

OpenRSD: Towards Open-prompts for Object Detection in Remote Sensing Images

Ziyue Huang¹, Yongchao Feng¹, Ziqi Liu¹, Shuai Yang¹, Qingjie Liu^{1,2*}, Yunhong Wang^{1,2}

¹State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China

²Hangzhou Innovation Institute, Beihang University, Hangzhou, China

{ziyuehuang, fengyongchao, liuzigi2002, buaa.yangshuai, qingjie.liu, yhwang}@buaa.edu.cn

Abstract

Remote sensing object detection has made significant progress, but most studies still focus on closed-set detection, limiting generalization across diverse datasets. Open-vocabulary object detection (OVD) provides a solution by leveraging multimodal associations between text prompts and visual features. However, existing OVD methods for remote sensing (RS) images are constrained by small-scale datasets and fail to address the unique challenges of remote sensing interpretation, include oriented object detection and the need for both high precision and real-time performance in diverse scenarios. To tackle these challenges, we propose OpenRSD, a universal open-prompt RS object detection framework. OpenRSD supports multimodal prompts and integrates multi-task detection heads to balance accuracy and real-time requirements. Additionally, we design a multi-stage training pipeline to enhance the generalization of model. Evaluated on seven public datasets, OpenRSD demonstrates superior performance in oriented and horizontal bounding box detection, with real-time inference capabilities suitable for large-scale RS image analysis. Compared to YOLO-World, OpenRSD exhibits an 8.7% higher average precision and achieves an inference speed of 20.8 FPS. Codes and models are available at: <https://github.com/floatingstarZ/OpenRSDt>.

1. Introduction

As an important task of remote sensing (RS) image interpretation, remote sensing object detection has witnessed significant advancements in recent years [29, 43]. Nonetheless, the majority of existing studies have concentrated on closed-set object detection, which aims to optimize performance on a fixed set of categories within specific datasets by refining model architectures [43] or loss functions [45]. These models frequently exhibit poor generalization ca-

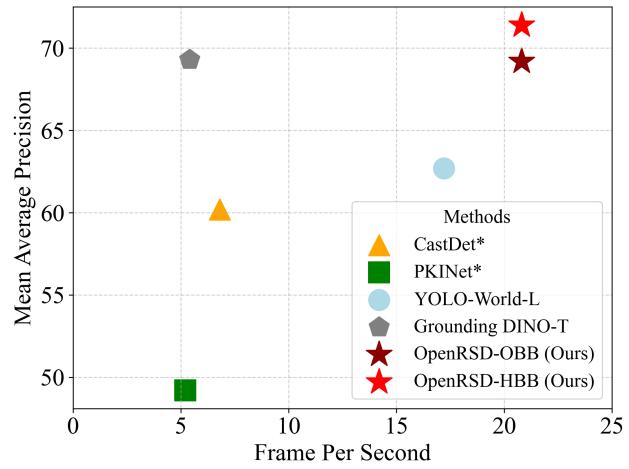


Figure 1. We compare the precision and inference speed of OpenRSD against other methods, with inference speed tested on a single 2080Ti GPU.

pabilities across diverse datasets and are limited to detecting objects belonging to predefined categories. Open-vocabulary object detection (OVD) [7, 24] in natural scenes present a viable solution to this limitation by establishing multimodal associations between text prompts and visual features, thereby relaxing the limitations of predefined categories. Furthermore, they leverage large-scale datasets such as image-text pairs [30], high-quality detection datasets [20], and grounding data [24] for training, enabling robust generalization across various scenes.

Recent studies have explored OVD models for RS images [22, 32, 41]; however, these efforts are constrained by the use of small-scale datasets, which are inadequate to support robust generalization. Furthermore, they fail to address the unique requirements of RS scenarios, such as detecting small objects and precisely object orientations. This gap motivates us to develop a universal RS object detector with functional flexibility and strong cross-domain detection capabilities, serving as a foundational model for RS image in-

*indicates the corresponding author.

terpretation. To this end, we propose a multimodal prompt-based object detector and introduce a multi-stage training pipeline to strengthen its generalization.

We improve upon the RTMDet [29] framework and propose the **OpenRSD**, which achieves the best performance in both overall accuracy and inference speed, as shown in Fig. 1. A prompt construction module is developed, utilizing pretrained models include SkyCLIP [40] and DINOv2 [31] to extract text and image prompt embeddings. The image prompts aim to reduce reliance on expert knowledge, which is particularly important since categories in RS images are often rare in everyday contexts. By incorporating random sampling during training, the detector is enabled to identify relevant object based on arbitrary prompts. We design two detection heads with multi-task: alignment head and fusion head. The alignment head performs classification by computing the similarity between detection features and prompt embeddings, offering faster inference and easy scalability for expanding the number of categories. The fusion head introduces cross-modal interactions, achieving better detection performance. Meanwhile, we integrate both horizontal and oriented bounding box regression tasks into one framework. Both heads are trained simultaneously with minimal impact on training time, offering a flexible design to meet diverse application requirements in RS scenarios.

To fully leverage existing datasets and enable the detector to robust detect objects in general scenarios, we propose a multi-stage training pipeline. Existing RS datasets are fragmentation and diversity, encompassing numerous specialized datasets, such as HRSC2016 [25] for ship and SpaceNet [37] for building. This heterogeneity poses substantial challenges for jointly trained detectors, as models trained on individual datasets often fail to generalize effectively to other scenarios, leading to severe false positives and false negatives. To address these challenges, we first pretrain the detector using both labeled and unlabeled data to adapt to RS data and the detection task. Subsequently, we finetune detector and employ it to generate pseudo-labels, and leverage SkyCLIP [40] for filtering. Finally, we continue fine-tuning detector using the complete self-annotated dataset integrated with labeled data, endowing it with strong generalization performance.

Our contributions are summarized as follows: (1) We propose an **Open-Prompt Remote Sensing object Detector** (OpenRSD) that supports both oriented and horizontal object detection using either image or text prompts. To balance inference speed and high-precision detection, we introduce an alignment head and a fusion head, each specifically designed to address these distinct requirements. (2) We design a multi-stage training pipeline consists of pre-training, finetuning, and self-training to enhance the detector’s generalization and cross-domain detection capabilities. Furthermore, we construct a large-scale training dataset,

ORSD+, which comprises over 470k images spanning 200 categories. (3) We evaluate OpenRSD on seven public datasets. For oriented object detection, OpenRSD demonstrates superior average performance. For horizontal bounding box detection, OpenRSD achieves performance comparable to the high-precision baseline model, Grounding DINO [24], but with significantly higher inference speed. It runs at an inference speed of 20.8 FPS, making it well-suited for large-scale RS image analysis.

