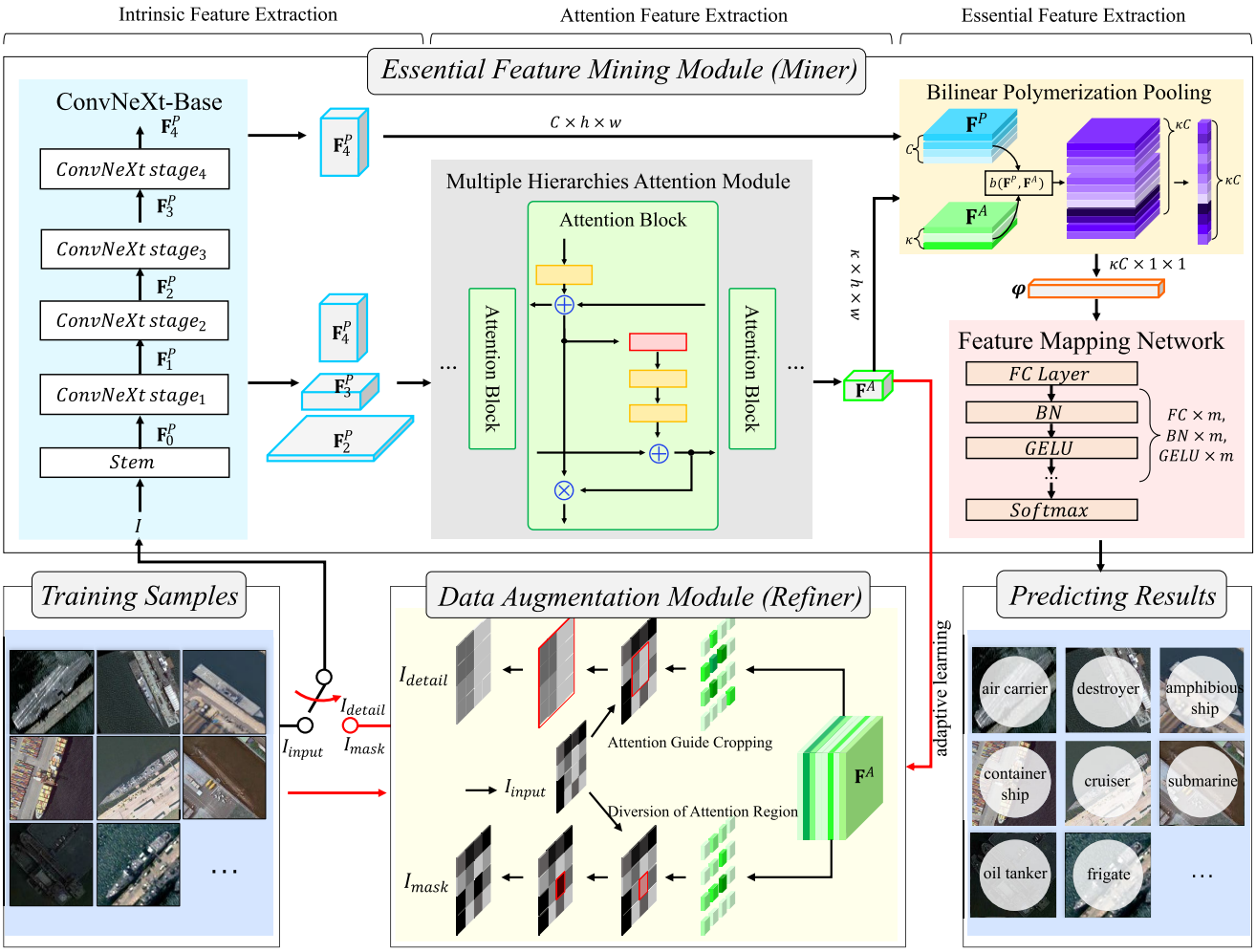

Fig. 2. Overview of the EFM-Net based on convolutional neural network. The EFM-Net consists of the essential feature mining module (Miner) and the data augmentation module (Refiner). The intrinsic features of the input images are generated through the ConvNeXt-Base. Features of different layers are input into multiple hierarchies attention module (MHAM) to generate attention features. The BPP fuses the intrinsic feature and the attention feature, and outputs the aggregational feature to the FMN for decoupling. In addition, the attention features from the input images flow along the red line into Refiner for adaptive learning. The Refiner modifies the input images based on the feature, generates two new images, reinputs them into the network, and calculates along the black line. Best viewed in color with zoomed-in.

adjacent channel weights to obtain the attention vector $\boldsymbol{v}$

$$\mathbf{v}_l = (\mathbf{v}_{l-1} + \widetilde{\mathbf{v}}_l)/2 \tag{3}$$

and pool the input feature map

$$\mathbf{F}_l^A = \mathbf{v}_l \odot \widetilde{\mathbf{F}}_l^P \tag{4}$$

where $\widetilde{\mathbf{v}}_l$ is the channel weight vector of the lth layer, $\mathbf{v}_l$ is the attention vector of the lth layer, and $\odot$ represents the channelwise product. The attention feature $\mathbf{F}_l^A$ has the same size with $\widetilde{\mathbf{F}}_l^P$.

The dual-path structure between layers is to find the most recognizable features. To reduce the computational cost, the top-layer attention features $\mathbf{F}_4^A$, which combine the information with all layers, are taken as the output of the MHAM. We recorded the output attention feature as $\mathbf{F}^A$ with the shape $\kappa \times h \times w$.

3) Bilinear Polymerization Pooling: By simply stacking the attention features with the intrinsic features, we hope that the classifier can recognize the target from the various categories. However, the effect is limited due to the disregard for the association between features. In addition, many repetitive calculations make rough stacking inefficient [43]. Inspired by Lin et al. [13], we cleverly combine the attention features and intrinsic features and propose a novel feature BPP.

The mathematical model used by bilinear pooling is to expand the features quadratically

$$f = \sum_{j=1}^{N}\sum_{k=1}^{M} u_j W_{jk} v_k = \mathbf{u}^T \mathbf{W}_i \mathbf{v} + b \tag{5}$$

where $\mathbf{u} \in \mathbb{R}^{L_1}$ and $\mathbf{v} \in \mathbb{R}^{L_2}$ are input feature vectors, $\mathbf{W}_i \in \mathbb{R}^{L_1 \times L_2}$ is a weight matrix of the output f, and b is the bias. This regularization idea of interacting two matrices brings a lot of computation. When the length of two input vectors is the same, we can treat the parameter W simply as the average pooling, and the formula can be written for

$$f = \frac{1}{L}\mathbf{u}^T \mathbf{v} = \frac{1}{L}\mathbb{1}^T(\mathbf{u} \circ \mathbf{v}) \tag{6}$$

where $\mathbb{1} \in \mathbb{R}^L$ denotes a column vector of ones and $\circ$ denotes the Hadamard product (elementwise multiplication).

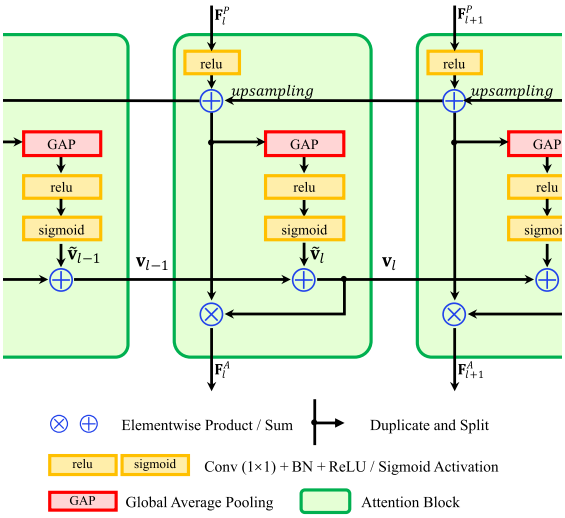

Fig. 3. Attention mechanism in the multiple hierarchies attention module (MHAM). The purpose of applying channel attention to a multilevel network is to capture the common features that are of interest to all hierarchical feature maps.

The regulation can also handle cases where the inputs are tensors. We denote the two input tensors as $\mathbf{U} \in \mathbb{R}^{N\times L}$ and $\mathbf{V} \in \mathbb{R}^{M\times L}$, and calculate the fusion vector through the operator above

$$f_j^k = \frac{1}{L}(\mathbb{1}^T(\mathbf{U}_j \circ \mathbf{V}_k)) \tag{7}$$

where $\mathbb{1} \in \mathbb{R}^L$ denotes a column vector of ones, $j = 1, \ldots, N$, and $k = 1, \ldots, M$. Fusion feature is denoted as $\mathbf{f}^k = \{f_1^k, \ldots, f_j^k, \ldots, f_C^k\}^T$, then converting Hadamard product back to matrix product

$$\mathbf{f}^k = \frac{1}{L}\sum_{l=1}^{L}\left(\mathbf{U}\mathbf{V}_k^T\right) = \Phi\left(\mathbf{U}\mathbf{V}_k^T\right) \tag{8}$$

where $k = 1, \ldots, M$; there are $\mathbf{f}^1, \ldots, \mathbf{f}^M$, and $\Phi(\cdot)$ represents the average pooling, which can be written for

$$\mathbf{f} = \Phi(\mathbf{F}) = \frac{1}{L}\sum_{l=1}^{L}\mathbf{F}_l \tag{9}$$

where $\mathbf{F}$ is the feature of $M \times zL$. f is spliced into a feature vector $\mathbf{f} \in \mathbb{R}^M$ as the fusion feature. In order to maintain the integrity of fusion features, each fusion feature $\mathbf{f}^k$ is spliced to obtain the final feature vector. We denote this fusion process as BPP. The complete formula is defined as follows:

$$b(\mathbf{U}, \mathbf{V}) = \begin{pmatrix} \Phi(\mathbf{U}\mathbf{V}_1^T) \\ \cdots \\ \Phi(\mathbf{U}\mathbf{V}_k^T) \\ \cdots \\ \Phi(\mathbf{U}\mathbf{V}_M^T) \end{pmatrix} \tag{10}$$

where b represents the BPP operator.

