Weakly Supervised Part-Based Method for Combined Object Detection in Remote Sensing Imagery

Wanli Qian, Zhiyuan Yan, Member, IEEE, Zicong Zhu, Student Member, IEEE, and Wenxin Yin, Member, IEEE

Abstract—Deep learning methods have reached considerable achievement on remote sensing object detection in recent years. However, most methods are designed for single object detection, such as vehicles and ships, and have limited detection capabilities for the combined object with large scale and complex part structure. In this article, we propose a part-based topology distillation network (PTDNet) for accurate and efficient combined object detection in remote sensing imagery. Specifically, a part-based feature module is designed to extract the key parts information of a combined object in a weakly supervised manner. Besides, to balance the accuracy and efficiency of the model, with considering the topology structure of multiple parts in combined objects, a lightweight network training method based on partial topological feature distillation is proposed to improve the model performance without additional parameters. Experiments show that the PTDNet outperforms the state of the art methods and achieves 65.4% mean average precision and 84.1% accuracy for combined object detection.

Index Terms—Combined object detection, knowledge distillation, part-based topological structure, remote sensing, weakly supervised learning.

I. INTRODUCTION

OBJECT detection is an important research area in remote sensing image interpretation. While existing models focus mainly on a single object, such as vehicles, ships, and aircraft, the combined objects detection, usually consisting of small and medium-sized compound facilities built for a specific purpose, is also a critical topic. Such combined objects include power plants that produce and transmit electricity, sewage treatment plants that remove pollutants in urban sewage, and waste-to-energy plants which combust waste to produce electricity. These combined objects usually have a significant impact on social development. Hence, it is of great significance to develop detection methods for combined objects.

While we still need to formalize the concept of combined object detection, we found previous research in the computer vision field for the human body or facial recognition instructive to our research on combined objects. We can divide the human body or facial recognition approaches into general feature extraction and multitop feature fusion methods. The first category uses a novel network structure or loss function to achieve high precision. Zhang et al. [1] used the region proposal network (RPN) to generate candidate boxes and feature maps, then used them to train a cascaded boosted forest classifier, which proves the effectiveness of the Faster R-CNN framework. The second category treats the combined objects as parts combined following specific geometric properties. These methods try to exploit each part’s inherent properties, thereby constructing the effective internal relationship between detectors and combined objects. The most representative work is the Deformable Part Model (DPM) proposed by Felzenszwalb et al. [2], which constructed a human body model through three parts: 1) overall structure; 2) partial structure; and 3) deformable model. However, although the human body or facial recognition brings a few good insights, the problem of insufficient data for combined objects in remote sensing imagery greatly hinders the research in this field.

Some common combined objects in remote sensing images are shown in Fig. 1, including the sewage treatment plant, thermal power plant, waste treatment plant, and some dedicated fixed facilities. Specifically, the sewage treatment plant in Fig. 1(a) comprises a circular sedimentation tank marked in red and an arc-shaped sedimentation tank marked in blue. Fig. 1(b) shows a thermal power plant consisting of a coal yard marked in red, a pool marked in blue, and a chimney or cooling tower marked in green. The waste treatment plant in Fig. 1(c) consists of multiple waste parking areas and Fig. 1(d) shows the dedicated fixed facilities, including multiple red-marked launchers and transportation channels connected. Detecting these combined objects is difficult for the following reasons.

1) The combined object often has variable layouts. Take the thermal power plant as an example, as shown in Fig. 2, the coal yards (marked as red), the pools (marked as blue), and the cooling towers (marked as green) are distributed differently in various instances of the thermal power plant. In other words, although the components are similar, the layout of each part is variable and susceptible to large size changes. Therefore, it is not easy to model and describe these combined objects uniformly.

2) The detection of combined objects tends to have high false alarm rate. The combined object often appears with
Fig. 1. Common parts in remote sensing images. The sewage treatment plant in the top left consists of a circular sedimentation tank marked in red and an arc-shaped sedimentation tank marked in blue. The thermal power plant in the top right consists of a coalfield marked in red and a pool marked in blue, chimneys and cooling towers marked in green. The waste treatment plant in the bottom left consists of multiple garbage parking areas. The dedicated fixed facilities in the bottom right include multiple launch racks marked in red and are connected by transportation channels.

Fig. 2. Samples of thermal power plants with irregular components. Components include chimneys marked by green, coal yards marked by red, and pools marked by blue.

Fig. 3. Comparison between the existing method and our method.

complicated background in remote sensing images. Besides, the shape of the combined object is often irregular. These factors make it hard to learn the accurate feature representation of the combined object, and it is easy to get high false alarm rate.

3) The contradiction between the high precision and fast speed of combined objects detection. The combined objects often occupy a large area with complex composition in large-scale remote sensing images. Effective feature learning often requires a heavy model with a large number of parameters. Lightweight model is hard to obtain accurate feature representation constrained by a limited number of parameters. However, high-complexity deep models often need more computing resources, and the training and inference time is too long, which is not conducive to the practical application of combined object detection.

This article proposes a part-based topology distillation network (PTDNet) for combined object detection in remote sensing imagery in response to the above problems. We aim to make full use of the object’s local features to improve the precision of the model while leveraging computational efficiency. As shown in Fig. 3, considering the variable layout of the combined objects, we design a clustering-based module to achieve the unsupervised part feature extraction. Moreover, models containing complex components usually require more computing resources and time. We reduce computing costs by introducing the knowledge distillation technique. Specifically, the part-based joint loss function is introduced in the knowledge transfer process of the teacher–student network. The above-mentioned designs and strategies effectively alleviated the problems of multiscale inputs, high false alarm rate, and difficulty in refined labeling of combined targets. Our innovations and achievements can be summarised as follows.

