Exploring Model Compression Limits and Laws: A Pyramid Knowledge Distillation Framework for Satellite-on-Orbit Object Recognition

Yanhua Pang, *Graduate Student Member, IEEE*, Yamin Zhang, Yi Wang, Xiaofeng Wei, and Bo Chen, *Member, IEEE*

***Abstract*— Extremely constrained storage and computational resources are one of the difficulties of satellite-on-orbit computing, which leads to over-parametric high-performance models not performing properly on-orbit. Knowledge distillation (KD) is an effective method for model compression; yet, there is a gap in the study of the limits and laws of KD-based model compression. To bridge this gap, we propose a novel KD framework, pyramid KD (PKD) and define a knowledge explosion and knowledge offset. Specifically, the pyramid distillation framework is built by stacking multiple sets of deep mutual learning (DML) models, with the smaller models on the top of the larger ones, and the overall structure is like a pyramid; hence, it is called PKD. To avoid knowledge explosion, we design a hybrid online–offline smooth distillation (HOSD) strategy by combining online distillation and offline distillation and reducing the difference between models. To avoid knowledge offset, we design an adaptive multiteacher distillation method to obtain multiteacher weighted knowledge by adaptively learning the weight of each teacher's knowledge. We introduce an evolutionary algorithm to automatically find the optimal PKD configuration. We conduct ablation experiments and compare PKD with state-of-the-art distillation methods using ResNet series networks and VGG series networks as base models on Aircraft and FGSC-23 datasets, respectively. The experimental results show the effectiveness and advancement of PKD and reveal the law that the object recognition accuracy varies with the model compression rate.**

***Index Terms*— Knowledge distillation (KD), model compression, object recognition, remote sensing image, satellite-on-orbit computing.**

I. Introduction

SATELLITE-ON-ORBIT computing can reduce the pressure on satellite-to-ground data transmission bandwidth, improve the timeliness of information acquisition and greatly improve satellite application efficiency [1]. However, the computing resources and storage resources on the satellite are extremely limited by factors, such as satellite heat dissipation and power consumption, which makes it difficult for the over-parameterized state-of-the-art deep learning models to operate normally on the satellites [2]. Many scholars have studied various methods for model compression acceleration, and knowledge distillation (KD) is an effective model compression method. KD generally uses a larger model as a teacher model to guide a smaller student model. Although the student model, which incorporates knowledge from the teacher's model, has lower model complexity and smaller model size, its performance is comparable to that of the teacher's model [3].

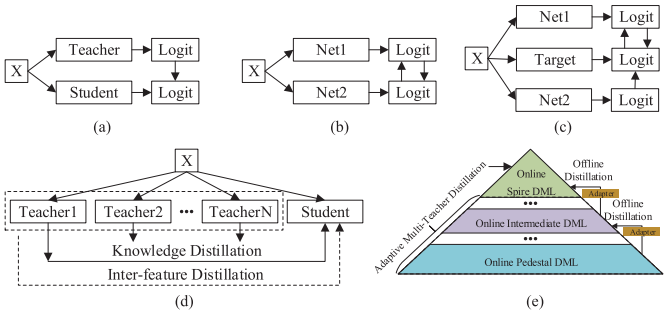


Fig. 1. Several variants of KD. (a) Classical offline distillation. (b) Online distillation (DML). (c) Online–offline distillation. (d) Multiteacher distillation. (e) Pyramid distillation.

There are many studies on the application of model compression methods based on KD on remote sensing images [4], [5], and the distillation scheme of most research works on current KD is offline, as shown in Fig. 1(a). The biggest advantage of offline distillation is that it is easy to implement [6]. Offline distillation is mainly divided into two stages: 1) large-capacity teacher model training to obtain a pretrained teacher model and 2) loading the teacher pretraining model in stage 1 to assist in guiding the student model training process. Most researchers focus on the design of knowledge [3], [7] and the improvement of the loss function for feature matching [8], [9], while little research has been done on the structure of the teacher model and the relationship between the teacher model and the student model. The student model obtained by the distillation method of offline distillation is extremely dependent on the prior knowledge of the teacher model. Therefore, the structure of the teacher model and the relationship between the teacher model and the student model determines the degree

Manuscript received 25 October 2023; revised 3 December 2023; accepted 27 December 2023. Date of publication 1 January 2024; date of current version 10 January 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFF0503900, in part by the Natural Science Foundation of Guangdong Province under Grant 2022A1515010113, and in part by the Natural Science Foundation of Shenzhen City under Grant GXWD20220811163556003. *(Corresponding author: Bo Chen.)*

The authors are with the Institute of Space Science and Applied Technology, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China (e-mail: gm.yanhuapang@gmail.com; skdzym@sina.com; wangyi2021@hit.edu.cn; weixiaofeng@pku.edu.cn; hitchenbo@hit.edu.cn).

Digital Object Identifier 10.1109/TGRS.2023.3348470

to which the student model absorbs the knowledge of the teacher model [8]. Yang et al. [10] proposed the category correlation and adaptive KD for compact cloud detection in remote sensing images. Zhang et al. [11] distilled the backbone and head of the detection network separately and further improved the regression of difficult samples by modifying the loss function. Li et al. [12] proposed a novel dual KD model combining dual attention and spatial structure to solve the contradiction between the accuracy of the convolutional neural network (CNN) and a large amount of model parameters. Yang et al. [13] proposed the statistical sample selection and multivariate knowledge mining for lightweight detectors in remote sensing imagery. Xu et al. [14] adopted vision transformer as a teacher model and utilized its advantages in contextual information extraction to guide a small CNN student model with local feature extraction advantages, their experiments yielded good distillation performance.

Online distillation is a single-stage end-to-end distillation scheme with efficient parallel computing, as shown in Fig. 1(b). Online distillation allows multiple models to be trained simultaneously, without a specific teacher model and student model, and these models learn from each other, resulting in a small model that often performs better in DML than that model trained with itself alone [15]. Various variants of online distillation have been studied in recent years. Guo et al. [16] utilized ensemble soft logits of multiple models to enhance the generalization ability of the model. Chen et al. [17] introduced auxiliary peers and group leaders into DML to enhance the diversity of features mined by the models. Chung et al. [18] introduced the idea of adversarial training into DML and trained multiple models simultaneously through the discriminator using knowledge of both category probability and feature map. Distillation lacks effective supervision using online distillation alone, and offline distillation can compensate for this deficiency, so the combination of online and offline distillations can further improve the robustness of the model.

With the in-depth research on KD techniques, Su et al. [19] found that the student model obtained by using both online distillation and offline distillation for KD is more robust, as shown in Fig. 1(c). Two different distillation methods, online distillation and offline distillation, are fused in a single distillation framework, and this method not only has the advantage of efficient and parallel training of online distillation but also gains the supervision guidance of high-capacity models in offline distillation. Specifically, multilevel online mutual learning can not only learn intermediate features but also advanced features, while benefitting from the prior knowledge of pretrained models as supervision to enhance the efficiency of student model feature extraction. The combination of online distillation and offline distillation is the trend of future research work on KD. There is limited research on the distillation method of multiple-layer online–offline distillation stacking, which is prone to the problem of knowledge offset.

Multiteacher distillation is generally considered as a variation of offline distillation, as shown in Fig. 1(d), with multiple teachers guiding student model at multiple levels [20]. The solid line in Fig. 1(d) represents multiple teachers guiding student models separately [21]. However, there is no association between multiple teacher models in this distillation method, which tends to distill duplicated knowledge to student models, causing the problem of inefficient distillation. The dashed line in Fig. 1(d) represents the average knowledge of multiple teachers' models as internal features for distillation of student model [22], which avoids not only the problem of inefficient distillation but also brings another question of whether the average knowledge can represent the degree of importance of each teacher's knowledge. In this article, we use an adaptive learning approach to learn the importance of teachers' knowledge and then use the combined weighted knowledge to guide the student model distillation training.

The aforementioned model compression approaches based on KD have shown promising experimental results. However, there are still areas that can be improved upon. First, due to significant differences in model structure and size between the student model and the teacher model, the student model is unable to fully absorb the knowledge from the teacher model, i.e., knowledge explosion. Second, the use of multiple teacher models in KD [as shown in Fig. 1(d)] can result in knowledge shift during knowledge transfer, i.e., knowledge offset. Finally, there is a lack of research on the laws and limits of model compression, particularly in scenarios with extremely limited computational resources.

To solve the above problems, we propose a novel KD, PKD, which aims to explore model compression limits for satellite-on-orbit computing. Our main contributions are summarized as follows.