Given the intrinsic features, $\mathbf{F}^P \in \mathbb{R}^{C\times h\times w}$, and given the attention feature, $\mathbf{F}^A \in \mathbb{R}^{\kappa\times h\times w}$, flattening to the 2-D tensors, which is demoted as $\tilde{\mathbf{F}}^P \in \mathbb{R}^{C\times L}$ and $\tilde{\mathbf{F}}^A \in \mathbb{R}^{\kappa\times L}$, $L = h \times w$. $\tilde{\mathbf{F}}^P$ and $\tilde{\mathbf{F}}^A$ polymerize and pool to obtain essential feature vector through BPP operator b

$$\boldsymbol{\varphi} = b(\tilde{\mathbf{F}}^P, \tilde{\mathbf{F}}^A) = \begin{pmatrix} \Phi(\tilde{\mathbf{F}}^P(\tilde{\mathbf{F}}^A)_1^T) \\ \cdots \\ \Phi(\tilde{\mathbf{F}}^P(\tilde{\mathbf{F}}^A)_k^T) \\ \cdots \\ \Phi(\tilde{\mathbf{F}}^P(\tilde{\mathbf{F}}^A)_\kappa^T) \end{pmatrix} \tag{11}$$

where $\boldsymbol{\varphi} \in \mathbb{R}^{\kappa C}$ represents the essential feature vector.

To understand BPP from the perspective of channel level, we can think that $\mathbf{F}^A$ is the feature description of $\mathbf{F}^P$. We use each layer of $\mathbf{F}^A$ to regularize the input feature $\mathbf{F}^P$ and obtain the discriminative descriptors in the higher level essential feature space. Each channel of the attention feature $\mathbf{F}^A$ output by MHAM can be considered as an accurate reflection of a feature of the backbone feature. The BPP feature fusion method can associate the attention features with the intrinsic features. Through the feature interaction, we obtain the essential feature vector, which combines with the discriminative image descriptors.

The computational time complexity of fully bilinear pooling (FBP) and BPP for the features of $C \times h \times w$ is given as follows:

$$\begin{aligned} \Omega(\text{FBP}) &= hwC^2 \\ \Omega(\text{BPP}) &= hwC \end{aligned} \tag{12}$$

where FBP represents the fully bilinear pooling. The detailed channel dimension number, computational complexity, and parameter amount of more fusion methods are compared in the discussion of Section IV.

4) Feature Mapping Network: Different fine-grained features of the target usually have small differences. After the fusion features are obtained, the space where the feature vectors are located is generally chaotic. In mathematical form, the chaos manifests in the coordinate system. The unit vectors $\boldsymbol{\eta} = \{\eta_1, \eta_2, \ldots, \eta_m\}$ describing the feature vectors are generally not mutually orthogonal. There must be a set of orthonormal bases in the finite high-dimensional space. We design an FMN, hoping to learn a nonlinear mapping $\mathcal{T}$

$$\boldsymbol{\xi} = \mathcal{T}(\boldsymbol{\eta}; w^{\text{fc}}) \tag{13}$$

where w^{fc} is the mapping parameter and the space basis vectors $\boldsymbol{\xi} = \{\xi_1, \xi_2, \ldots, \xi_m\}$ are mutually orthogonal. There will be no feature entanglement between features. The learned mapping matrix $\mathcal{T}$ transforms the eigenvector $\boldsymbol{\varphi}$

$$\boldsymbol{\varphi}^{\text{sef}} = \mathcal{T}(\boldsymbol{\varphi}; w^{\text{fc}}) \tag{14}$$

where $\boldsymbol{\varphi}^{\text{sef}}$ represents the separated essential feature vector. In the process of FMN, feature vectors maintain dimensional consistency in the Euclidean space. Because FMN does not compress the essential feature vector, it will not cause information loss. In experiments, we fit this transformation by adding two mapping blocks consisting of fully connected layers with input dimensions equal to output dimensions. Too many mapping blocks may lead to an increase in the number of parameters, and the data are relatively insufficient, making it difficult for the network to fit. The parameter m of FMN in Fig. 2 is set to 2. In the training phase, the learning rate of

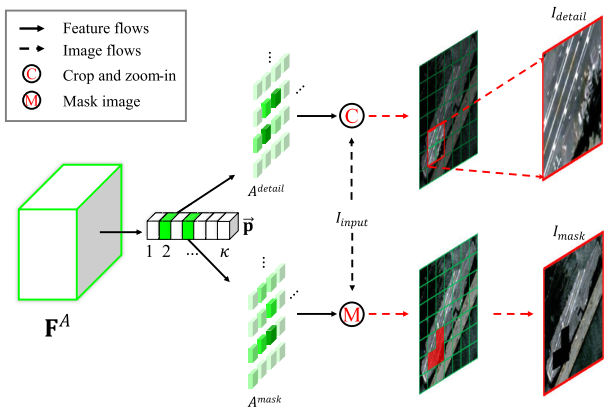

Fig. 4. Detailed process diagram of the Refiner. $\mathbf{F}^A$ represents the attention feature output by the MHAM. After passing through two branches, the input image is processed into a detail image and a mask image. The operation "C" means that A^{detail} is bilinearly interpolated to the length and width of the input image. After the attention map, $\mathbf{F}^A$ is binarized, and the minimum circumscribed rectangle formed by the points that are not 0 is used to map the input image. "M" means that A^{mask} is bilinearly interpolated to the input size and then the dot product of the input image. The blue box in the figure represents the selected area, and the red block represents the deleted area.

FMN is set to one percent of others. The separated essential feature vector is fed into the $softmax$ layer for classification.

C. Data Augmentation Module (Refiner)

Optical remote sensing images have unique attributes compared with natural images, such as multiple imaging angles, diverse scenes, and arbitrary orientation of objects. These interference factors will lead to the fluctuation of training loss, which is not conducive to the convergence of the model.

To deal with the problem, we design the Refiner as the data augmentation method in the training phase. An unsupervised Refiner finds the features of important channels in the attention features. Based on these attention slice features, the Refiner amplifies important regions for region focusing and searches more hotspots for region diversification. To select important channels, the Refiner sums the elements in each channel of the attention feature $\mathbf{F}^A$ to get the overall response vector $\vec{p}$ of each channel

$$\vec{\mathbf{p}} = \Phi(\mathbf{F}^A) \tag{15}$$

where $\mathbf{F}^A$ represents the attention features generated by MHAM. Then, we randomly select two channels of features in $\mathbf{F}^A$ according to the probability vector $\vec{\mathbf{p}}$, denoted as A^{detail} and A^{mask}, respectively. Then, the detail image and the mask image are obtained through the two branches of attention guide cropping and diversion of attention region. The detailed process of the Refiner is shown in Fig. 4. Note that this step is only used in the training phase; we will remove this part in the test phase of the experiment.

1) Attention Guide Cropping: For locating and magnifying critical regions, we use a flexible point guidance method. The size and position of the chosen region are determined by taking the smallest circumscribed rectangle formed by the points whose value exceeds the threshold in A^{detail}. The parts of the input image mapped by chosen region are cropped and enlarged. For the feature map A^{detail} (A^d), after bilinear interpolation to the size of the input image, the boundary coordinates of the minimum circumscribed rectangle are obtained by the following formula:

$$\begin{aligned}
\theta_A &= \max(A^d) \cdot \theta_{\text{detail}} \\
t_{\text{xmin}} &= \arg\min_i A^d(i, j) \geq \theta_A \\
t_{\text{xmax}} &= \arg\max_i A^d(i, j) \geq \theta_A \\
t_{\text{ymin}} &= \arg\min_j A^d(i, j) \geq \theta_A \\
t_{\text{ymax}} &= \arg\max_j A^d(i, j) \geq \theta_A
\end{aligned} \tag{16}$$

where $t_{\text{xmin}}, t_{\text{xmax}}, t_{\text{ymin}}$, and t_{ymax} represent the lower and upper boundaries of the coordinates x and y of the region, respectively, and θ_{detail} is a hyperparameter that ensures the Refiner can locate the target. We obtain the region bounding box of the input image and crop the patch from the input image

$$\begin{aligned}
I_{\text{detail}} = \{(x, y) | &t_{\text{xmin}} \leq x \leq t_{\text{xmax}} \\
&t_{\text{ymin}} \leq y \leq t_{\text{ymax}}, \quad (x, y) \in I\}
\end{aligned} \tag{17}$$

where I_{detail} represents the detail image and I represents the input image. The detail image is fed into the network again to extract more detailed features.

2) Diversion of Attention Region: A more accurate detail image is obtained through attention guide cropping. The receptive field of the network is limited to the local area, and some areas with certain important features are forced to give up. Considering that the regions containing the discriminative feature will not be concentrated in a single area, it is necessary to design a mechanism to discover more hotspots. Refiner set a mask for covering the region selected by the attention feature to deal with this situation. A mask is a binary data matrix of the same size as the input image, with 0 representing the covered area and 1 representing the retained area

$$\text{Mask}(x, y) = \begin{cases} 1, & A^m(x, y) < \max(A^m) \cdot \theta_{\text{mask}} \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

where A^m is the abbreviation of A^{mask} and θ_{mask} is a hyperparameter that ensures the Refiner can locate the target, and x and y represent the coordinates of the feature. Then, mask the input image with the Mask

$$I_{\text{mask}} = I_{\text{input}} \odot \text{Mask} \tag{19}$$

where I_{mask} represents the mask image and is fed back into the network for forwarding propagation. The mask image allows the network to comprehensively learn those important features in the next forward propagation. This can help the network to locate other regions with similar features to the masked region.