1) We propose a PTDNet for combined object detection in remote sensing imagery. It can utilize the key part information and distill according to the topology structure to get accurate and efficient detection for combined objects.

2) A part-based feature module (PFM) is designed to extract the key parts information of combined object in remote sensing scenes. It can achieve fine extraction of local features by clustering strategy, thereby effectively improving the accuracy of combined object detection.

3) To balance the accuracy and complexity of our model, with taking into account the topology structure of multiple parts in combined objects, we propose a lightweight network training method based on partial topological feature distillation. The lightweight network transfers the local topology information learned by the teacher network through the learning strategy of knowledge distillation. In this way, we can maintain the inference performance of the model without additional parameters.
The organization of the rest of this article is as follows. Section II introduces the related research works. Then, Section III describes our method in detail. The experiment results are shown in Section IV. Finally, Section V concludes the article and discusses future work.

II. RELATED WORKS

This section will introduce the research progress of related work from the following three aspects: 1) deep learning-based object detection and recognition methods; 2) combined object modeling methods; and 3) weakly supervised learning methods.

A. Object Detection and Recognition Method Based on Deep Learning

The main idea of object detection algorithm based on convolutional neural network (CNN) is to use deep CNN to extract object features. Then, calculate the features which contain position information to obtain the predicted position of the object. Object detection algorithms based on CNNs can be divided into single-stage object detection and dual-stage object detection.

The most representatives dual-stage object detection algorithms are regions with convolutional neural networks (RCNN) [3], spatial pyramid pooling (SPPNet) [4], Fast RCNN [5], Faster RCNN [6], etc. Such algorithms usually include candidate region extraction and prediction procedures. The candidate area extraction stage screens the candidate area through a dedicated area extraction module that aims to distinguish the foreground and background before prediction and balance positive and negative samples through sample selection. A CNN is used to construct features in each candidate area, determining the object position and category in the prediction stage. RCNN [3] extracted candidate object regions through the selective search method, obtaining the feature vector of each object region based on the CNN. Then, determine the object category through training support vector machine (SVM) [7]. More recently, several methods have been proposed inspired by the high-performance RCNN model. Cheng et al. [8] proposed to learn a rotation-invariant CNN (RicCNN) model in R-CNN framework used for multiclass geospatial object detection. Furthermore, in their more recent work, Cheng et al. [9] proposed the rotation-invariant and Fisher discriminative CNN (RIFD-CNN) model by imposing a rotation-invariant regularizer and a Fisher discrimination regularizer on the CNN features. Long et al. [10] proposed an unsupervised score-based bounding box regression method based on RCNN framework to achieve high accuracy in high-resolution images.

The above methods inspired by RCNN have achieved high precision at the cost of efficiency. To solve the high computational cost in RCNN related methods, SPPNet [4] proposed a spatial pyramid pooling method, which extracted the features of all candidate regions on the entire image and improved the efficiency of the network. Fast RCNN algorithm [5] proposed region of interest (ROI) pooling which calculated the overall feature vector of the CNN for the input image and achieved higher performance compared to the SVM classification model used in previous models. Based on Fast RCNN, Faster RCNN [6] designed a candidate region extraction network (RPN) [6] that generated candidate region boxes which further accelerated the calculation speed of the model. In addition, Cascade RCNN [11] cascaded multiple detection networks to improve Intersection over Union (IoU) stage by stage. This method reduced the mismatch between the candidate frames of the network in the training phase and the prediction phase. While improving detection performance, this method has a longer run time. To sum up, the two-stage detection algorithms generally use a dedicated module to obtain candidate regions. The features extracted in each region are highly correlated with the object, so the detection is considerably accurate, but the overall running speed is slow.

Single-stage object detection algorithms are represented by you only look once (YOLO) series [12]–[14], single shot multi-box detector (SSD) [15], RetinaNet [16], etc. RetinaNet is one of the most representative single-stage methods in recent years. It aims to address the foreground–background class imbalance problem by a novel focal loss, which reshapes the cross entropy loss to down-weights the loss assigned to well-classified examples. These algorithms do not require a dedicated candidate region extraction module. However, they directly extract the feature of the input image through a CNN to determine whether there is an object in the corresponding spatial location, then locate the object area.

Liu et al. [17] used the rotatable bounding box embedded into an SSD framework to explicitly parameterize the rotation of the bounding boxes. By using YOLOv2 architecture, Liu et al. [18] predicted the orientation of the object bounding boxes directly for detecting arbitrary-oriented ships. In addition, hard example mining [19], [20], multifeature fusion [21], transfer learning [22], nonmaximum suppression [23], etc., are often used in remote sensing object detection to enhance model performance. Without the extraction process for candidate regions, single-stage detection algorithms usually have a simple structure and the features used are not closely related to the object. Therefore, the detection speed is usually faster, but the detection accuracy is limited.

With more remote sensing image datasets such as UC merced land use dataset (UCMD) [24], EuroSAT [25], dataset for object detection in aerial images (DOTA) [26], and detection in optical remote (DIOR) [27] coming into view, many researchers have introduced object detection algorithms based on CNNs into remote sensing image object detection tasks. R2CNN [28] was initially designed for the detection task of rotating text, which used a slanted rectangular candidate frame to fit narrow and long objects. It is now widely used in the detection task of ships, vehicles, and other rotated objects in remote sensing images. Based on R2CNN, multi-category rotation detector for small, cluttered and rotated objects (SCRDet) [29] expressed the slanted rectangular candidate frame by five parameters, including the rotation angle, and proposed the inclined IoU calculation based on triangulation. Context-aware detection network (CAD-Net) [30] integrated a global context network and a pyramid local context network to obtain the global scene level and local object-level context information.
However, combined objects have diverse shapes and boundaries than common remote sensing image objects such as vehicles, ships, and airplanes. They usually include multiple parts with independent spatial locations and various dedicated functions, making feature representation difficult. Therefore, we have unsatisfactory performance when the traditional remote sensing object detection algorithm is applied to combined objects.