2. Related Work

2.1. Open-Set Object Detection

Open-set object detection models utilize multimodal prompts [16, 24] to detect relevant objects. Early works [12] adhered to strict OVD settings [48], training in base classes and testing in unseen classes, using conventional datasets like COCO [23] for research. Recent studies have attempted to enhance generality by expanding available datasets through multitask learning [24, 46] or self-training [30], and by conducting large-scale image-text alignment pre-training. In terms of model architecture, GLIP [20], Grounding DINO [24], and OV-DINO [39] introduced deep cross-modal fusion, while YOLO-World incorporated lightweight feature interaction modules during multi-scale feature fusion. DetCLIP [46] and OWL-ViT [30] adopted simple feature alignment combined with large-scale pre-training to improve model generalization. Cross-modal fusion helps enhance feature alignment and information interaction, improving detection performance, but increases the difficulty of detecting large vocabularies. To meet the requirements of both efficiency and accuracy in RS scenarios, our proposed OpenRSD integrates both architectures into an efficient detector, enhancing the model’s generality across different scenarios. Additionally, we introduce a multi-prompt design to reduce reliance on expert knowledge in remote sensing.

2.2. Object Detection in Remote Sensing

Real RS scenarios present challenges such as extreme object scales [5], diverse object variations [36], and arbitrary object orientations [42]. Existing works often focus on closed-set object detection, improving structures [2] and loss functions [44] but failing to detect new categories. Recently, some studies have extended detectors to the realm of open vocabulary. DescReg [47] and CastDet [22] use alignment-based approaches to achieve open-vocabulary detection, while OVA-DETR [41] and LAE-DINO [32] introduce deep cross-modal fusion. However, constrained by dataset scale, training methods, or model architecture, these models struggle to handle real RS scenarios. Our proposed OpenRSD improves upon existing state-of-the-art RS object detector [29], making it most suitable for the task. Further-

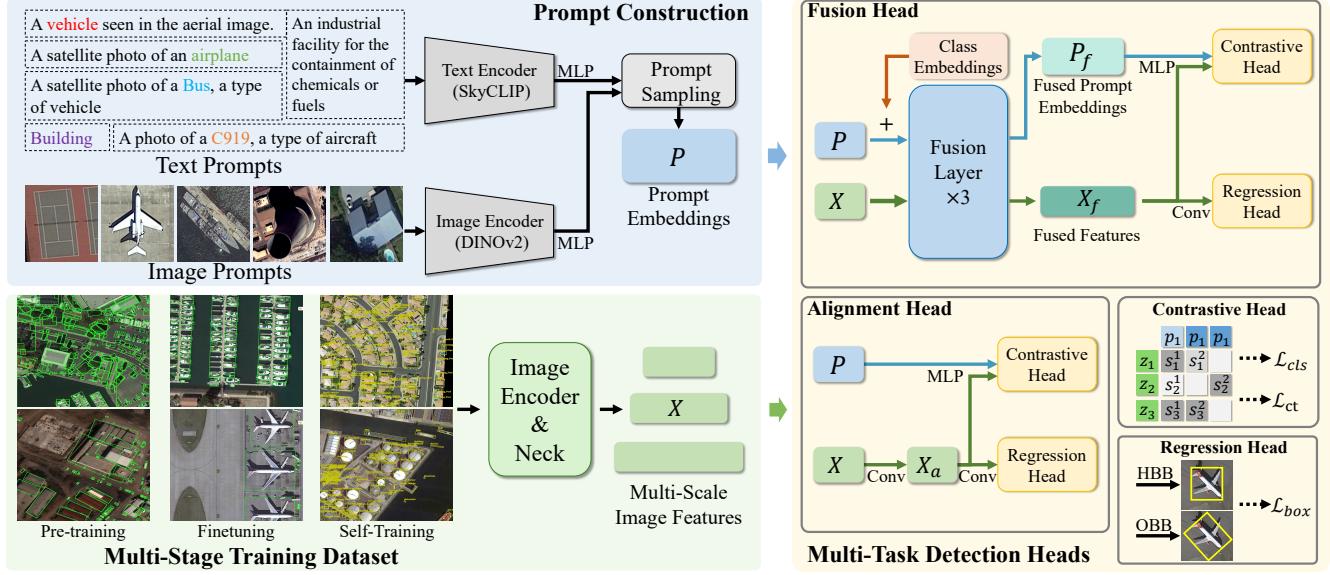


Figure 2. OpenRSD consists of three components: multi-scale image feature encoding, prompt construction, and multi-task detection heads. The prompt construction randomly samples multiple text or image prompts for each category and encodes them into prompt embeddings. The multi-task detection heads leverage the correlation between prompt embeddings and image features to perform detection, seamlessly integrating fusion-based and alignment-based open-prompt structures within one framework. This design enables support for a wide range of prompt-based classification and regression tasks. The alignment head offers higher speed and greater vocabulary scalability, while the fusion head achieves better precision, allowing the model to adapt to different application scenarios. To enhance generalization, we further employ a multi-stage training pipeline.

more, a multi-stage training pipeline is adopted to enhance the model’s generalization and cross-domain capabilities.

3. Method

3.1. Overview

We propose OpenRSD, an RS object detector capable of detecting objects based on either image or text prompts. As illustrated in Fig. 2, the model comprises an image encoder and a neck for extracting multi-scale image features X , a prompt construction module, and two detection heads with multi-task that operate on prompt embeddings P . Specifically, the detection heads include two parallel components: the alignment head and the fusion head, which employ feature alignment and cross-modal fusion, respectively, to effectively utilize prompts for detection. Additionally, we introduce a multi-stage training pipeline to significantly improve the model’s cross-dataset generalization.

3.2. Prompt Construction

OpenRSD supports both text and image prompts, achieves open-prompt object detection. Text prompts enable the detector to perform open-vocabulary object detection, while image prompts allow the detector to perform detection based on images even in the absence of specific category names or descriptions, thereby reducing the need for expert

knowledge, especially for fine-grained categories such as aircraft models. We utilize the text encoder from SkyCLIP [40], pre-trained on large-scale RS image-text pairs, to extract text prompt embeddings, and employ DINOv2 [31] to extract image prompt embeddings. The prompt embeddings are stored offline to reduce training and inference time.