3) Learning of Refiner: After each parameter update of the Miner, the Refiner guides the network to perform two more forward and backward propagation for refinement. When the Miner is underfit, we hope this refinement to increase appropriately to speed up the model convergence. Miner no longer needs fine-tuning of the Refiner when it is about to fit. Thus, we design additional coefficients to the categorical

Algorithm 1 Feature Miner Training Strategy

Require: Input image I_{input}

1: Obtain $\mathbf{F}^P$ by Backbone Network
2: Obtain $\mathbf{F}^A$ by MHAM with $\mathbf{F}^P$
3: Obtain $\boldsymbol{\varphi}^{esf}$ by Equation 11 and Equation 14 with $\mathbf{F}^P$ and $\mathbf{F}^A$
4: Predict and calculate $loss_1$: $loss_1 = \mathcal{L}_{ce}(I_{input}, y^I)$
5: Crop and enlarge the region to obtain I_{detail} by Equation 17 with I_{input}
6: Mask the region to obtain I_{mask} by Equation 19 with I_{input}
7: Predict and calculate the soft entropy loss: $loss_2 = \mathcal{L}_{sce}(I_{detail}, y^I)$, $loss_3 = \mathcal{L}_{sce}(I_{mask}, y^I)$
8: **return** $loss_{total} = loss_1 + loss_2 + loss_3$

cross-entropy loss for the Refiner learning. Specifically, given input I and its label y^I, the loss is defined as

$$\mathcal{L}_{\text{sce}}(\boldsymbol{I}, \boldsymbol{y^I}) = -\sum_{i}^{\mathcal{C}} y_i (1 - \mathcal{M}(\boldsymbol{I})_i)^{\gamma} \log(\mathcal{M}(\boldsymbol{I})_i) \quad (20)$$

where $\mathcal{C}$ is the number of classes, $\mathcal{M}(\cdot)$ represents the process of the Miner, and γ is a hyperparameter. $\mathcal{M}(\boldsymbol{I})_i$ is the probability that image $\boldsymbol{I}$ belongs to the ith class. We denote this loss function as the **soft-cross-entropy loss** (SCE loss). γ is used to control the slope of the loss curve in different segments. When $\gamma = 0$, the loss function becomes the cross-entropy loss. With the increase in γ, the gradient of the loss function curve is smaller, and the loss fluctuation is more gentle when approaching convergence. Since three images (input image, detail image, and mask image) forward propagations are done separately in the training phase, the total classification loss is the sum of the three losses

$$\begin{aligned} \text{loss}_1 &= \mathcal{L}_{\text{ce}}(I_{\text{input}}, y^I) \\ \text{loss}_2 &= \mathcal{L}_{\text{sce}}(I_{\text{detail}}, y^I) \\ \text{loss}_3 &= \mathcal{L}_{\text{sce}}(I_{\text{mask}}, y^I) \\ \text{loss}_{\text{total}} &= \text{loss}_1 + \text{loss}_2 + \text{loss}_3 \end{aligned} \quad (21)$$

where $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss, $\text{loss}_{\text{total}}$ represents the total loss of the network, loss_1 represents the calculation loss of the input image forward propagation to the classification network, loss_2 and loss_3 represent the forward propagation loss of the detail image and the mask image, and I_{input}, I_{detail}, and I_{mask} represent the input image, the detail image, and the mask image. The proposed Refiner can autonomously learn to amplify regions with discriminative features. Even if does nothing, it will not negatively affect the backbone.

In this article, we improve the work based on feature enhancement so that the enhanced attention features can be fully learned. The proposed structure has the ability to mine the essential feature from the fine-grained features. In terms of training strategy, maximizing the posterior probability can greatly promote the ability of the backbone to extract features. The whole training process of Miner is shown in Algorithm 1. In addition, our pipeline is not limited to a certain backbone network, so this work can be updated with a better feature extractor.

TABLE I
STATISTICS OF THE FINE-GRAINED REMOTE SENSING IMAGE CLASSIFICATION DATASETS USED IN OUR EXPERIMENTS

Dataset	#Classes	#Training	#Testing
FGSC-23 [16]	23	3256	825
FGSCR-42 [17]	42	4693	4627
Aircraft-16	16	6355	2133

IV. EXPERIMENTS

In this section, we present the datasets, the setting of experimental hyperparameters, comparisons with the baseline methods, results with different hyperparameters, and the discussion.

A. Datasets

To test the effect of our proposed pipeline on fine-grained classification in remote sensing images, we conduct extensive experiments on two publicly available datasets (FGSC-23 [16] and FGSCR-42 [17]) and a self-made dataset (Aircraft-16). The details are shown in Table I.

1) FGSC-23: The fine-grained ship classification dataset [16], FGSC-23, is a high-resolution optical fine-grained remote sensing image classification dataset, including 23 types of ships and 4052 samples. Each target is given a class label, an aspect ratio label, and a distribution direction label. Compared with the existing optical remote sensing image ship target recognition dataset, the FGSC-23 dataset has the characteristics of diverse image scenes, fine classification, and complete labels. Fig. 5 displays examples of 23 fine-grained classes in FGSC-23.

2) FGSCR-42: Fine-grained ship classification in remote-sensing with 42 classes [17], FGSCR-42, contains 42 fine-grained categories of ten broad categories, specifically, fine-grained categories of ships, such as the Kitty-Hawk-class aircraft carrier, the Arleigh-Burke-class destroyer, and the mega-yacht. The images of this dataset are composed of sliced images of the object detection datasets DOTA [44], HRSC2016 [45], NWPUVHR-10 [46], and so on. The slices are obtained by extending the data in the target frame label attached to these datasets to the surrounding by a certain pixel and then cutting them out from the input image. Compared with FGSC-23, the dataset category is further subdivided, specific to the type of the ship. In the experiment, we follow the division method of the original article and divide the entire dataset into the training set and test set. Samples of ship slices from FGSCR-42 are presented in Fig. 6.

3) Aircraft-16: To verify the generalization of the model, we build a fine-grained classification dataset of aircraft targets, namely Aircraft-16. Our dataset has a similar volume to the above two open-source datasets. Compared with other fine-grained classification datasets of aircraft targets, such as MTARSI [47] or Raleplane [48], Aircraft-16 has diversified scenarios, a sufficient number of samples, and fine-grained class labels. We select and crop a total of 8488 slices of aircraft

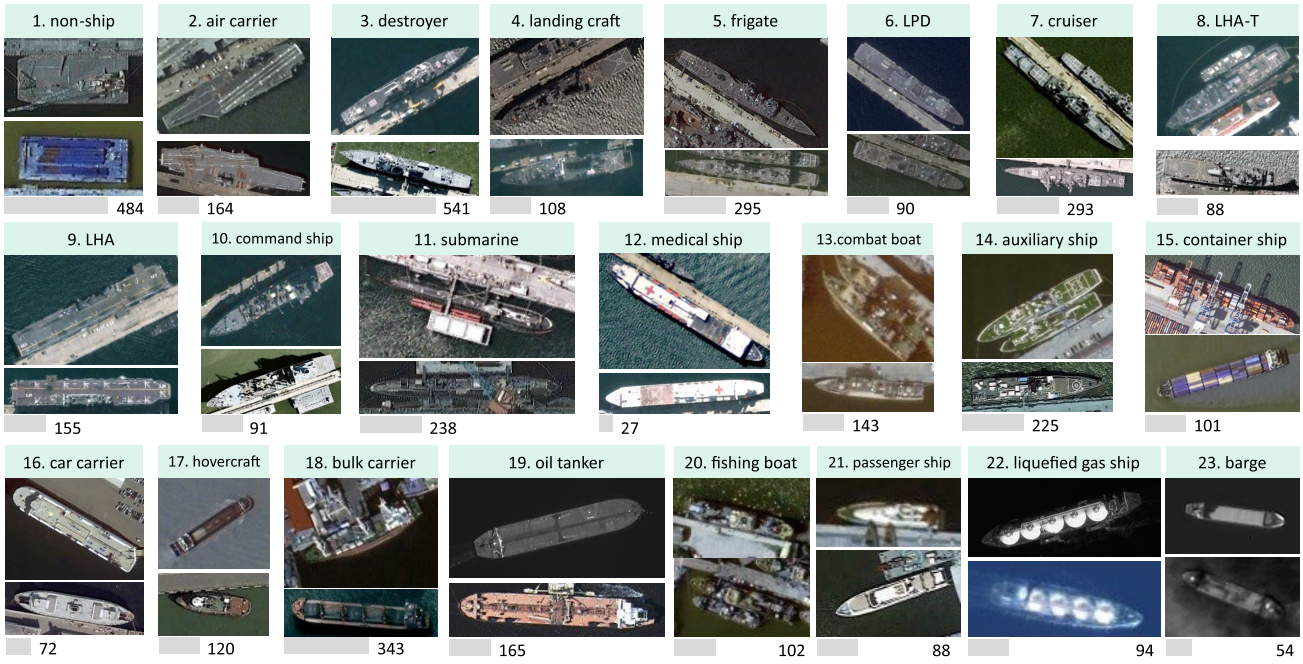

Fig. 5. Samples of 23 fine-grained ship classes in FGSC-23 [16]. The numbers in the light green squares represent the category labels. The gray progress bar at the bottom represents the amount of the various categories in this dataset, and the number represents the actual sample quantity. LPD is short for the amphibious transport dock, "LHA-T" is short for the Tarawa-class amphibious assault ship, and LHA is short for the Amphibious assault ship. The dataset has many similar subcategories (such as #5 and #7, and #6 and #8), and the color of the same category object changes greatly. Some targets (such as #8 and #22) are seriously disturbed by the scene.

targets in 16 categories from various datasets (Rareplane [48], FAIR1M [49], and so on) with a resolution of 0.5–1.5 m for aircraft target detection. These slice images are original images without augmentation, ensuring the data's diversity and authenticity. Second, the label only contains the fine-grained category, such as B-52, E-3, and Boeing 747. Sample images for each category are shown in Fig. 7. To simplify the expression, we label it as type-0 to type-15. With the consideration of the scene (such as a sunny airport and a dimly lit runway), we randomly select targets from all samples to form the training set and the test set. The ratio between the training set and the test set is about 8:2. Aircraft-16 is a private dataset in our previous work [50], and now, we are making it public.

