

# Exploring Efficient Open-Vocabulary Segmentation in the Remote Sensing

Bingyu Li<sup>1,3\*</sup>, Haocheng Dong<sup>1,3</sup>, Da Zhang<sup>2,3</sup>, Zhiyuan Zhao<sup>3</sup>, Junyu Gao<sup>2,3</sup>, Xuelong Li<sup>3†</sup>

<sup>1</sup>Department of Electronic Engineering and Information Science, University of Science and Technology of China, China

<sup>2</sup>School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, China

<sup>3</sup>Institute of Artificial Intelligence (TeleAI), China

libingyu0205@mail.ustc.edu.cn, haocheng-dong@mail.ustc.edu.cn,

dazhang@mail.nwp.edu.cn, tuzixini@163.com, gjy3035@gmail.com, xuelong\_li@ieee.org

Open-Vocabulary Remote Sensing Image Segmentation (OVRSIS), an emerging task that adapts Open-Vocabulary Segmentation (OVS) to the remote sensing (RS) domain, remains underexplored due to the absence of a unified evaluation benchmark and the domain gap between natural and RS images. To bridge these gaps, we first establish a standardized OVRSIS benchmark (**OVRSISBench**) based on widely-used RS segmentation datasets, enabling consistent evaluation across methods. Using this benchmark, we comprehensively evaluate several representative OVS/OVRSIS models and reveal their limitations when directly applied to remote sensing scenarios. Building on these insights, we propose **RSKT-Seg**, a novel open-vocabulary segmentation framework tailored for remote sensing. RSKT-Seg integrates three key components: (1) a Multi-Directional Cost Map Aggregation (RS-CMA) module that captures rotation-invariant visual cues by computing vision-language cosine similarities across multiple directions; (2) an Efficient Cost Map Fusion (RS-Fusion) transformer, which jointly models spatial and semantic dependencies with a lightweight dimensionality reduction strategy; and (3) a Remote Sensing Knowledge Transfer (RS-Transfer) module that injects pre-trained knowledge and facilitates domain adaptation via enhanced upsampling. Extensive experiments on the benchmark show that RSKT-Seg consistently outperforms strong OVS baselines by +3.8 mIoU and +5.9 mACC, while achieving 2x faster inference through efficient aggregation. Our code is [here](#).

## Introduction

Semantic segmentation, a classic task in computer vision, aims to achieve pixel-level category prediction (Chen et al. 2017). Traditional semantic segmentation models are based on manually annotated datasets, which usually cover only a limited number of categories. Similarly, traditional remote sensing image segmentation has long focused on a fixed set of predefined categories (Kotaridis and Lazaridou 2021; Diakogiannis et al. 2020). However, as application demands grow, especially in scenarios like tracking new urban infrastructure or identifying rare geological features, the shortcomings of this limited-category-based approach have be-

come evident. This has spurred the development of open-vocabulary remote sensing image segmentation (OVRSIS) (Ye, Zhuge, and Zhang 2025).

OVRSIS, builded on the general OVS concept(Li et al. 2025; Cho et al. 2024; Xie et al. 2024b; Xu et al. 2023), is tailored to the unique characteristics of remote sensing imagery. By leveraging the power of VLMs and cross-modal learning (Radford et al. 2021; Jia et al. 2021), OVRSIS breaks the constraints of traditional training categories, enabling the segmentation of novel classes not present in the original training data. This significantly improves the adaptability and generalization ability of remote sensing image segmentation.

However, unlike the recent advances in OVS (Xu et al. 2023; Radford et al. 2021; Jia et al. 2021), its extension to the remote sensing (RS) domain (OVRSIS) remains largely underexplored. A critical bottleneck hindering progress in this emerging field is the absence of a standardized evaluation benchmark. Most prior works assess their models on limited datasets or under inconsistent experimental setups, making it difficult to draw fair comparisons or systematically analyze model behavior in remote sensing scenarios. To address this issue, we construct a unified benchmark for OVRSIS, named **OVRSISBench**, by reformulating several widely-used remote sensing segmentation datasets under open-vocabulary settings. By leveraging established RS datasets with open-vocabulary configurations, OVRSIS-Bench retains the domain-specific challenges of traditional RS tasks while incorporating the flexibility and generalization demands of open-vocabulary segmentation. This benchmark facilitates fair, consistent, and scalable evaluation of OVS models in realistic RS environments.

Using this **OVRSISBench** benchmark, we comprehensively evaluate several representative OVS models and observe a significant drop in performance when these models are directly transferred from natural images to the RS domain (see Fig. fig. 1(a-b)). Furthermore, we survey recent works on OVRSIS model and evaluate them under the proposed **OVRSISBench**. Our experiments reveal that while some approaches demonstrate incremental improvements compared with OVS model, they often fail to achieve effective knowledge transfer to the RS domain. For example, OVRS (Cao et al. 2024), which introduces only minor modifications to OVS model (Cho et al. 2024), shows per-

\*Work done during an internship at TeleAI

†Corresponding author

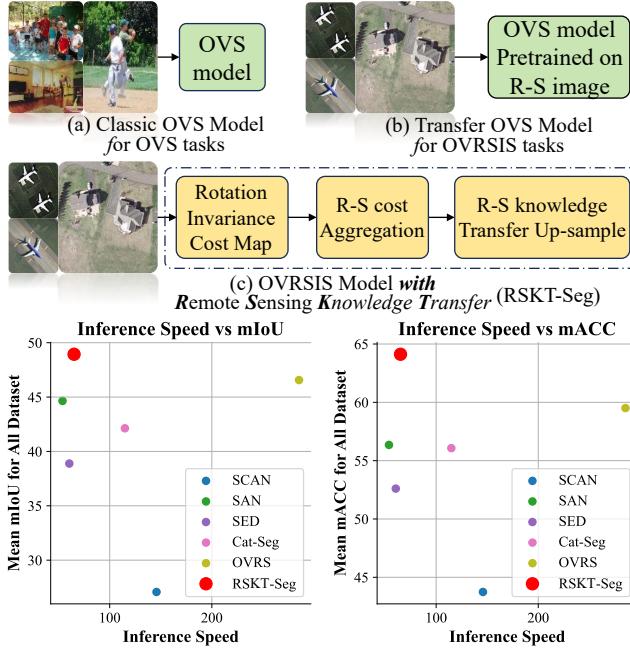


Figure 1: (a-c): Comparison of RSKT-Seg with classic OVS and OVRSIS model. (d): Comparison of RSKT-Seg with different models in terms of inference speed against mean Intersection over Union (mIoU) on the left and against mean Accuracy (mACC) on the right.

formance gains but lacks mechanisms to incorporate RS-specific priors, leading to limited adaptability. Similarly, models such as GSNet (Ye, Zhuge, and Zhang 2025), although partially incorporating RS knowledge, overlook key characteristics of RS imagery such as large-scale context, object rotation, and spectral diversity.

Overall, these findings highlight two key limitations of existing OVS and OVRSIS approaches: (1) classic OVS models exhibit limited transferability when applied directly to open-vocabulary tasks in the remote sensing domain; (2) existing OVRSIS methods fail to adequately model remote sensing-specific characteristics such as rotation invariance and large-scale spatial context.

To fill these gaps, we propose **RSKT-Seg**, an efficient framework for open-vocabulary segmentation in remote sensing that achieves both high segmentation accuracy and fast inference speed (see Fig. fig. 1(d)). First, we introduce a *Multi-Direction Remote Sensing Cost Map Aggregation* (RS-CMA) module to capture the rotation-invariant characteristics of RS images by computing vision-text similarities from multiple directions. Second, we design an efficient *Cost Map Fusion* (RS-Fusion) strategy that simultaneously considers spatial and class-wise interactions, while incorporating a dimensionality reduction mechanism to accelerate inference without sacrificing performance. Finally, we propose a *Remote Sensing Knowledge Transfer* (RS-Transfer) upsampling module that leverages pre-trained model knowledge to facilitate effective domain adaptation to RS imagery. Our main contributions are summarized as follows:

- **Benchmark.** We construct **OVRSISBench**, an unified

benchmark for open-vocabulary remote sensing image segmentation.

- **Evaluation.** Based on OVRSISBench, we comprehensively evaluate representative OVS models and recent OVRSIS methods. Our analysis reveals their potential limitations and provides a standardized reference for future research.
- **Framework.** We propose **RSKT-Seg**, an efficient and effective framework for open-vocabulary segmentation in remote sensing, which achieves both high segmentation accuracy and fast inference speed (shown in fig. 1(d)).

## Related Works

### Semantic Segmentation

Semantic segmentation, a crucial task in computer vision, focuses on pixel-level classification. Over the years, it has seen significant progress with various methods and models. Fully convolutional networks (FCNs) (Long, Shelhamer, and Darrell 2015) were an early milestone. As end-to-end models, they enabled direct pixel-wise predictions. Later, SegNet (Badrinarayanan, Kendall, and Cipolla 2017) and U-Net (Ronneberger, Fischer, and Brox 2015) evolved from FCNs. These encoder-decoder architectures effectively captured low-level and high-level features, gaining popularity. ResNet is widely used as a feature encoder in semantic segmentation, forming the basis of many models (Lin et al. 2018). Recently, Vision Transformers (ViT) (Dosovitskiy 2020) have brought new ideas. By using self-attention, ViT models capture long-range dependencies. Adaptations like SETR (Zheng et al. 2021) improve efficiency. New paradigms based on ViT have emerged for semantic segmentation (Li et al. 2024b; Hu et al. 2024), which requires precise pixel-level feature handling (Xie et al. 2024a; Zhu et al. 2024; Li et al. 2024a). Notable works such as MaskFormer (Cheng, Schwing, and Kirillov 2021) and Mask2Former (Cheng et al. 2022) unify pixel-level and mask classification, enhancing performance. SegFormer addresses resolution issues by removing positional encodings and presenting multi-resolution features (Xie et al. 2021). We summarize works about OVS in appendix A.