Specifically, text prompts include category names, descriptions, or other textual information related to the categories, as illustrated in Fig. 2. We use GPT-4 [1] to generate 10 to 15 text prompts to simulate potential inputs, enhancing the diversity of text prompts. For image prompts, we enlarge the ground truth bounding boxes in the training set by 1.25 times to include necessary visual context information, then crop and resize them to 224×224 pixels, obtaining a large number of RS object images. Due to the considerable variation in the spatial resolution of RS images, certain objects, such as small vehicles or ships, may appear blurred and lack distinguishable features. To address this issue, we sample images from each dataset to construct a classification dataset, and then train a simple classifier using DINOv2 [31] combined with a two-layer MLP. For each category, we select the top 100 images with the highest classification confidence to form the final image prompt set. The prompt embeddings are mapped into 256-dimensional embedding space through two separate multi-layer perceptrons (MLP) for joint training.

During each iteration, we randomly select one type of prompt for training. We first identify the annotated categories present in the image as positive categories. Simultaneously, we randomly sample negative categories to enhance robustness against irrelevant prompts. OpenRSD supports using multiple prompts for robust detection. For each category, 3 to 7 prompt embeddings are randomly sampled and fed into the detection heads for further processing.

3.3. Alignment Head

Alignment heads are suitable for rapid detection in large vocabulary scenarios. It first uses a convolution layer to obtain its own image features X_a . Then, we employ the decoupled head structure [29] for regression and contrastive classification, where the classification branch outputs the 256-dimensional embeddings. Subsequently, we utilize a shared MLP to map embeddings into the same semantic space, generating output embeddings $Z = \{z_i\}_{i=1}^N \in \mathbb{R}^{N \times 256}$ and prompt embeddings $P = \{p_j\}_{j=1}^M \in \mathbb{R}^{M \times 256}$, where N and M denote the number of embeddings, respectively. Let $L = \{l_j\}_{j=1}^M$ denote the ground truth labels of the prompt embeddings. We compute the maximum normalized inner product between z_i and the intra-class prompt embeddings as the logits for class c , denoted as s_i^c :

$$s_i^c = \max_{j \in \{j|l_j=c\}} \left(\alpha \frac{z_i \cdot p_j}{\|p_j\|} + \beta \right) \quad (1)$$

Here, α and β are adjusted during training to ensure stability [7]. Finally, the logits for each class are aggregated, and the classification loss \mathcal{L}_{cls} [29] is computed.

In addition to the classification loss, a contrastive loss is introduced to further stabilize the feature alignment. Specifically, for each prompt embedding p_j , we select the most similar prediction embedding $z_i^* = \arg \max_{z_i} \frac{p_j \cdot z_i}{\|p_j\| \|z_i\|}$, and apply a supervised contrastive loss [17]:

$$\mathcal{L}_{\text{ct}} = \sum_i -\frac{1}{|P(i)|} \sum_{j \in P(i)} \log \frac{e^{(z_i^* \cdot p_j / \tau)}}{\sum_{k \in A(i)} e^{(z_k^* \cdot p_j / \tau)}} \quad (2)$$

Here, $A(i)$ denotes the set of indices excluding i , $P(i)$ is the subset of $A(i)$ where the predicted and ground truth labels match, and τ is the temperature parameter, set to 0.1. Total loss for the alignment head is:

$$\mathcal{L}_{\text{aln}} = \mathcal{L}_{\text{ct}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{box}} \quad (3)$$

where \mathcal{L}_{box} is the regression loss [29], which includes both horizontal and oriented box regression.

3.4. Fusion Head

The fusion head introduces a lightweight cross-modal fusion to facilitate interaction between prompt embeddings and detector features, enabling improved detection performance. Additionally, we introduce a class embedding to

explicitly differentiate between distinct categories. Specifically, for each image, we gather the corresponding annotated categories and assign a random category ID. This ID is then mapped to a learnable class embedding, which is added to all prompt embeddings of that class. The class embedding enables the fusion head to dynamically manage detection tasks at varying granularities, preventing conflicts between tasks. The fusion module comprises three layers, each defined as follows:

$$P' = \text{LN}(\text{MHCA}(P_i, X_i) + P) \quad (4)$$

$$P_{i+1} = \text{LN}(\text{MLP}(P') + P') \quad (5)$$

$$X' = \text{LN}(\text{MHCA}(X_i, P_{i+1}) + X_i) \quad (6)$$

$$X_{i+1} = \text{LN}(\text{MLP}(X') + X') \quad (7)$$

where MHCA is multi-head cross-attention, LN is layer normalization, and MLP is multi-layer perceptron. The classification, supervised contrastive, and regression losses are computed same to the alignment head, using the fused features X_f and fused prompt embeddings P_f , with their sum denoted as \mathcal{L}_{fus} . The overall loss for OpenRSD is:

$$\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{fus}} + \mathcal{L}_{\text{aln}} \quad (8)$$

3.5. Multi-Stage Training

To ensure the detector to handle diverse prompt inputs and varying RS scenarios, we introduce a multi-stage training pipeline, as illustrated in Fig. 3. In the first stage, we pretrain the detection modules using both labeled and unlabeled RS datasets. For labeled data, we use seven RS datasets covering common categories. Oriented bounding boxes are converted to horizontal ones via minimum enclosing rectangles for horizontal box regression. We use the Million-AID dataset [26] as unlabeled data. Region proposals are generated by SAM [18], cropped, and processed with DINOv2 [31] to extract object embeddings [15]. These embeddings are clustered, and cluster centroids serve as prompt embeddings during training.

In the second stage, we expand to ten datasets with varying category granularity, and apply an empirical sampling rate during training. Although performance improves, the detector still struggles with objects like ships and buildings and fails to generalize to mixed prompts across datasets. These issues primarily because RS datasets often restrict to specific categories, such HRSC2016 [25] and SpaceNet [37] dataset, resulting in significant domain gaps that limit the model's generalization. In addition, the lack of training on mixed prompts causes inconsistency in real-world scenarios. To address these issues, we adopt a simple self-training strategy.