To sum up, deep learning based algorithms have achieved fruitful results in object detection tasks for remote sensing images of large scenes. Nevertheless, conventional remote-sensing detection algorithms yield subpar results when facing rich object types and complex background information, especially the combined object containing multiple parts. Therefore, constructing a remote sensing image object detection algorithm suitable for combined objects based on CNNs is an important research focus.

### B. Combined Object Modeling Algorithm

In current deep object detection research, the common combined object is the human face and body, as they can all be seen as a combination of multiple independent parts, which constitute a more semantically advanced object. The following survey in this subsection investigates the human body and face-detection algorithms. The related approaches can be divided into standard feature extraction methods and multipart feature fusion methods.

Many methods use common feature extraction methods and innovative network structure or loss functions that do not utilize the object’s topological information. Zhang et al. [1] used the regional feature extraction network to generate candidate boxes and convolutional feature maps and then used these results to train the cascaded boosted forest classifier, which verifies the effectiveness of the Faster R-CNN framework. HyperLearner [31] is composed of Faster R-CNN and a channel feature module, which stitches the output features of multiple convolutional layers with the image segmentation channel features. These methods have achieved decent results.

Compared with common methods, the multipart-based method takes geometric information of the combined objects into account and fully uses the inherent relationships between the distinct parts. Han et al. [32] proposed a powerful part-based convolutional neural network (P-CNN) for fine-grained visual categorization, which achieves state of the art performance. These methods focus on constructing an effective part detector and modeling. Many researchers divide combined goals into specific parts. Generally, the human body is divided into three overlapping parts: 1) the upper; 2) middle; and 3) lower parts, and these image subblocks are input into the CNN respectively to obtain the characteristics of these parts. Finally, the characteristics of the human body are combined. Different researchers may adopt different segmentation methods. In the article [33], the human body is divided into the following four parts:

1. Face;
2. Legs;
3. Left limb;
4. Right limb.

At the same time, there is also related research on automatic detection of effective parts. The most representative work, the DPM proposed by Felzenszwalb et al. [2], is composed of three components: 1) the overall structure; 2) the partial structure; and 3) the deformation model of the part relative to the overall structure. The overall structure and the partial structure correspond to low-resolution and high-resolution information. Felzenszwalb et al. [34] were inspired by the performance of cascade modeling in face detection tasks and proposed a cascade-based DPM.

Based on the simple idea that the combined object contains multiple parts, and according to the spatial position correspondence between the original image and the feature map, some researchers [35] also divide the feature map into blocks and directly using the block features as the reference unit to perform subsequent processing on the image. This kind of method transforms the part information of the original image into the block information in feature dimension, making feature selection more flexible.

Because the combined objects of remote sensing images have complex internal structure, high modeling difficulty, and little public data, there is little research on combined objects in remote sensing images. The usual method [36] tackled the detection task of dedicated fixed facilities. It obtains several remote sensing images with a large coverage area through sliding windows and sends these slices to the CNN for binary classification. Then, K-means [37] spatial clustering is used with the spatial position corresponding to each slice and the confidence of the classification, and finally synthesizing the entire original image to obtain the detection result of the special fixed facility. This method has the following shortcomings: 1) The method directly uses Alexnet [38], ResNet [39], VGG [40] and other networks for feature extraction and does not consider the specific characteristics of the object; 2) The method is compared against expert judgment results which make it hard for quantitative analysis with other methods; 3) The detection process of this method is not in an end-to-end fashion, which is not suitable for the future adjustment of the model.

In summary, the research on remote sensing image combined object detection and recognition is still in the preliminary stage, with many unresolved questions. Hence, constructing a special modeling algorithm based on a CNN suitable for combined objects of remote sensing images is of great significance.

### C. Weakly Supervised Learning Algorithm

Due to the inflated cost of the data labeling process, it is difficult for many tasks to obtain strong supervision information of all the truth labels. Therefore, there is a pressing need for constructing predictive models based on weakly supervised learning. According to the difference of supervision information, weakly supervised learning can be divided into incomplete supervision, inexact supervision, and inaccurate supervision.

Incomplete supervision means that only part of the training data is labeled, while most remaining data are unlabeled. The two main methods to solve the problem of incompletely supervised
learning are: 1) active learning and 2) semisupervised learning. Active learning assumes that the truth labels of unlabeled data can be obtained by querying “human experts.” Active learning tends to select the most valuable unlabeled data given a small amount of labeled data and a large amount of unlabeled data to query. Active learning models are usually based on either the amount of information [41] or representative construction [42].

The value of data selection is usually determined through the amount of information, which measures the degree to which unlabeled data can reduce the uncertainty of the statistical model and representative construction [43], and the probability that the sample can represent the input distribution of the model. Recently, some methods have begun to try both characteristics to determine the value of data selection [43]. Semisupervised learning attempts to automatically use labeled data and unlabeled data to improve learning performance without manual intervention. The main assumption of semisupervised learning is that similar data inputs can obtain similar outputs. Commonly used are generative methods [44], graph-based methods [45], [46], low-density segmentation, low-density separation methods [47], [48], and disagreement methods [49], [50].

The authors in [51] proposed a dynamic curriculum learning strategy by feeding images with increasing difficulty that matches with the current detection ability of the object detector. Additionally, an instance-aware focal loss function for detector learning is added to balance the bias effect of the progressive training scheme. Other examples include random noise learning [52]–[55], dynamic curriculum learning [51], and so on. Common ideas include identifying and correcting wrong samples [53] or inferring truth labels in crowd-sourcing mode [54], [55].