### Open-Vocabulary Remote Sensing Image Segmentation

OVRSIS is an emerging field in a rapid development stage, and preliminary exploratory research work has been carried out. In the process of migrating from the OVS field to the remote sensing field, OVRS (Cao et al. 2024) takes into account the basic characteristics of the remote sensing field, makes some fundamental improvements on cat-seg, and has achieved certain results. GSNet (Ye, Zhuge, and Zhang 2025) introduces a specialist-generalist model, which can better integrate remote sensing domain knowledge into the model. SegEarth-OV (Li et al. 2024c) proposes a train-free model, which shows better performance than the OVS model on 17 datasets. Overall, as an emerging field, in OVRSIS, few existing models can consider remote sensing features and introduce remote sensing domain knowledge for migration. Most of them only modify and improve the OVS

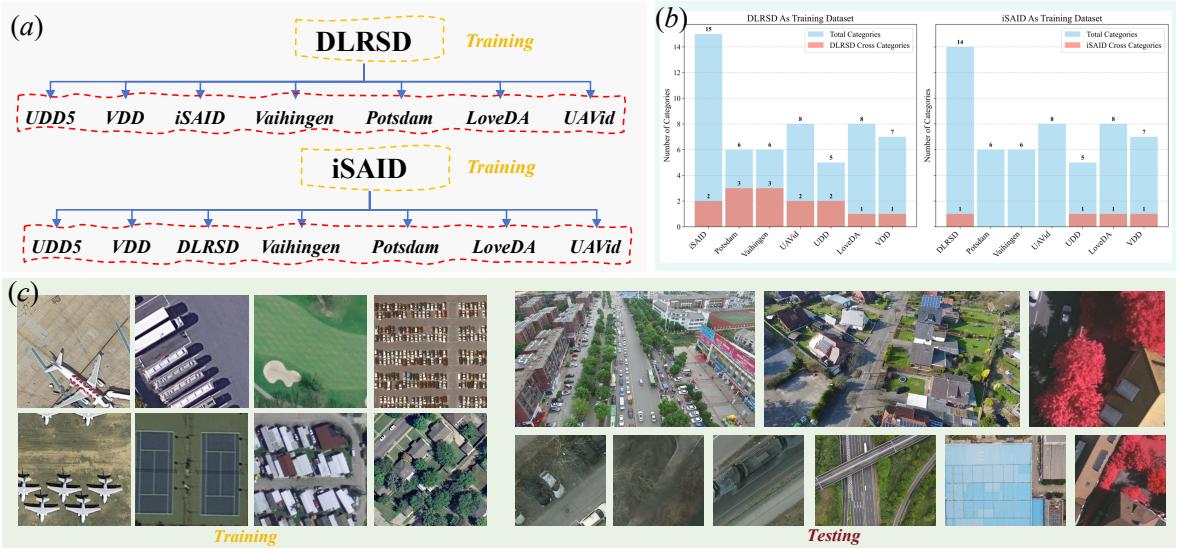


Figure 2: **Schematic diagram of OVRSISBench** (a) Dataset division based on the open-vocabulary protocol (b) Vocabulary (class) overlap number between training and test datasets under two division scenarios (c) Examples display of training and test sets. The more information is in the appendix.

model. In view of this, we propose the RSKT-Seg model. By considering the rotational invariance of remote sensing images and introducing remote sensing domain knowledge, this model has obtained satisfactory performance on multiple datasets.

### Benchmark: OVRSISBench

To promote standardized evaluation in the emerging field of open-vocabulary remote sensing image segmentation (OVRSIS), we construct **OVRSISBench**, a unified and scalable benchmark that reformulates several widely-used remote sensing datasets under an open-vocabulary setting.

### Benchmark Construction

**Dataset Construction** **OVRSISBench** follows an open-vocabulary paradigm like (Cao et al. 2024), where the model must generalize to previously unseen classes based on text descriptions (fig. 2(a)). To construct this benchmark, we adapt 8 representative remote sensing datasets: DLRSD (Chaudhuri et al. 2017), iSAID (Yao et al. 2021), Potsdam, Vaihingen, UAVid(Lyu et al. 2020), UDD5(Chen et al. 2018), LoveDA(Wang et al. 2021) and VDD(Cai et al. 2025). These datasets cover a diverse range of scenes, including urban layouts, agricultural regions, and high-resolution aerial imagery, the examples of the datasets are illustrated in the fig. 2(c). We introduce the datasets in the appendix C.

**Dataset Usage and Splits** As shown in fig. 2(a), we utilize DLRSD and iSAID as training sets as (Cao et al. 2024) due to their large scale and diversity, enabling robust learning of remote sensing visual patterns. For evaluation, we test across all 8 datasets to ensure the model’s generalization under varying scene distributions and resolutions. The training and evaluation splits are consistent across all methods to ensure fair comparison.

### Open-Vocabulary Protocol Analysis

To rigorously assess the open-vocabulary generalization capability of the models, particularly its performance on unseen classes. We adopt a cross-dataset transfer protocol in our **OVRSISBench**. In this setting, the models are trained on the DLRSD and iSAID datasets and evaluated on the separate target datasets with partially disjoint category sets. This protocol aligns with the standard open-vocabulary segmentation (OVS) setting(Xu et al. 2022; Cho et al. 2024). As shown in fig. 2(b), we further perform a statistical analysis of the cross-category distribution to substantiate the open-vocabulary nature of our setup.

### Evaluation on OVRSISBench

**Model Selection and Experiments** We first evaluated the performance of several classic OVS model on **OVRSISBench**, and subsequently selected recently published OVRSIS methods for evaluation. For the evaluation metrics, we adopted mean Intersection over Union (mIoU) and mean Accuracy (mACC). We introduce the Evaluation metric in the appendix D. We utilized the pre-trained ViT-B/16 (Dosovitskiy 2020) and ViT-L/14@336 (Radford et al. 2021) as the vision-language models.

**Analysis of Existing Model Limitations** Table 1 shows classic OVS methods and existing OVRSIS methods perform poorly overall, with clear room for improvement. Classic OVS methods, designed for natural scenes, lag behind dedicated OVRSIS methods under the same backbone. In contrast, although existing OVRSIS methods perform better than classic OVS methods, they still have limitations in fully exploiting remote sensing domain knowledge.

### Method: RSKT-Seg

Although previous OVRSIS models have achieved certain advancements, as we analyzed in the introduction section,

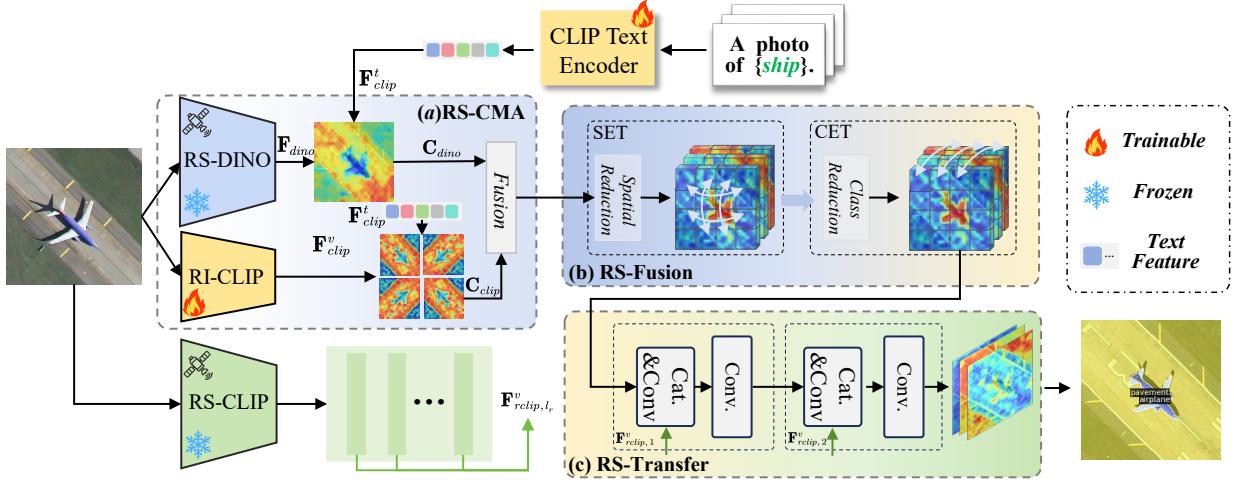


Figure 3: **The overall framework of RSKT-Seg** includes: (a) the overall procedure of RS-CMA module; (b) the workflow of the RS-Fusion Module; (c) the framework of the RS-Transfer Upsample. The more detailed framework is in appendix J

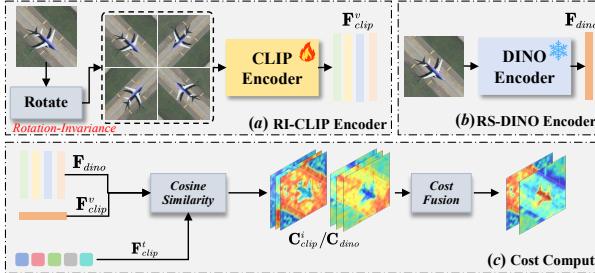


Figure 4: (a) Multi-rotation feature encoding using CLIP (b) feature encoding using RS-DINO and (c) cost map construction using CLIP and DINO. These modules do not introduce learnable parameters.

we attribute the low accuracy observed in the tests on OVR-SISBench to the lack of effective transfer of remote sensing domain knowledge. Therefore, we have designed the **RSKT-Seg** with remote sensing domain knowledge transfer capabilities.

## Overall Architecture of RSKT-Seg

The overall framework of our proposed RSKT-Seg is shown in Figure fig. 3(a). We design the OVR-SIS architecture with remote sensing knowledge transfer. The preparatory knowledge for training pre-trained VLMs such as CLIP has been extensively introduced in previous methods (Hu et al. 2024). The definition of OVR-SIS is the same as that of OVS. We will not elaborate on it here, and researchers can refer to the corresponding sections of relevant literature (Li et al. 2025; Cho et al. 2024; Xu et al. 2023).

### Remote Sensing Cost Map Aggregation (RS-CMA)

As shown in fig. 2(c), unlike natural images, remote sensing (RS) imagery often captures scenes from top-down perspectives, where object orientation is arbitrary due to aerial rotation or satellite orbit variations. Consequently, the same semantic class (e.g., “bridge”, “airplane”) may appear under drastically different orientations across samples. This

intrinsic variability poses a significant challenge for open-vocabulary recognition. To address this, we explicitly introduce **rotation-invariance** through our *Multi-Direction Remote Sensing Cost Map Aggregation (RS-CMA)* module.

Specifically, we augment the input image  $\mathbf{I}$  with the height  $H$  and width  $W$  in four directions, obtaining  $\mathbf{I}_i$  ( $i = 0, 1, 2, 3$ ) with the same shape (shown in fig. 4(a)). Among them,  $\mathbf{I}_0$  is the original orientation, and 1, 2, 3 respectively represent the enhanced images in the orientations rotated by  $90 \times i$ .