1) To bridge the gap in the study of the limits and laws of KD-based model compression, we propose a novel KD framework, PKD. The framework automatically and flexibly adjusts the number of pyramid layers to obtain models with different accuracies corresponding to different compression ratios (CRs), thereby drawing the accuracy-CR curve and obtaining the influence of the model CR on the variation law of model accuracy.
2) To solve the problem that the compact student models cannot fully absorb the knowledge of the large-capacity teacher models, named as knowledge explosion, we innovatively design a hybrid online–offline smooth distillation (HOSD) strategy, i.e., the models of the PKD framework are trained with both online distillation and offline distillation, and they are very similar in structure and size from bottom to top, both within and between layers.
3) To solve the knowledge offset problem, i.e., the compact student model fails to learn effective knowledge due to knowledge offset resulting from knowledge transfer between multiple teachers, we design an adaptive weighted multiteacher distillation method (AWMD). Specifically, the layers near the bottom offline guide the layers near the top to perform online deep mutual learning (DML) through adaptive learning teachers' weights. All the models except the Spire models are used as a teacher models group, and the Spire models are offline guided by the adaptive learning teachers' weights for online DML.

The rest of this article is organized as follows. Section II details our proposed PKD. Section III demonstrates the experimental results on the ground server and the satellite board to evaluate the effectiveness of the proposed approach. Finally, Section IV draws the conclusion.

With the in-depth research on KD techniques, Su et al. [19] found that the student model obtained by using both online distillation and offline distillation for KD is more robust, as shown in Fig. 1(c). Two different distillation methods, online distillation and offline distillation, are fused in a single distillation framework, and this method not only has the advantage of efficient and parallel training of online distillation but also gains the supervision guidance of high-capacity models in offline distillation. Specifically, multilevel online mutual learning can not only learn intermediate features but also advanced features, while benefitting from the prior knowledge of pretrained models as supervision to enhance the efficiency of student model feature extraction. The combination of online distillation and offline distillation is the trend of future research work on KD. There is limited research on the distillation method of multiple-layer online–offline distillation stacking, which is prone to the problem of knowledge offset. The method proposed in this article employs an AWMD strategy, which effectively addresses this problem.

II. Methodology

In this section, we will elaborate on the research motivation of PKD, the PKD framework, the HOSD method, the AWMD, the total loss training algorithm, the exploration of compression limits, and laws of distillation models based on particle swarm optimization algorithm and satellite-on-orbit object recognition based on PKD.

A. Motivation

The research motivation of PKD mainly comes from two aspects: the shape of the pyramid and the teaching scene of teachers and students in real life. First, the purpose of KD for model compression is to obtain high-precision compact student models with high CR. Naturally, there is a novel idea of stacking the high-capacity teacher models and the compact student model of KD in a pyramidal structure with a large bottom and a small top. The top of the pyramid is a small spike and the student model at the corresponding position is also small. We can distill a compact student model with a larger CR with the help of the pyramid frame and then explore the limit of model compression by adjusting how much of the middle layer of the pyramid. Second, in the actual situation where teachers teach students, on the one hand, there is a generation gap between older teachers and younger students, which prevents students from fully understanding and absorbing the knowledge taught by teachers. The corresponding situation in KD is knowledge explosion. The intuitive manifestation of knowledge explosion is that, given a student model, the performance gain in KD based on a large teacher model is equal to or even less than the performance gain in KD based on a medium-sized teacher model. The solution in this article is to use an HOSD method to reduce the difference between the teacher model and student model and to mix online and offline distillation methods to improve the learning efficiency of student models. On the other hand, when knowledge is transmitted among generations of teachers, each teacher adds to the knowledge more or less his own understanding, which is not the knowledge itself, so knowledge happens offset when teachers teach students, resulting in students not fully understanding and absorbing the knowledge itself. The corresponding situation in KD is knowledge offset. The intuitive manifestation of knowledge offset is that, given a student model, the distilled performance gain based on unweighted multiteacher distillation is smaller than the distilled performance gain based on weighted multiteacher distillation. The solution in this article is to use AWMD to adaptively learn the weight of each teacher's knowledge and then obtain the weighted combined knowledge of all teachers to guide the student model training process to avoid the generation of knowledge offset.

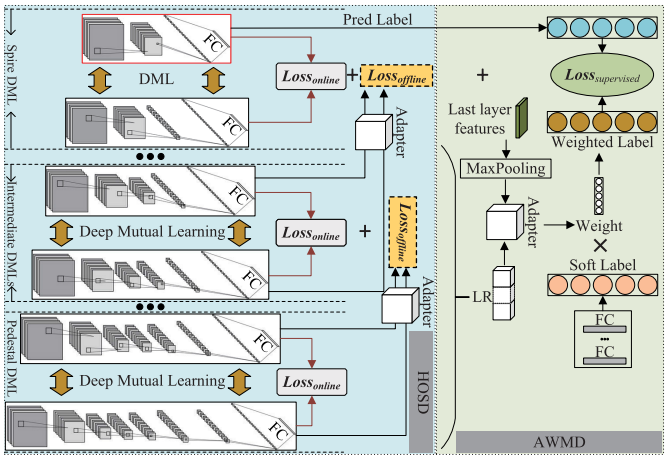


Fig. 2. Framework of the proposed PKD.

B. Pyramid Knowledge Distillation (PKD) Framework

The PKD framework, as shown in Figs. 1(e) and 2, is inspired by the shape of a pyramid and aims to obtain a compact student model with a high CR like a pyramid Spire by using PKD. The overall framework of PKD is similar to that of the pyramid structure. The PKD framework is divided into three parts: the Spire DML, the intermediate DMLs, and Pedestal DML. The small model is above the large model, the Pedestal models are the largest, and the Spire models are the smallest.

We assume that the pyramid has a total of N layers, and the Spire layer is the Nth layer and is represented as L_N, and the Pedestal layer is the 1st layer and is represented as L_1. There are two models, one large and another small, for DML between each layer. The large model is represented as L_{nl}, and the small model is represented as L_{ns}, where n represents the nth layer. The number of network layers of a deep learning model is represented by the letter H. I in (1) and (2) represents the network layers interval, I_{intra} represents the intralayer interval of the PKD framework and I_{inter} represents the interlayer

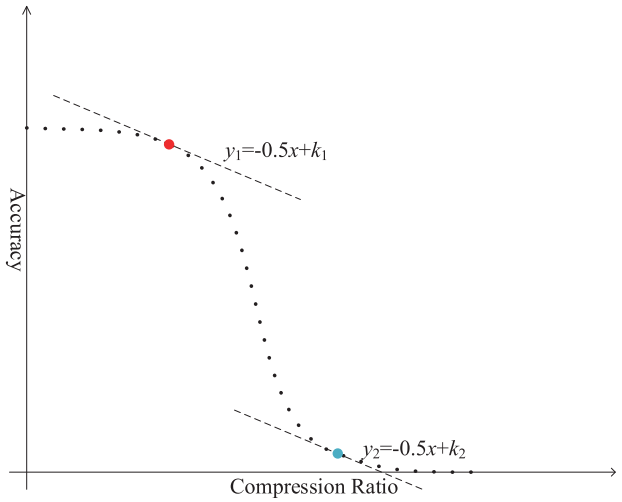


Fig. 3. Accuracy-CR curve schematic, the red dots represent the point of optimal CR and accuracy. The X-axis is the CR, which represents the degree of model compression; the Y-axis is the accuracy, which represents the performance of model recognition; the two diagonal lines represent the auxiliary line with a slope of -0.5, when the slope of the curve is less than -0.5, it is considered that the accuracy changes drastically, and when it is greater than -0.5, it is considered that the change is slow. The slope of the auxiliary line is set artificially for ease of analysis and does not affect the final laws.

interval of the PKD framework

$$I_{\text{intra}} = H(L_{nl}) - H(L_{ns}) \tag{1}$$

$$I_{\text{inter}} = H(L_{(n-1)s}) - H(L_{nl}). \tag{2}$$

The number of pyramid layers N and the network layers interval I are both artificially adjustable parameters of the PKD framework, they are the key parameters used to control the difference between teacher models and student models, and also the parameters to adjust the model CR. The larger the network layer number N, the smaller the network layer interval I. We assume that the Spire student model is a two-layer simple neural network $S(L_{1s}) = 2$, then the number of neural network layers of the teacher model in the Nth layer is

$$H(L_{Nl}) = H(L_{1s}) + 2 \times N \times I \tag{3}$$

where 2 means that each layer of the pyramid has two models in DML. The intermediate layers of the pyramid are trained by an HOSD method to obtain a middle-layer teacher model. The Pedestal models use online DML to train the Pedestal teacher models. The Spire models are guided by the offline knowledge weighted by all the teacher models and the offline guidance of the knowledge weighted by a group of teacher models closest to the Spire model for online DML, and finally, a compact Spire student model with high CR is obtained.

A series of models with different CRs are obtained by automatically adjusting the number of pyramid layers N and the network layers interval I by using an improved particle swarm optimization algorithm, and the accuracy-CR curve is drawn by means of the Boltzmann fitting algorithm; the schematic is shown in Fig. 3. The motivation behind the design of Fig. 3 is to obtain mathematical expressions that describe the relationship between different CRs and accuracy by fitting curves to the data. This allows for an exploration of the laws and limits of model compression. We can obtain the limits and laws of model compression based on KD from Fig. 3. The slope of the auxiliary line is set artificially for ease of analysis and does not affect the final laws.