Inaccurate supervision means that the training data only give coarse-grained object labels, which is inconsistent with the task. Feng et al. [56] provided a robust self-supervised adversarial and equivariant network to learn complementary and consistent visual patterns for weakly supervised object detector. Differing from fine-grained object interpretation, inaccurate supervision refers to multinstance learning [57]–[59], which takes a multi-instance package as the training unit. Each package contains multiple specific instances. If there is one positive instance in the package, it becomes a positive package. If a package contains all negative instances, the package is a negative one. In the multi-instance learning task, we are given the label of the instance package while the label of the specific instance is unknown. The purpose is to predict the label of the unknown package. An important assumption for inaccurate supervision is that the labels in the training data are not always true values.

In summary, the combined object interpretation includes object-level annotations, but without part-level annotations, which is comparable to the case of inaccurate supervision. However, the relationship between individual components within the combined objects is quite different from the instance-package relationship in inaccurate supervision. Therefore, achieving part-based object interpretation based on weakly-supervised learning is still a significant challenge.

III. Method

A. Overview

Our innovations mainly consist of PFMs and teacher–student knowledge distillation networks.

The overall model architecture is shown in Fig. 4. The proposed PTDNet contains two models: 1) the teacher and 2) the lightweight student models.

The teacher network containing PFM is based on RetinaNet utilizing ResNet-50 [39], which is elaborated in Fig. 5. The student network chooses ResNet-18 [39] as the backbone to reduce the computational costs and parameters. The proposed architecture has three losses: 1) classification loss; 2) regression loss; and 3) distillation loss.

The teacher network first sends images to the Feature Pyramid Network (FPN) with ResNet-50 as the backbone to derive the global image feature. Then, the extracted features are sent to the classification and regression heads to generate the classification and localization results, respectively. Meanwhile, PFM applies the K-means algorithm to cluster the extracted features. The spatially-correlated features can be clustered into individual groups during the clustering process. We then introduce part classiﬁcation and regression loss to improve the detection results. In order to avoid high part-level annotation cost, PTDNet manages to transfer the topological information acquired from the teacher network to the student network via distillation.

B. PFM Designing

It is diﬃcult for large-scale remote sensing images to obtain numerous data with detailed annotations. The contour and relative position relationship of its corresponding components is undoubtedly a powerful basis for discrimination for the combined object. The experiments further validate the above point of view by extracting the component features of the combined object, thereby enhancing its overall features. This section proposes PFM to obtain the component features inside each object automatically. A joint loss function is designed for feature extraction without component-level annotations based on a weakly supervised method.

1) PFM Architecture: To handle the problem of multiscale objects in combined object detection, as illustrated in Fig. 5, we adopt the FPN structure with the layers P3, P4, and P5 in the network backbone to generate multiscale feature maps for objects of various sizes and shapes. Then, we learn hierarchical part information with these layers to guide the detection of the multiscale combined object. For P3, P4, and P5 extracted by the CNN, the proposed PFM applies K-means algorithm to obtain the cluster centers on each channel, and the peak responses obtained by these clusters correspond to the respective internal parts of the combined object. PFM extracts part features inside the object without part-level annotations, so that this work can be regarded as the inaccurate supervision task in weakly supervised learning.

The classification loss and regression loss are calculated separately with the features of the current component position,
Fig. 4. PTDNet structure. PTDNet contains two subnetworks, the teacher network and the lightweight student network. The teacher network uses ResNet-50 RetinaNet as the backbone. RetinaNet attaches two subnetworks, Class subnet for classifying anchor boxes and Box subnet for regressing anchor boxes to ground-truth object boxes. In order to reduce Params and run time, we choose ResNet-18 as the backbone of the student network. We also incorporated distillation loss during training. The total loss considers both regularization and classification loss and the similarity of the image feature output from the teacher and student network.

Fig. 5. Architecture of network including detection head with PFM. This framework consists of deep feature extraction subnetwork, Global prediction subnetwork and PFM. In deep feature extraction, the CNN produces multiscale feature maps including irregular shapes and semantic meaning of the object. Global prediction subnetwork includes class subnet for predicting label for anchors and box subnet for regressing from anchors to bounding boxes. PFM uses K-means to cluster extracted features into groups. Each group consists of features with spatially correlated patterns, meaning that each group is likely to be a component of thermal power plants.

The most important feature of a combined object is that there are multiple spatially dispersed distinct parts inside, which can be utilized to enhance the ability for overall feature representation. Therefore, if we map the region with a more significant response activation value of the depth feature of the combined object to the original image space, it should be consistent with the obvious part of the object. Therefore, this method attempts to cluster the multiscale depth feature maps into several groups through the K-means clustering algorithm [37], then the clustering center of each group corresponds to each unique part of the thermal power plant.

2) Part-Based Joint Loss Function: In this section, we propose our joint loss function based on parts. For combined object, the overall loss $L$ is composed of global loss $L_{\text{global}}$ and part loss $L_{\text{part}}$ in interest areas as shown in (1), we add an adjustable parameter $\alpha_{\text{part}}$ to keep the
balance between them

\[ L = L_{\text{global}} + \alpha L_{\text{part}}. \]  

(1)

For global loss \( L_{\text{global}} \), its loss function is defined as the sum of classification loss function \( L_{\text{class}} \) and regression loss function \( L_{\text{reg}} \). As in (2), where \( p_i \in [0, 1] \) is the estimated probability of the object category, \( p_i^* \in \{0, 1\} \) is the true value label of the preselection box, i.e., when the box is the foreground object, \( p_i^* = 1 \), when it is the background, \( p_i^* = 0 \). \( k_i \) represents the four parameterized coordinate vectors of the preselection box, \( k_i^* \) represents coordinate vector of the true position of the box. \( \lambda \) is used to balance classification loss \( L_{\text{class}} \) and regression loss \( L_{\text{reg}} \). Here we set the value of \( \lambda \) to 1

\[ L_{\text{global}} = \frac{\sum_{i} L_{\text{class}}(p_i, p_i^*)}{N_{\text{class}}} + \lambda \frac{\sum_{i} p_i^* L_{\text{reg}}(k_i, k_i^*)}{N_{\text{reg}}}. \]  