As fig. 4(a-b), we use the CLIP image encoder  $\mathcal{E}_{clip}^v$  to encode  $\mathbf{I}_i$ , yielding  $\mathbf{F}_{clip}^{v,i} = \mathcal{E}_{clip}^v(\mathbf{I}_i)$ . To embed remote-sensing domain knowledge, we employ a pre-trained DINO encoder  $\mathcal{E}_{dino}$  on a large-scale dataset (Ye, Zhuge, and Zhang 2025). Encoding  $\mathbf{I}_0$  with  $\mathcal{E}_{dino}$ , we get  $\mathbf{F}_{dino} = \mathcal{E}_{dino}(\mathbf{I}_0)$ .

Given a set of classes  $\mathcal{C} \in \mathbb{R}^{N_t}$ , we obtain text embeddings  $\mathbf{F}_{clip}^t = \mathcal{E}_{clip}^t(\mathcal{C}) \in \mathbb{R}^{N_t \times C_f}$ . Then, for each  $\mathbf{I}_i$ , we compute a cost volume  $\mathbf{C}_{clip}^i \in \mathbb{R}^{(H_f \times W_f) \times N_t}$  using cosine similarity:

$$\mathbf{C}_{clip}^i(j, n) = \frac{\mathbf{F}_{clip}^{v,i}(j) \cdot \mathbf{F}_{clip}^t(n)}{\|\mathbf{F}_{clip}^{v,i}(j)\| \|\mathbf{F}_{clip}^t(n)\|} \quad (1)$$

where  $j$  represents 2D spatial positions of the image embedding and  $n$  is an index for a class.

Similarly, for the DINO-based features, we compute a cost volume  $\mathbf{C}_{dino} \in \mathbb{R}^{(H_f \times W_f) \times N_t}$  for  $\mathbf{I}$  as:

$$\mathbf{C}_{dino}(j, n) = \frac{\mathbf{F}_{dino}(j) \cdot \mathbf{F}_{clip}^t(n)}{\|\mathbf{F}_{dino}(j)\| \|\mathbf{F}_{clip}^t(n)\|} \quad (2)$$

Next, we apply a cost map fusion function across the four rotated variants and the DINO-based cost map to generate a rotation-invariant and domain-aware fused cost map (as shown in fig. 4(c)):

$$\mathbf{C}_{fused} = \text{Fusion}(\mathbf{C}_{clip}^i, \mathbf{C}_{dino}), \quad i = 0, 1, 2, 3 \quad (3)$$

The fused cost map is further organized under multiple prompt templates, forming a tensor of shape  $H_f \times W_f \times$

$N_t \times P$ , where  $P$  denotes the number of templates.  $\mathbf{C}_{fused}$  is projected into  $\mathbf{C}_s$  with the shape  $H_f \times W_f \times N_t \times C_f$  by a linear layer.

### Efficient Cost Map Fusion (RS-Fusion)

The cost map  $\mathbf{C}_s$  is used to characterize the correlation between visual features and text features. Building upon this, we gradually predict the segmentation map by enhancing the cost map's **spatial discriminative ability** ( $H_f W_f$  dimension) and **class discriminative ability** ( $N_t$  dimension). To achieve this, we have designed two dedicated modules: the Spatial Enhancement Transformer (SET) and the Class Enhancement Transformer (CET), as illustrated in fig. 3(b) and (c). This idea aligns with that presented in (Cho et al. 2024). To further strengthen the integration of remote sensing domain knowledge, we embed knowledge from the pre-trained DINO model together with the cost map. Additionally, considering the requirement for faster inference speed, we have developed two feature dimension reduction methods and a lightweight transformer architecture.

**Spatial Enhancement Transformer(SET)** To enhance the semantic representation with richer spatial information, we concatenate the cost map with intermediate-level features from both CLIP and DINO. These features are first projected to a unified dimension and then passed through a convolutional layer  $\mathcal{R}_s$  that reduces the spatial resolution, yielding a compressed tensor for fast inference (as shown in fig. 3(b)):

$$\mathbf{C}_{sr} = \mathcal{R}_s([\mathbf{C}_s; \mathbf{F}_{clip}^v; \mathbf{F}_{dino}]) \quad (4)$$

Here,  $\mathcal{R}_s$  denotes the spatial reduction module and  $[.;.;.]$  is concatenate operation. Then the reduced feature  $\mathbf{C}_{sr}$  is used as key and value in a Transformer block designed to aggregate spatial context via cross-attention manner:

$$\mathbf{C}_{so} = \text{Transformer}_s(\mathbf{C}_s, \mathbf{C}_{sr}, \mathbf{C}_{sr}) \quad (5)$$

**Class Enhancement Transformer(CET)** As shown in fig. 3(b), we further refine the fused features  $\mathbf{C}_{so}$  in the class dimension by integrating the text features  $\mathbf{F}_{clip}^t$  from CLIP. After applying average pooling  $\text{Avg}(\cdot)$  to downsample the spatial resolution:

$$\mathbf{C}_{cr} = \text{Avg}([\mathbf{C}_{so}; \mathbf{F}_{clip}^t]), \quad (6)$$

we reshape the cost map to match the shape of repeated text features. The two are concatenated and passed through a second Transformer block to process the class dimension:

$$\mathbf{C}_{co} = \text{Transformer}_c(\mathbf{C}_{cr}, \mathbf{C}_{cr}, \mathbf{C}_{cr}) \quad (7)$$

This allows the model to capture interactions across different categories.

Through iterative processing across  $N$  layers, the aforementioned SET and CET enhance the features of the cost map in both spatial and class dimensions. This strengthens the discriminability of the cost map for objects belonging to different categories as well as for objects within the same class, ultimately forming the fused cost map  $\mathbf{C}_{agg}$ . The analysis for Dimension Reduction is in the appendix B.

### RS-Transfer Upsample

The fused cost map  $\mathbf{C}_{agg}$  provides spatial and class-discriminative signals but lacks detailed textures due to the low spatial resolution. To address this, as shown in fig. 3(c), we integrate intermediate-layer features from RemoteCLIP vision encoder  $\mathbf{F}_{rclip}^v$ , CLIP vision encoder  $\mathbf{F}_{clip}^v$ , and DINO encoder  $\mathbf{F}_{dino,l_d}$  during the upsampling process. We concatenate these features with the upsampled cost map  $\mathbf{C}_{agg}$  and pass them through a projection module:

$$\mathbf{C}_{agg} = \text{Proj}([\mathbf{C}_{agg}; \mathbf{F}_{rclip,l_r}^v; \mathbf{F}_{clip,l}^v; \mathbf{F}_{dino,l_d}]) \quad (8)$$

Here, Proj is a feature projection module. After  $N_d$  layers of refinement, the final prediction  $\mathbf{O}$  is generated in the original image resolution, with shape  $N_t \times H \times W$ .

### Training Loss

We adopt standard cross-entropy loss for supervision. Given a one-hot segmentation mask  $\mathbf{M} \in \mathbb{R}^{H \times W \times N_t}$  and the prediction  $\mathbf{O}$ , the loss is defined as:

$$\mathcal{L}_{ce} = \text{CrossEntropyLoss}(\mathbf{O}, \mathbf{M}) \quad (9)$$

Minimizing this loss encourages the model to produce accurate pixel-level class predictions.

## Experiment

### Experimental Analysis of RSKT-Seg

**Implementation Details of RSKT-Seg** Our approach is implemented based on the PyTorch and Detectron2 (Wu et al. 2019) frameworks. The number of layers in RSKT-Upsample  $N_d$  is configured as 2. When using the ViT-B/16 architecture as encoder,  $l$  is set to the values  $\{3, 7\}$ , and when using the ViT-L/14 architecture,  $l$  is set to  $\{7, 15\}$ . We use DINO ViT-B/32 pretrained in (Ye, Zhuge, and Zhang 2025) and  $l_d$  is set to  $\{3, 7\}$ .  $l_r$  for RemoteCLIP ViT-B/16 is set to  $\{3, 7\}$ . AdamW is utilized as the optimizer, with an initial learning rate of  $2 \times 10^{-4}$  and a weight decay of  $1 \times 10^{-4}$ . The batch size is fixed at 8. All the experiments are carried out on 4 Nvidia 4090 GPUs, each having 24GB of memory. Our experimental results is that the optimal number of layers  $N$  is 5 when using DLRSD as the training set, and 2 when using iSAID.

**Results of the Proposed Method on Different Datasets** Table 1 shows that RSKT-Seg outperforms both classic OVS methods and existing OVRSIS methods across 8 OVRSIS datasets, demonstrating superior effectiveness in remote sensing open-vocabulary segmentation. Under different training datasets (DLRSD and iSAID) and backbone configurations (ViT-B and ViT-L), it consistently leads in key metrics like m-mIoU and m-mACC. RSKT-Seg exhibits strong robustness when switching training datasets, excelling in diverse scenarios such as complex land cover (Potsdam), small objects (UAVid), and challenging datasets (VDD, UDD5). Its performance scales effectively with stronger backbones, with ViT-L bringing significant gains over ViT-B. It overcomes the limitations of classic OVS methods in handling remote sensing-specific challenges.

Table 1: **Main Results of the RSKT-Seg on 8 OVRSIS Datasets.** Evaluation metrics: mIoU, mACC. Best results are bold, second-best underlined. m-mIoU and m-mACC denote average values across all datasets.

