

# Quality-Driven Curation of Remote Sensing Vision-Language Data via Learned Scoring Models

Dilxat Muhtar Enzhuo Zhang Zhenshi Li Feng Gu Yanglangxing He  
Pengfeng Xiao Xueliang Zhang✉  
*Nanjing University*

pumpkindilxat@gmail.com, Zenzhuo@smail.nju.edu.cn  
{xiaopf, zx1}@nju.edu.cn

>Data Model Code

## Abstract

*Vision-Language Models (VLMs) have demonstrated great potential in interpreting remote sensing (RS) images through language-guided semantic understanding. However, the effectiveness of these VLMs critically depends on high-quality image-text training data that captures rich semantic relationships between visual content and language descriptions. Unlike natural images, RS lacks large-scale interleaved image-text pairs from web data, making data collection challenging. While current approaches rely primarily on rule-based methods or flagship VLMs for data synthesis, a systematic framework for automated quality assessment of such synthetically generated RS vision-language data is notably absent. To fill this gap, we propose a novel score model trained on large-scale RS vision-language preference data for automated quality assessment. Our empirical results demonstrate that fine-tuning CLIP or advanced VLMs (e.g., Qwen2-VL) with the top 30% of data ranked by our score model achieves superior interpretation accuracy compared to both full-data fine-tuning and CLIP-score-based ranking approaches. Furthermore, we demonstrate applications of our scoring model for reinforcement learning (RL) training and best-of-N (BoN) test-time scaling, enabling significant improvements in VLM performance for RS tasks.*

## 1. Introduction

The advancement of artificial intelligence has consistently demonstrated the benefits of scaling across three crucial dimensions: data volume, computational resources, and model complexity [12, 16]. In visual understanding, vision-language models (VLMs) have particularly benefited from diverse training data, exhibiting evolutionary patterns that

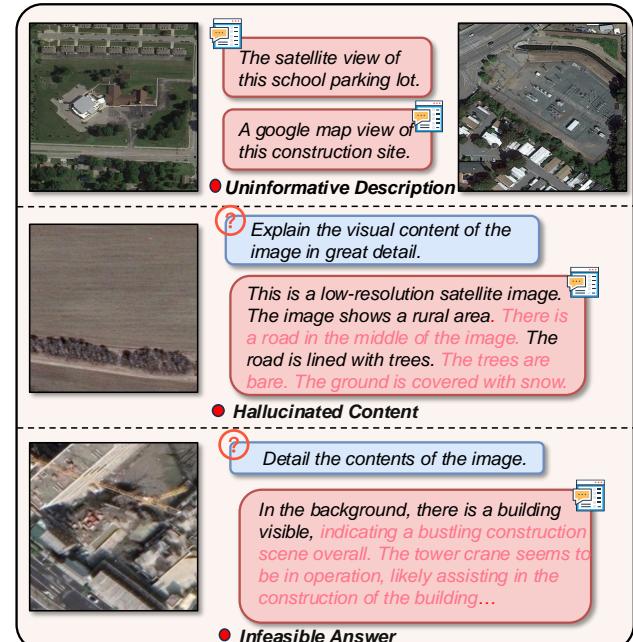


Figure 1. Examples from common RS vision-language datasets highlighting quality issues. These samples illustrate three prevalent problems: uninformative descriptions that lack meaningful content about the image features, hallucinated answers that introduce factual inaccuracies, and responses to infeasible questions that cannot be answered based on the visible images

mirror human cognitive development in environmental perception and interpretation. From the foundational CLIP model [29] to recent integrations with large language models (LLMs) [5, 8, 20, 38], VLMs have achieved remarkable success, emerging as indispensable tools for real-world applications ranging from intelligent agents [52] to autonomous driving systems [26].