(2)

Regression loss function in RetinaNet is the standard smooth-

\[ L_{\text{reg}} = \text{Smooth} L_1(k = k_i^* - k_i) = \begin{cases} 0.5 k^2, & \text{if } |k| < 1 \\ |k| - 0.5, & \text{otherwise}. \end{cases} \]  

(3)

Classification loss function \( L_{\text{class}} \) is the softmax loss of the two category labels, the foreground and the background. In the RetinaNet, the classification loss function \( L_{\text{class}} \) is a two-category focal loss function, which is designed based on the possible category imbalance problem of the cross-entropy loss during the training process. While \( \alpha \in [0, 1] \) is a weighting factor to adjust the class imbalance problem. \( \gamma \in [0, 5] \) is an adjustable parameter to smoothly control difficult and easy samples in the dataset. \( P_i \in [0, 1] \) represents the probability that the class label is predicted to be 1. It is defined by (5), in which \( y \) represents the ground truth

\[ L_{\text{class}} = -\alpha y(1 - p_i)^{\gamma} \log(p_i) \]  

(4)

\[ p_i = \begin{cases} \frac{p}{y}, & \text{if } y = 1 \\ 1 - p, & \text{otherwise}. \end{cases} \]  

(5)

For \( L_{\text{part}} \), we first need to identify the area of interest. First, the probability of being distributed in \( N \) target categories at each spatial location can be regarded as a set of vectors with dimensions NA, WH, where A represents the candidate boxes, W represents the width, and H represents height. The dimension 4 A, WH represents the set of four relative offset coordinates between the candidate box and the truth box for each of the candidate boxes at each spatial position. Then, we calculate the sum of \( k \) cluster centers which are of dimensions NA, k and 4 A, k. The cluster center obtained by clustering should mainly correspond to multiple components of the combined target. We define \( L_{\text{part}} \) similar to \( L_{\text{global}} \), in (6), where the parameter \( \lambda_{\text{part}} \) is added to balance \( L_{\text{part}} \) and \( L_{\text{reg}} \). It is defaulted to 1

\[ L_{\text{part}} = \frac{1}{N_{\text{part}}} \sum_{k} L_{\text{part}}(p_i, p_i^*) + \lambda_{\text{part}} \frac{1}{N_{\text{reg}}} \sum_{k} p_i^* L_{\text{reg}}(t_i, t_i^*). \]  

(6)

In loss function, \{\( p_i \)\} represents the probability of being distributed in \( N \) object categories at each spatial location, which can be regarded as a set of vectors with dimensions NA, WH, where W represents width and H represents height. \{\( t_i \)\} represents the set of four relative offset coordinates between the candidate box and the truth box for each of the A candidate boxes at each spatial position. The dimension can be expressed as 4 A, WH. Therefore, we can get \{\( p_i \)\} and \{\( t_i \)\} by the dimension of NA, k and 4 A, k by using the part-based loss function in (6) to calculate \( p_i \) and \( t_i \) in the k cluster centers.

The cluster centers obtained by clustering should mainly correspond to multiple parts of the combined object. A parameter-adjusted cross-entropy loss is used for brevity as part-based classification loss (7). The loss function can also be regarded as a focal loss when \( \gamma = 0 \). Furthermore, for part-based regression loss, \( L_{\text{reg}} \) should be the same as smooth \( L_1 \) loss used in \( L_{\text{reg}} \)

\[ L_{\text{class}} = CE(p_i) = -\alpha \gamma \log(p_i). \]  

(7)

C. Partial Topological Feature Distillation Networks

The combined objects, such as power plants and sewage treatment plants, often occupy a large area with complex composition in large-scale remote sensing images. Effective feature learning often requires heavy model with a large number of parameters. Lightweight model is usually hard to obtain accurate feature representation constrained by a limited number of parameters. However, heavy deep models often need more computing resources, and the training and inference time is too long, which is not conducive to the practical application of combined object detection. To weigh the accuracy and complexity of deep model, knowledge distillation has become an effective solution in recent years. However, most of the existing distillations are designed for single object, such as vehicles, ships, and aircraft, which have difference with the combined objects. In fact, the multiple parts in combined objects usually have apparent topological distribution, some examples are shown in the Figs. 6 and 7. It is of vital importance to consider internal topological structures between each part in a combined object. If we regard a complex
combined object as a graph structure, the parts in the object can correspond to the nodes and edges due to the differences in their visual features and relative positions.

Considering the topology structure of the multiple parts in combined objects, we propose a lightweight network training method based on partial topological feature distillation. As shown in Fig. 4, both global image feature produced by CNN backbone and predictions obtained from classification and regression subnets are utilized as the training guidance. As a result, the lightweight network transfers the global and local topology information learned by the teacher network through the learning strategy of knowledge distillation. In training process, we first train a complex teacher network with strong expressive ability. Then, the parameters of the teacher are frozen and used to guide the student network training. The goal of the distillation training is as follows:

\[
L = L_{\text{feat}} + L_{\text{pred}}
\]

(8)

\[
L_{\text{feat}} = \frac{F_T F_S}{\|F_T\| \ast \|F_S\|}
\]

(9)

\[
L_{\text{pred}} = \alpha_{ts}(L_{T\text{reg}} + L_{T\text{class}}) + L_{\text{reg}} + L_{\text{class}}
\]

(10)

where \(L_{\text{feat}}\) measures the similarity of the features output by the student network and teacher network. \(L_{\text{reg}}\) and \(L_{T\text{class}}\) represent the loss of regression and classification between the student network and teacher network, respectively. \(\alpha_{ts}\) is the weight factor. Besides, \(L_{\text{reg}}\) and \(L_{\text{class}}\) represent the loss of regression and classification between the student network and ground truth. In this way, we quantify the difference in topological characteristics by measuring the similarity of deep features and predictions, and pass the topological information from teacher to student via distillation.