Method	Backbone	Type	DLRSD as Training Dataset																	
			DLRSD		iSAID		LoveDA		Potsdam		UAvid		UDD5		Vaihingen		VDD		Mean of All Datasets	
			mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	m-mIoU	m-mACC
SCAN <sub>CVPR2024</sub>	ViT-B	OVS	48.52	68.68	34.18	49.05	18.23	40.12	20.22	34.70	18.56	30.31	32.12	40.45	5.38	22.54	26.25	43.67	25.43	41.19
SAN <sub>CVPR2023</sub>	ViT-B	OVS	85.73	91.03	30.63	44.03	23.15	48.26	22.34	44.98	22.34	37.56	36.87	47.21	31.92	45.36	34.76	52.42	36.96	51.36
SED <sub>CVPR2024</sub>	ConvNeXt-B	OVS	85.13	91.36	21.54	36.28	21.32	45.17	19.47	33.40	20.12	33.34	34.65	44.10	29.40	49.38	31.43	50.25	32.88	47.91
Cat-Seg <sub>CVPR2024</sub>	ViT-B	OVS	85.84	91.44	23.56	38.48	25.45	50.32	26.79	44.72	24.56	39.20	38.23	49.56	32.32	49.65	36.18	54.30	36.62	52.21
OVRS <sub>TGRS2025</sub>	ViT-B	OVRIS	85.98	91.52	39.09	54.43	28.67	52.10	27.47	42.07	25.23	40.18	39.10	50.65	33.71	49.01	37.34	55.15	39.57	54.51
GSNet <sub>AAAI2025</sub>	ViT-B	OVRIS	84.12	90.53	42.00	59.19	29.32	53.02	26.46	43.20	25.42	40.70	40.05	51.72	35.15	52.62	38.10	56.01	40.08	55.87
RSKT-Seg	ViT-B	OVRIS	<b>90.60</b>	<b>94.89</b>	<b>44.04</b>	<b>61.23</b>	<b>32.49</b>	<b>55.67</b>	<b>34.53</b>	<b>50.71</b>	<b>28.14</b>	<b>42.86</b>	<b>42.99</b>	<b>57.81</b>	<b>37.16</b>	<b>54.81</b>	<b>41.22</b>	<b>58.09</b>	<b>43.90</b>	<b>59.51</b>
SCAN <sub>CVPR2024</sub>	ViT-L	OVS	52.42	72.43	44.28	67.25	23.17	35.36	27.45	39.22	20.28	34.43	34.14	43.25	15.23	29.45	29.24	45.57	30.78	45.87
SAN <sub>CVPR2023</sub>	ViT-L	OVS	86.45	91.25	49.56	67.25	25.33	37.54	37.75	46.28	23.53	38.14	37.23	48.45	39.22	48.33	35.83	53.25	41.80	53.81
SED <sub>CVPR2024</sub>	ConvNeXt-L	OVS	87.68	91.24	51.23	68.24	24.55	36.83	29.35	37.95	21.33	35.64	35.73	45.15	39.02	58.62	32.53	51.34	40.18	53.13
Cat-Seg <sub>CVPR2024</sub>	ViT-L	OVS	88.68	93.34	53.34	70.86	28.64	38.73	30.75	49.03	25.73	40.54	40.24	51.65	42.30	60.65	39.14	55.85	44.23	57.58
OVRS <sub>TGRS2025</sub>	ViT-L	OVRIS	<b>88.85</b>	<b>93.64</b>	<b>52.65</b>	<b>69.59</b>	<b>31.53</b>	<b>59.82</b>	<b>36.44</b>	<b>50.17</b>	<b>24.13</b>	<b>34.83</b>	<b>40.82</b>	<b>54.24</b>	<b>43.50</b>	<b>63.31</b>	<b>37.23</b>	<b>56.34</b>	<b>44.39</b>	<b>60.24</b>
GSNet <sub>AAAI2025</sub>	ViT-L	OVRIS	86.02	91.48	53.73	71.57	<b>32.52</b>	<b>60.23</b>	<b>37.85</b>	<b>52.35</b>	24.22	35.03	40.92	<b>57.04</b>	<b>44.13</b>	62.38	37.34	57.04	<b>44.56</b>	<b>60.89</b>
RSKT-Seg	ViT-L	OVRIS	<b>93.49</b>	<b>96.49</b>	<b>54.32</b>	<b>71.72</b>	<b>33.23</b>	<b>57.41</b>	<b>38.44</b>	<b>56.69</b>	<b>25.72</b>	<b>40.89</b>	<b>42.10</b>	56.39	42.69	63.29	<b>39.69</b>	<b>58.60</b>	<b>46.21</b>	<b>62.69</b>
iSAID as Training Dataset																				
Method	Backbone	Type	iSAID as Training Dataset																	
			DLRSD		iSAID		LoveDA		Potsdam		UAvid		UDD5		Vaihingen		VDD		Mean of All Datasets	
			mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	mIoU	mACC	m-mIoU	m-mACC
SCAN <sub>CVPR2024</sub>	ViT-B	OVS	16.09	38.25	62.34	76.48	12.56	32.10	<b>18.25</b>	33.17	9.87	15.30	8.45	28.60	8.72	27.20	15.32	35.10	18.95	35.78
SAN <sub>CVPR2023</sub>	ViT-B	OVS	18.82	42.36	85.43	90.36	18.45	40.25	14.82	34.84	12.32	18.32	12.11	32.45	16.23	34.38	22.10	40.25	25.04	41.65
SED <sub>CVPR2024</sub>	ConvNeXt-B	OVS	21.48	45.15	93.31	96.66	20.10	42.30	5.78	17.52	14.80	18.90	10.23	31.50	9.36	21.62	17.32	38.12	24.05	38.97
Cat-Seg <sub>CVPR2024</sub>	ViT-B	OVS	20.41	44.08	94.16	96.72	23.50	41.55	15.23	37.17	15.47	23.57	12.10	38.55	14.03	38.61	19.62	<b>51.38</b>	26.82	46.45
OVRS <sub>TGRS2025</sub>	ViT-B	OVRIS	21.06	45.48	<b>94.60</b>	<b>96.87</b>	25.30	42.32	15.57	38.94	16.22	24.05	11.90	38.80	14.66	38.68	21.42	51.25	27.59	47.05
GSNet <sub>AAAI2025</sub>	ViT-B	OVRIS	<b>26.20</b>	<b>57.07</b>	90.00	93.60	26.80	42.84	15.12	36.16	15.93	24.33	12.44	39.62	12.45	41.15	22.22	42.33	27.87	47.14
RSKT-Seg	ViT-B	OVRIS	<b>24.80</b>	<b>55.94</b>	93.16	96.37	<b>28.07</b>	<b>46.58</b>	<b>20.28</b>	<b>46.71</b>	<b>17.15</b>	<b>34.78</b>	<b>13.94</b>	<b>43.47</b>	<b>17.47</b>	<b>50.84</b>	<b>25.34</b>	47.18	<b>30.03</b>	<b>52.73</b>
SCAN <sub>CVPR2024</sub>	ViT-L	OVS	21.44	53.26	64.28	85.46	18.50	38.20	28.32	<b>52.47</b>	14.32	26.00	16.22	32.51	14.23	34.25	20.18	36.70	24.69	44.86
SAN <sub>CVPR2023</sub>	ViT-L	OVS	20.54	49.32	87.22	92.54	22.33	44.22	24.72	<b>56.54</b>	15.01	26.50	21.35	37.66	22.49	50.78	26.42	42.30	30.01	49.98
SED <sub>CVPR2024</sub>	ConvNeXt-L	OVS	23.80	50.36	94.32	96.84	23.24	45.13	11.85	23.87	15.21	26.80	22.42	39.15	12.61	<b>25.73</b>	28.50	44.20	28.99	44.01
Cat-Seg <sub>CVPR2024</sub>	ViT-L	OVS	28.80	59.56	<b>94.77</b>	<b>96.96</b>	25.11	48.25	23.90	49.49	16.10	26.90	24.32	42.77	21.74	51.25	30.16	47.20	33.11	52.80
OVRS <sub>TGRS2025</sub>	ViT-L	OVRIS	<b>32.25</b>	60.35	<b>94.86</b>	<b>97.06</b>	<b>27.98</b>	49.01	26.39	50.15	<b>16.24</b>	27.00	31.88	55.32	<b>28.80</b>	<b>54.20</b>	31.01	<b>55.30</b>	<b>36.18</b>	56.05
GSNet <sub>AAAI2025</sub>	ViT-L	OVRIS	31.50	<b>63.05</b>	93.11	95.98	27.21	49.53	<b>28.50</b>	52.00	15.98	27.10	<b>32.24</b>	56.20	25.10	<b>52.80</b>	32.07	55.21	35.71	<b>56.48</b>
RSKT-Seg	ViT-L	OVRIS	<b>31.57</b>	<b>61.66</b>	93.96	96.63	<b>29.62</b>	<b>51.01</b>	<b>28.57</b>	51.69	<b>17.36</b>	<b>35.75</b>	<b>34.01</b>	<b>57.38</b>	<b>25.55</b>	52.66	<b>33.40</b>	<b>56.50</b>	<b>36.76</b>	<b>57.91</b>

## Ablation Study On RSKT-Seg

Our ablation studies are conducted under the same default settings. All reported average values are obtained from inference results on the four datasets: DLRSD, iSAID, Potsdam, and Vaihingen.

Table 2: The Effectiveness of Proposed Components.

Dataset	R-I Cost Map	DINO Cost Map	RS-Transfer	RS-Fusion	m-mIoU	m-fwIoU	m-mACC
DLRSD	✓	✓			46.42	51.51	60.23
		✓			48.06	53.00	61.72
			✓		47.47	54.11	62.16
				✓	47.68	52.92	61.60
iSAID	✓				48.72	54.70	62.56
		✓			48.76	54.94	63.11
			✓		48.94	<b>55.46</b>	<b>64.12</b>
				✓	35.51	35.54	57.36
	✓				38.42	37.96	58.86
		✓			37.59	37.66	58.42
			✓		35.36	36.30	57.91
				✓	38.93	38.86	58.47
	✓				39.07	40.31	61.24
		✓			39.80	<b>41.59</b>	<b>62.55</b>

**The Effectiveness of Different Components** Table 2 validates the effectiveness of each proposed component. Adding R-I Cost Map or DINO Cost Map alone improves metrics on datasets, with R-I Cost Map performing slightly better. Combining them yields further gains (e.g., DLRSD m-mIoU: 48.72 vs. 46.42). Incorporating RS-Transfer brings marginal improvements and adding RS-Fusion achieves the best results, boosting performance.

**The importance of introducing remote sensing knowledge** The table 3 compares the performance of "rsDINO" and "natureDINO" on the DLRSD dataset. When the "DINO" component is added to the model, it improves the model's performance significantly. Evidently, the "rsDINO" pre-trained on remote sensing data is more effective in en-

Table 3: The Effectiveness of DINO with Remote Sensing Knowledge. "natureDINO" denotes pretrained DINO weights in (Caron et al. 2021), "rsDINO" denotes pretrained DINO weights on large-scale Remote Sensing Dataset in (Ye, Zhuge, and Zhang 2025).

Dataset	rotate	rsDINO	natureDINO	remoteclip	m-mIoU	m-fwIoU	m-mACC
DLRSD	✓				47.17	52.26	60.48
		✓			<b>48.22</b>	<b>54.86</b>	<b>62.41</b>
			✓		46.46	53.87	62.19
				✓	<b>49.47</b>	<b>55.45</b>	<b>62.81</b>
	✓				46.90	54.96	62.40
		✓			<b>49.37</b>	<b>55.29</b>	<b>62.74</b>
			✓		47.17	54.11	61.70
				✓			

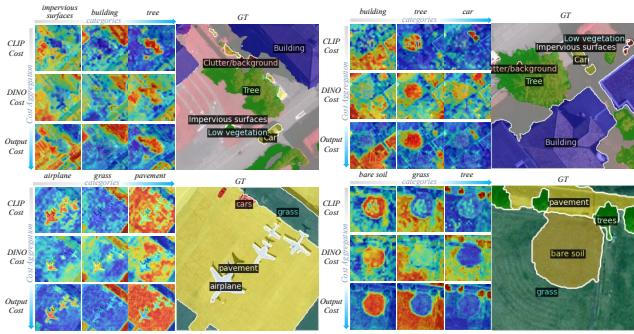
hancing the model's performance for the DLRSD dataset.