B. Implementation Details and Baselines

1) Implementation Details: To comprehensively discuss the validity and accuracy of the proposed method, the other two backbones, VGG and ResNet, are investigated in the experiment. The images after preprocessing are fed into the backbone, and the essential feature vector obtained through the forward propagation is input to the fully connected layer to calculate the cross-entropy loss. At the same time, the Refiner generates a detail image and a mask image for the second forward propagation and calculates the SCE losses. Finally, we use the Back Propagation algorithm to fine-tune the parameters. Our experiments use class labels for supervised training.

The method of image preprocessing is as follows: the input image is resized to 256 × 256 by bicubic interpolation and then randomly and centrally cropped to 224 × 224, with a 50% probability of random horizontal flipping, and only interpolation and center crop are needed in the test phase. The learning rate is initialized to 1e-3, and the batch size is fixed at 16. In particular, the learning rate of FMN in Section III-B4 is set to 1e-5. The total epoch number is set by 100. For the optimizer, we use the AdamW with β_1 is 0.9, β_2 is 0.999, weight decay is 1e-8, and eps (epsilon) is 1e-8.

The settings of the hyperparameters proposed in this article are given as follows. The channels of the attention block, κ, in Section III-B2, are set to 16. The hyperparameter γ in (20) is set to 0.5. θ_{detail} in (16) is set to 0.5, and θ_{mask} in (18) is set to 0.25.

Experiments are performed on Ubuntu 16.04 system containing two NVIDIA RTX 3090 GPUs for accelerated computing, and the model is built based on the Pytorch framework.

2) Baselines: We list the notation of all baselines involved in the comparison and a brief implementation of their experiments.

The backbone networks are given as follows. We use the official code and the following hyperparameter settings for the experiment: the batch size is 16, the learning rate is 0.01, the input image size is 448, the number of iterations is 100, and the pretraining model is not loaded.

1) *VGG [18]:* Replace the large kernels of 5 × 5 and 7 × 7 with stacks of convolutional layers with tiny kernels of 3 × 3 and appropriately increase the network depth to 16 or 19 layers.
2) *ResNet [15]:* Through residual learning, the feature degradation mechanism caused by the deepening of the network level is overcome.
3) *Inception-v3 [19]:* The regularization of the model is achieved by smoothing the labels, computational

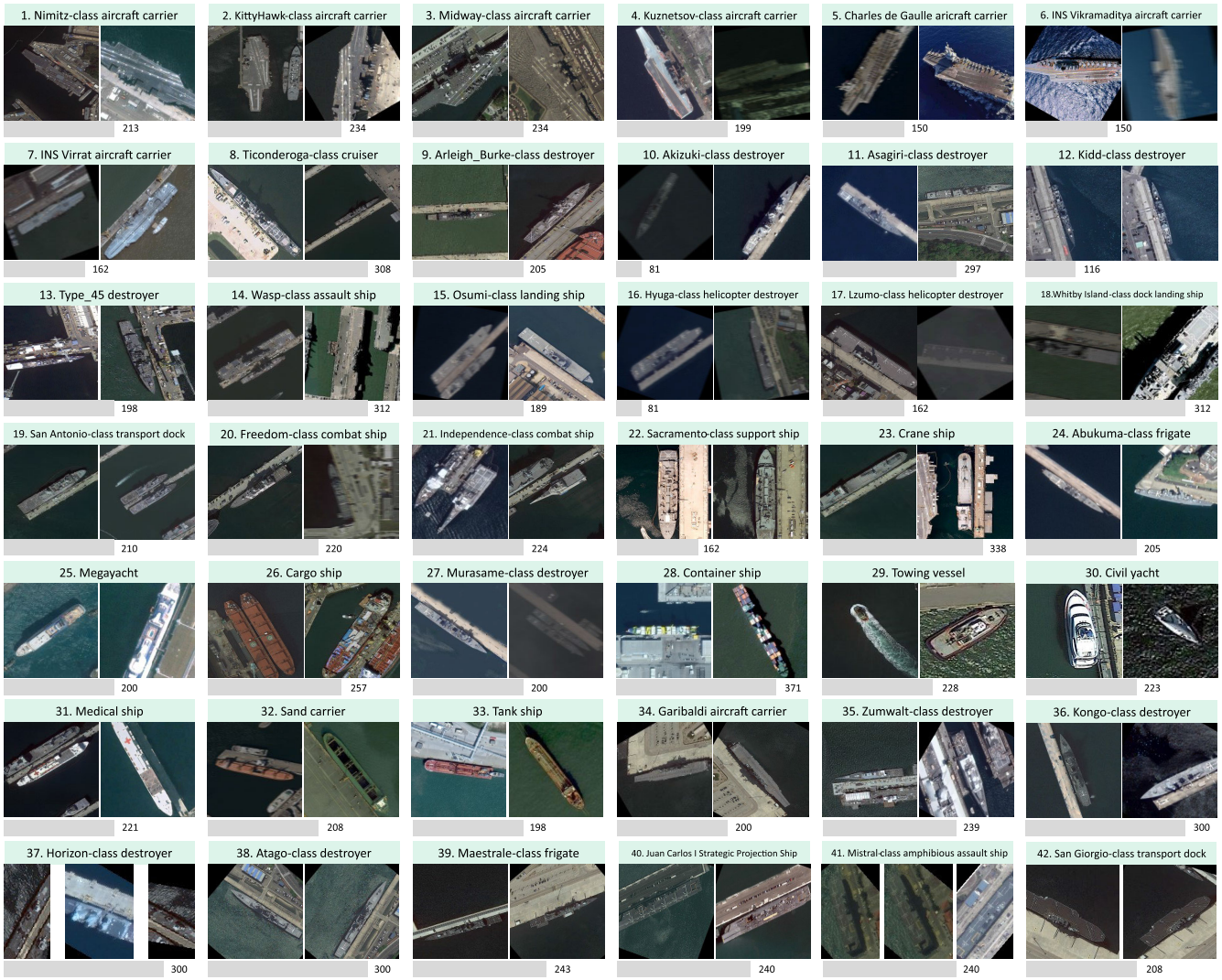

Fig. 6. Samples of 42 fine-grained ship classes in FGSCR-42 [17]. The gray progress bar at the bottom represents the amount of the various categories in this dataset, and the number represents the actual sample quantity. The dataset has sufficient categories, various samples, and rich scenes. It is an excellent dataset for target fine-grained classification in remote sensing images.

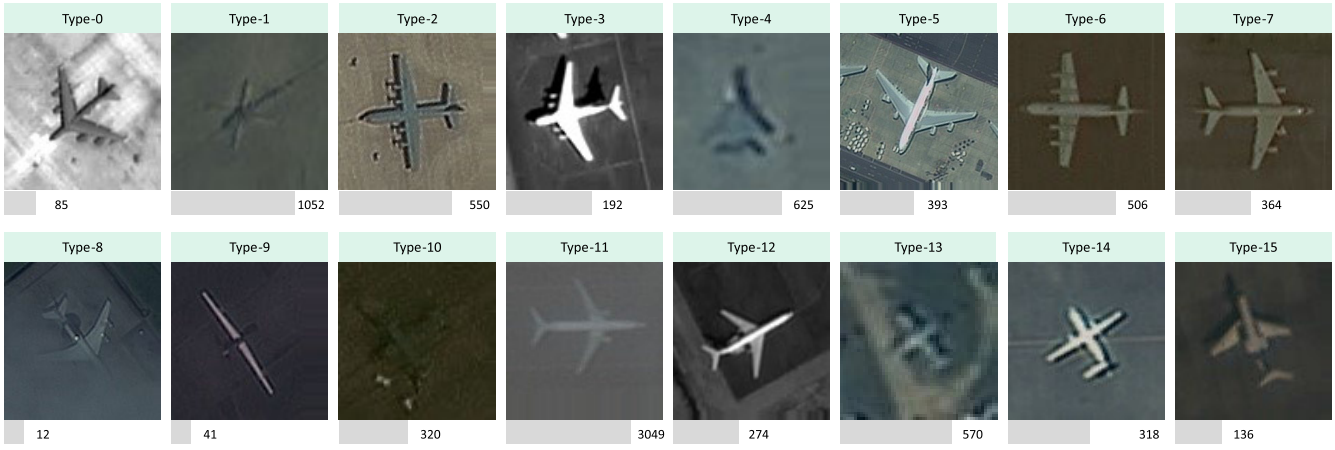

Fig. 7. Samples of 16 fine-grained aircraft classes in our Aircraft-16. The gray progress bar at the bottom represents the amount of the various categories in this dataset, and the number represents the actual sample quantity. Images are seriously disturbed by visible light imaging conditions. Difficult samples include cloud occlusion (such as Type-8), small targets (such as Type-13), and unbalanced samples (such as Type-8 and Type-9).

complexity is reduced by decomposing the convolution, and the detection of small objects is easier with a lower receptive field.