IV. EXPERIMENTS

A. Experimental Settings

The experiments use an NVIDIA GeForce RTX2080 GPU with Pytorch deep learning framework. RetinaNet [16] with ResNet-50 [39] is used as the baseline method, which is initialized by the pretrained model on the ImageNet dataset [60]. In the part-based joint loss function, the global loss function (2) is consistent with the loss function of RetinaNet. Additionally, the stochastic gradient method is used in the training process with a momentum of 0.9. The network model has been trained with a total of 80 epochs. The initial learning rate of the network is set to 0.001 and decreased by a tenth in every 10 000 steps. The ratio of negative samples to positive samples is set to 3 in the training phase to suppress negative samples.

Dataset objects that include thermal power plants and special fixed facilities are prepared for the experiments. The ratio between the training set and the test set is 7:3. There are more than 2000 thermal power plants in the dataset and 3000 special fixed facilities. All samples are 900*900 pixels, and the resolution is 0.60 m.

B. Evaluation Metrics

The experiments use the mean average precision (mAP) to measure the performance of listed methods. Frames per second (FPS), floating-point operations per second (FLOPs), and model parameters (Params) are applied to evaluate the computational costs in space and time. Specifically, \(mAP^{(0.5:0.95)}\) is applied as the main detection evaluation metric. Typically, when IoU is over 0.5, its prediction matches the true value and fails to match otherwise. As for the recognition metrics, the experiment uses the overall accuracy and confusion matrix. The accuracy is the percentage of the number of correctly identified objects on the test set to the total number of image objects, which is a reasonable measure due to the scale of the test set. The confusion matrix is the percentage matrix dividing each category into other categories. It shows the classification effect and demonstrates the network’s ability to recognize each category

\[
\text{IoU}(A, B) = \frac{A \cap B}{A \cup B}
\]

(11)

Depending on this, we could categorize predicted bounding boxes into true positive (TP), true negative, false positive (FP), and false negative (FN), and precision (p) and recall (r) are defined as

\[
p = \frac{TP}{TP + FN}
\]

(12)

\[
r = \frac{TP}{TP + FP}
\]

(13)

\[F_1 = \frac{2pr}{p + r}
\]

(14)

mAP is then defined as

\[
mAP = \frac{1}{|c|} \sum_{k \in c} \frac{TP(k)}{TP(k) + FP(k)}
\]

(15)
Fig. 8. Heat map visualization of feature representation. Ablation experiments show that PFM could help focus on interest areas during the training process. The first column is the input image; the second column is the heat-map visualization of baseline RetinaNet; the third column is the heat-map visualization of network with our PFM; the last column shows the topological structures inside the combined object with the same color mapping as Fig. 2.

| α_{part} | mAP  | max F1 | Thermal power plant | Dedicated fixed facility | Classification average |
|----------|------|--------|----------------------|--------------------------|------------------------|
| 0        | 0.632| 0.646  | 0.777                | 0.850                    | 0.819                  |
| 0.10     | 0.640| 0.667  | 0.780                | 0.866                    | 0.829                  |
| 0.25     | 0.658| 0.671  | 0.787                | 0.874                    | 0.837                  |
| 0.5      | 0.654| 0.670  | 0.788                | 0.877                    | 0.839                  |
| 0.75     | 0.651| 0.641  | 0.792                | 0.852                    | 0.826                  |
| 0.99     | 0.654| 0.612  | 0.778                | 0.846                    | 0.817                  |

Here \( c \) stands for all categories, and \( TP(k) \) means the number of true positives in class \( k \). We can see from the table that the joint loss function makes the deep network focus on the parts in the object, and the detection and recognition performance is significantly improved, which confirms the effectiveness of the part information in the description of the combined object.

C. Hyperparameter Selection

The hyperparameters in our article are discussed in this part, including \( \alpha_{\text{part}} \) in (1), \( \alpha_{fl} \) in (4), \( \alpha_{ce} \) in (7), and \( \alpha_{ts} \) in (8). Fig. 10 shows the mAP of combined object detection of each hyperparameter with different values.

To balance the global loss and local part loss, \( \alpha_{\text{part}} \) ranges from 0 to 1 in the experiment to study its influence. Table I and Fig. 10(a) show the performance of the proposed method when \( \alpha_{\text{part}} \) varies. It can be seen that the results demonstrate a trend of convex function, and the best performance is obtained in the interval \([0.25, 0.5]\). When it is close to 1, the performance is significantly reduced. The reason may be that the part feature extraction is still under optimized. Based on the above analysis, \( \alpha_{\text{part}} \) is set to 0.25 in our experiments.

Fig. 10(b) shows the performance of the proposed method when \( \alpha_{ts} \) varies. It can be seen that the mAP turns out an undulate curve with the peak point around 0.4 when \( \alpha_{ts} \) ranges from 0.05 to 0.95. Thus, we set \( \alpha_{ts} \) to 0.4 in our experiments.
Fig. 9. Performance of the student network on the training set and test set of the combined object detection. The blue curve and orange curve represent the mAP and the classification accuracy of the student network on the training set, respectively. While the gray curve and the yellow curve represent the mAP and the classification accuracy of the student network on the test set, respectively.