Table 4: The Influence of Different Cost Map Fusion Strategies.

Fusion	m-mIoU	m-fwIoU	m-mACC



</



**Figure 5: Comparasion of different cost map and effectiveness of efficient cost map aggregation (vertical) on different classes(horizontal).**

**Table 5: The Influence of Different Layers of the Cost Aggregation Module.** We test on 8 datasets in this experiment.

Dataset/ N	1	2	3	4	5	6
DLRSD m-mIoU	46.21	46.87	<b>49.25</b>	48.84	<b>51.58</b>	48.20
DLRSD m-mACC	61.35	62.36	64.06	63.11	<b>65.41</b>	64.11
iSAID m-mIoU	38.13	<b>39.09</b>	<b>38.93</b>	37.22	37.84	37.34
iSAID m-mACC	61.32	<u>62.21</u>	<b>62.47</b>	60.17	60.29	60.24

DLRSD and iSAID datasets, with metrics like m-mIoU, m-fwIoU, and m-mACC. Based on the experimental results, we selected N=5 for the DLRSD dataset and N=2 for the iSAID dataset as the optimal configurations for the Cost Aggregation Module

**Table 6: The Ablation Study of Different Fine-tuning Strategies.** The trainable CLIP and the frozen DINO demonstrate better performance.

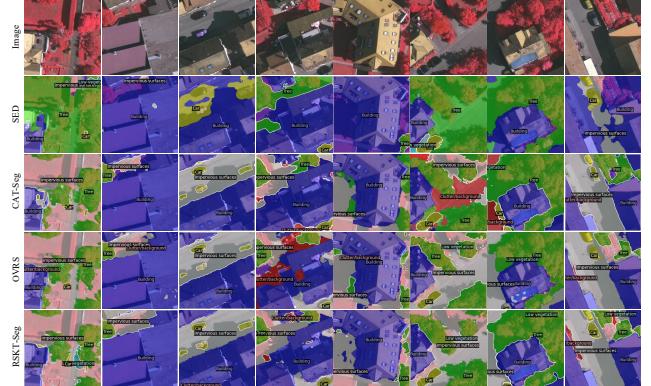
Dataset	CLIP	DINO	m-mIoU	m-fwIoU	m-mACC
iSAID	frozen	frozen	28.80	29.70	45.69
	frozen	attention	27.42	29.31	46.79
	attention	frozen	<b>39.07</b>	<b>40.31</b>	<b>61.24</b>
	attention	attention	37.96	38.98	61.21

**The impacts of different finetuning strategies** This subsection analyzes the impacts of different finetuning strategies using data from table 6 on the iSAID dataset. The key components are CLIP and DINO, each with frozen and attention states. when CLIP is in attention and DINO frozen (highlighted), the model achieves the best performance among all combinations.

**The analysis of inference speed** This table presents the inference speed (in ms) and FPS across multiple datasets for different methods. Among them, RSKT-Seg stands out for its significantly improved inference speed compared to Cat-Seg and OVRS. RSKT-Seg only needs 65.11 ms on average. This shows that RSKT-Seg can achieve faster inference, which is beneficial for real-time applications.

**Table 7: The Comparative Experiment on Inference Speed and FPS.**

Method	DLRSD	iSAID	Potsdam	Vaihingen	Mean (ms) ↓	FPS ↑
SCAN	148.27	143.24	143.36	148.46	145.83	6.86
SAN	54.15	49.30	50.64	61.06	53.79	18.59
SED	61.91	63.13	58.50	58.35	60.47	16.54
Cat-Seg	125.08	118.02	108.14	108.27	114.88	8.70
OVRS	301.60	292.33	273.84	273.72	285.37	3.50
RSKT-Seg	<b>69.95</b>	<b>64.90</b>	<b>62.88</b>	<b>62.70</b>	<b>65.11</b>	<b>15.36</b>



**Figure 6: The comparison of the segmentation visualization of different models.** The model (ViT-L) trained on DLRSD is used to perform inference on Vaihingen.

## Visualization

### Cost map visualization on diffetent categories

As seen in fig. 5, cost maps from CLIP Cost and DINO Cost show how models assign costs to classes like impervious surfaces, trees, and buildings. These heat-like visualizations reveal areas of high or low classification confidence. RS-Fusion, depicted vertically in fig. 5 across different classes, refines these maps, enhancing segmentation accuracy. More Cost visualization figures are shown in appendix E.

### Comparison of visualization

As depicted in fig. 6, a comparison is made among the segmentation visualizations of multiple models. These models were trained on the DLRSD dataset and then tested on the Vaihingen dataset. Each row corresponds to a different model, while the columns represent various image samples. RSKT-Seg demonstrates notable effectiveness in this comparison. Compared to other models, RSKT-Seg shows a relatively lower rate of misclassification and a more precise identification of object boundaries.

## Conclusion

In this paper, we address OVRSIS challenges via **OVRSIS-Bench**, a unified benchmark. Evaluations show classic OVS methods perform poorly in remote sensing, and existing OVRSIS methods lack RS-specific modeling. Thus, we propose the **RSKT-Seg** that integrates RS-CMA, RS-Fusion, and RS-Transfer, outperforming baselines with 2x faster inference, advancing OVRSIS research. In the appendix H, we carefully analyzed the gap of open-vocabulary segmentation from natural images to remote-sensing images, as well as the **limitations** of our model, and our **future work**.

## Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 62306241, and in part by grants from the Innovation Foundation for Doctor Dissertation of Northwestern Polytechnical University (No.CX2025109).

## A: More related works

### Open-Vocabulary Segmentation

Semantic segmentation models traditionally rely on predefined datasets, limiting their scalability and adaptability to novel scenarios. The advent of pretrained Vision-Language Models (VLMs) has introduced a new paradigm for segmentation. However, these models, pretrained at the image level, lack the fine-grained pixel-text alignment required for effective segmentation, posing a significant challenge for direct application. To address this issue, researchers have explored two main strategies: two-stage and single-stage approaches. Two-stage methods (Liang et al. 2023; Xu et al. 2022; Zhou, Loy, and Dai 2022) utilize external mask generators to refine segmentation predictions. However, the limited accuracy of these generated masks reduces the overall effectiveness and generalization of such approaches. In contrast, single-stage models integrate mask generation directly within the segmentation process. FC-CLIP (Yu et al. 2023) and its enhanced version (Jiao et al. 2025) incorporate trainable mask predictors within the CLIP framework, improving segmentation accuracy. More advanced techniques, such as Cat-Seg (Cho et al. 2024) and SED (Xie et al. 2024b), further refine segmentation performance by leveraging similarity matrices as pseudo-masks. Recent advancements have introduced novel mechanisms to enhance segmentation effectiveness. Side adapters, proposed in (Xu et al. 2023), facilitate efficient feature fusion, while frequency-domain modules (Xu et al. 2024) improve model generalization. Additionally, (Shan et al. 2024) presents an adaptive integration of SAM and CLIP outputs, optimizing segmentation efficiency. These ongoing innovations continue to refine vision-language alignment, advancing both accuracy and adaptability in open-vocabulary segmentation.

## B: The analysis of computational efficiency

Table 8: The Comparative Experiment on the efficiency and computational complexity of the model.

Method	train time(ms/iteration)	total param.(M)	trainable param(M)
SED	8.60	180.76	89.59
Cat-Seg	9.60	<b>154.29</b>	127.55
OVRS	18.53	154.32	127.57
RSKT-Seg	<b>7.96</b>	296.15	<b>59.89</b>

The presented table 8 compares the computational efficiency of RSKT-Seg with multiple models, taking into account training time per iteration and parameter complexity. Among them, RSKT-Seg clearly demonstrates its superiority. It achieves the shortest training time of merely 7.96 ms/iteration. Although it has a relatively large total parameter count of 296.15 M, the significantly fewer trainable parameters, only 29.89 M, substantially enhance its training efficiency. This allows for rapid convergence during training. In contrast, Cat-Seg, having the fewest total parameters

(154.29 M), lags in training speed with 9.60 ms/iteration. OVRS is the slowest, with a training time of 18.53 ms/iteration. SED remains at an average level. Overall, RSKT-Seg's remarkable efficiency makes it an excellent choice, especially in resource-constrained scenarios. Furthermore, for the effectiveness analysis of the dimensional reduction strategy in efficient cost-map aggregation, please refer to the Appendix. C.

## B: Computational Complexity Analysis

### Analysis for Spatial Dimension Reduction

The original spatial size of the features is  $H_f \times W_f$ . After the operation of the 2D convolutional layer  $\mathcal{R}_s$ , the spatial size becomes  $\frac{H_f}{r_1} \times \frac{W_f}{r_1}$ , where  $r_1$  is the reduction ratio in the spatial dimension.

In the cross-attention mechanism of the Transformer, the computational complexity of calculating the attention scores is typically proportional to the square of the sequence length. Before the spatial dimension reduction, assume the sequence length is  $L_1 = H_f \times W_f$ . After the reduction, the sequence length becomes  $L_2 = \frac{H_f \times W_f}{r_1^2}$ .

The computational complexity of the cross-attention mechanism is  $O(L^2d)$ , where  $L$  is the sequence length and  $d$  is the feature dimension. The computational complexity before reduction is  $C_1 = O((H_f \times W_f)^2 d_c)$ , and after reduction, it is  $C_2 = O((\frac{H_f \times W_f}{r_1^2})^2 d_c)$ .

The reduction ratio of the computational complexity is:

$$\frac{C_2}{C_1} = \frac{O((\frac{H_f \times W_f}{r_1^2})^2 d_c)}{O((H_f \times W_f)^2 d_c)} = \frac{1}{r_1^4} \quad (10)$$

### Analysis for Class Dimension Reduction

In the RS Class Fusion Transformer, an average pooling layer  $\mathcal{R}_c$  is used to reduce the class dimension of the features.

In the self-attention mechanism of the Transformer, the computational complexity of calculating the attention scores is also proportional to the square of the sequence length. Before the class dimension reduction, assume the sequence length is  $L_3 = H_f \times W_f$ . After the reduction, the sequence length becomes  $L_4 = \frac{H_f \times W_f}{r_2^2}$ .