4) *ResNeXt [21]:* Change the single-branch architecture of each block of ResNet [15] to a wider set of multibranch architectures.

TABLE II

COMPARISON EXPERIMENTAL RESULTS ON FGSC-23, FGSCR-42, AND AIRCRAFT-16. THE "INPUT SIZE" REPRESENTS THE LENGTH × WIDTH OF THE IMAGE FED INTO THE NETWORK. THE NUMBERS BELOW DATASETS INDICATE TEST OA (%) ON THIS DATASET

Method	Input size	FGSC-23	FGSCR-42	Aircraft-16
VGG-16 (2014) [18]	448×448	82.0	77.4*	84.5*
Bilinear-CNN (2015) [13]	448×448	84.0*	89.5*	85.1
ResNet (2015) [15]	448×448	84.6	84.8*	85.0*
Inception-v3 (2016) [19]	448×448	83.9*	77.1	76.3
ResNeXt (2017) [21]	448×448	74.1	89.2*	83.4
DenseNet (2017) [20]	448×448	84.0*	88.7	84.8
DCL (2019) [52]	448×448	87.5	93.0*	89.0
TASN (2019) [34]	448×448	85.8	93.5*	86.3
AC-Net (2020) [43]	448×448	69.8	88.4	81.5
API-Net (2020) [32]	448×448	88.7	97.6	89.6
BAT (2020) [51]	448×448	60.9	96.00	80.4
PMG-V2 (2021) [33]	448×448	88.1	98.0	89.7
FFVT (2021) [31]	384×384	87.6	97.9	89.2
SIM (2022) [53]	448×448	86.3	97.9	89.5
EA-Net (2022) [7]	224×224	91.8*	94.1*	-
EFM-Net(Ours)	224×224	**92.6**	**99.3**	**92.4**

"-" represents no available experimental result.

5) *DenseNet [20]:* Connect each layer to every other layer in a feed-forward manner.

We reproduce some bilinear or trilinear methods on three datasets as comparison methods. The method-specific parameters in the experiment are set according to the original code, and other parameters are listed after the method introduction.

1) *B-CNN [13]:* The output features of the two feature extraction networks are fused by the bilinear pooling, and the result can represent higher order features. The hyperparameters in the experiment are set as follows: the batch size is 16, the learning rate is 0.01, the input image size is 448, and the number of epochs is 100.
2) *TASN [34]:* Convert the extracted features into attention maps using trilinear products. Hyperparameters setting: the batch size is 16, the initial learning rate is 0.0008, the input image size is 448, and the training epoch is 360.
3) *BAT [51]:* Bilinear and attention attached with the Vision Transformer for simulating global attention and local attention. Hyperparameters setting: the batch size is 16, the initial learning rate is 0.1, the input image size is 448, and the training epoch is 100. When the epoch is equal to 30, 60, and 90, the learning rate decreases to one-tenth of the last epoch. The hyperparameters mentioned in this article follow the author's setting.

The methods of attention mechanism are given as follows.

1) *DCL [52]:* By disrupting the reconstruction learning of the original image divided into small regions, the network is forced to pay attention and learn the local difference regions. Hyperparameters setting: the batch size is 16, the initial learning rate is 0.0008, the input image size is 448, and the training epoch is 360.
2) *AC-Net [43]:* By building a binary tree, learning features are at different levels from coarse to fine. Hyperparameters setting: the batch size is 16 and the input image size is 448. The training is divided into two stages. The first stage is 60 iterations, and the learning rate is 1. The second stage is 200, and the learning rate is 0.001.
3) *PMG-v2 [33]:* Train the network to learn image features at different granularity levels and gradually fuse multi-granularity features. Hyperparameters setting: the batch size is 4, the training epoch is 200, and the input image size is 448. The learning rate of the feature extractor is set to 5e-4, and other parameters are set to 5e-3.
4) *API-Net [32]:* Compare features between different images and learn the difference between two kinds of features to achieve fine-grained image classification. Hyperparameters setting: the batch size is 100, the training epoch is 100, the learning rate is 0.01, and the input image size is 448.
5) *EA-Net [7]:* The filter aggregation mechanism is used to filter out a part of the features with the highest weight for the classification network decision.

There are also some methods that use self-attention.

1) *FFVT [31]:* Through the mutual attention weight selection module, the Transformer's attention to the local-,

TABLE III

ABLATION STUDIES FOR COMPONENTS. THE EFFECTIVENESS OF THE MINER AND THE REFINER IS LISTED. THE "IMP." REPRESENTS THE AVERAGE PERCENTAGE OF THE INCREASE IN OA OVER THE THREE DATASETS

Component		Dataset			Imp.
Miner	Refiner	FGSC-23	FGSCR-42	Aircraft-16	
×	×	90.68	98.94	89.32	-
×	✓	91.28	99.05	91.61	1.00
✓	×	91.39	99.03	91.65	1.04
✓	✓	**92.61**	**99.44**	**92.36**	1.82

TABLE IV

INFLUENCE OF FEATURE CHANNEL NUMBERS (I.E., κ) IN MHAM. EXPERIMENTS ARE CARRIED OUT ON FGSC-23, AIRCRAFT-16, AND FGSCR-42, AND κ IS SET TO 8, 16, 32, AND 64, RESPECTIVELY. THE BEST OA OF EACH DATASET IS MARKED IN BOLD

Backbone	OA(%)											
	FGSC-23				FGSCR-42				Aircraft-16			
	8	16	32	64	8	16	32	64	8	16	32	64
VGG-19	87.64	87.27	**88.85**	88.00	97.32	**97.47**	97.23	97.34	88.80	89.40	**89.73**	89.50
ResNet-50	**89.70**	88.97	88.24	88.24	98.31	98.49	**98.70**	98.37	90.01	**91.28**	91.23	90.95
ConvNeXt-Base	91.52	**92.61**	92.00	91.88	99.39	**99.44**	99.31	99.29	90.76	**92.36**	91.89	90.95

TABLE V

INFLUENCE OF HYPERPARAMETER IN CLASSIFICATION LOSS ON THREE FINE-GRAINED REMOTE SENSING IMAGE CLASSIFICATION DATASETS. γ REPRESENTS THE HYPERPARAMETER IN (20). ALL THE NUMBERS ARE THE OA (%) OF THE TEST SET

value of the γ	FGSC-23	FGSCR-42	Aircraft-16
0	91.76	98.87	91.89
0.5	**92.61**	**99.44**	**92.36**
1.0	91.64	99.09	91.98

TABLE VI

MODEL PARAMETERS, FLOATING POINT OPERATIONS, AND THE OA OF BACKBONE AND EFM-NET ON FGSC-23 [16] DATASET ARE COMPARED

Model	Params(M)	GFlops	OA(%)
VGG-19	20.04	19.57	87.9
EFM-Net (VGG-19)	22.16	19.57	88.9
ResNet-50	23.51	4.13	84.6
EFM-Net (ResNet-50)	32.00	4.15	89.7
ConvNeXt-Base	87.51	15.35	90.0
EFM-Net (ConvNeXt-Base)	91.76	15.37	92.6

TABLE VII

COMPARISON OF DIFFERENT BILINEAR POOLING METHODS IN TERMS OF DIMENSION OF FUSION FEATURE, COMPUTATIONAL COMPLEXITY, AND PARAMETERS. C, h, AND w DENOTE THE DIMENSION, HEIGHT, AND WIDTH OF THE OUTPUT FEATURE BY THE BACKBONE, RESPECTIVELY. OTHERS ARE HYPERPARAMETERS. $d = 8192$, $m = 100$, $g = 256$, AND $k = 16$. HYPERPARAMETER κ IS SET TO 16

Method	Dim.	Comp.	Param.
Full Bilinear Pooling [13]	C^2	$O(hwC^2)$	0
CBP-TS [54]	d	$O(hw(C + d * logd))$	$2C$
LRBP [55]	m^2	$O(hwmC + hwm^2)$	mC
IBP [56]	C^2	$O(hwC^2 + C^3)$	0
GP-256 [57]	kg	$O(hwCg + g^3)$	gC
BPP(ours)	κC	$O(hwC)$	0

medium-, and low-level features is improved, and the Transformer's selection of tokens is guided. Hyperparameters setting: the batch size is 8, the training step is 2000, the learning rate is 0.02, and the input image size is 384.

2) *SIM [53]:* Mining the spatial and background relationships of important plates within the object range with the help of the Transformer's self-attention weight, and identifying the target with multilevel features. Hyperparameters setting: the batch size is 5, the training step is 10000, the learning rate is 0.03, and the input image size is 448.

C. Comparisons With State-of-the-Art Methods

In our experiments, the overall accuracy (OA) and the accuracy rate (AR) of each category are the basic metrics for evaluating the performance of the model. OA represents the proportion of correctly classified images to the number of the test set, and AR is the proportion of correctly classified images to a single class. In addition to the OA, we list the input image size of our network and other methods. The metric of AR can be found in Fig. 8. The detailed analysis is given as follows.