In the third stage, we use the fine-tuned detector to complete the annotations of the labeled dataset, aiming to comprehensively cover all objects within the predefined 200

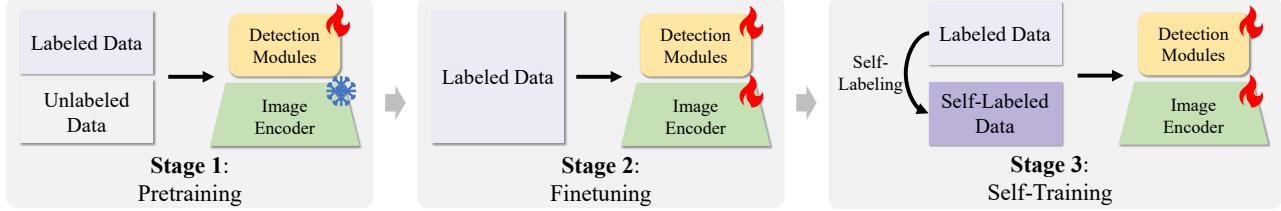


Figure 3. The multi-stage training pipeline includes pretraining, fine-tuning, and self-training. Pretraining trains only the detection modules to adapt to RS detection task. Fine-tuning stage enables the detector to detect arbitrary objects in RS images. Self-training enhances cross-scenario generalization.

categories. Each image is then assigned an independent category list to enrich prompt semantics. First, we perform detection using prompt embeddings constructed from each dataset, setting a score threshold of 0.3 to select high-confidence detection results, and only retain detection boxes un-overlap with the ground truth. For detections with high overlap with ground truth, we check whether the predicted category and the ground truth category belong to the same parent class by a pre-defined category tree [26]. Predicted categories with different parent classes from the ground truth are recorded as hard negative categories and added to the category list. Next, we merge all predictions, and apply class-agnostic NMS to remove duplicate detection results.

For objects with pixels larger than 16×16 , we use SkyCLIP [40] for further filtering. Specifically, we extract the textual features of predicted categories and the visual features of boxes using SkyCLIP’s text encoder and image encoder, respectively, and calculate their cosine similarity. Based on the filtering visualization results, we remove predictions with a similarity score below 0.24. The retained predicted categories are added to the category list of the corresponding image. Finally, we continue training the finetuned model using a combination of the self-labeled dataset and the original labeled dataset. The self-labeled dataset is sampled at half the rate of the original labeled dataset to preserve original capabilities while improving cross-domain performance.

4. Experiments

4.1. Datasets

We use 11 datasets for training and evaluation, which can be categorized into three major groups:

General Datasets. DIOR-R [4] dataset contains 20 categories. The training and validation sets consist of 11,725 images, each with a size of 800×800 pixels. We use the training and validation sets for training and evaluate on the test set. DOTA-v1.0 [42] and DOTA-v2.0 [10] dataset contain 15 and 18 categories, respectively, include diverse RS images with varying scenes. For DOTA-v1.0, we clip images into 800×800 pixels with a stride of 600 [45]. For

DOTA-v2.0, we clip images into $1,024 \times 1,024$ pixels with an overlap of 500 and apply multi-scale augmentation. We train on the training set and evaluate on the validation set.

Specific Datasets. WHU-Mix [28] is a large-scale building detection dataset containing 43,778 images with spatial resolutions ranging from 0.09 to 2.5 meters. We train on the training and validation sets and evaluate on the test set. SpaceNet [37] is a building detection dataset covering multiple cities. We merge four cities to construct a training set of 10,152 images and evaluate on the merged test set. GLH-Bridge [21] is a bridge dataset, in which images are clipped into $1,024 \times 1,024$ pixels with an overlap of 200.

Fine-Grained Datasets FAIR1M-2.0 [36] dataset includes 37 fine-grained categories for aircraft, ships, scenes, and vehicles. We clip images into 800×800 pixel patches with a stride of 400, using validation set for evaluation. Xview [19] dataset involves 60 fine-grained categories. We randomly select 700 images as the training set and 146 images for testing, clip images into 800×800 pixels with an overlap of 200, HRSC2016 [25] and ShipRSImageNet [49] are fine-grained ship detection datasets. We train on the training and validation sets and evaluate on the test set. FMoW [8] dataset contains 62 foreground classes, covering common scenes and fine-grained categories of buildings.

The multi-stage training utilizes these datasets, as detailed in Tab. 2. Among them, ORSD-Pre is used for pre-training, while ORSD+ serves as the fine-tuning dataset. ORSD+ contains a total of 474,058 images across 200 categories.

4.2. Experimental Setup

OpenRSD. We use DINOv2-ViT-L [31] to extract image prompt embeddings and the text encoder of SkyCLIP-ViT-L [40] to extract text prompt embeddings. These embeddings are extracted offline and stored as fixed prompt dictionaries. We implement OpenRSD based on the RTMDet-L [29] in MMrotate [50] framework. The model is trained using the Adam optimizer [27] with an initial learning rate of 2.5×10^{-4} , a weight decay of 5×10^{-2} , and a cosine annealing scheduler that reduces the learning rate to 5% of its initial value mid-training. Training is performed on four 3090Ti

Table 1. Comparison on seven public datasets. The evaluation includes both oriented bounding box (OBB) and horizontal bounding box (HBB) detection, using AP50 as the evaluation metric. Models marked with * are trained using the proposed ORSD+ dataset. Inference speed, measured in FPS, is evaluated on a single 2080Ti.