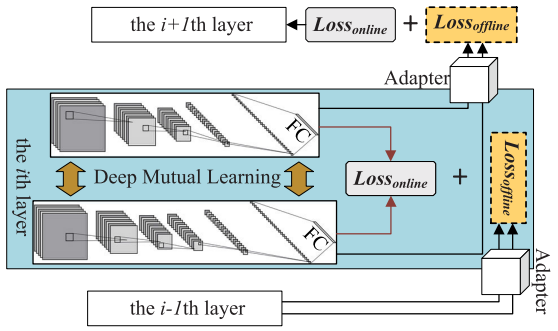


Fig. 4. HOSD schematic.

C. HOSD

The intermediate DMLs contain $(N-2)$ layers of PKD, each of which adopts an HOSD method to reduce the difference between the teacher model and the student model and avoid the occurrence of knowledge explosion.

The schematic of HOSD is shown in Fig. 4. Assuming that this is the ith layer of PKD, the teacher model used in the offline distillation part is the two models trained on the $i-1$th layer of PKD. The two teacher models of offline distillation are adaptively weighted to guide the i layer online DML.

In the online stage, we define a batch of C types of remote sensing images as $X = \{x_1, x_2, \ldots, x_n\}$, and the corresponding labels are $Y = \{y_1, y_2, \ldots, y_n\}$. The M intermediate features of the teacher model and the student model can be expressed as $\{f_i{}^T\}_{i=1}^M$ and $\{f_i{}^S\}_{i=1}^M$, respectively. The loss of the ith block pair between teacher and student can be expressed as follows:

$$\mathcal{L}_{\text{KL}}^m\left(X, f_i^T, f_i^S\right) = \text{KL}\left(f_i^T, f_i^S\right), \quad m \in M. \tag{4}$$

Therefore, the loss of intermediate features is as follows:

$$\mathcal{L}^{\text{Inter}} = \sum_{i=1}^{M} \mathcal{L}_{\text{KL}}^{\text{block}}\left(X, f_i^T, f_i^S\right). \tag{5}$$

HOSD matches the teacher model and student model in the high layer using the $L2$ loss function between its logits a_T and the student's logits a_S, which benefits both networks from collaborative learning. The logits $\mathcal{L}^{\text{Logits}}$ loss enables the teacher and student networks to learn high-level knowledge, which is computed as follows:

$$\mathcal{L}^{\text{Logits}} = \|a_T - a_S\|_2^2. \tag{6}$$

The teacher model and student model are trained directly to perform classification tasks on the remote sensing image datasets with the cross-entropy (CE) loss, which is utilized to adjust the neural network's weights via drawing the model predictions toward to ground-truth label. The ith output of these two networks and CE loss $\mathcal{L}_{\text{CE}}$ from probabilistic outputs

to one-hot labels y are given as follows:

$$o_T = \text{softmax}(a_T) = \frac{\exp(a_i^T)}{\sum_j \exp(a_j^T)} \tag{7}$$

$$o_S = \text{softmax}(a_S) = \frac{\exp(a_i^S)}{\sum_j \exp(a_j^S)} \tag{8}$$

$$\mathcal{L}^{\text{CE}} = \mathcal{H}(o_T, y) + \mathcal{H}(o_S, y) \tag{9}$$

where $\mathcal{H}$ represents the CE function. The loss of online distillation is

$$\mathcal{L}^{\text{online}} = \alpha \mathcal{L}^{\text{Inter}} + \beta \mathcal{L}^{\text{Logits}} + \mathcal{L}^{\text{CE}}. \tag{10}$$

The loss in the offline stage includes the CE loss and KD loss of the classification. Since the CE loss is already included in the online stage, only the KD loss needs to be calculated in the offline stage, so the loss in the offline stage is

$$\mathcal{L}_{\text{offline}} = \mathcal{L}^{\text{KD}} = \gamma \tau^2 \mathcal{H}(\tau(O_{T_{i-1}}), \tau(O_S)) \tag{11}$$

where γ is the balanced weight coefficient of $\mathcal{L}_{\text{offline}}$, τ is the temperature parameter to control the importance of soft label, $\tau(O_S)$ is the KD loss to match the student's softened logits, and $\tau(O_{T_{i-1}})$ is the teachers' weighted softened logits of the $i-1$th layer of PKD via CE loss. The weighted softened logits are obtained using AWMD for the two teacher models at $i-1$th layer.

D. AWMD

Multiteacher KD increases the diversity of knowledge but brings about the problem of knowledge offset, which makes it difficult for the student model to effectively absorb the knowledge itself. We innovatively propose AWMD to solve this problem. Specifically, we introduce a latent representation (LR) to describe the knowledge of multiple teachers, as shown on the right side of Fig. 2 with a light green background.

LR is similar in function to the latent factor model commonly used in recommender systems, where each user or item corresponds to a latent factor used to summarize its implied characteristics. We assume that the factor corresponding to the tth teacher is $\boldsymbol{\theta}_t \in \mathbb{R}^d$, where d is the dimension of the factor, and $t \in \{1, 2, \ldots, n\}$ with n teachers. The output of any layer of the student model can be used as a representation of instances, and the high-level features output by the last convolutional layer are the best choice for the representation of instances. In this work, we assume that the tensor of the ith remote sensing image is expressed as $\mathbf{B}_i \in \mathbb{R}^{CHW}$ where C, H, and W correspond to the number of channel, height, and width of student's feature map, respectively. To facilitate subsequent calculations, a max-pooling operation with a kernel size of $s = H \times W$ is used to make the tensor representation of the remote sensing image have the same dimension as the teacher factor

$$\boldsymbol{\delta}_i = \text{Max-Pooling}(\mathbf{B}_i, s) \tag{12}$$

where $\delta_i \in \mathbb{R}^C$ and d is set to be equal to C for simplicity.

The importance weight of the tth teacher model for the ith image is calculated as follows:

$$\gamma_{t,i} = \boldsymbol{v}^{\text{T}}(\boldsymbol{\theta}_t \odot \boldsymbol{\delta}_i) \tag{13}$$

where $\boldsymbol{v}$ is a global parameter vector to be learned and $\odot$ denotes element-wise product. Larger $\gamma_{t,i}$ denotes the teacher is more important with regard to the image. From (13), we can observe the interaction between the representations of the teacher model, and the image is captured by the elementwise product operation. It computes their similarities in each dimension. $\boldsymbol{v}$ determines whether or not the value in each dimension has a positive effect on the score. We further normalize the importance weight through the softmax function defined as follows:

$$w_{t,i} = \text{softmax}(\gamma_{t,i}) = \frac{\exp(\gamma_{t,i})}{\sum_{t'=1}^{m} \exp(\gamma_{t',i})}. \tag{14}$$

We use weighted addition operation to acquire the integrated soft-target $\tilde{\mathbf{y}}_i^T$, the formula is as follows:

$$\tilde{\mathbf{y}}_i^T = \sum_{t=1}^{m} w_{t,i} \times \tilde{\mathbf{y}}_{t,i}^T \tag{15}$$

where $\tilde{\mathbf{y}}_{t,i}^T$ is the soft-target generated by the tth teacher for the ith image and m represents the number of teacher models.

E. Total Loss Training Algorithm

In this section, we describe the PKD total loss training algorithm in detail to clearly demonstrate the complete process of model distillation based on the PKD framework. The original intention of PKD design is to explore the limit of model compression based on the KD method, so the total loss is the loss of PKD Spire model training, that is,

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{online}} + \mathcal{L}_{\text{offline}} + \mathcal{L}_{\text{supervised}} \tag{16}$$

where $\mathcal{L}_{\text{online}}$ comes from (10), $\mathcal{L}_{\text{offline}}$ comes from (11), $\mathcal{L}_{\text{supervised}}$ represents the supervised loss formed by adaptively weighting the knowledge of all teacher models, and its calculation formula is as follows: given a triplet of examples denoted by (i, j, k), an angle-based metric is used to measure the structural relation between these examples, which is given as follows:

$$\Delta(x_i, x_j, x_k) = \cos \angle x_i x_j x_k = \langle \mathbf{e}^{ij}, \mathbf{e}^{kj} \rangle \tag{17}$$

where $\mathbf{e}^{ij}$ and $\mathbf{e}^{kj}$ are the normalized vector differences, i.e., $\mathbf{e}^{ij} = (x_i - x_j / \|x_i - x_j\|_2)$ and $\mathbf{e}^{kj} = (x_k - x_j / \|x_k - x_j\|_2)$

$$\mathcal{L}_{\text{supervised}} = \sum_{(i,j,k)} l_\delta(\Delta(\tilde{\mathbf{y}}_i^T, \tilde{\mathbf{y}}_j^T, \tilde{\mathbf{y}}_k^T), \quad \Delta(\tilde{\mathbf{y}}_i^S, \tilde{\mathbf{y}}_j^S, \tilde{\mathbf{y}}_k^S)) \tag{18}$$

where l_δ is the Huber loss which provides robust regression, compared with standard mean square loss. Note that $\tilde{\mathbf{y}}_i^T$, $\tilde{\mathbf{y}}_j^T$, and $\tilde{\mathbf{y}}_k^T$ are all integrated soft targets computed by (15) for different image examples, $\tilde{\mathbf{y}}_i^S$ represents the output of the student model with i remote sensing image as input.