1) Performance on FGSC-23: In fact, FGSC-23 [16] contains coarse-grained categories, such as nonship and destroyer classes. Nevertheless, our method still performs very well. Table II summarizes the classification results of the current excellent methods on three datasets. "*" represents that the results are from other articles [7], [50]. It can be found that the self-attention methods [31], [53] using Vision Transformer do not perform well on this dataset, and our result is much better than that of FFVT. Some methods using bilinear or trilinear product performance perform poorly, and BAT [51] is even difficult to converge. Other methods based on attention mechanisms, such as API-Net [32] and PMG-v2 [33], are not as effective as ours. Our method based on ConvNeXt-Base [14] achieves the competitive result and exceeds the second EA-Net [7] by 0.8%. In addition, only using features extracted by backbone for classification is applied in the experiments, but the unsatisfactory results indicate the importance of feature enhancement.

2) Performance on FGSCR-42: As a fine-grained image classification dataset with over 9000 remote sensing images and 42 classes, these numbers are nearly twice as large as FGSC-23, reflecting the greater challenge of the FGSCR-42 [17]. Our approach still leads the rest. As shown in Table II, our model topped the charts with a 1.3% advantage over the second on the FGSCR-42. It is worth mentioning that EA-Net [7] is not outstanding enough in this dataset; meanwhile, our method has a strong generalization performance, especially in this large dataset. The OA in Table II indicates the

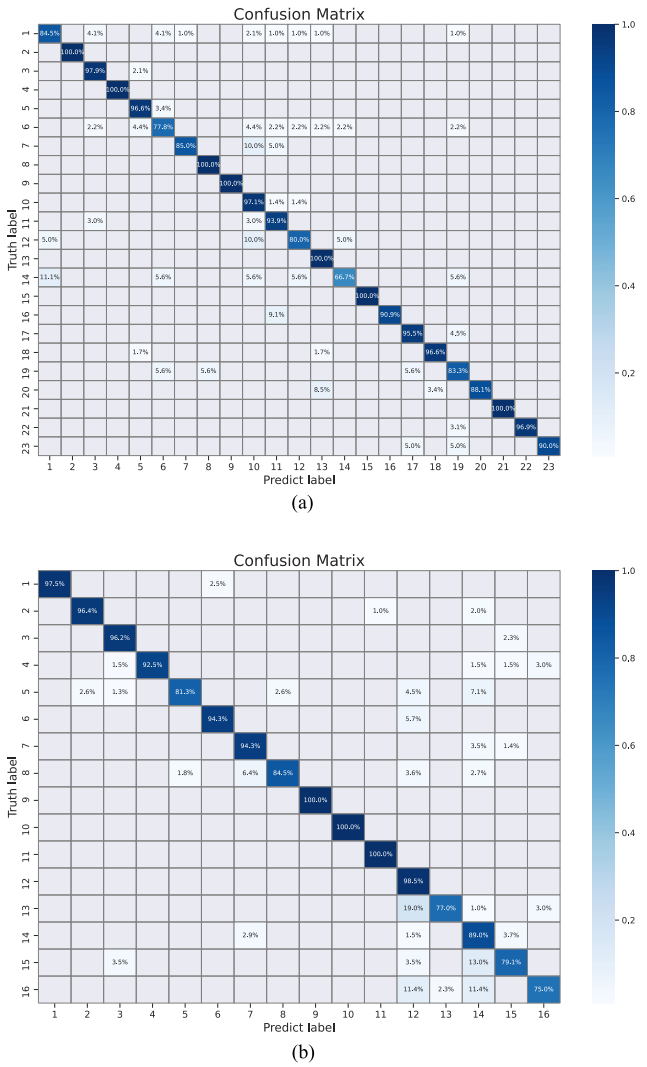

Fig. 8. Confusion matrices of EFM-Net on (a) FGSC-23 and (b) Aircraft-16 datasets. The number in the square on the diagonal of the matrix represents the AR of each category. The darker the color of the square, the closer the AR value is to 100%. The color of the squares on the diagonal lines of the confusion matrix for both datasets is very dark, indicating that the classification effect is ideal.

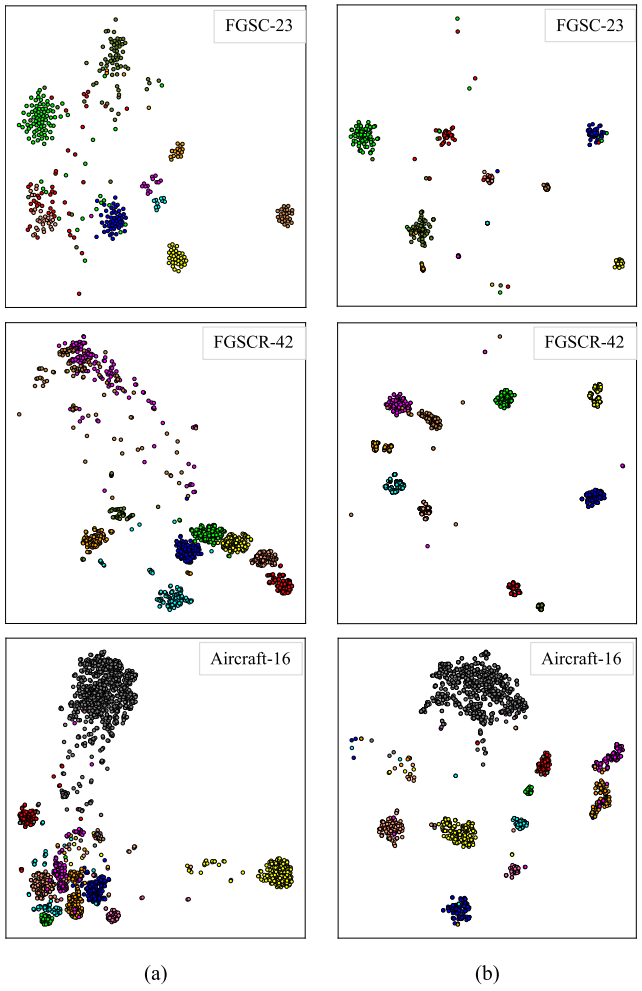

Fig. 9. Essential feature and the intrinsic feature distribution visualization using t-NSE [58]. The t-NSE parameter settings are unified as follows: perplexity is 30, the learning rate is 200, and early exaggeration is 12. Select ten categories with a larger number of each dataset for feature distribution visualization. The color and category correspondences in (a) and (b) are consistent. (a) and (b) Visualization of intrinsic features and essential features, respectively. It is better to zoom in and view in color.

effectiveness of the mechanism for essential feature learning in EFM-Net.

3) Performance on Aircraft-16: We take the open-source approach on the previous datasets and test their model's performance on our dataset of aircraft targets. We show the results of comparison methods on the Aircraft-16 dataset in Table II for analysis. Compared with ships, aircraft have more distinct rigid and detailed structural features, such as the explicit features of the aircraft's engine and the shape of its wings. As shown in Table II, owing to the lack of detailed features and texture, the classification performance of the backbone network is obviously poor. Our method obtains the most discriminative feature and achieves satisfactory classification results. Our model achieves a maximum OA of 92.4%, which is 2.7% higher than the second. The proposed method still performs very well on the fine-grained classification dataset of aircraft targets. Therefore, the outstanding results show that our method performs well on datasets with different target morphologies and label annotation granularities.

D. Ablation Study

In this section, we will verify the effectiveness and necessity of each architecture designed in our work through a series of ablation experiments and explain the reasons behind our design and choice.

1) Effectiveness of the Components: We conduct ablation experiments on three datasets, FGSC-23 [16], FGSCR-42 [17], and Aircraft-16, and test the classification performance under only the backbone, adding the Miner, adding the Refiner, and the complete model. Considering the possible overfitting effect caused by the excessive model complexity, we only select the output features of the last three stages to build multiple hierarchies features.

The results are shown in Table III. When the Miner and the Refiner are all removed, only the intrinsic features enter