The computational complexity before reduction is  $C_3 = O((H_f \times W_f)^2 d_c)$ , and after reduction, it is  $C_4 = O((\frac{H_f \times W_f}{r_2^2})^2 d_c)$ .

The reduction ratio of the computational complexity is:

$$\frac{C_4}{C_3} = \frac{O((\frac{H_f \times W_f}{r_2^2})^2 d_c)}{O((H_f \times W_f)^2 d_c)} = \frac{1}{r_2^4} \quad (11)$$

## C: Datasets introduction

### DLRSD

It is a large-scale remote sensing image dataset, consisting of a total of 7002 images across 17 categories. In the experimental configuration, 5601 images are designated for training, and the remaining 1401 images are used for validation.

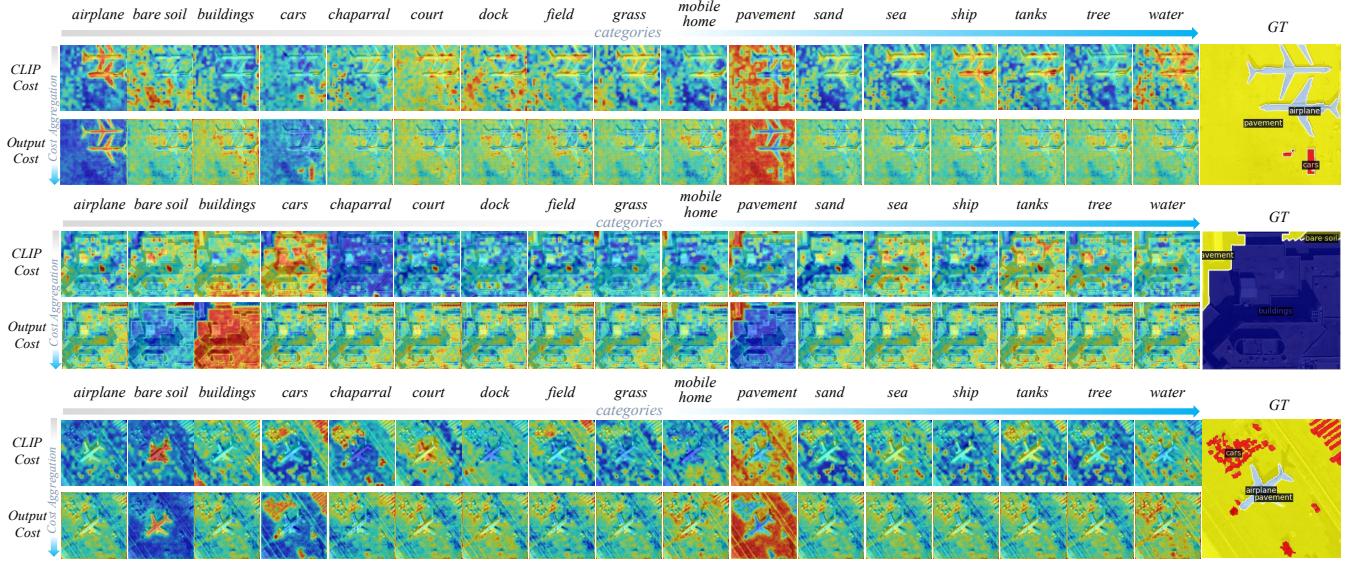


Figure 7: Comparing the Effectiveness of Cost Maps and Cost Aggregation on Different Categories using the DLRSD Dataset.

## iSAID

It contains 24,439 images across 15 categories. In the experimental setup, 18,076 images are used for training, and the remaining 6,363 images are employed to validate the model’s performance.

## Potsdam

The Potsdam dataset is a large-scale and extensively used dataset in the realm of remote sensing images. With 20,102 images spread across 6 categories, it has been a staple for numerous research projects and model evaluations.

## Vaihingen

The Vaihingen dataset is a commonly used remote sensing image dataset in the academic and research community. Comprising 2254 images across 6 categories, it provides a valuable resource for studying and developing remote sensing applications.

## UAVid

UAVid is a high-resolution dataset captured from unmanned aerial vehicles, designed for urban scene understanding. It consists of 300 images annotated with 8 semantic categories. In our experiments, 200 images are used for training and 100 for validation.

## UDD5

The UDD5 dataset contains urban driving scenes collected from a bird’s-eye view, with pixel-wise annotations across 5 semantic categories. It includes 4,198 images in total, with 3,149 images used for training and 1,049 for validation.

## LoveDA

LoveDA is a large-scale remote sensing dataset with diverse land cover types, collected from rural and urban scenes. It

includes 18,000 images annotated with 7 classes. For training, 14,168 images are used, while the remaining 3,832 are reserved for validation.

## VDD

VDD is a recently introduced dataset focused on visual domain adaptation for remote sensing tasks. It consists of 7,992 images labeled across 7 categories. In our setup, 5,994 images are used for training, and 1,998 images are used for validation.

## D: Evaluation metric

Consistent with existing OVRSIS methods, we use the mean Intersection over Union (mIoU), Frequency Weighted Intersection over Union (fwIoU), and Mean Accuracy (mACC) as evaluation metrics. The mIoU is computed as the average IoU across all classes, and is expressed as Equation eq. (12):

$$mIoU = \frac{1}{n} \sum_{i=1}^n \frac{TP_i}{TP_i + FP_i + FN_i} \quad (12)$$

where  $n$  is the number of classes,  $TP_i$  is the number of true positives for class  $i$ ,  $FP_i$  is the number of false positives for class  $i$ , and  $FN_i$  is the number of false negatives for class  $i$ . The fwIoU metric, which accounts for the frequency of each class and offers a weighted measure, is calculated as Equation eq. (13):

$$fwIoU = \sum_{i=1}^n \frac{TP_i + FN_i}{\sum_{j=1}^n (TP_j + FN_j)} \times \frac{TP_i}{TP_i + FP_i + FN_i} \quad (13)$$

The mACC, which evaluates the per-class accuracy, is given by Equation eq. (14):

$$mACC = \frac{1}{n} \sum_{i=1}^n \frac{TP_i}{TP_i + FN_i} \quad (14)$$

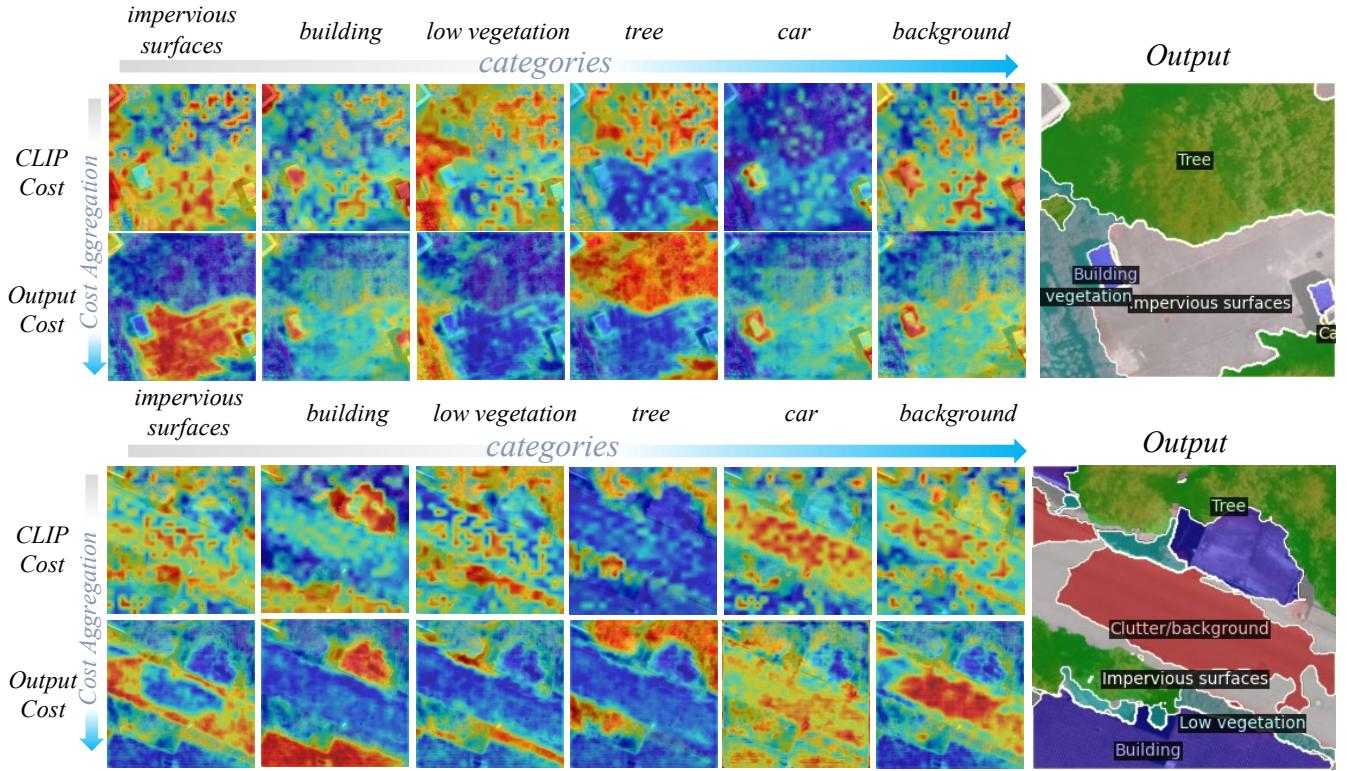


Figure 8: Comparing the Effectiveness of Cost Maps and Cost Aggregation on Different Categories using the Vaihingen Dataset.

## E: More cost map visualization

The fig. 7 and fig. 8 offer insightful comparisons of cost maps and cost aggregation across different object categories. fig. 7 focuses on the DLRSD dataset, displaying cost maps from CLIP Cost and Output Cost Aggregation for a wide range of categories such as airplane, buildings, cars, etc. Similarly, fig. 8 examines the Vaihingen dataset, presenting cost maps for categories like impervious surfaces, building, and tree. By comparing CLIP Cost and Output Cost Aggregation, it becomes evident how cost aggregation refines the initial cost maps, potentially improving the accuracy of object segmentation. Moreover, cost maps become more discriminative among different categories after efficient aggregation.