	Backbone	DIOR-R	DOTA-v1.0	DOTA-v2.0	FAIR-1M-2.0	WHU-Mix	SpaceNet	HRSC2016	Average	FPS
<i>OBB Evaluation</i>										
RetinaNet-OBB [34] _[CVPR17]	ResNet-50	51.7	61.3	60.0	35.9	77.3	43.2	41.2	52.9	20.6
RoI Transformer [9] _[CVPR19]	ResNet-50	61.6	73.1	64.1	43.5	79.6	45.7	51.9	59.9	11.2
S ² ANet [13] _[TGRS21]	ResNet-50	61.3	70.8	63.8	44.0	79.5	45.5	64.2	61.3	16.9
R ³ Det-KFIoU [45] _[ArXiv22]	ResNet-50	57.2	66.6	61.1	41.4	78.5	44.1	50.8	57.1	18.2
Oriented R-CNN [43] _[ICCV21]	ResNet-50	59.7	70.2	63.0	43.2	79.3	44.9	57.2	59.6	7.0
Oriented R-CNN [43] _[ICCV21]	Swin-T	65.8	76.0	67.4	45.0	79.3	44.6	79.1	65.3	6.9
MTP [38] _[JSTARS24]	ViT-B	72.1	76.8	67.8	44.2	80.0	46.7	80.1	66.8	4.7
RTMDet-R-M [29] _[ArXiv22]	RTMDet-M	69.4	74.5	61.4	46.9	79.9	47.3	79.5	65.6	35.7
RTMDet-R-L [29] _[ArXiv22]	RTMDet-L	70.1	71.5	66.7	47.6	79.9	47.4	79.9	66.2	23.8
CastDet* [22] _[ECCV24]	ResNet-50	63.3	72.1	63.1	37.7	77.7	45.0	74.1	61.9	6.8
PKINet* [3] _[CVPR24]	PKINet-T	56.7	64.9	59.6	34.6	62.6	34.6	41.3	50.6	5.2
OpenRSD* (Text Prompt)	RTMDet-L	73.7	76.9	70.1	46.1	79.7	50.2	88.1	69.3	20.8
OpenRSD* (Image Prompt)	RTMDet-L	73.7	76.7	67.8	44.1	79.6	50.2	88.6	68.7	20.8
<i>HBB Evaluation</i>										
YOLO-World-L [7] _[CVPR24]	YOLOv8-L	73.2	70.8	58.0	41.1	82.6	53.7	59.4	62.7	17.2
Grounding DINO-T [24] _[ECCV24]	Swin-T	78.7	78.7	71.8	46.0	85.4	56.5	68.0	69.3	5.4
OpenRSD* (Text Prompt)	RTMDet-L	76.7	77.7	71.8	45.8	85.0	55.4	87.7	71.4	20.8
OpenRSD* (Image Prompt)	RTMDet-L	76.7	77.3	69.8	43.9	84.7	55.4	87.3	70.7	20.8

Table 2. The datasets used in the multi-stage training pipeline. ORSD-Pre dataset is used for pretraining, and ORSD+ dataset is used for finetuning and self-training.

Name	Datasets
ORSD-Pre	DOTA-v2.0 [10], DIOR-R [4], FAIRIM-2.0 [36], SpaceNet [37], Xview [19], HRSC2016 [25], GLH-Bridge [21], Million-AID [26]
ORSD+	DOTA-v2.0 [10], DIOR-R [4], FAIRIM-2.0 [36], SpaceNet [37], SpaceNet [37], Xview [19], HRSC2016 [25], GLH-Bridge [21], fMoW [8], WHU-Mix [28], ShipRSImageNet [49]

GPUs with a batch size of 16. Pretraining involves 360k iterations with input of 896×896 pixels, while fine-tuning and self-training each consist of 288k iterations on images with input of 832×832 pixels.

Compared Methods. We compare our method with SOTA oriented object detection methods, including RetinaNet-OBB [34], RoI-Transformer [9], S²ANet [13], R³Det-KFIoU [45], Oriented R-CNN [43], MTP [38], and RTMDet [29]. These models are trained and tested on their respective datasets. To ensure a fair comparison, we also include OVD methods that could jointly trained using ORSD+, namely CastDet* [22] and PKINet* [3]. We also compare with two powerful OVD methods, Grounding-DINO [24] and YOLO-World [7], in horizontal bounding box detection. Since these two detectors have been sufficiently pre-trained, fine-tuning them on each dataset can demonstrate their optimal performance [32].

The training schedule is adjusted based on the model architectures and datasets: the default schedule consists of 12 epochs, while single-stage detectors, including RetinaNet-OBB, RTMDet, S²ANet, and YOLO-World, follow a 36-epoch schedule. For the HRSC2016 dataset, the schedule is tripled. Both CastDet* and PKINet*, same as OpenRSD,

are trained for 288k iterations. The average precision at IoU 0.5 (AP50) in DOTA [42] is used as the evaluation metric.

4.3. Main Results

Tab. 1 compares the performance of various models across seven datasets. For OpenRSD, we report the results using text and image prompts separately, with the number of prompts set to 7. For oriented object detection (OBB), OpenRSD achieves the best average performance, obtaining either the best or second-best precision across most datasets while maintaining a real-time inference speed of 20.8 FPS. Notably, on widely-used datasets such as DIOR-R, DOTA-v1.0, and DOTA-v2.0, OpenRSD outperforms the baseline RTMDet-R-L by margins of 3.5%, 5.2%, and 3.5%, respectively. OpenRSD also outperform the MTP, which relies on ViT-B [11] large-scale pretraining, across most datasets, while functioning as a prompt-based detector that requires no fine-tuning. It demonstrates significant improvements on the HRSC2016 dataset, excelling in both coarse-grained and fine-grained RS object detection tasks. While CastDet*, which also employs multi-dataset joint training, showing improvement compared to models fine-tuned on individual datasets, its framework design limitations resulted in inferior performance compared to OpenRSD. CastDet*, which also employs multi-dataset joint training, achieves limit improvement compared to baseline Oriented R-CNN, due to structural limitations and training procedures.

For horizontal box detection (HBB), OpenRSD exhibit a substantial accuracy advantage over YOLO-World-L, which also built on the YOLO architecture. Specifically, OpenRSD outperform YOLO-World-L by 4.0%, 6.5%, 13.8%, and 4.6% on DIOR-R, DOTA-v1.0, DOTA-v2.0,

Table 3. Evaluation of cross-dataset performance. All models, except OpenRSD, are fully trained on DOTA-v2.0. Models marked with * are trained using the proposed ORSD+ dataset.

Methods	VEDAI	CORS	DOSR	SODA-A	Average
RetinaNet-OBB [34]	60.9	71.6	76.3	54.5	65.8
RoI Transformer [9]	62.2	72.2	90.1	63.0	71.9
S ² ANet [13]	62.2	78.5	89.7	62.5	73.2
Oriented R-CNN [13]	62.1	72.2	90.1	62.9	71.8
RTMDet-R-M [29]	66.8	78.4	90.4	62.9	74.6
RTMDet-R-L [29]	67.8	78.6	89.9	67.1	75.9
CastDet* [22]	66.3	72.1	90.2	60.1	72.2
PKINet* [3]	64.1	72.1	90.3	60.3	71.7
OpenRSD	69.7	79.8	90.6	72.3	78.1

Table 4. Ablation study on model components.