Furthermore, as shown in Fig. 10(c), when $\alpha_{fl}$ ranges from 0.05 to 0.95, the model achieves a high mAP value around 0.2–0.3. While, as shown in Fig. 10(d), when $\alpha_{ce}$ ranges from 0.05 to 0.95, the results have a faint fluctuation between 0.2 and 0.5. The reason may be that $\alpha_{ce}$ is in the inner layer in the loss calculation and the influence is significantly weakened after the weighted calculation of the outer layer. As a result, $\alpha_{fl}$ and $\alpha_{ce}$ both set to 0.25 is an acceptable decision for our datasets.

D. Ablation Experiments

1) Effect of PFM: The proposed PFM is trained for exploring the influence of key component features to the feature representation. As shown in Fig. 5, experiments use the RetinaNet detection framework by the addition of PFM. Experimental results are listed in the first row (RetinaNet) and fourth row (network with our PFM) in Table III. We can see that network with our PFM based on weakly supervised clustering increases mAP by 2.2% with respect to RetinaNet, which indicates the feature representation ability enhancement of PFM. Fig. 8 demonstrates PFM could help focus on interest areas during the training process.

2) Effect of Part Topological Features Distillation: Using network with our PFM as the teacher network, a teacher–student framework is constructed following Fig. 4. Distillation loss of PTDNet is defined in (8) containing loss functions of deep features ($L_{feat}$) and predictions ($L_{pred}$). Performances on the test dataset are shown in Table II, which indicates that student networks achieve comparable accuracy with the teacher network. Besides, the addition of $L_{feat}$ contributes to the knowledge transfer of combined object feature representation. However, the FLOPs of the student network (PTDNet) is only 78% of the teacher network, which demonstrates the reduction of the reasoning network complexity.

The performance of the student network on the training set and test set of the combined object detection are shown in Fig. 9. Specifically, the blue curve and orange curve represent the mAP and the classification accuracy of the student network on the training set. While the gray curve and the yellow curve represent the mAP and the classification accuracy of the student network on the test set. It can be seen that the growth of metric on the test set is slower than the training set, and the final performance on the test set is lower than the training set.

E. Performance Comparison

Performance comparison studies are conducted with various remote sensing object detection models on the dataset. In this article, the most classic and advanced single-stage methods, RetinaNet and YOLOv5, as well as the two-stage methods,
TABLE III

| Method               | mAP   | Precision | FPS (s) | Parameter number |
|----------------------|-------|-----------|---------|------------------|
| RetinaNet [16]       | 0.632 | 0.819     | 19.61   | 34.67M           |
| Faster-RCNN [6]      | 0.635 | 0.836     | 10.3    | 26.97M           |
| Cascade-RCNN [11]    | 0.641 | 0.861     | 5.33    | 93.75M           |
| YOLOv5 [61]          | 0.648 | 0.847     | 25.76   | 21.52M           |
| Network with our PFM | 0.654 | 0.858     | 16.24   | 35.28M           |
| PTDNet               | 0.637 | 0.841     | 23.44   | 21.9M            |

Fig. 12. Detection results visualization. Blue bounding boxes show the ground truth of detection and red bounding boxes are the prediction outputs.

Faster R-CNN and Cascade R-CNN, are selected as our comparison method. The experimental results are shown in Table III and Fig. 12. The mAP and precision in the table, respectively, represent the detection and recognition performance of each method. FPS and parameter number describe the time and space costs of the methods.

As shown in Fig. 12, the performances of two-stage Faster-RCNN and multistage Cascade-RCNN have minor mAP improvements and less efficiency than single-stage RetinaNet, however with several examples of false alarms, especially, Cascade-RCNN. This could be because that complicated multistage models are not easy to optimize, especially for the combined objects. Network with our PFM based on weakly supervised clustering increases mAP by 2.2% with respect to RetinaNet, which indicates PFM can enhance the feature representation ability by attention on key components. Using the network with our PFM as a teacher network, our lightweight student network PTDNet reduces the Params by 37.93% with comparable detection and recognition performances to the complicated models. It is worth mentioning that YOLOv5 has better accuracy and higher efficiency than RetinaNet, Faster-RCNN, Cascade-RCNN, and even our PTDNet. The reason may be that a large number of cost-free training tricks (such as various data augmentations, improved losses, etc.) are used in YOLOv5, which are beneficial to improve the accuracy. However, these training tricks are not used in our PTDNet. Thus, it is difficult to control for variables when comparing YOLOv5 with our methods. In the future, we will apply the useful tricks of YOLO in our method to further improve the performance.

V. CONCLUSION

In this article, we propose a PTDNet for combined object detection in remote sensing imagery. It can utilize the key part
In the future, the PFM for combined objects with various scales and orientations in remote sensing images can be further optimized. Besides, we will pay attention to enhance the feature representation of combined objects based on graph convolutional network due to its strong ability on topology. In a word, we will work to further improve the performance of the combined object detection.

REFERENCES

[1] L. Zhang, L. Lin, X. Liang, and K. He, “Is faster R-CNN doing well for pedestrian detection?,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 443–457.

[2] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2014.

[4] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in Proc. Eur. Conf. Comput. Vis., 2014, pp. 346–361. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-10578-9_23.

[5] R. Girshick, “Fast R-CNN,” in Proc. Int. Conf. Comput. Vis. (ICCV), Dec. 2015.

[6] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in Advances in Neural Information Processing Systems, D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015. [Online]. Available: https://proceedings.neurips.cc/paper/2015/file/14bad6b114875e45bba02a821ed38046-Paper.pdf.

[7] C. Cortes and V. Vapnik, “Support vector networks,” Mach. Learn., vol. 20, pp. 273–295, 1995.