## F: Comparison of RSKT-Seg with Different Models at Each Category

The performance of RSKT-Seg is comprehensively evaluated by comparing it with several other models at the granularity of each category. table 9 and table 10 present the per-class mIoU (mean Intersection over Union) values on the iSAID and Vaihingen datasets respectively, where all models are trained on the DLRSD dataset.

In the iSAID dataset (table 9), RSKT-Seg outperforms other models in most categories. For instance, in the “ship” category, RSKT-Seg achieves an mIoU of 62.37, which is higher than SCAN’s 55.47, SAN’s 50.89, SED’s 40.63, Cat-Seg’s 42.78, and OVRS’s 58.36. Similar advantages are

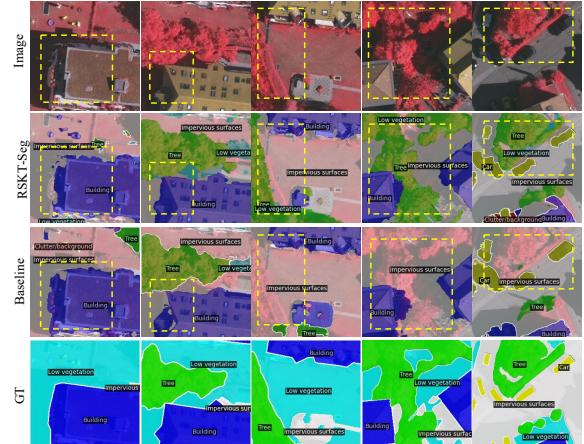


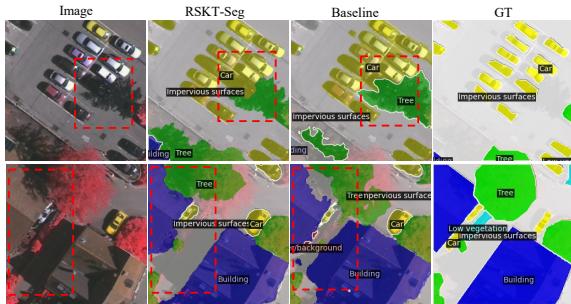
Figure 9: Comparison of visualization of RSKT-Seg and baseline

**Table 9: Comparision of per-class mIoU on iSAID dataset.** We use the model trained on DLRSD dataset and perform inference on the iSAID dataset.

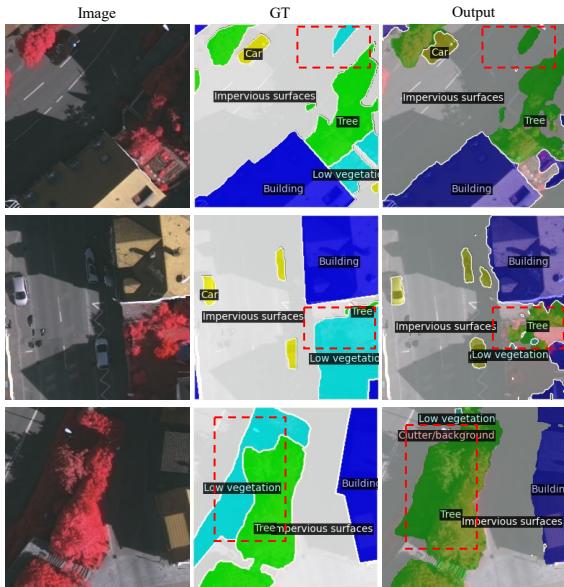
Model	Backbone	ship	storage tank	baseball diamond	tennis court	basketball court	ground track field	bridge	large vehicle	small vehicle	helicopter	swimming pool	roundabout	soccer ball field	plane	harbor	mIoU
SCAN <sub>CVPR2024</sub>	ViT-B	55.47	75.68	25.94	28.73	9.35	28.89	15.68	50.45	55.67	1.36	30.45	9.67	52.67	75.89	40.67	34.18
SAN <sub>CVPR2023</sub>	ViT-B	50.89	70.92	22.31	25.18	8.56	25.23	13.76	45.89	50.12	1.08	27.68	8.23	47.89	70.12	35.23	30.63
SED <sub>CVPR2024</sub>	ConvNeXt-B	40.63	60.15	18.98	20.67	6.48	18.45	10.92	35.21	40.67	0.92	22.12	6.78	37.21	60.45	28.78	21.54
Cat-Seg <sub>CVPR2024</sub>	ViT-B	42.78	62.87	19.76	21.34	6.92	19.87	10.43	36.78	41.34	0.87	23.78	6.54	38.67	62.98	29.45	23.56
OVRS <sub>CVPR2024</sub>	ViT-B	58.36	78.21	28.12	30.49	10.23	30.76	16.89	52.67	57.89	1.56	32.45	10.34	55.45	78.63	42.34	39.09
RSKT-Seg	ViT-B	62.37	82.74	30.46	33.06	11.08	33.33	18.16	58.29	62.04	1.75	35.71	11.27	60.21	83.99	46.27	42.05

**Table 10: Comparision of per-class mIoU on Vaihingen dataset.** We use the model trained on DLRSD dataset and perform inference on the Vaihingen dataset.

Method	Backbone	Impervious surfaces	Building	Low vegetation	Tree	Car	Clutter/background	mIoU
SCAN <sub>CVPR2024</sub>	ViT-B	5.60	9.64	1.15	7.28	4.04	4.55	5.38
SAN <sub>CVPR2023</sub>	ViT-B	33.26	56.36	7.13	43.29	24.21	27.23	31.92
SED <sub>CVPR2024</sub>	ConvNeXt-B	30.50	52.34	6.37	40.00	22.13	25.08	29.40
Cat-Seg <sub>CVPR2024</sub>	ViT-B	33.46	57.19	7.32	43.77	24.44	27.74	32.32
OVRS <sub>CVPR2024</sub>	ViT-B	35.06	59.32	7.72	45.45	25.81	28.89	33.71
RSKT-Seg	ViT-B	35.27	59.57	7.54	46.71	26.57	30.34	34.33



**Figure 10: Failure case.** The presence of shadows causes misclassification.



**Figure 11: limitation.** The model can not distinguish the height of remote sensing image (Low vegetation and tree).

also observed in categories like “storage tank” and “plane”. The overall mIoU of RSKT-Seg reaches 42.05, significantly higher than the other models, indicating its superiority in handling complex object categories in remote-sensing images. Regarding the Vaihingen dataset (table 10), RSKT-Seg also shows excellent performance. In the “Impervious surfaces” category, it attains an mIoU of 35.27, surpassing SCAN, SED, and being slightly ahead of SAN, Cat-Seg, and OVRS. In the “Tree” category, RSKT-Seg’s mIoU of 46.71 is higher than the values of other models. The overall mIoU of RSKT-Seg for the Vaihingen dataset is 34.33, demonstrating its effectiveness in segmenting different surface types in this dataset. Overall, these comparisons highlight the strong adaptability and high-accuracy segmentation ability of RSKT-Seg across different datasets and object categories.

## G: Comparison with the baseline model Cat-Seg

We compared RSKT-Seg with the basic Cat-Seg model in the open-vocabulary segmentation (OVS) task for natural images in table 11. The results show that our model generally outperforms Cat-Seg in all evaluation metrics, which fully demonstrates the effectiveness of our proposed remote sensing domain adaptation method.

## H: The Effectiveness of Remote Sensing Knowledge Transfer (RSKT)

The fig. 9 compares RSKT-Seg and the baseline model. RSKT-Seg excels in object segmentation precision, more accurately outlining building boundaries, clearly distinguishing vegetation types, and consistently delineating impervious surfaces compared to the baseline. It also demonstrates a superior overall scene understanding, with more coherent results aligning better with the actual scene. In contrast, the baseline shows misclassification, less-defined edges, and disjointed segmentation. This visual comparison strongly indicates that RSKT-Seg outperforms the baseline, highlighting its effectiveness in remote sensing domain adaptation,

Table 11: Comparison Results between RSKT-Seg and Cat-Seg

DLRSD as Training Dataset																		
Method	Backbone	DLRSD			iSAID			Potsdam			Vaihingen			Mean of All Datasets				
		mIoU	fwIoU	mACC	mIoU	fwIoU	mACC	mIoU	fwIoU	mACC	mIoU	fwIoU	mACC	m-mIoU	m-fwIoU	m-mACC		
Cat-Seg	ViT-B	85.84	86.79	91.44	23.56	27.40	38.48	26.79	31.28	44.72	32.32	38.71	49.65	42.13	46.05	56.07		
RSKT-Seg	ViT-B	87.88	89.62	93.31	42.05	52.53	59.77	31.49	36.55	49.35	34.33	43.14	54.06	48.94	55.46	64.12		
△	ViT-B	(+2.04)	(+2.83)	(+1.87)	(+18.49)	(+25.13)	(+21.29)	(+4.70)	(+5.27)	(+4.63)	(+2.01)	(+4.43)	(+4.41)	(+6.81)	(+9.41)	(+8.05)		
Cat-Seg	ViT-L	88.68	89.38	93.34	53.34	64.49	70.86	35.78	36.82	49.06	42.30	49.57	60.65	55.03	60.07	68.48		
RSKT-Seg	ViT-L	91.51	93.33	95.41	54.17	64.59	70.82	39.05	41.93	53.59	47.52	55.92	63.92	58.06	63.94	70.94		
△	ViT-L	(+2.83)	(+3.95)	(+2.07)	(+0.83)	(+0.10)	(-0.04)	(+3.27)	(+5.11)	(+4.53)	(+5.22)	(+6.35)	(+3.27)	(+3.03)	(+3.87)	(+2.46)		
iSAID as Training Dataset																		
Method	Backbone	DLRSD			iSAID			Potsdam			Vaihingen			Mean of All Datasets				
		mIoU	fwIoU	mACC	mIoU	fwIoU	mACC	mIoU	fwIoU	mACC	mIoU	fwIoU	mACC	m-mIoU	m-fwIoU	m-mACC		
Cat-Seg	ViT-B	20.41	23.06	44.08	94.16	94.52	96.72	15.23	18.62	37.17	14.03	18.86	38.61	35.96	38.77	54.15		
RSKT-Seg	ViT-B	28.89	31.23	58.63	92.33	91.90	95.56	21.91	23.95	46.48	20.88	24.22	50.37	41.00	42.83	62.76		
△	ViT-B	(+8.48)	(+8.17)	(+14.55)	(-1.83)	(-2.62)	(-1.16)	(+6.68)	(+5.33)	(+9.31)	(+0.85)	(+5.36)	(+11.76)	(+5.04)	(+4.06)	(+8.61)		
Cat-Seg	ViT-L	28.80	29.02	59.56	94.77	95.03	96.96	23.90	28.85	49.49	21.74	26.14	51.25	42.30	44.76	64.32		
RSKT-Seg	ViT-L	31.73	33.69	61.40	94.07	94.14	96.72	29.71	35.48	53.02	26.67	34.10	54.55	45.55	49.35	66.42		
△	ViT-L	(+2.93)	(+4.67)	(+1.84)	(-0.70)	(-0.89)	(-0.24)	(+5.81)	(+6.63)	(+3.53)	(+4.93)	(+7.96)	(+3.30)	(+3.25)	(+4.59)	(+2.10)		

which is crucial for applications like urban planning and environmental monitoring.