Alignment Head	Fusion Head	Class Embeddings	DIOR-R	DOTA-v2.0	FAIR1M-2.0
✓			71.5	68.8	45.4
✓	✓		73.1	69.5	45.3
✓	✓	✓	72.9	70.4	45.9
			SpaceNet	HRSC2016	Average
✓			47.8	84.3	63.3
✓	✓		47.8	73.1	61.8
✓	✓	✓	47.5	84.6	64.3

Table 5. Detection performance with different types of head.

Head Type	DIOR-R	DOTA-v1.0	DOTA-v2.0	FAIR1M-2.0
Alignment	71.4	76.4	68.2	45.2
Fusion	72.9	76.6	70.4	45.9
	WHU-Mix	SpaceNet	HRSC2016	Average
Alignment	79.7	47.8	80.5	67.0
Fusion	79.4	47.5	84.6	68.2

and FAIR1M-2.0, respectively, while achieving comparable inference speed. The performance of OpenRSD on most datasets is on par with the high-precision Grounding-DINO-T method. Notably, OpenRSD is four times faster than Grounding-DINO-T, making it more suitable for large-scale RS image analysis.

Tab. 3 evaluates the cross-domain performances of various models. We select four datasets, i.e. VEDAI [33], CORS [35], DOSR [14], and SODA-A [6]. These datasets are excluded from the training process, as test sets and standardized their categories to align with the categories in DOTA-v2.0 [10]. The compared methods are multi-scale trained on DOTA-v2.0, ensuring their generalization on common RS object detection. As shown in Tab. 3, RTMDet demonstrates best average performance, highlighting the importance of model structure. Compared to RTMDet-L baseline, OpenRSD achieves an average improvement of 2.2%, with particularly notable gains of 5.2% in the small object detection dataset SODA-A [6].

4.4. Ablation Studies

We conduct a comprehensive analysis of the modules and training strategy of OpenRSD.

Table 6. Comparison of performance under different prompt types and quantities.

Prompt Type	#Number of Prompts				
	1	3	5	10	20
DOTA-v2.0					
Image Prompt	65.7	67.8	67.8	68.2	68.0
Text Prompt	70.1	70.2	70.1	70.1	70.1
HRSC2016					
Image Prompt	87.9	87.7	88.1	88.1	88.0
Text Prompt	88.1	88.3	88.6	88.0	87.7
FAIRIM-2.0					
Image Prompt	44.0	44.0	44.1	44.1	44.0
Text Prompt	46.0	45.9	46.1	46.1	46.0

Modules. Tab. 4 examines the impact of individual modules. Introducing the alignment head alone achieves near-optimal performance in the building detection (SpaceNet) while remaining compatible with fine-grained detection (FAIR1M-2.0 and HRSC2016). The fusion head improves performance on DIOR-R and DOTA-v2.0 by 1.6% and 0.7%, respectively, though at the cost of reduced fine-grained recognition capability. By incorporating class embeddings, OpenRSD achieves greater training stability and better performance across almost all dataset. Tab. 5 compares detection performance when using different head types for inference after joint training. The fusion head outperform the alignment head by an average margin of 1.2%, demonstrating that deep interactions between image features and prompt embeddings enhance the model’s generalization capability from multi-dataset joint training.

Prompts. Tab. 6 explores the impact of different prompt types and quantities. Unlike few-shot fine-tuning, all parameters remain frozen, with only the number of prompts being adjusted; thus, the improvements are not significant. Compared to image prompts, which require additional semantic extraction, text prompts inherently carry semantic information. It exhibits stronger generalization and greater stability when only use one prompt. Image prompts, on the other hand, mitigate the randomness as the number increases, leading to a gradual performance improvement.

Tab. 7 analyzes the impact of multi-stage training pipeline. During the pretraining stage, we combine labeled and unlabeled data to pretrain the detection module, enabling it to adapt to RS object detection. Although the pre-trained detector exhibits poor detection performance due to the frozen backbone and the presence of noise in the unlabeled data, it is beneficial for enhancing the final performance. Comparing the results in row 2 and 3 of Tab. 7, pretraining provides an average improvement of 1.5%, particularly enhancing semantic discrimination capabilities, with performance gains of 1.9%, 1.3%, and 3.8% on DIOR-R, FAIR1M-2.0, and HRSC2016, respectively. Subsequent self-training using self-labeled dataset further boost detection performance. Experiments reveal that unfreezing the

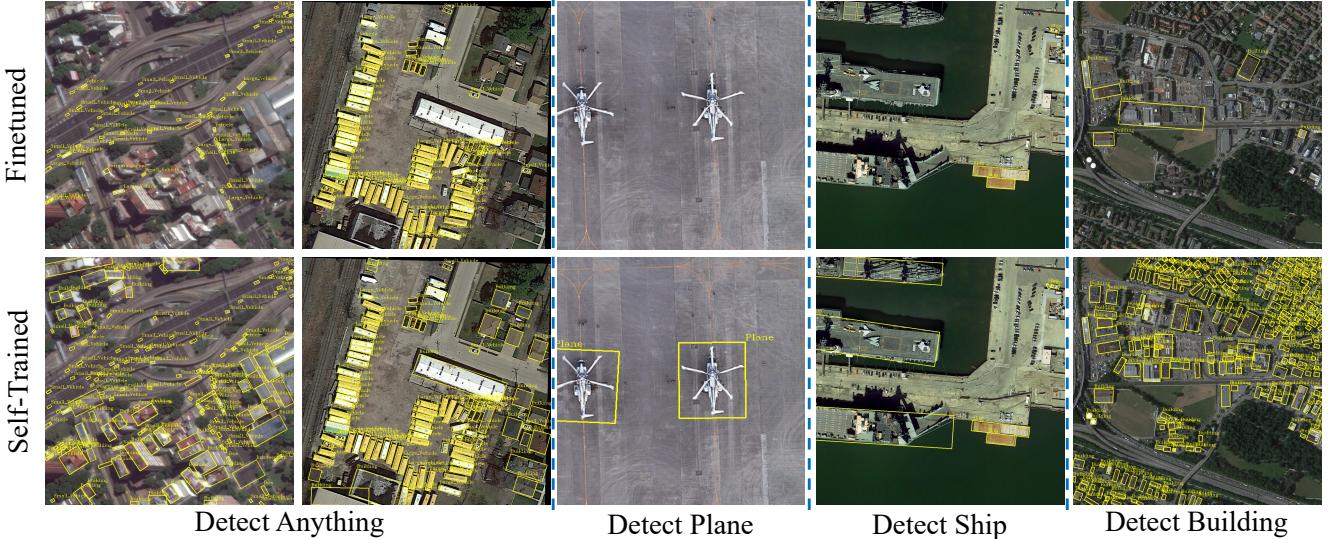


Figure 4. Visualization results on the DOTA-v2.0 [10] validation set before (the top row) and after self-training (the bottom row), demonstrating four prompts: detecting any objects, detecting plane, detecting ship, and detecting buildings.