Despite these advancements, VLMs demonstrate limited capability in interpreting remote sensing (RS) images due to

✉ Corresponding Author

substantial domain shifts in data distributions [17, 19]. This performance gap stems primarily from a critical bottleneck: the scarcity of large-scale vision-language data for RS domains, which, unlike natural images, lacks the abundant, naturally aligned image-text pairs readily available through web crawling. Current approaches to bridge this gap include: (1) binary classification models filtering RS data from general vision-language datasets like DataComp [51], (2) rule-based construction of vision-language pairs using OpenStreetMap tags [39], and (3) automated data generation through flagship VLMs [17, 18, 22, 24, 27]. While these methods advance holistic RS understanding, they introduce significant quality concerns: rule-based approaches often produce informationally sparse or semantically inconsistent pairs, while VLM-generated data risks propagating hallucinatory content that misrepresents the actual imagery and incorrectly infers answers to questions that cannot be determined from the given images (Fig. 1). The absence of robust, automated quality assessment framework consequently constrains further improvements in RS-specific VLMs [18].

To address this critical gap, we propose ScoreRS – a learned quality scoring model for vision-language data in RS domains. In developing ScoreRS, we first construct both image-caption and vision instruction preference datasets by leveraging existing RS-specific VLMs and robust open-source VLMs as policy models, while employing both rule-based evaluation and flagship VLMs as judges for preference dataset construction. Through a three-stage progressive training process with the constructed data, we evaluate ScoreRS’s effectiveness in both data quality ranking and its practical applications as a large VLM reward model for GRPO [33] training and best-of-n (BoN) selector. Our results demonstrate that fine-tuning CLIP and Qwen2VL [38] on just 30% of the data, ranked using ScoreRS scores, outperforms models trained on either the complete dataset or CLIP score-ranked data. Furthermore, evaluation on the challenging RS-specific vision-language benchmark VG-DOIR [47] and LHRS-Bench [24] reveals that ScoreRS serves as an effective BoN selector for improving results when scaling VLMs with multiple generated samples at test time, and can be integrated with rule-based rewards for GRPO reinforcement training to enhance model capabilities.

The main contributions of our work can be summarized as follows:

1. We introduce the first large-scale RS-specific preference dataset, comprising pairwise preference pairs for both image-captions and vision instructions, alongside ScoreRS—a novel data scoring model specifically designed for quality control of RS vision-language data.
2. We demonstrate that models fine-tuned on RS vision-language data ranked by ScoreRS scores consistently

outperform those trained on either the complete dataset or CLIP score-ranked data.

3. We empirically validate ScoreRS’s effectiveness through two key applications: (1) as a reward model for GRPO training, and (2) as a BoN selector for scaling VLM-generated samples at test time. Both applications yield significant improvements in RS VLM performance across multiple challenging benchmarks.

## 2. Method

To establish quality control for RS vision-language datasets, we propose a scoring framework that quantitatively evaluates data quality through a learned model. This framework enables data curation through score-based ranking and selection of high-quality samples.

Formally, we define a scoring function  $f_\theta : \mathcal{T} \times \mathcal{I} \rightarrow \mathbb{R}$ , parameterized by  $\theta$ , where  $\mathcal{T}$  and  $\mathcal{I}$  denote the text and image spaces, respectively. Given an image-text pair  $(I, T) \in \mathcal{I} \times \mathcal{T}$ , which may consist of either a simple caption or a multi-turn conversation associated with the image, the model outputs a scalar quality score  $s = f_\theta(I, T)$ . The scoring function is trained to assign higher scores to better-aligned image-text pairs  $(I, T)$  through pairwise preference learning. For each image  $I$ , we collect pairs of text  $(T^+, T^-)$  where  $T^+$  is preferred over  $T^-$ . This preference dataset is formally defined as  $\mathcal{D} = (I_i, T_i^+, T_i^-)_{i=1}^{|\mathcal{D}|}$ . The pairwise preferences are modeled using the Bradley-Terry model [2], which defines the probability of  $T^+$  being preferred over  $T^-$  given image  $I$  as:

$$p(T^+ \succ T^- | I) = \frac{\exp(f_\theta(I, T^+))}{\exp(f_\theta(I, T^+)) + \exp(f_\theta(I, T^-))}, \quad (1)$$

where  $\succ$  denotes the preference relation. The model parameters  $\theta$  are optimized by minimizing the empirical negative log-likelihood loss [25, 37]:

$$\mathcal{L}(\theta) = -\mathbb{E}_{(I, T^+, T^-) \sim \mathcal{D}} \log(\sigma(f_\theta(I, T^+) - f_\theta(I, T^-))), \quad (2)$$

where  $\sigma(\cdot)$  denotes the sigmoid function.