We describe PKD in detail in Algorithm 1. First, we perform online DML training on the Pedestal models; second, we perform online DML on the Intermediate models and receive offline KD of the model on the upper layer simultaneously; finally, we perform DML on the Spire models and receive offline KD of its upper layer models and supervision of adaptive weighted knowledge of all teacher models.

Algorithm 1 Training the PKD Algorithm

Input: Training set x, label set y; parameters: α, β and τ; **Output:** Training loss $\mathcal{L}_{total}$

PKD = [Spire DML, Intermediate DMLs, Pedestal DML]
for models in PKD **do**
  Select models from PKD
  if models==Pedestal DML **then**
    repeat Train Pedestal DML
    input images set x_i and its label y randomly
    compute the Pedestal DML loss with Eq. 10
    until maximum iterations, save the models P.
  else if models==Intermediate DMLs **then**
    Training Intermediate DMLs
    for model in Intermediate DMLs **do**
      repeat train the teacher T_i to get a_{Ti}
      calculate the online loss with Eq. 10
      if i==1, $i \in [1, 2, \ldots, n-1]$ **then**
        obtain the soft label p_l of the models P
        calculate the offline loss with p_l and Eq. 11
      else
        obtain the soft label l_{i-1} of the models $i-1$
        calculate the offline loss with Eq. 11
      update weights in the teacher i
  else
    repeat train the Spire DML
    train the teacher T to get a_T
    obtain the soft label l_{n-1}, l_n except Spire DML
    calculate the online loss, obtain $\mathcal{L}_{online}$;
    calculate the offline loss, obtain $\mathcal{L}_{offline}$;
    calculate the supervised loss, obtain $\mathcal{L}_{supervised}$
    obtain $\mathcal{L}_{total}$ with Eq. 16
    update the parameters
    until both models T_s and S_s are trained
return obtain models T_s and S_s

F. Exploration of Compression Limits and Laws of KD Based on Improved Particle Swarm Optimization Algorithm

To explore the exploration of the limits and laws of model compression based on KD, we introduce a particle swarm algorithm to automatically find the best the layer N of PKD and the model interval I. Specifically, to control the variable at a lesser level, we set CR to a constant value. CR $\in (0, 1)$, to plot the model accuracy-CR curve, CR will be set to multiple discrete values from 0 to 1. For ease of understanding, we use CR $= 0.25$ as an example for the subsequent description. We set the fitting function as the operation of the testing model, so that the higher the accuracy of the model, the better the corresponding combination of N and I can be

$$\text{fit}_{N_i}(x) = f_n^i(x) \quad \text{CR} = 0.25. \tag{19}$$

We set the position to the layer N of PKD, and the velocity to the model interval I. The specific algorithm is shown in Algorithm 2.

The optimal N I configuration of PKD is obtained through Algorithm 2, and the accuracy CR fitting curve is drawn according to the experimental results data, and we can find the approximate variation law of the model compression according to the trend of the fitting curve. And obtain the fitting function expression y, calculate the first derivative of the fitting function y to obtain y', according to y' can judge the change law of the fitting curve y, we think that when the absolute value of the fitting function y slope is greater than 0.5, it will change drastically, so if $y' = -0.5$, the best CR x_1 can be obtained by solving the equation; that is, if the CR is less than x_1, the fitting function y changes slowly; if the CR is greater than x_1, the fitting function y changes drastically. In this way, we get the limit of model compression from the perspective of mathematical analysis.

Algorithm 2 Improved Particle Swarm Optimization

Input: Number of particles M.
Output: Best position $gBest$, Best velocity $Ibest$, Acc

for each particle i **do**
  Initialize velocity I_i and position N_i for particle i
  Training $PKD_{N_i}^{I_i}$ with Algorithm 1
  Evaluate particle i with Eq. 19
  Save $Acc[n, i]$ and set $pBesti = N_i$
$gBest = minpBesti$
while not stop **do**
  for $i = 1$ to M **do**
    Update the velocity and position of particle i
    Training $PKD_{N_i}^{I_i}$ with Algorithm 1
    Save Acc$[n, i]$
    if fit$(N_i)>$ fit$(pBesti)$ **then**
      $pBesti = N_i$
    if fit$(pBesti) >$ fit$(gBest)$ **then**
      $gBest = pBesti, Ibest = I_i$
  Print $gBest$, $Ibest$, $Acc[gBest, Ibest]$

G. Satellite-on-Orbit Object Recognition Based on PKD

The process of satellite in-orbit target recognition based on PKD is mainly divided into four steps, as shown in Fig. 5.

1) *S1:* Choose the appropriate model M according to the accuracy-CR curve, which not only meets the accuracy requirements of on-orbit object recognition but also meets the extremely limited resources on the satellite. We consider that the optimal CR and accuracy model is obtained when the accuracy is greater than 75% and the variation of the accuracy with the CR is not drastic, i.e., the absolute value of the slope of the accuracy-CR curve is close to or even equal to 0.5.
2) *S2:* Train PKD on the ground server using historical remote sensing imagery datasets and save the trained model M.
3) *S3:* Upload model M to the satellite for on-orbit verification and on-orbit inferencing.
4) *S4:* Regularly update the fine-tuned model M using newly acquired remote sensing imagery datasets.

III. Experimental Settings

In this section, the proposed PKD trained on the ground server inferences on the satellite imitated by NVIDIA Jetson TX2 (TX2), the results of the experiment, including the ablation study and some comparison experiments of state-of-the-art KD methods, will be presented and analyzed.

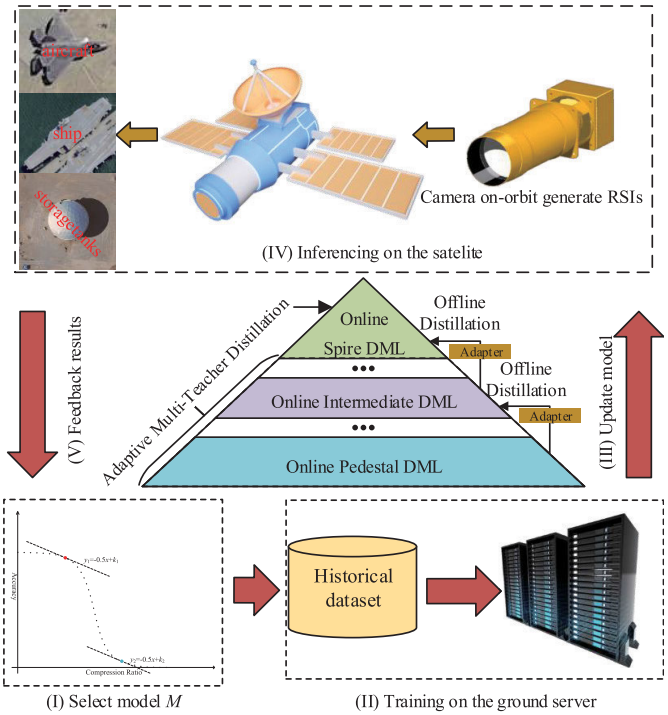


Fig. 5. Flowchart of satellite-on-orbit object recognition based on PKD.

A. Data Description

To ensure the fairness of experimental comparisons, we performed the same data preprocessing operations on all experiments during the model training phase. The specific operations include: first, randomly cropping and resizing the images to 224 × 224; second, randomly flipping the images horizontally; and finally, normalizing the image data based on the mean and standard deviation. We divided the datasets into training, validation, and testing sets in a ratio of 3:1:1.

1) Aircraft: The Aircraft dataset [23] contains 30 types of aircraft on the WorldView-3 satellite images with a resolution of 0.31 m, which is marked by us. The Aircraft dataset has a total of 4378 samples, the size of each sample is 224 × 224.

2) FGSC-23: Fine-Grained Ship Collection-23 (FGSC-23) [24] is a high-resolution optical remote sensing image dataset for fine-grained ship object recognition. It consists of 23 ship categories with a total of 4052 ship instances. The data are sourced from high-resolution Google Earth and GF-2 satellite images.

B. Base Model Network

We choose ResNet series networks as our base model networks for our PKD framework, as it is better suited for models of the same type but with different depths, while also taking into account computational cost and model performance. To highlight the performance advantages of PKD and make it as smooth as possible, we set up PKD ($I = 2$ and $N = 26$), a total of 52 models with layer intervals I of 2 and pyramid layers N of 26, ranging from ResNet8 to ResNet110. There are also other different combinations of the number of pyramid layers N and the network layers interval I, such as PKD ($I = 3$ and $N = 9$), PKD ($I = 6$ and $N = 5$), and PKD ($I = 8$ and $N = 3$).