## I: Limitation and Future Work

The performance of the model is not without **limitations**, as illustrated in fig. 10 and fig. 11. fig. 10 showcases a failure case where the model is misled by elements such as shadows. In the segmentation results of both RSKT-Seg and the baseline, we can observe that the presence of shadows causes misclassification. For example, in the red-boxed areas, the model incorrectly segments regions affected by shadows, failing to accurately distinguish between impervious surfaces, trees, and cars. The shadowed areas interfere with the model's ability to correctly identify object boundaries and categories, leading to sub-optimal segmentation outcomes.

fig. 11 further highlights another limitation of the model. It demonstrates that the model struggles to differentiate between low vegetation and trees based on height. In the marked regions of the output compared to the ground truth (GT), there are evident misclassifications. The model confuses low vegetation with trees, which is a significant shortcoming. This inability to accurately discern between these two categories based on height-related features is an area that requires further improvement.

In **future work**, we will consider introducing the depth modality to enhance the sense of distance, enabling the model to identify differences highly relevant to height and distinguish the impact of shadows on the segmentation results.

## J: Detailed Framework

## References

- Badrinarayanan, V.; Kendall, A.; and Cipolla, R. 2017. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12): 2481–2495.
- Cai, W.; Jin, K.; Hou, J.; Guo, C.; Wu, L.; and Yang, W. 2025. Vdd: Varied drone dataset for semantic segmentation. *Journal of Visual Communication and Image Representation*, 109: 104429.
- Cao, Q.; Chen, Y.; Ma, C.; and Yang, X. 2024. Open-vocabulary remote sensing image semantic segmentation. *arXiv preprint arXiv:2409.07683*.
- Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging Properties in Self-Supervised Vision Transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*.
- Chaudhuri, B.; Demir, B.; Chaudhuri, S.; and Bruzzone, L. 2017. Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method. *IEEE Transactions on Geoscience and Remote Sensing*, 56(2): 1144–1158.
- Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4): 834–848.
- Chen, Y.; Wang, Y.; Lu, P.; Chen, Y.; and Wang, G. 2018. Large-scale structure from motion with semantic constraints of aerial images. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, 347–359. Springer.
- Cheng, B.; Misra, I.; Schwing, A. G.; Kirillov, A.; and Giridhar, R. 2022. Masked - attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1290–1299. IEEE.
- Cheng, B.; Schwing, A.; and Kirillov, A. 2021. Per - pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34: 17864–17875.

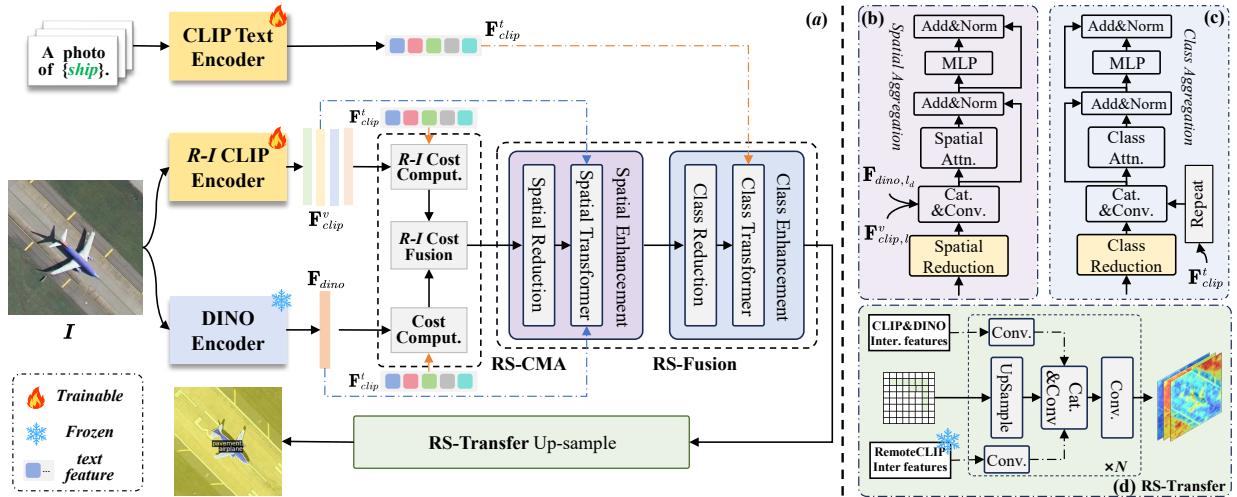


Figure 12: **The overall framework of RSKT-Seg** includes: (a) the overall procedure of RSKT-Seg; (b) the workflow of the RS Spatial Fusion Transformer; (c) the operation flow of the RS Class Fusion Transformer; (d) the framework principle of the Remote Sensing Knowledge Transfer Upsample.

Cho, S.; Shin, H.; Hong, S.; Arnab, A.; Seo, P. H.; and Kim, S. J. 2024. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4113–4123.

Diakogiannis, F. I.; Waldner, F.; Caccetta, P.; and Wu, C. 2020. ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 162: 94–114.

Dosovitskiy, A. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Hu, Z.; Gao, J.; Yuan, Y.; and Li, X. 2024. Contrastive Tokens and Label Activation for Remote Sensing Weakly Supervised Semantic Segmentation. *IEEE Transactions on Geoscience and Remote Sensing*.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, 4904–4916. PMLR.

Jiao, S.; Zhu, H.; Huang, J.; Zhao, Y.; Wei, Y.; and Shi, H. 2025. Collaborative vision - text representation optimizing for open - vocabulary segmentation. In *Proceedings of the European Conference on Computer Vision*, 399–416.

Kotaridis, I.; and Lazaridou, M. 2021. Remote sensing image segmentation advances: A meta-analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 173: 309–322.

Li, B.; Zhang, D.; Zhao, Z.; Gao, J.; and Li, X. 2024a. StitchFusion: Weaving Any Visual Modalities to Enhance Multimodal Semantic Segmentation. *arXiv preprint arXiv:2408.01343*.

Li, B.; Zhang, D.; Zhao, Z.; Gao, J.; and Li, X. 2024b. U3M: Unbiased Multiscale Modal Fusion Model for Multimodal Semantic Segmentation. *arXiv preprint arXiv:2405.15365*.

Li, B.; Zhang, D.; Zhao, Z.; Gao, J.; and Li, X. 2025. FGAsen: Fine-Grained Pixel-Text Alignment for Open-Vocabulary Semantic Segmentation. *arXiv preprint arXiv:2501.00877*.

Li, K.; Liu, R.; Cao, X.; Bai, X.; Zhou, F.; Meng, D.; and Wang, Z. 2024c. Segearth-ov: Towards training-free open-vocabulary segmentation for remote sensing images. *arXiv preprint arXiv:2410.01768*.

Liang, F.; Wu, B.; Dai, X.; Li, K.; Zhao, Y.; Zhang, H.; Zhang, P.; Vajda, P.; and Marculescu, D. 2023. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7061–7070.

Lin, B.; Xie, J.; Li, C.; and Qu, Y. 2018. Deeptongue: Tongue segmentation via ResNet. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1035–1039. IEEE.

Long, J.; Shelhamer, E.; and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3431–3440.

Lyu, Y.; Vosselman, G.; Xia, G.-S.; Yilmaz, A.; and Yang, M. Y. 2020. UAVid: A semantic segmentation dataset for UAV imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 165: 108 – 119.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 8748–8763. PMLR.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III*, 234–241. Springer.

- Shan, X.; Wu, D.; Zhu, G.; Shao, Y.; Sang, N.; and Gao, C. 2024. Open - vocabulary semantic segmentation with image embedding balancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 28412–28421.
- Wang, J.; Zheng, Z.; Ma, A.; Lu, X.; and Zhong, Y. 2021. LoveDA: A Remote Sensing Land-Cover Dataset for Domain Adaptive Semantic Segmentation. In Vanschoren, J.; and Yeung, S., eds., *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran Associates, Inc.
- Wu, Y.; Kirillov, A.; Massa, F.; Lo, W.-Y.; and Girshick, R. 2019. Detectron2. <https://github.com/facebookresearch/detectron2>.
- Xie, B.; Cao, J.; Anwer, R. M.; Xie, J.; Nie, J.; Yang, A.; and Pang, Y. 2024a. Multi - query and multi - level enhanced network for semantic segmentation. *Pattern Recognition*, 156: 110777.
- Xie, B.; Cao, J.; Xie, J.; Khan, F. S.; and Pang, Y. 2024b. Sed: A simple encoder-decoder for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3426–3436.
- Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34: 12077–12090.
- Xu, M.; Zhang, Z.; Wei, F.; Hu, H.; and Bai, X. 2023. Side adapter network for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2945–2954.
- Xu, M.; Zhang, Z.; Wei, F.; Lin, Y.; Cao, Y.; Hu, H.; and Bai, X. 2022. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, 736–753. Springer.
- Xu, W.; Wang, C.; Feng, X.; Xu, R.; Huang, L.; Zhang, Z.; Guo, L.; and Xu, S. 2024. Generalization boosted adapter for open - vocabulary segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*.
- Yao, X.; Cao, Q.; Feng, X.; Cheng, G.; and Han, J. 2021. Scale-aware detailed matching for few-shot aerial image semantic segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 60: 1–11.
- Ye, C.; Zhuge, Y.; and Zhang, P. 2025. Towards Open-Vocabulary Remote Sensing Image Semantic Segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Yu, Q.; He, J.; Deng, X.; Shen, X.; and Chen, L.-C. 2023. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. *Advances in Neural Information Processing Systems*, 36: 32215–32234.
- Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P. H.; et al. 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6881–6890.
- Zhou, C.; Loy, C. C.; and Dai, B. 2022. Extract free dense labels from CLIP. In *European Conference on Computer Vision*, 696–712. Springer.
- Zhu, C.; Li, L.; Wu, Y.; and Sun, Z. 2024. Saswot: Real - time semantic segmentation architecture search without training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 7722–7730.