Given the absence of RS specific vision-language preference datasets, we start by establishing a data collection pipeline for acquiring both image-caption and vision instruction preference pairs. Subsequently, we detail the construction of our scoring function  $f_\theta$  by leveraging pre-trained VLMs and employing a progressive training strategy.

### 2.1. Preference Data Construction

#### 2.1.1. Image-Caption Preference Dataset

Following established practices in VLM training [20, 24, 27, 38], we start with training ScoreRS to judge high-quality image-caption alignments. Since RS images exhibit

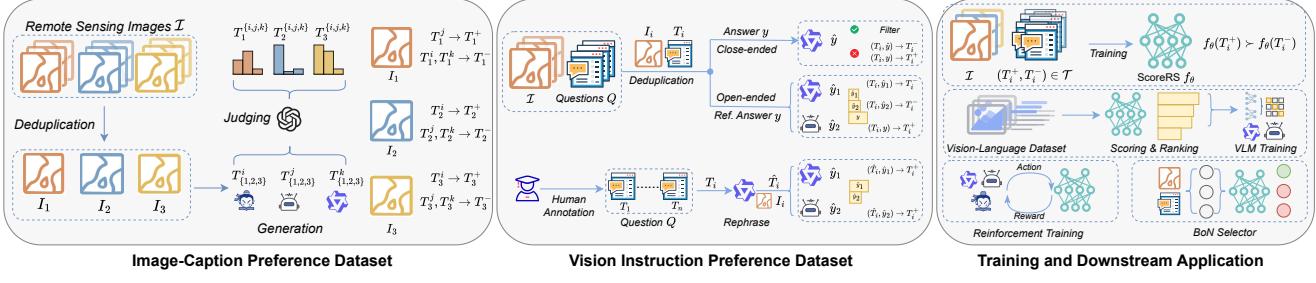


Figure 2. Our data pipeline for generating pairwise preference datasets and the training and downstream application of our ScoreRS model.  $I_i \in \mathcal{I}$  represents a RS image, and  $T_i \in \mathcal{T}$  represents an image caption, question, or conversation associated with the image

variations across geographical locations [24, 39], we leverage the LHR-S Align dataset [24] as our image source. This dataset comprises 1.15M orthorectified RS images from major global urban areas, enabling ScoreRS to learn from diverse geographical contexts worldwide.

**Deduplication** We implement a rigorous image deduplication process to further ensure image quality and representativeness: (1) Feature extraction using the SSCD copy detection model [10, 28] to compute image embeddings. (2) Similarity computation through cosine distance computation between image embeddings, with a carefully tuned threshold of 0.65 (empirically determined through experiments across the range 0.6–0.9). (3) Duplicate grouping via connected-components algorithm, preserving one image-text pair per component. Finally, this deduplication pipeline yields 76K distinctive representative RS images.

**Pairwise Preference Generation** Our caption pairwise preference generation process employs three VLMs: the RS-specific LHR-S-Bot-Nova [18] and two general-purpose models, Qwen2VL-7B [38] and InternVL-2.5-8B [5]. We generate captions for each image using same prompts and sampling parameters across all models. The resulting captions are then evaluated by GPT-4o along five dimensions: *accuracy, completeness, conciseness, objectivity, and spatial clarity*. For each image, we compute the mean score across these five dimensions for each caption. The caption with the highest average score is designated as the positive example ( $T^+$ ), while the remaining two captions serve as negative examples ( $T^-$ ). After removing results with parsing errors, we construct a dataset containing 72K image-caption preference pairs. To validate the reliability of GPT-4o as a preference judge, we conducted a human evaluation study on a random sample of 1,000 pairs. Human experts independently annotated these samples to identify positive and negative captions. The results showed 92.6% agreement (926 out of 1,000 samples) between human judgments and GPT-4o’s assessments. This high level of agreement validates our choice of GPT-4o as a reliable judge for generating pairwise preferences.