To further emphasize the generalization capability of PKD on conventional deep networks, we also choose VVG series networks as base model networks. The last three fully connected layers of the VVG series networks contribute significantly to the overall parameter count of the VGG network. Consequently, the introduction of additional convolutional layers within the VGG network leads to a considerably low parameter growth rate. As a result, the parameter count of the Spire student model becomes over 90% of the Pedestal teacher model's parameter count, limiting the model CR to less than 10%. Furthermore, the computational cost associated with VGG series networks is substantial. Considering the aforementioned aspects, we reconfigure the last three fully connected layers of the VGG network into a single fully connected layer, with an input filter count set at 128. This adjustment not only mitigates the computational expenses linked with the VGG series networks but also magnifies the impact of introducing convolutional layers to the overall parameter count of the VGG network. In effect, this expansion of the exploration scope of the model CR is achieved. We set up PKD ($I = 1$ and $N = 52$), a total of 52 models with layer intervals I of 1 and pyramid layers N of 52, ranging from VGG6 to VGG57. There are also other different combinations of the number of pyramid layers N and the network layers interval I.

C. Performance Metric

In this work, we are concerned with the compression results of the model after KD, including the CR, the compressed model accuracy and model size, and its inference latency and throughput on satellite boards. The specific calculation formula is as follows:

$$\mathrm{CR} = \frac{p_o - p_c}{p_o} \times 100\% \tag{20}$$

where p_o is the original parameter of model and p_c is the parameter of compressed model

$$\mathrm{Accuracy} = \frac{\sum_{i=1}^{x} t_{pi} + t_{ni}}{\sum_{i=1}^{x} t_{pi} + t_{ni} + f_{pi} + f_{ni}} \tag{21}$$

where t_p, t_n, f_p, and f_n refer to true positives, true negatives, false positives, and false negatives in class i, respectively. Herein, x is the number of classes to be classified.

D. Training Details

All experimental model training is carried out on the ground server. The server configuration is as follows: 8 × 10-core Intel Xeon[1] Gold 6148 at 2.40-GHz CPU, 8 × NVIDIA GeForce RTX 3090 GPUs, and an operating system of 64-bit CentOS Linux release 7.9.2009, the programming language is Python-3.7.13, and the deep learning framework is Pytorch-1.12.1.

The development environment configuration of TX2 is as follows: the operating system is 64-bit Ubuntu 18.04.5 LTS,

[1]Registered trademark.

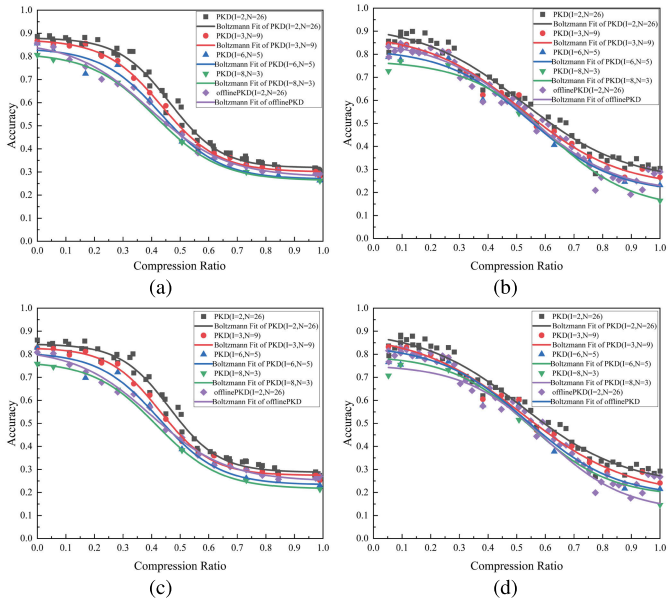


Fig. 6. Experimental results of offlinePKD and PKD for different combinations of the number of pyramid layers N and the network layers interval I. (a) ResNets on Aircraft. (b) VGGs on Aircraft. (c) ResNets on FGSC-23. (d) VGGs on FGSC-23.

the Jetpack version is Jetpack 5.1, the programming language is Python-3.6.9, and the deep learning framework is Pytorch-1.7.0.

E. Ablation Study

To highlight the contribution of the PKD proposed in this article in addressing knowledge explosion, knowledge offset, and the advantages of such design, we set up related ablation experiments.

1) *Study of the HOSD:* Discuss the advantages of using HOSD compared with offlinePKD (a method that uses only offline distillation based on the PKD framework) and the influence of PKD with different combinations of the number of pyramid layers N and the network layers interval I on the experimental results.
2) *Study of the AWMD:* Discuss the advantages of using AWMD versus not using AWMD.

1) Study of the HOSD: Knowledge explosion is specified in the PKD framework as follows; for the same CR, there are higher and lower model accuracies, and lower model accuracies are specified when knowledge explosion occurs. The larger the number of pyramid layers N and the smaller the network layers interval I, means that there is less difference in model size and model depth between the teacher models involved in distillation, and therefore, it is difficult for knowledge explosion to occur in this case. As shown in Fig. 6, the larger the number of pyramid layers $N$and the smaller the network layers interval I the better the distillation performance of PKD, such as PKD ($I = 2$ and $N = 26$), which has a fitting curve on the upper side of all curves.

Fig. 7 presents a structural comparison between offlinePKD and PKD, from which it can be observed that offlinePKD lacks the in-layer online distillation compared to PKD, while

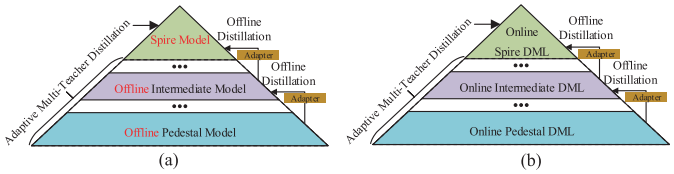


Fig. 7. Structural comparison between offlinePKD and PKD. (a) Schematic of offlinePKD. (b) Schematic of PKD.

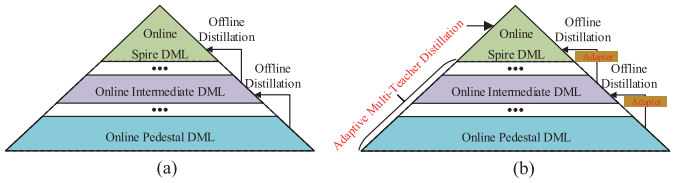


Fig. 8. Structural comparison between PKD* and PKD. (a) Schematic of PKD*. (b) Schematic of PKD.

other configurations remain the same. On both the Aircraft and FGSC-23 datasets, we observe this phenomenon in both experiments based on ResNets and VGGs. From the experimental results of the accuracy-CR curves shown in Fig. 6, the accuracy-CR curve of offlinePKD ($I = 2$ and $N = 13$) are below those of PKD ($I = 2$ and $N = 13$). Therefore, under the same CR, PKD has higher precision compared to offlinePKD, indicating that the combination of online and offline distillation alleviates the problem of knowledge explosion, where compact student models cannot fully absorb the knowledge from large teacher models.

On both the Aircraft and FGSC-23 datasets, we observe this phenomenon in both experiments based on ResNets and VGGs. Based on the experimental results shown in Fig. 6 of the accuracy-CR curves, it can be observed that as the network layers interval I decreases and the number of pyramid layers N increases, the corresponding accuracy-CR curve shifts upward. This indicates that for a given CR, the precision is higher, and for a given precision, the CR is greater when the network layers interval I is smaller, the number of pyramid layers N is larger, and the structure of PKD is smoother. The experimental results indicate that HOSD is capable of effectively addressing the problem of knowledge explosion.

2) Study of the AWMD: Fig. 8 presents a structural comparison between PKD* without AWMD and PKD with AWMD. As shown in Fig. 8, PKD* lacks AWMD compared with PKD, while other configurations are the same as PKD. On both the Aircraft and FGSC-23 datasets, we observe the knowledge offset phenomenon in both experiments based on ResNets and VGGs. Based on the experimental results shown in Fig. 9, most of the accuracy CR curves of PKD are above those of PKD*, and a small part is close to those of PKD* under different combinations of the number of pyramid layers N and the network layers interval I, indicating that PKD achieves higher accuracy CR than PKD* at the same accuracy, and PKD achieves a greater CR at the same accuracy. The experimental results demonstrate that AWMD is effective in addressing the issue of knowledge offset caused by knowledge transfer among multiple teachers when the compact student model fails to learn effective knowledge.