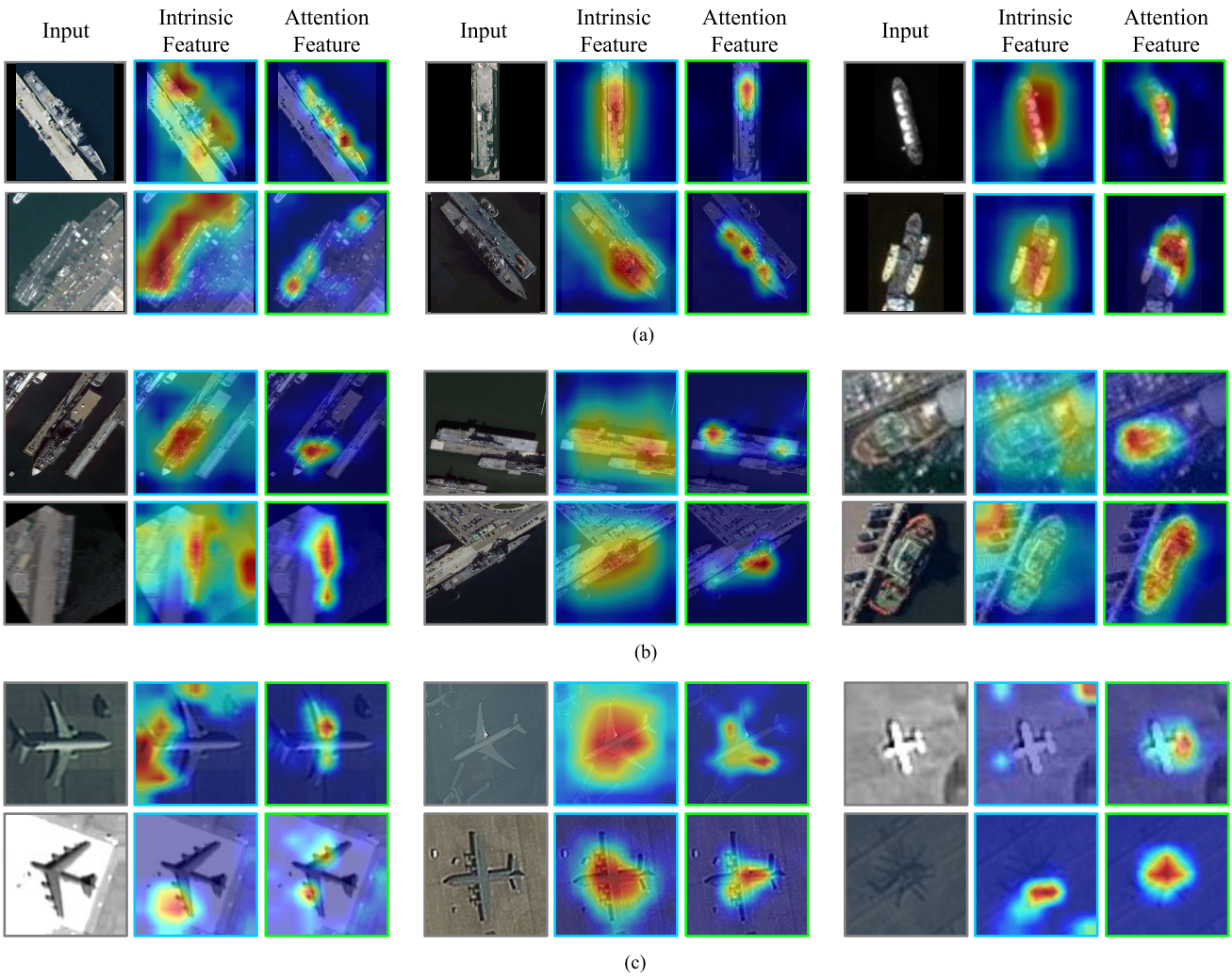

Fig. 10. Difference between the intrinsic feature and the attention feature on activation maps. Images are from three datasets: (a) FGSC-23, (b) FGSCR-42, and (c) Aircraft-16. The visualization results are obtained via the Grad-CAM [59]. "Intrinsic Feature" and "Attention Feature" represent the activation maps of the output features by ConvNext-Base and MHAM. These feature maps have the same size. Intrinsic features focus on a small area, which is not accurate enough and contains background and interference. The areas where essential features are concerned are relatively concentrated and accurate.

the classification network, and the result is not outstanding over other methods. After adding the Miner, the average OA is improved by 1.04% on the three datasets, which indicates that the Miner can facilitate the intrinsic features and enhance fine-grained features. In addition, the model with only Refiner has an average improvement of 1.00% on all datasets. It is worth noting that the model with a Refiner has improved by 2.33% on Aircraft-16, which verifies the importance of the data augmentation mechanism. The classification performance of the model after adding a Refiner is already better than most existing methods. This ablation experiment demonstrates that the two components that we propose are well-compatible and can be optimized in a complementary manner, ultimately enabling the EFM-Net to obtain the essential features.

2) Decision of the Channel Numbers in MHAM: The channel number of intrinsic features needs to be unified before entering the MHAM. In this process, the number of feature channels affects both the richness of input information and the amount of network computation. To study the effect of the number of channels in the attention network, we conduct the following experiments.

We choose VGG-19 [18], ResNet-50 [15], and ConvNeXt-base [14] as the feature extractor, respectively, and test the effect of channel number κ on classification on three datasets. As shown in Table IV, for VGG-19 and ResNet-50, the value of κ on different datasets has a greater impact on the classification performance. This indicates that the features extracted by VGG and ResNet are confusing. When ConvNeXt-base is used as the backbone, $\kappa = 16$ achieves the best results on each dataset. The more generalized ConvNeXt-base ensures the consistency of the classification performance of the model on different datasets. That is why, we combine ConvNeXt with our network.

3) Effectiveness of the SCE Loss: The hyperparameter γ of the SCE loss in (20) represents the degree of deviation from the cross-entropy loss curve. When $\gamma = 0$, the SCE loss is equivalent to cross-entropy loss. The experimental results are shown in Table V. From the experimental results, the

classification effect is relatively better in different datasets when $\gamma = 0.5$, which verifies the rationality of the preset value. It proves that the effectiveness of SCE loss is universal. Our method with SCE loss improves OA by at least 0.5% on each dataset, which verifies the effectiveness of the new loss function.

E. Discussion

1) Analysis of Model Complexity: As shown in Table VI, the flops of our model are basically equal to the backbone network. The number of model parameters is increased relative to the number of backbones, but the increase is negligible relative to the number of backbone parameters. It indicates that neither BPP nor FMN network will increase the complexity of the model.

The BPP performs a bilinear product on the two homologous features, including the attention features by the MHAM and the intrinsic features. Through the strengthening and filtering of MHAM, a large number of repeated response channels can be filtered out, and the fusion feature compressed into κ channels only contains the target attention feature. In this process, not only the computational complexity is reduced by avoiding repeated channels but also the interference to the classification is removed. Intrachannel average pooling in BPP suppresses the interference of peak response from the background. The comparison of the bilinear pooling method is shown in Table VII.

2) Analysis of the AR: The difficulty of the target fine-grained classification task lies in: 1) low interclass bias and high intraclass variance and 2) unbalanced samples. The confusion matrix is used for analyzing the AR in each class. In this way, similar categories, as well as low sample size categories, can be observed in Fig. 8. It shows the classification confusion matrix of EFM-Net on typical datasets FGSC-23 and Aircraft-16. In Fig. 8(a), #1 (nonship) and #14 (auxiliary ship) have a large degree of confusion. It is difficult to learn because the essential features of nontarget classification are different from ship target features. In Fig. 8(a), #6 (LPD) contains ships that are an insufficiently segmented class, resulting in difficulties for the model to learn accurate features. In addition, most AR is basically above 90%, and one-third of the categories are reaching 100%. In Fig. 8(b), #1, #6, #8, and #9 with only slight morphological differences are precisely distinguished by EFM-Net, and both #9 and #10 with uneven sample sizes can be accurately distinguished. The confusion matrices indicate the effectiveness of Refiner in feature learning for classes with unbalanced samples.

3) Analysis of the Essential Feature: t-NSE [58] is often used for 2-D visualization of high-dimensional features, so we use this tool in experiments. t-NSE visualizes the intrinsic feature vector in Section III-B2 and essential feature vector obtained by Miner in Section III-B3. We pick ten categories with more samples, and the points with the same color in (a) and (b) correspond to the same category. As can be seen from Fig. 9(a), the intraclass variance of the intrinsic feature vectors in FGSC-23 [16] is large, which makes the distribution of the same class more scattered, and the corresponding essential feature has a larger distance between classes and more clustering within a class. Especially, in FGSCR-42, the blue, green, and yellow point sets in (a) are close together, but, in (b), these sets have obvious boundaries. This also implies the excellent solution of the feature decoupling effect of FMN in Section III-B4. Similarly, the classes clustered in the Aircraft-16 and FGSCR-42 eigenfeatures are separated. Feature space visualization shows the effectiveness of EFM-Net in extracting features with high interclass separability. In addition, essential features can better distinguish different categories and generalize objects of the same category than intrinsic features.

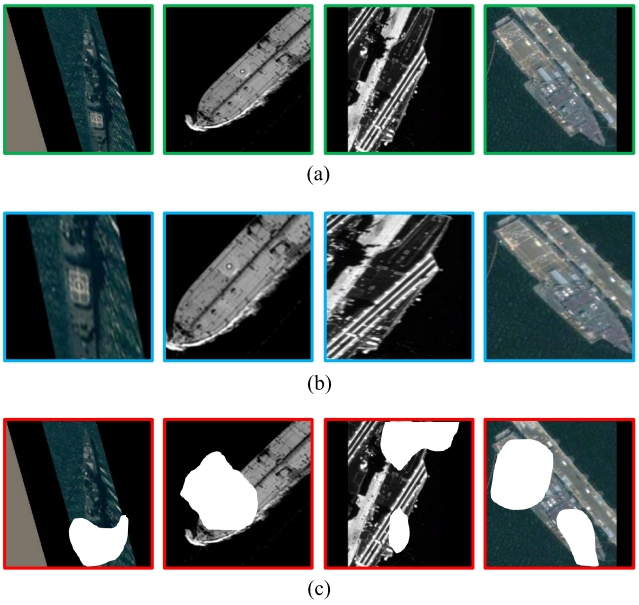

Fig. 11. Visualization of the input image from FGSC-23 [16], detail image, and mask image output by Refiner during training. The image in the green box in (a) represents the input image, the image in the blue box in (b) represents the detail image, and the image in the red box in (c) represents the mask image.

4) Analysis of the MHAM: As shown in Fig. 10, we use Grad-CAM [59] to generate an activation map of the intrinsic features from the backbone network and the attention features from MHAM. In the activation map, the red and blue regions represent the attention and inhibition of our model, respectively. From the activation maps, we can observe that the area concerned by the backbone is discrete and loose, accompanied by more noise, and is not sensitive to the features of small targets. The attention features focus on the region of interest while removing interference. The area of attention on FGSC-23 [16] and FGSCR-42 [17] is elongated, and using interchannel attention helps to focus on important areas. Attention regions on Aircraft-16 are close to point-like, relying on low-level locations and precise localization of detailed information. Therefore, the MHAM used in Miner can well solve the problems of noise interference and small target loss. In addition, the essential features that are generated by fusing these two features contain the advantages of both.