### 2.1.2. Vision Instruction Preference Dataset

Vision instruction data consists of conversations about images [20]. To construct a RS-specific vision instruction preference dataset that enables ScoreRS to evaluate responses to diverse user queries, we prioritized collecting a broad spectrum of question types and conversation scenarios. We aggregate vision instruction data from multiple sources: GeoChat [17] (306K samples), LHR-S-Instruct [24] (39.8K samples), and a subset of SkysenseGPT [22] (381K samples) as the source of our vision instruction preference dataset.

**Deduplication** We employ a two-stage similarity-based filtering process to ensure question diversity: (1) Text similarity: let  $Q = \{T_1, T_2, \dots, T_n\}$  be the set of all questions extracted from the conversations. We compute their semantic embeddings using the BGE [3] model, where for any two queries  $T_i, T_j \in Q$ :  $\text{sim}_{\text{text}}(T_i, T_j) = \cos(\text{BGE}(T_i), \text{BGE}(T_j))$ ; (2) Image similarity: For query pairs where  $\text{sim}_{\text{text}}(T_i, T_j) > 0.65$ , we further examine their corresponding images  $I_i, I_j$  using SSCD [28] embeddings:  $\text{sim}_{\text{image}}(I_i, I_j) = \cos(\text{SSCD}(I_i), \text{SSCD}(I_j))$ . When both  $\text{sim}_{\text{text}}(T_i, T_j)$  and  $\text{sim}_{\text{image}}(I_i, I_j)$  exceed 0.65, we retain only one question-image sample from the pair to minimize redundancy.

**Pairwise Preference Generation** After obtaining the deduplicated set of conversations, we categorized them into two types: close-ended questions with definitive answers (e.g., vision-question answering, classification, and visual grounding tasks) and open-ended questions requiring interpretive responses (e.g., visual reasoning). For close-ended questions, which can be objectively verified against specific correct responses, we prompted the Qwen2VL-7B model to generate an answer  $\hat{y}$ . If  $\hat{y}$  differs from the provided answer  $y$ , we extract the sample into our vision instruction preference dataset, treating the generated answer as the negative target  $T^-$  and the provided answer as the positive target  $T^+$ . For open-ended questions, we first employ LHR-S-Bot-Nova and Qwen2VL-7B to generate answers  $\hat{y}_1$  and  $\hat{y}_2$ , respectively, given the input image and

question. Subsequently, we prompt Qwen2VL-72B to evaluate the generated answers  $\hat{y}_1$  and  $\hat{y}_2$ , as well as the answer  $y$  from the source dataset, across five dimensions: *accuracy, completeness, clarity, relevance, and hallucination*. Qwen2VL-72B assigns scores (0-5) for each dimension and provides an overall confidence score (0-5) for the evaluation. We select samples where the confidence score exceeds 3 and designate the answer with the highest average score across the five evaluation dimensions as the positive target ( $T^+$ ). Both remaining answers are treated as negative targets ( $T^-$ ), which simultaneously enhances the diversity of negative samples and increases our training data volume. We do not directly consider the answer from the source dataset as  $T^+$  because many of them are generated by older VLMs and are often too short, meaningless, or contain incorrect statements. To manage costs associated with the large volume of data, we employ Qwen2VL-72B rather than GPT-4o as our evaluator. After filtering out parsing errors and applying human-defined quality rules, we curate a dataset of 26K vision instruction preference pairs. The detailed prompts used in this process are provided in Sec. 7.2.