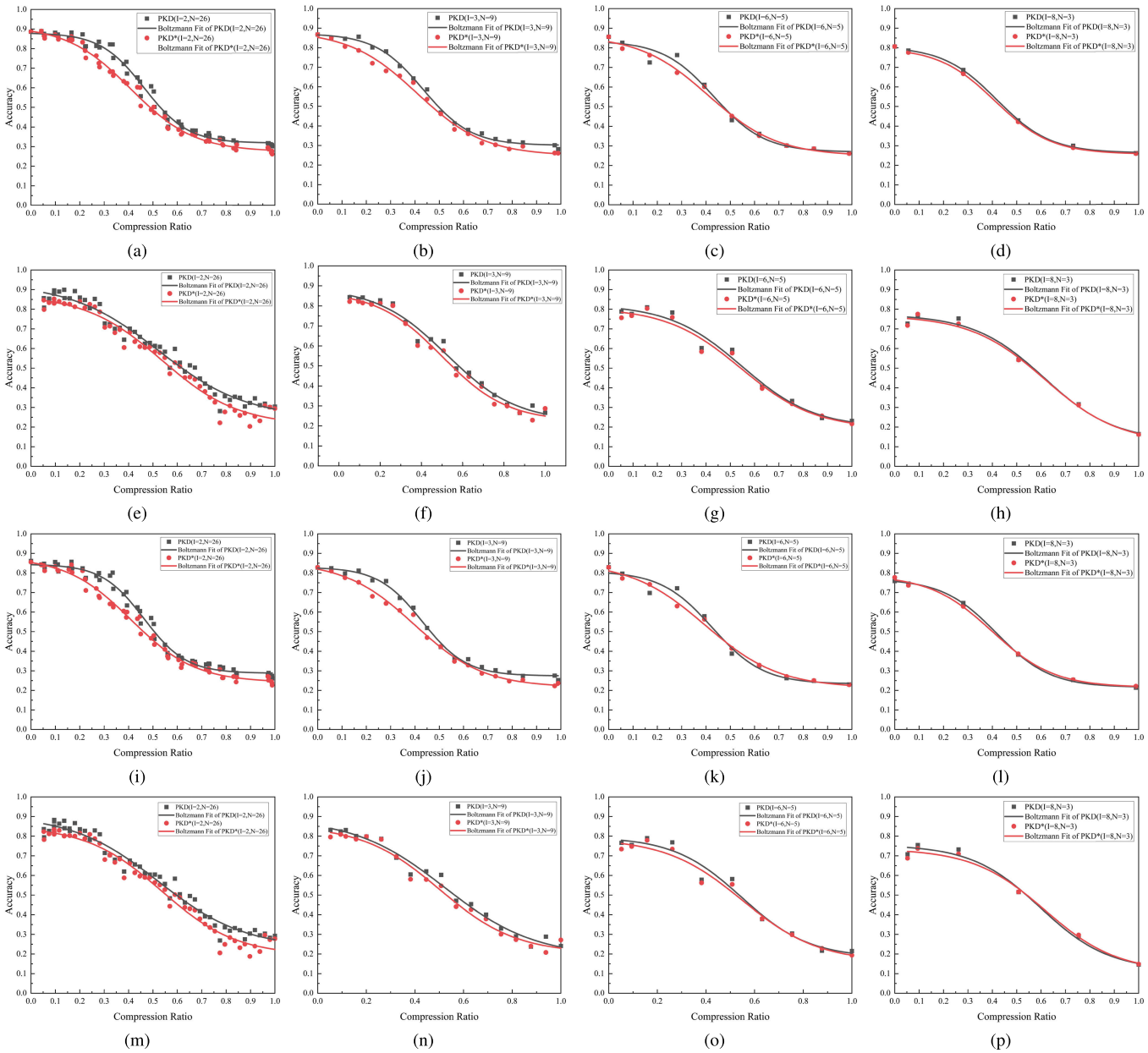


Fig. 9. Experimental results of PKD with AWMD and PKD* without AWMD under different combinations of the number of pyramid layers N and the network layers interval I. (a)–(d) ResNets on Aircraft. (e)–(h) VGGs on Aircraft. (i)–(l) ResNets on FGSC-23. (m)–(p) VGGs on FGSC-23.

F. Comparison With State-of-the-Art KD Methods

To further prove the advantages of the PKD method in addressing knowledge explosion, knowledge offset and exploring the limits and laws of compression based on KD model, we use the state-of-the-art KD methods (collaborative consistent knowledge distillation (CKD) [25], dual KD (DKD) [12], and pairwise similarity KD (PSKD) [26]), to carry out comparative experiments. To enhance the persuasiveness of comparative experiments, we have configured CRs based on the model compression limits described in Section III-F2. Specifically, we have set CRs at the optimal level, below the optimal level, and above the optimal level. According to (20), it is evident that CR is a discrete variable. Therefore, in practical experiments, the optimal CR employed is not determined by solving the functional equation as outlined in Section III-F2 for the theoretical optimal CR. Instead, it is selected as the discrete CR that approximates the theoretical optimal CR most closely.

1) Analysis of Comparative Experimental Results: Based on the comparative experimental results shown in Fig. 10, the green bar represents the object recognition accuracy without using KD. The orange bar represents the object recognition accuracy after using KD. Fig. 10(a) presents the experimental results of ResNets on the Aircraft dataset with the optimal CR of 0.21. The CKD method results in a 1.13% growth in ResNets' performance, the DKD method leads to a 3.37% growth, the PSKD method yields a 0.08% growth, and the PKD method results in a remarkable 15.11% growth in ResNets' performance, making it the most effective distillation method among those considered for comparison.

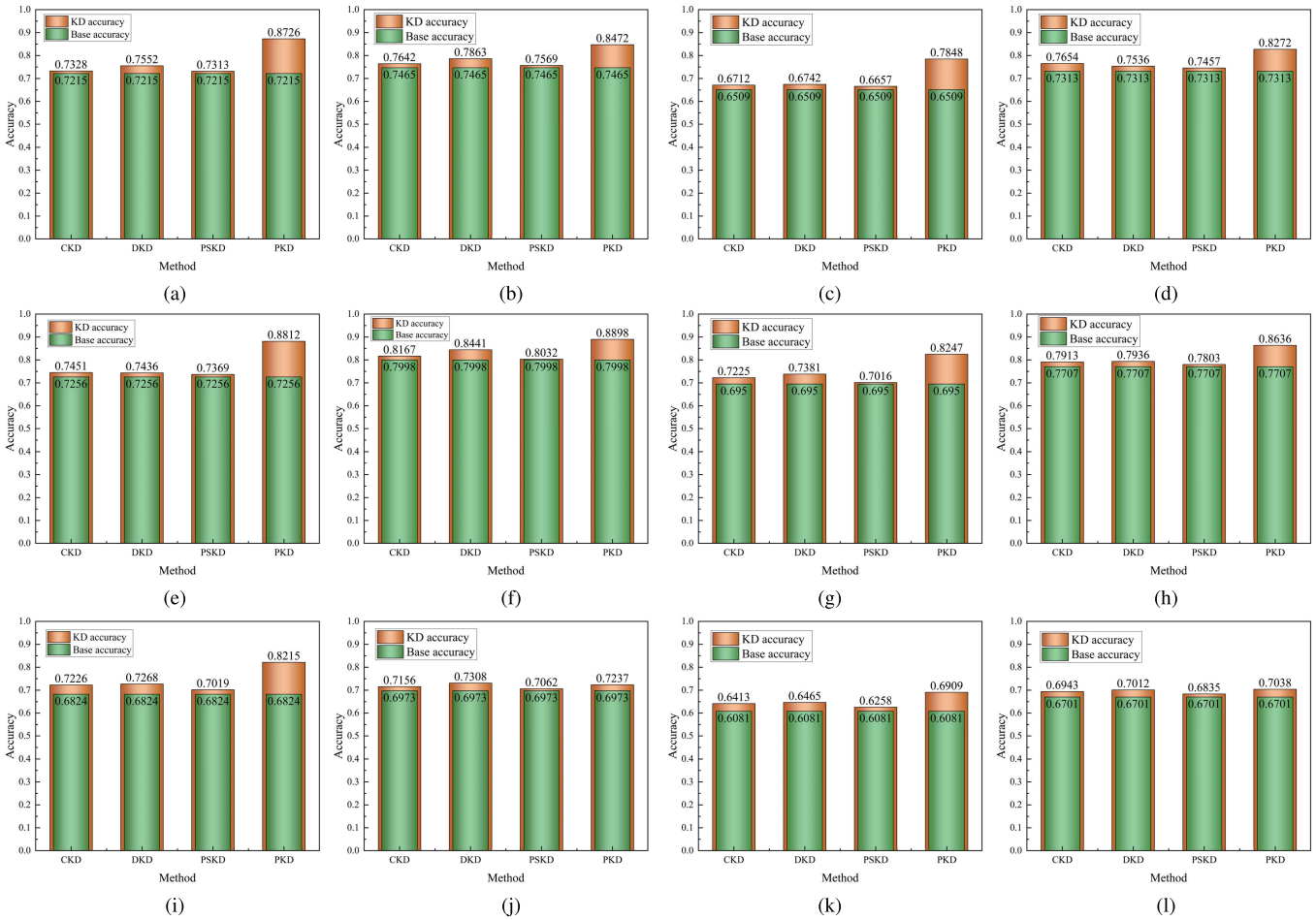


Fig. 10. Experimental results compared with state-of-the-art KD methods. The compression rates for (a)–(d) were set at the optimal level. The compression rates for (e)–(h) were set at below the optimal level. The compression rates for (i)–(l) were set at above the optimal level. (a) ResNets on Aircraft, CR = 0.21. (b) VGGs on Aircraft, CR = 0.22. (c) ResNets on FGSC-23, CR = 0.26. (d) VGGs on FGSC-23, CR = 0.22. (e) ResNets on Aircraft, CR = 0.10. (f) VGGs on Aircraft, CR = 0.11. (g) ResNets on FGSC-23, CR = 0.15. (h) VGGs on FGSC-23, CR=0.11. (i) ResNets on Aircraft, CR = 0.33. (j) VGGs on Aircraft, CR = 0.32. (k) ResNets on FGSC-23, CR = 0.38. (l) VGGs on FGSC-23, CR = 0.32.