Table 7. Ablation study of the multi-stage training strategy. For self-training, we compare self-labeling with the labeled dataset, followed by freezing the backbone or full-parameter fine-tuning. We also test self-labeling with the Million-AID [26] dataset and full-parameter fine-tuning.

Pre-training	Fine-tuning	Self-training	DIOR-R	DOTA-v1.0	FAIR-1M-2.0	SpaceNet	HRSC2016	Average
✓			69.2	69.2	21.5	43.3	33.0	47.2
	✓		70.8	76.3	44.6	47.7	80.8	64.0
✓	✓		72.9	76.6	45.9	47.5	84.6	65.5
✓	✓	Freezing Backbone	73.7	75.6	46.0	48.0	85.0	65.7
✓	✓	Full-Parameters	73.6	76.7	46.0	50.3	87.8	66.9
✓	✓	Using Million-AID	73.2	76.0	45.7	47.8	83.6	65.3

backbone for full-parameter fine-tuning led to more significant improvements, with an overall gain of 1.4% compared to the stage before self-training. Additionally, we test whether incorporating an external unlabeled dataset for self-training leads to further improvements. Specifically, we select 250k images from the Million-AID [26] with resolutions exceeding 512×512 pixels for self-training, as shown in the last row of the Tab. 7. Results indicate that more data dose not lead to further performance gains, as the scale of the existing labeled datasets was already sufficient for achieving robust RS object detection.

Fig. 4 compares the visualization results before and after self-training for four different prompts: detecting any objects, detecting planes, detecting ships, and detecting buildings. To achieve detection of any objects, we use the alignment head to identify all potential categories, and then use the fusion head for further detection. After self-training, the model exhibits fewer false positives, higher recall, and more precise localization. Moreover, self-training effectively improves the detection performance of planes, ships, and buildings across different scenarios.

5. Conclusion

In this paper, we propose OpenRSD, an open-prompt RS object detection method. By incorporating multi-task detection heads and a multi-stage training pipeline, OpenRSD achieves robust generalization across diverse datasets and scenarios. The integration of alignment and fusion heads allows for flexible and efficient detection, balancing speed and precision. Extensive evaluations on seven public datasets demonstrate that OpenRSD outperforms state-of-the-art models in both oriented and horizontal detection, while maintaining real-time inference.

Acknowledgment

This work was supported by the National Key Research and Development Program of China under Grant 2023YFB3906102, the Emergency Management Research and Development Project of Zhejiang Province (No. 2024YJ018), the National Natural Science Foundation of China under Grant 62176017, and the Fundamental Research Funds for the Central Universities.

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 3
- [2] Xinhao Cai, Qiuxia Lai, Yuwei Wang, Wenguan Wang, Zeren Sun, and Yazhou Yao. Poly kernel inception network for remote sensing detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27706–27716, 2024. 2
- [3] Xinhao Cai, Qiuxia Lai, Yuwei Wang, Wenguan Wang, Zeren Sun, and Yazhou Yao. Poly kernel inception network for remote sensing detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27706–27716, 2024. 6, 7
- [4] Gong Cheng, Jiabao Wang, Ke Li, Xingxing Xie, Chunbo Lang, Yanqing Yao, and Junwei Han. Anchor-free oriented proposal generator for object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2022. 5, 6
- [5] Gong Cheng, Xiang Yuan, Xiwen Yao, Kebing Yan, Qinghua Zeng, Xingxing Xie, and Junwei Han. Towards large-scale small object detection: Survey and benchmarks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2
- [6] Gong Cheng, Xiang Yuan, Xiwen Yao, Kebing Yan, Qinghua Zeng, Xingxing Xie, and Junwei Han. Towards large-scale small object detection: Survey and benchmarks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):13467–13488, 2023. 7
- [7] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1, 4, 6
- [8] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6172–6180, 2018. 5, 6
- [9] Jian Ding, Nan Xue, Yang Long, Gui-Song Xia, and Qikai Lu. Learning roi transformer for oriented object detection in aerial images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2849–2858, 2019. 6, 7
- [10] Jian Ding, Nan Xue, Gui-Song Xia, Xiang Bai, Wen Yang, Michael Yang, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Object detection in aerial images: A large-scale benchmark and challenges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. 5, 6, 7, 8
- [11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 6
- [12] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *International Conference on Learning Representations*, 2022. 2
- [13] Jiaming Han, Jian Ding, Jie Li, and Gui-Song Xia. Align deep features for oriented object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2021. 6, 7
- [14] Yaqi Han, Xinyi Yang, Tian Pu, and Zhenming Peng. Fine-grained recognition for oriented ship against complex scenes in optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–18, 2021. 7
- [15] Ziyue Huang, Yongchao Feng, Qingjie Liu, and Yunhong Wang. Mutdet: Mutually optimizing pre-training for remote sensing object detection. In *European Conference on Computer Vision*, pages 1–17. Springer, 2025. 4
- [16] Qing Jiang, Feng Li, Tianhe Ren, Shilong Liu, Zhaoyang Zeng, Kent Yu, and Lei Zhang. T-rex: Counting by visual prompting. *arXiv preprint arXiv:2311.13596*, 2023. 2
- [17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020. 4
- [18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 4
- [19] Darius Lam, Richard Kuzma, Kevin McGee, Samuel Doolley, Michael Laielli, Matthew Klaric, Yaroslav Bulatov, and Brendan McCord. xview: Objects in context in overhead imagery. *arXiv preprint arXiv:1802.07856*, 2018. 5, 6
- [20] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022. 1, 2
- [21] Yansheng Li, Junwei Luo, Yongjun Zhang, Yihua Tan, Jin-Gang Yu, and Song Bai. Learning to holistically detect bridges from large-size vhr remote sensing imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7778–7796, 2024. 5, 6
- [22] Yan Li, Weiwei Guo, Xue Yang, Ning Liao, Dunyun He, Jiaqi Zhou, and Wenxian Yu. Toward open vocabulary aerial object detection with clip-activated student-teacher learning. In *European Conference on Computer Vision*, pages 431–448. Springer, 2025. 1, 2, 6, 7
- [23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 2
- [24] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang,

- Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024. 1, 2, 6
- [25] Zikun Liu, Hongzhen Wang, Lubin Weng, and Yiping Yang. Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds. *IEEE geoscience and remote sensing letters*, 13(8):1074–1078, 2016. 2, 4, 5, 6
- [26] Yang Long, Gui-Song Xia, Shengyang Li, Wen Yang, Michael Ying Yang, Xiao Xiang Zhu, Liangpei Zhang, and Deren Li. On creating benchmark dataset for aerial image interpretation: Reviews, guidances, and million-aid. *IEEE Journal of selected topics in applied earth observations and remote sensing*, 14:4205–4230, 2021. 4, 5, 6, 8
- [27] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 5
- [28] Muying Luo, Shunping Ji, and Shiqing Wei. A diverse large-scale building dataset and a novel plug-and-play domain generalization method for building extraction. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16:4122–4138, 2023. 5, 6
- [29] Chengqi Lyu, Wenwei Zhang, Haian Huang, Yue Zhou, Yudong Wang, Yanyi Liu, Shilong Zhang, and Kai Chen. Rtmddet: An empirical study of designing real-time object detectors. *arXiv preprint arXiv:2212.07784*, 2022. 1, 2, 4, 5, 6, 7
- [30] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 2
- [31] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 3, 4, 5
- [32] Jiancheng Pan, Yanxing Liu, Yuqian Fu, Muyuan Ma, Jiaohao Li, Danda Pani Paudel, Luc Van Gool, and Xiaomeng Huang. Locate anything on earth: Advancing open-vocabulary object detection for remote sensing community. *arXiv preprint arXiv:2408.09110*, 2024. 1, 2, 6
- [33] Sébastien Razakarivony and Frédéric Jurie. Vehicle detection in aerial imagery: A small target detection benchmark. *Journal of Visual Communication and Image Representation*, 34:187–203, 2016. 7
- [34] T-YLPG Ross and GKHP Dollár. Focal loss for dense object detection. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2980–2988, 2017. 6, 7
- [35] Tianjun Shi, Jinnan Gong, Shikai Jiang, Xiyang Zhi, Guangzhen Bao, Yu Sun, and Wei Zhang. Complex optical remote-sensing aircraft detection dataset and benchmark. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–9, 2023. 7
- [36] Xian Sun, Peijin Wang, Zhiyuan Yan, Feng Xu, Ruiping Wang, Wenhui Diao, Jin Chen, Jihao Li, Yingchao Feng, Tao Xu, et al. Fair1m: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 184:116–130, 2022. 2, 5, 6
- [37] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232*, 2018. 2, 4, 5, 6
- [38] Di Wang, Jing Zhang, Minqiang Xu, Lin Liu, Dongsheng Wang, Erzhong Gao, Chengxi Han, Haonan Guo, Bo Du, Dacheng Tao, et al. Mtp: Advancing remote sensing foundation model via multi-task pretraining. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2024. 6
- [39] Hao Wang, Pengzhen Ren, Zequn Jie, Xiao Dong, Chengjian Feng, Yinlong Qian, Lin Ma, Dongmei Jiang, Yaowei Wang, Xiangyuan Lan, et al. Ov-dino: Unified open-vocabulary detection with language-aware selective fusion. *arXiv preprint arXiv:2407.07844*, 2024. 2
- [40] Zhecheng Wang, Rajarie Prabha, Tianyuan Huang, Jiajun Wu, and Ram Rajagopal. Skyscript: A large and semantically diverse vision-language dataset for remote sensing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5805–5813, 2024. 2, 3, 5
- [41] Guotong Wei, Xia Yuan, Yu Liu, Zhenhao Shang, Kelu Yao, Chao Li, Qingsen Yan, Chunxia Zhao, Haokui Zhang, and Rong Xiao. Ova-detr: Open vocabulary aerial object detection using image-text alignment and fusion. *arXiv preprint arXiv:2408.12246*, 2024. 1, 2
- [42] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Beßongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3974–3983, 2018. 2, 5, 6
- [43] Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. Oriented r-cnn for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3520–3529, 2021. 1, 6
- [44] Hang Xu, Xinyuan Liu, Haonan Xu, Yike Ma, Zunjie Zhu, Chenggang Yan, and Feng Dai. Rethinking boundary discontinuity problem for oriented object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17406–17415, 2024. 2
- [45] Xue Yang, Yue Zhou, Gefan Zhang, Jirui Yang, Wentao Wang, Junchi Yan, Xiaopeng Zhang, and Qi Tian. The kfiou loss for rotated object detection. *arXiv preprint arXiv:2201.12558*, 2022. 1, 5, 6
- [46] Lewei Yao, Renjie Pi, Jianhua Han, Xiaodan Liang, Hang Xu, Wei Zhang, Zhenguo Li, and Dan Xu. Detclipv3: Towards versatile generative open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27391–27401, 2024. 2
- [47] Zhengqing Zang, Chenyu Lin, Chenwei Tang, Tao Wang, and Jiancheng Lv. Zero-shot aerial object detection with visual description regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6926–6934, 2024. 2

- [48] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14393–14402, 2021. [2](#)
- [49] Zhengning Zhang, Lin Zhang, Yue Wang, Pengming Feng, and Ran He. Shipsimagenet: A large-scale fine-grained dataset for ship detection in high-resolution optical remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:8458–8472, 2021. [5](#), [6](#)
- [50] Yue Zhou, Xue Yang, Gefan Zhang, Jiabao Wang, Yanyi Liu, Liping Hou, Xue Jiang, Xingzhao Liu, Junchi Yan, Chengqi Lyu, et al. Mmrotate: A rotated object detection benchmark using pytorch. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 7331–7334, 2022. [5](#)