During the process of collecting the question set  $Q$ , we observe that most data sources lack RS application-specific questions (e.g., questions related to agricultural and disaster analysis). To enhance ScoreRS’s performance on domain-specific content, we systematically develop a specialized question set comprising five manually crafted questions for each of 12 expert-defined RS image analysis categories. We then randomly sample 35K RS images from our deduplicated image dataset. For each image, we randomly select a question from the manually designed question set and prompt Qwen2.5-13B to rephrase the question, increasing the diversity of the manually generated questions. Subsequently, we employ LHRs-Bot-Nova and Qwen2VL-7B to generate answers given the image and rephrased question. Finally, we use the same prompt to elicit GPT-4o to evaluate the generated answers and select the highest-scoring answer as the positive target  $T^+$ , while treating the remaining answers as negative targets  $T^-$ . This process yield 33K RS-specific vision instruction preference data points after removing entries with parsing errors. The manually designed questions and categories can be found in Sec. 7.2.

## 2.2. Training and Application

**Training** Our ScoreRS model is initialized with Qwen2VL-7B, with the language head replaced by a linear layer to output a scalar score, following the standard value-head-based reward model [7, 37]. We do not consider generative score modeling due to efficiency concerns. Similar to the standard recipe for training large VLMs [20, 38], we implement a multi-stage training procedure to gradually train ScoreRS to distinguish better vision-language pair. In the first stage, we train the newly introduced value head

using a pure text preference dataset, UltraFeedback [6], to provide a good initialization for the value head. Then, we unfreeze the ViT and value head and train ScoreRS on our image-caption preference dataset to enable ScoreRS to better understand RS images. Finally, we unfreeze the LLM and conduct full-parameter training with our vision instruction preference dataset and the additional RLHF-V [46] dataset. This training approach enables ScoreRS to effectively identify high-quality outputs and assign higher scores to better responses across diverse RS scenarios.

**Application** Beyond basic data selection using ScoreRS, we also explore its usage for RL training and BoN selection. We primarily discuss implementing ScoreRS for RL, as its applications to data selection and BoN selection are straightforward. Our RL framework is based on group relative policy optimization (GRPO) [33], chosen for its computational efficiency and ease of hyperparameter tuning. RL training methods typically require a reward model to evaluate each action trajectory and update the model parameters to favor outputs with higher scores (i.e., responses more aligned with human preferences). While DeepSeek-R1 [11] demonstrated that rule-based rewards for close-ended questions with verifiable answers are effective for RL training, this approach is insufficient for RS applications. In the RS image understanding domain, most questions are open-ended, such as ‘Describe the urban development patterns visible in this satellite imagery,’ requiring nuanced interpretations rather than definitive answers. These questions also usually do not have verifiable answers, as different interpretations can lead to multiple acceptable responses. To address this challenge, we introduce a novel reward computation method that leverages ScoreRS for evaluating open-ended responses while incorporating rule-based rewards for close-ended questions, creating an approach that handles both question types. Specially, for close-ended questions, we employ a binary reward system (0 or 1) based on exact match or intersection over union (IoU) with the ground-truth answer. For open-ended questions, where we have a reference answer  $y$  (typically sampled from standard vision instruction datasets), we compute the reward  $r$  for a predicted answer  $\hat{y}$  using the following formulation:

$$r = \begin{cases} 1 - \exp(-(f_\theta(\hat{y}) - f_\theta(y)) \times \beta), & \text{if } f_\theta(\hat{y}) > f_\theta(y) \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where  $\beta > 0$  is a hyperparameter controlling the reward sharpness. We use this reference-based approach because it accelerates learning. The reason behind this approach is that the reference answer serves as a baseline, allowing the policy model to improve upon it by generating responses that receive higher scores. This method focuses on continuous improvement rather than selecting the least problematic