Fig. 10(e) presents the experimental results of ResNets on the Aircraft dataset with a CR of 0.1 below the optimal CR. The CKD method results in a 1.95% growth in ResNets' performance, the DKD method leads to a 1.8% growth, the PSKD method yields a 1.13% growth, and the PKD method results in a remarkable 15.56% growth in ResNets' performance. The reduction in CRs has not yielded significant accuracy gains for CKD, DKD, PSKD, and PKD; in fact, DKD's accuracy has shown a slight decrease. PKD's performance continues to outshine that of the comparative methods in the distillation effects. Although PKD's accuracy has improved by 0.45% in comparison to that of PKD with the best CR, it has reduced CRs by 11%. This trade-off is not justifiable for satellites operating under severe computational resource constraints. Fig. 10(i) presents the experimental results of ResNets on the Aircraft dataset with a CR of 0.33 above the optimal CR. The CKD method results in a 4.02% growth in ResNets' performance, the DKD method leads to a 4.44% growth, the PSKD method yields a 1.95% growth, and the PKD method results in a remarkable 13.91% growth in ResNets' performance, making it the most effective distillation method among those considered for comparison. As the CR increases, the accuracy gains for CKD, DKD, and PSKD do not exhibit a significant improvement, while the accuracy gain for PKD decreases by 1.2%. Through the analysis of Fig. 10(a), (e), and (i), it becomes evident that the variation in CR has a minor impact on the accuracy gains of CKD, DKD, and PSKD, but exerts a more pronounced effect on the accuracy gain of PKD. This observation is further validated in the last three columns of the comparative results as shown in Fig. 10. In summary, while CKD, DKD, and PSKD distillation methods all improve object recognition accuracy, their improvements are lower than PKD distillation. It is evident that the PKD KD framework has an advantage over existing KD frameworks.

2) Exploring the Limits and Laws of Compression Based on KD Model: Through the comparative analysis of the results of the ablation experiments shown in Fig. 6, we found that the accuracy CR curve of PKD ($I = 2$ and $N = 26$) is the highest, indicating that compared with other models at the same CR, its accuracy is the highest; and compared with other models at the same accuracy level, its CR is the highest. Therefore, PKD ($I = 2$ and $N = 26$) is the optimal distillation method. This also reveals a pattern: the smaller the network layers interval I, the larger the number of pyramid layers N, and the smoother the structure of PKD, the better the KD effect.

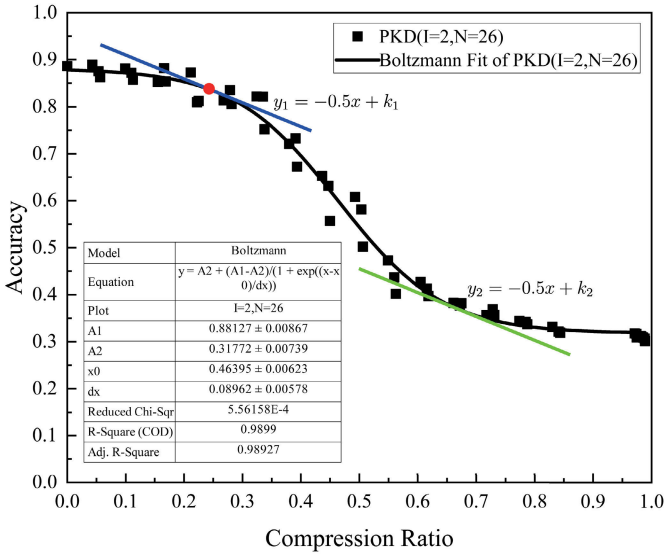
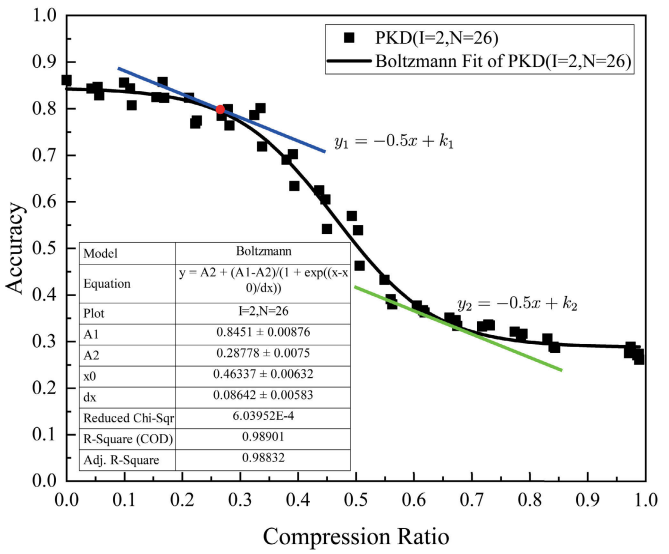


Fig. 11. [ResNets on Aircraft] Accuracy-CR curve of PKD ($I = 2$ and $N = 26$).



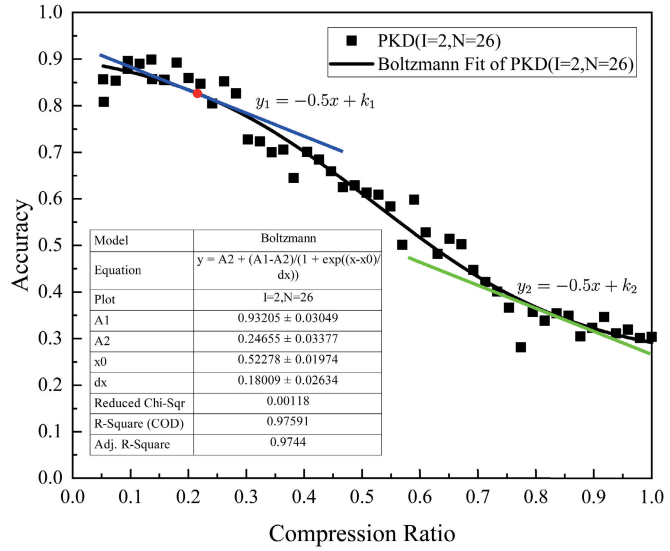


Fig. 12. [VGGs on Aircraft] Accuracy-CR curve of PKD ($I = 2$ and $N = 26$).

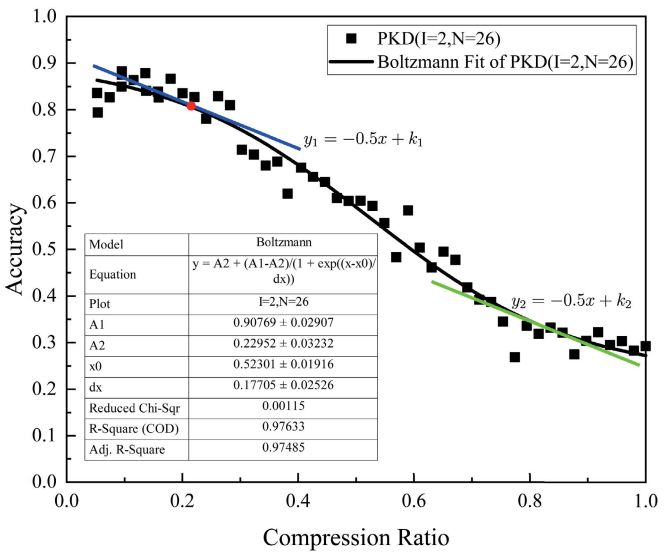


Fig. 14. [VGGs on FGSC-23] Accuracy-CR curve of PKD ($I = 2$ and $N = 26$).

We have utilized the Boltzmann algorithm to fit the experimental data of PKD ($I = 2$ and $N = 26$) and obtained the fitting results shown in Figs. 11–14. From the accuracy CR curve of PKD in Fig. 11, it can be observed that the accuracy of object recognition decreases slowly when the CR is low; as the CR continues to increase, the accuracy of object recognition suddenly drops; however, as the CR continues to increase, the decrease in accuracy of object recognition becomes slow again. To describe this pattern more scientifically, we conducted a mathematical analysis. The fit function has been determined as follows:

$$y = A_2 + \frac{A_1 - A_2}{1 + e^{(x - x_0)/dx}}. \tag{22}$$

Substituting the fitting parameters A_1, A_2, x_0, and dx into (22) yields

$$y = 0.31772 + \frac{0.56355}{1 + e^{(x - 0.46395)/0.08962}}. \tag{23}$$

Taking the first derivative of (23) yields

$$y' = -\frac{56335\, e^{\frac{50000\,x}{4481} - \frac{46395}{8962}}}{8962\left(e^{\frac{50000\,x}{4481} - \frac{46395}{8962}} + 1\right)^2}. \tag{24}$$

It is evident that $y' < 0$, hence the function y is monotonically decreasing. This indicates that as the CR increases in KD based on PKD, its object recognition accuracy will gradually decrease. According to Fig. 11, it can be observed that the curve of function y descends most rapidly when the slope is less than -0.5. Therefore, we set $y' = -0.5$, i.e.,

$$-\frac{56335\, e^{\frac{50000\,x}{4481} - \frac{46395}{8962}}}{8962\left(e^{\frac{50000\,x}{4481} - \frac{46395}{8962}} + 1\right)^2} = -0.5 \tag{25}$$

and solving (25) to obtain $x_1 = 0.2534$ and $x_2 = 0.6745$.