5) Visualization of the Refiner: To illustrate the effectiveness of our Refiner, we save the detail image and mask image

of the Refiner output at a certain moment in the training as images, as shown in Fig. 11. It is obvious that the image in Fig. 11(b) is enlarged compared to the image in Fig. 11(a), in which background and invalid target features are removed. This enables the model to more accurately locate the region where the essential features of the target are located, which is conducive to attention-gathering. The image in Fig. 11(c) has increased occlusion compared to the input image. This occlusion is not regular but varies with the feature map. This masking mechanism prevents the model from overfitting and releases the problem that excessively focused attention is detrimental to the generalization of the model. The random coverage reflects the flexibility of network adaptive learning.

V. Conclusion

In this article, we propose the EFM-Net for target fine-grained classification in remote sensing images, which explores extracting the essential feature of the target. The network consists of the Miner and the Refiner. The Miner can effectively filter, strengthen the intrinsic features, and fuse them with the attention feature to obtain the most discriminative feature. The Refiner can locate more regions containing essential features. The Miner and the Refiner can complement each other, and the attention feature is beneficial to mine and locate discriminative regions, thereby promoting Miner to enhance features more effectively. A large number of experiments show that the model can accurately extract and refine features and strong generalization performance on three datasets: FGSC-23, FGSCR-42, and Aircraft-16. The results show the necessity and importance of the proposed modules. Visualizations further show the interpretability of our method. In the future, we will continue to tap the potential of Refiner based on excellent Miner and optimize the learning strategy of EFM-Net.

References

[1] Y. Ding, Y. Chong, S. Pan, Y. Wang, and C. Nie, "Spatial–spectral unified adaptive probability graph convolutional networks for hyperspectral image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Sep. 23, 2021, doi: 10.1109/TNNLS.2021.3112268.

[2] S. Huang, H. Zhang, and A. Pizurica, "Subspace clustering for hyperspectral images via dictionary learning with adaptive regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5524017.

[3] B. Niu, Z. Pan, J. Wu, Y. Hu, and B. Lei, "Multi-representation dynamic adaptation network for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5633119.

[4] S. Chen, H. Wang, F. Xu, and Y. Q. Jin, "Target classification using the deep convolutional networks for SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4806–4817, Aug. 2016.

[5] Y. Han, X. Yang, T. Pu, and Z. Peng, "Fine-grained recognition for oriented ship against complex scenes in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–18, 2021.

[6] Y. Nie, C. Bian, and L. Li, "Adap-EMD: Adaptive EMD for aircraft fine-grained classification in remote sensing," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[7] 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.

[8] W. Liang, J. Li, W. Diao, X. Sun, K. Fu, and Y. Wu, "FGATR-Net: Automatic network architecture design for fine-grained aircraft type recognition in remote sensing images," *Remote Sens.*, vol. 12, no. 24, p. 4187, Dec. 2020.

[9] K. Fu, W. Dai, Y. Zhang, Z. Wang, M. Yan, and X. Sun, "MultiCAM: Multiple class activation mapping for aircraft recognition in remote sensing images," *Remote Sens.*, vol. 11, no. 5, p. 544, Mar. 2019.

[10] P. Shamsolmoali, M. Zareapoor, J. Chanussot, H. Zhou, and J. Yang, "Rotation equivariant feature image pyramid network for object detection in optical remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2021.

[11] W. Zhang, L. Jiao, Y. Li, Z. Huang, and H. Wang, "Laplacian feature pyramid network for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2021.

[12] C. Zhang, W. Jiang, Y. Zhang, W. Wang, Q. Zhao, and C. Wang, "Transformer and CNN hybrid deep neural network for semantic segmentation of very-high-resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4408820.

[13] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear CNN models for fine-grained visual recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1449–1457.

[14] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," 2022, *arXiv:2201.03545*.

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

[16] 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.

[17] 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, p. 747, Feb. 2021.

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

[19] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[21] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1492–1500.

[22] H. Zhang et al., "ResNeSt: Split-attention networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 2736–2746.

[23] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10012–10022.

[24] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[25] Q. Shi, W. Li, R. Tao, X. Sun, and L. Gao, "Ship classification based on multifeature ensemble with convolutional neural network," *Remote Sens.*, vol. 11, no. 4, p. 419, Feb. 2019.

[26] P. Qin, Y. Cai, J. Liu, P. Fan, and M. Sun, "Multilayer feature extraction network for military ship detection from high-resolution optical remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 11058–11069, 2021.

[27] L. Chen, Y. Wei, Z. Yao, E. Chen, and X. Zhang, "Data augmentation in prototypical networks for forest tree species classification using airborne hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4410116.

[28] G. Sumbul, R. G. Cinbis, and S. Aksoy, "Multisource region attention network for fine-grained object recognition in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, pp. 1–9, 2019.

[29] J. Fu, H. Zheng, and T. Mei, "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4438–4446.

[30] Z. Yang, T. Luo, W. Dong, Z. Hu, and L. Wang, *Learning to navigate for fine-grained classification*. Cham, Switzerland: Springer, 2018.

[31] J. Wang, X. Yu, and Y. Gao, "Feature fusion vision transformer for fine-grained visual categorization," 2021, *arXiv:2107.02341*.

[32] P. Zhuang, Y. Wang, and Y. Qiao, "Learning attentive pairwise interaction for fine-grained classification," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, 2020, pp. 13130–13137.

[33] R. Du, J. Xie, Z. Ma, D. Chang, Y.-Z. Song, and J. Guo, "Progressive learning of category-consistent multi-granularity features for fine-grained visual classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 9521–9535, Dec. 2022.

[34] H. Zheng, J. Fu, Z.-J. Zha, and J. Luo, "Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5012–5021.

[35] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, no. 2, 2012, pp. 84–90.

[36] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, *arXiv:1312.4400*.

[37] W. Liang, Y. Wu, M. Li, P. Zhang, Y. Cao, and X. Hu, "A feature fusion-net using deep spatial context encoder and nonstationary joint statistical model for high resolution SAR image classification," 2021, *arXiv:2105.04799*.

[38] E. Li, A. Samat, P. Du, W. Liu, and J. Hu, "Improved bilinear CNN model for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2020.

[39] D. Yu, H. Guo, Q. Xu, J. Lu, C. Zhao, and Y. Lin, "Hierarchical attention and bilinear fusion for remote sensing image scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 6372–6383, 2020.

[40] J. He et al., "Group bilinear CNNs for dual-polarized SAR ship classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[41] A. Fukui, D. Huk Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach, "Multimodal compact bilinear pooling for visual question answering and visual grounding," 2016, *arXiv:1606.01847*.

[42] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[43] R. Ji et al., "Attention convolutional binary neural tree for fine-grained visual categorization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10468–10477.

[44] G.-S. Xia et al., "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983.

[45] Z. Liu, L. Yuan, L. Weng, and Y. Yang, "A high resolution optical satellite image dataset for ship recognition and some new baselines," in *Proc. 6th Int. Conf. Pattern Recognit. Appl. Methods*, 2017, pp. 324–331.

[46] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[47] Z.-Z. Wu et al., "A benchmark data set for aircraft type recognition from remote sensing images," *Appl. Soft Comput.*, vol. 89, Apr. 2020, Art. no. 106132.

[48] J. Shermeyer, T. Hossler, A. V. Etten, D. Hogan, R. Lewis, and D. Kim, "RarePlanes: Synthetic data takes flight," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 207–217.

[49] X. Sun et al., "FAIR1M: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 184, pp. 116–130, Feb. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271621003269

[50] Y. Yi, Y. You, W. Zhou, and G. Meng, "MHA-CNN: Aircraft fine-grained recognition of remote sensing image based on multiple hierarchies attention," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2022, pp. 3051–3054.

[51] L. Chi, Z. Yuan, Y. Mu, and C. Wang, "Non-local neural networks with grouped bilinear attentional transforms," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11804–11813.

[52] Y. Chen, Y. Bai, W. Zhang, and T. Mei, "Destruction and construction learning for fine-grained image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5157–5166.

[53] H. Sun, X. He, and Y. Peng, "SIM-Trans: Structure information modeling transformer for fine-grained visual categorization," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 5853–5861.

[54] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell, "Compact bilinear pooling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 317–326.

[55] S. Kong and C. Fowlkes, "Low-rank bilinear pooling for fine-grained classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 365–374.

[56] T.-Y. Lin and S. Maji, "Improved bilinear pooling with CNNs," 2017, *arXiv:1707.06772*.

[57] X. Wei, Y. Zhang, Y. Gong, J. Zhang, and N. Zheng, "Grassmann pooling as compact homogeneous bilinear pooling for fine-grained visual classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 355–370.

[58] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 1–27, 2008.

[59] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

Yonghao Yi (Student Member, IEEE) received the B.S. degree from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, in 2021, where he is currently pursuing the M.S. degree with the School of Artificial Intelligence.

His research interests are in fine-grained image classification, especially in remote sensing image processing.

Yanan You (Member, IEEE) received the Ph.D. degree from the School of Electronic and Information Engineering, Beihang University, Beijing, China, in 2015.

He held a post-doctoral position at Beihang University from 2015 to 2017. He is currently an Associate Professor with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing. His research interests are in remote sensing image processing, deep learning, imaging detection, and intelligent perception.

Chao Li received the B.E. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2020, where he is currently pursuing the M.S. degree with the School of Artificial Intelligence.

His research interests are in remote sensing image processing, especially in image registration and local descriptor construction.

Wenli Zhou received the Ph.D. degree in engineering in signal and information processing from the Beijing University of Posts and Telecommunications, Beijing, China, in 2006.

She is currently an Associate Professor with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications. Her research interests include network traffic monitoring, user behavior analysis, telecommunications and Internet big data processing, and other research.