	NWPU@1	NWPU@5	EuroSAT@1	EuroSAT@5	fMoW@1	fMoW@5	AID@1	AID@5	SIRI-WHU@1	SIRI-WHU@5	WHU-RS19@1	WHU-RS@5	Avg.@1	Avg.@5
CLIP	65.31	93.23	42.14	89.20	<b>29.40</b>	<u>60.21</u>	64.11	91.21	58.11	85.17	86.24	99.21	57.55	86.37
RemoteCLIP (ALL)	65.70	93.89	42.74	86.54	18.14	44.63	<b>86.64</b>	<b>99.04</b>	72.67	96.63	<b>95.22</b>	<u>99.80</u>	63.52	86.76
RemoteCLIP (30% w. CLIP-Score)	<u>78.56</u>	97.37	<u>62.97</u>	<u>98.82</u>	26.71	58.05	83.31	98.25	<u>74.00</u>	98.29	94.33	99.72	<u>69.98</u>	91.75
RemoteCLIP (30% w. ScoreRS)	<b>78.58</b>	<b>97.54</b>	<u>63.67</u>	<b>99.01</b>	<u>29.29</u>	<b>60.70</b>	<u>85.14</u>	<u>98.41</u>	<b>74.21</b>	<b>98.87</b>	<u>94.95</u>	<b>100.00</b>	<b>70.97</b>	<b>92.42</b>

Table 1. Comparison of different finetuned CLIP models on classification tasks. Top-1 (@1) and top-5 (@5) classification accuracies are reported. Results are evaluated on official test sets when available; otherwise, the entire dataset is used for evaluation

	UCM T2I R@1	UCM T2I R@5	UCM I2T R@1	UCM I2T R@5	RSICD T2I R@1	RSICD T2I R@5	RSICD I2T R@1	RSICD I2T R@5	Avg. R@1	Avg. R@5
CLIP	29.44	66.84	36.67	85.24	5.31	17.09	3.75	11.99	15.78	45.29
RemoteCLIP (ALL)	37.66	80.11	<u>56.67</u>	<u>87.14</u>	12.49	35.74	<u>9.57</u>	<u>24.61</u>	25.19	56.90
RemoteCLIP (30% w. CLIP-Score)	<u>44.29</u>	<u>81.69</u>	56.19	<u>87.14</u>	<u>13.75</u>	<b>38.52</b>	8.78	23.24	<u>26.36</u>	<u>57.65</u>
RemoteCLIP (30% w. ScoreRS)	<b>44.56</b>	<b>82.09</b>	<b>57.90</b>	<b>88.40</b>	<u>13.90</u>	<u>38.02</u>	<b>9.59</b>	<b>24.88</b>	27.11	<b>58.35</b>

Table 2. Comparison of different finetuned CLIP models on cross-modal retrieval tasks. We report text-to-image (T2I) and image-to-text (I2T) performance using top-1 recall (R@1) and top-5 recall (R@5)

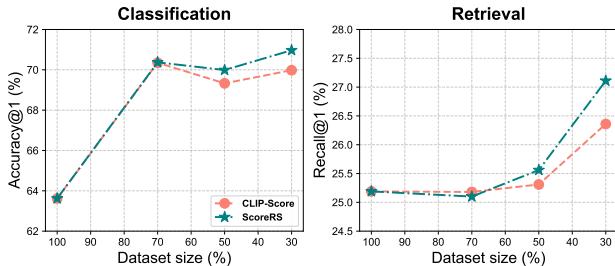


Figure 3. Classification and retrieval result using different percentages of data selected by CLIP-Score and ScoreRS. Results are reported as average scores across all evaluation datasets

option from multiple suboptimal alternatives that may be worse than the reference answer itself. We give our detailed unsuccessful attempts and our reasoning in Sec. 10.

### 3. Experiment

We conduct experiments to evaluate the effectiveness of ScoreRS across three key applications: (1) vision-language data selection for training VLMs, (2) RL training, and (3) BoN selector. Additionally, we analyze the impact of our proposed multi-stage training strategy and examine the quality of our curated preference dataset.