It can be seen from Fig. 11 that when the CR is less than x_1, the change in accuracy increases slowly as the CR increases. When the CR is greater than x_1, the change in accuracy increases dramatically as the CR increases. When the CR is greater than x_1, but less than x_2, the change in accuracy increases dramatically as the CR increases. When the CR is greater than x_2, the change in accuracy increases slowly as the CR increases. This pattern is consistent with the curve in Fig. 11.

Similarly, we have determined the optimal CRs through computation to be 0.2181 for Fig. 12, 0.2577 for Fig. 13, and 0.2219 for Fig. 14.

By analyzing the experimental results above, we have obtained the laws and limits of model compression based on KD.

1) The model compression method based on KD will gradually degrade network performance as the CR increases. According to (24), it can be deduced that $y' < 0$, hence the function y is monotonically decreasing.
2) The limit of model compression is the point where the accuracy CR curve begins to change drastically. In this article, this refers to the points where the slope of the accuracy CR curve is -0.5, and the CR is less than 0.5.

IV. Conclusion

This article proposes a new framework for KD, called PKD, to address the challenges of extremely constrained storage and computational resources in on-orbit satellite computing. This article defines two new concepts, knowledge explosion, and knowledge offset, to reflect the inability of the compact student model to fully absorb the large-capacity teacher model during the distillation process and the failure of the compact student model to learn effective knowledge due to knowledge shift caused by knowledge transfer between multiple teachers. The PKD framework uses a pyramid structure to stack multiple sets of DML models, with the smaller models on the top of the larger ones, and employs an HOSD and an adaptive multiteacher distillation method to avoid knowledge explosion and knowledge offset. This article introduces an evolutionary algorithm to automatically find the best model CR and tries to analyze the general laws of the compression limit. The experimental results on the Aircraft and FGSC-23 datasets show that PKD outperforms state-of-the-art distillation methods and reveal that the model compression method based on KD will gradually degrade network performance as the CR of the network increases; the limit of model compression is the point where the accuracy CR curve begins to change drastically. Overall, the proposed PKD framework is effective and advanced in model compression for on-orbit satellite computing.

Obviously, there are limitations to PKD as well: 1) there is still exploration needed for the end-to-end KD-based model compression limit that has not been implemented yet and 2) compared with other existing distillation methods, the ground training computational cost is high. In future work, we will further investigate these limitations, aiming to achieve exploration of model limits for any model in an end-to-end manner while reducing the training computational cost enabling it to be deployed on edge devices.

References

[1] G. Giuffrida et al., "The F-Sat-1 mission: The first on-board deep neural network demonstrator for satellite Earth observation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5517414.

[2] M. Ghiglione and V. Serra, "Opportunities and challenges of AI on satellite processing units," in *Proc. 19th ACM Int. Conf. Comput. Frontiers*. New York, NY, USA: Association for Computing Machinery, May 2022, p. 221.

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

[4] Y. Hu, X. Huang, X. Luo, J. Han, X. Cao, and J. Zhang, "Variational self-distillation for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5627313.

[5] Q. Zhao, Y. Ma, S. Lyu, and L. Chen, "Embedded self-distillation in compact multibranch ensemble network for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4506415.

[6] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *Int. J. Comput. Vis.*, vol. 129, no. 6, pp. 1789–1819, Jun. 2021.

[7] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "FitNets: Hints for thin deep nets," in *Proc. ICLR*, 2015, pp. 1–13.

[8] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 4, pp. 5191–5198.

[9] T. Li, J. Li, Z. Liu, and C. Zhang, "Few sample knowledge distillation for efficient network compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14627–14635.

[10] Z. Yang, Z. Yan, X. Sun, W. Diao, Y. Yang, and X. Li, "Category correlation and adaptive knowledge distillation for compact cloud detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5623318.

[11] Y. Zhang, Z. Yan, X. Sun, W. Diao, K. Fu, and L. Wang, "Learning efficient and accurate detectors with dynamic knowledge distillation in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5613819.

[12] D. Li, Y. Nan, and Y. Liu, "Remote sensing image scene classification model based on dual knowledge distillation," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4514305.

[13] Y. Yang, X. Sun, W. Diao, D. Yin, Z. Yang, and X. Li, "Statistical sample selection and multivariate knowledge mining for lightweight detectors in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5626414.

[14] K. Xu, P. Deng, and H. Huang, "Vision transformer: An excellent teacher for guiding small networks in remote sensing image scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5618715.

[15] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4320–4328.

[16] Q. Guo et al., "Online knowledge distillation via collaborative learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11017–11026.

[17] D. Chen, J.-P. Mei, C. Wang, Y. Feng, and C. Chen, "Online knowledge distillation with diverse peers," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 4, pp. 3430–3437.

[18] I. Chung, S. Park, J. Kim, and N. Kwak, "Feature-map-level online adversarial knowledge distillation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 2006–2015.

[19] T. Su, J. Zhang, Z. Yu, G. Wang, and X. Liu, "STKD: Distilling knowledge from synchronous teaching for efficient model compression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 10051–10064, Dec. 2023, doi: 10.1109/TNNLS.2022.3164264.

[20] S. You, C. Xu, C. Xu, and D. Tao, "Learning from multiple teacher networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 1285–1294.

[21] L. T. Nguyen, K. Lee, and B. Shim, "Stochasticity and skip connection improve knowledge transfer," in *Proc. 28th Eur. Signal Process. Conf. (EUSIPCO)*, Jan. 2021, pp. 1537–1541.

[22] S. Park and N. Kwak, "Feature-level ensemble knowledge distillation for aggregating knowledge from multiple networks," in *Proc. ECAI*, 2020, pp. 1411–1418.

[23] Y. Pang, Y. Zhang, Y. Wang, X. Wei, and B. Chen, "SOCNet: A lightweight and fine-grained object recognition network for satellite on-orbit computing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5632913.

[24] L. Yao, X. Zhang, Y. Lv, W. Wang, and M. Li, "FGSC-23: A large-scale dataset of high-resolution optical remote sensing image for deep learning-based fine-grained ship recognition," *J. Image Graph.*, vol. 26, no. 10, pp. 2337–2345, 2021.
[25] S. Xing, J. Xing, J. Ju, Q. Hou, and X. Ding, "Collaborative consistent knowledge distillation framework for remote sensing image scene classification network," *Remote Sens.*, vol. 14, no. 20, p. 5186, Oct. 2022.
[26] H. Zhao, X. Sun, F. Gao, and J. Dong, "Pair-wise similarity knowledge distillation for RSI scene classification," *Remote Sens.*, vol. 14, no. 10, p. 2483, May 2022.

Yanhua Pang (Graduate Student Member, IEEE) received the B.S. degree in automation from the Hunan University of Technology, Zhuzhou, China, in 2018, and the M.S. degree in instrumentation engineering from Yanshan University, Qinhuangdao, China, in 2021. He is currently pursuing the Dr.-Ing. degree in mechanical engineering with the Institute of Space Science and Applied Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China.

His research interests include lightweight neural networks, model compression, object detection and recognition based on remote sensing images, and satellite-on-orbit computing.

Yamin Zhang received the B.Eng. degree in remote sensing science and technology and the M.A.Eng. degree in surveying and mapping engineering from the Shandong University of Science and Technology, Qingdao, China, in 2013 and 2015, respectively.

He is currently a Research Assistant with the Harbin Institute of Technology (Shenzhen), Shenzhen, China. His research interests include space photogrammetry and deep learning of remote sensing.

Yi Wang received the B.S. degree in remote sensing science and technology from the PLA Information Engineering University, Zhengzhou, China, in 2010, the M.S. degree in photogrammetry and remote sensing from the Shandong University of Science and Technology, Qingdao, China, in 2013, and the Ph.D. degree in remote sensing science and technology from Chang'an University, Xi'an, China, in 2018.

He is currently a Post-Doctoral Fellow with the Harbin Institute of Technology (Shenzhen), Shenzhen, China. His research interests include space photogrammetry and deep learning of remote sensing.

Xiaofeng Wei received the Ph.D. degree in remote sensing science and technology from PLA Information Engineering University, Zhengzhou, China, in 2015.

Then, he studied as a Post-Doctoral Fellow at Peking University, Beijing, China. His research interests include remote sensing technology and global discrete grid systems.

Bo Chen (Member, IEEE) received the B.E. degree in aerial photogrammetry and the Ph.D. degree in photogrammetric engineering and remote sensing from PLA Information Engineering University, Zhengzhou, China, in 2002 and 2008, respectively.

Since 2020, he has been a Full Professor with the Institute of Space Science and Applied Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China. His research interests include satellite-on-orbit computing, geospatial big data, and spatial information engineering.