[8] 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.

[9] G. Cheng, J. Han, P. Zhou, and D. Xu, “Learning rotation-invariant and Fisher discriminative convolutional neural networks for object detection,” IEEE Trans. Image Process., vol. 28, no. 1, pp. 265–278, Jan. 2019.

[10] Y. Long, Y. Gong, Z. Xiao, and Q. Liu, “Accurate object localization in remote sensing images based on convolutional neural networks,” IEEE Trans. Geosci. Remote Sens., vol. 55, no. 5, pp. 2486–2498, May 2017.

[11] Z. Cai and N. Vasconcelos, “Cascade R-CNN: Delving into high quality object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 615–624.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 779–788.

[13] J. Redmon, “YOLO9000: Better, faster, stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 7262–7271.

[14] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018, arXiv:1804.02767.
[39] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

[40] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, arXiv:1409.1556.

[41] N. Abe. “Query learning strategies using boosting and bagging,” in Proc. 15th Int. Conf. Mach. Learn., 1998, pp. 1–9.

[42] H. T. Nguyen and A. Smeulders. “Active learning using pre-clustering,” in Proc. 21st Int. Conf. Mach. Learn., 2004, Art. no. 79.

[43] S.-J. Huang, R. Jin, and Z.-H. Zhou, “Active learning by querying informative and representative examples,” in Proc. Adv. Neural Inf. Process. Syst., 2010, pp. 892–900.

[44] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, “Text classification from labeled and unlabeled documents using EM,” Mach. Learn., vol. 39, no. 2, pp. 103–134, 2000.

[45] F. Wang and C. Zhang, “Label propagation through linear neighborhoods,” IEEE Trans. Knowl. Data Eng., vol. 20, no. 1, pp. 55–67, Jan. 2008.

[46] M. Hein and M. Maier, “Manifold denoising as preprocessing for finding natural representations of data,” in Proc. 22nd Nat. Conf. Artif. Intell., 2007, pp. 1646–1649.

[47] O. Chapelle and A. Zien, “Semi-supervised classification by low density separation,” in Proc. Int. Workshop Artif. Intell. Statist., 2005, pp. 57–64.

[48] Y.-F. Li, I. W. Tsang, J. T. Kwok, and Z.-H. Zhou, “Convex and scalable weakly labeled SVMs,” J. Mach. Learn. Res., vol. 14, no. 7, pp. 2151–2188, 2013.

[49] Z.-H. Zhou, Ensemble Methods: Foundations and Algorithms. Boca Raton, FL, USA: CRC Press, 2012.

[50] W. Wang and Z.-H. Zhou, “Theoretical foundation of co-training and disagreement-based algorithms,” 2017, arXiv:1708.04403.

[51] X. Yao, X. Feng, J. Han, G. Cheng, and L. Guo, “Automatic weakly supervised object detection from high spatial resolution remote sensing images via dynamic curriculum learning,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 1, pp. 675–685, Jan. 2021.

[52] B. Frénay and M. Verleysen, “Classification in the presence of label noise: A survey,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 5, pp. 845–869, May 2014.

[53] F. Muhlenbach, S. Lallich, and D. A. Zighed, “Identifying and handling mislabelled instances,” J. Intell. Inf. Syst., vol. 22, no. 1, pp. 89–109, 2004.

[54] V. C. Raykar et al., “Learning from crowds,” J. Mach. Learn. Res., vol. 11, no. 4, pp. 1297–1322, 2010.

[55] B. Frénay and S. Yu, “Eliminating spammers and ranking annotators for crowdsourced labeling tasks,” J. Mach. Learn. Res., vol. 13, no. 1, pp. 491–518, 2012.

[56] X. Feng, X. Yao, G. Cheng, J. Han, and J. Han, “SAENet: Self-supervised adversarial and equivariant network for weakly supervised object detection in remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–11, 2021.

[57] J. Foulds and E. Frank, “A review of multi-instance learning assumptions,” Knowl. Eng. Rev., vol. 25, no. 1, pp. 1–25, 2010.

[58] X.-S. Wei, J. Wu, and Z.-H. Zhou, “Scalable algorithms for multi-instance learning,” IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 4, pp. 975–987, Apr. 2017.

[59] J. Amores, “Multiple instance classification: Review, taxonomy and comparative study,” Artif. Intell., vol. 201, pp. 81–105, 2013.

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

[61] G. Jocher et al., “ultralytics/YOLOv5: V6.1 - TensorRT, TensorFlow edge TPU and OpenVINO export and inference,” Feb. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6229396

Wenxin Yin (Member, IEEE) received the B.Sc. degree in information engineering from the Beijing Institute of Technology, Beijing, China, in 2016, and the M.Sc. and Ph.D. degrees in signal and information processing from Aerospace Information Research Institute, Chinese Academy of Sciences (CAS), Beijing, China, in 2019.

She is currently an Assistant Professor with the Aerospace Information Research Institute. Her research interests include computer vision and remote sensing image analysis.

Zhiyu Yan (Member, IEEE) received the B.Sc. degree in microelectronics from Xiamen University, Xiamen, China, in 2016, and the M.Sc. degree in signal and information processing from Aerospace Information Research Institute, Chinese Academy of Sciences (CAS), Beijing, China, in 2019.

She is currently an Assistant Professor with the Aerospace Information Research Institute. Her research interests include computer vision and remote sensing image analysis.

Wenxin Yin received the B.Sc. degree in information engineering from the Beijing Institute of Technology, Beijing, China, in 2016, and the M.Sc. and Ph.D. degrees in communication and information systems from Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, China, in 2021.

She is currently an Assistant Professor with Aerospace Information Research Institute, Chinese Academy of Sciences. Her research interests include computer vision and remote sensing image analysis.