#### 3.1. Vision-Language Data Selection

##### 3.1.1. CLIP Finetuning

**Experimental Setting** To validate the effectiveness of ScoreRS for data selection, we apply our model to score and rank the training samples from RemoteCLIP [19], which are then used to finetune CLIP-ViT-L/14 [29]. We compare our approach against three baselines: (1) the CLIP model without finetuning, (2) CLIP finetuned on the complete, unfiltered dataset, and (3) CLIP finetuned on data filtered using CLIP-score. Detailed training hyperparameters are provided in Sec. 8.2.

**Main Result** We evaluate the finetuned models on both classification and retrieval tasks. For classification, we employ a fully zero-shot setting, ensuring that none of the test set categories appear in the finetuning datasets. The re-

sults, presented in Tab. 1 and Tab. 2, challenge the conventional wisdom of "more data yields better performance" in the context of RS-specific VLMs. Simple filtering using CLIP-score already yields improvements of 5% and 1% on classification and retrieval tasks, respectively. Our proposed ScoreRS filtering method further advances these improvements, achieving gains of 7% in classification and 2% in retrieval compared to training with the complete dataset, outperforming the CLIP-score-based filtering approach. These results support our initial hypothesis that current RS vision-language data are suboptimal, highlighting the necessity for automated quality control strategies.

We further evaluate different parameter-wise data selection methods by varying the selection thresholds. The results are presented in Fig. 3. Our analysis reveals distinct patterns for classification and retrieval tasks: retrieval tasks show higher sensitivity to noisy data, demonstrating more significant improvements with stricter filtering criteria, while classification tasks maintain reasonable performance even with relaxed selection criteria. Notably, quality filtering consistently improves model performance, aligning with previous research that highlights the crucial role of high-quality data in CLIP-style vision-language pretraining [44].

##### 3.1.2. Large VLMs Finetuning

**Experimental Setting** We finetune Qwen2VL-7B [38] using RS pretrained data and vision instruction datasets from VHM [27]. Following standard practices in large VLM training, we employ a two-stage finetuning approach: first training only the vision-language bridge layer using pretrained data, then training the underlying LLM using the vision instruction dataset. We investigate the effectiveness of ScoreRS in selecting high-quality data for both stages. As baselines, we compare our approach against models trained on the complete dataset and the data filtered using LongCLIP-L [48]. We opt for LongCLIP over the original CLIP model due to its ability to handle longer captions and conversations that are common in the chosen dataset. To reduce computational costs, we implement LoRA [13] for LLM finetuning in the second stage, using a rank of 8 un-

	RS Classification						RSVQA						Grounding	General Knowledge
	AID	METERML	NWPU	SIRI-WHU	WHU-RS19	Avg.	HR-Comp.	HR-Pres.	LR-Comp.	LR-Pres.	LR-R-U	Avg.	VG-DOIR	LHRS-Bench
Qwen2VL	66.13	63.54	62.35	70.79	87.40	70.04	75.60	63.30	75.47	62.00	73.00	69.87	11.87	64.78
Qwen2VL-FT	78.46	71.68	79.68	71.21	94.70	79.15	79.60	68.70	83.58	67.37	73.00	74.45	53.28	65.23
Qwen2VL-FT (CLIP-P30)	78.21	72.20	78.45	62.64	94.99	77.30	82.31	67.20	84.11	<b>69.43</b>	73.00	75.21	54.60	65.50
Qwen2VL-FT (ScoreRS-P30)	<b>79.30</b>	<b>72.61</b>	<b>80.29</b>	<b>72.63</b>	<b>95.40</b>	<b>80.05</b>	<b>85.30</b>	<b>70.60</b>	<b>85.47</b>	68.42	<b>76.00</b>	<b>77.16</b>	<b>55.22</b>	<b>66.24</b>
Qwen2VL-FT (CLIP-P30S30)	76.54	65.94	77.20	60.61	90.23	74.10	78.46	65.97	83.22	68.11	75.00	74.15	50.27	64.79
Qwen2VL-FT (ScoreRS-P30S30)	77.03	70.49	79.92	70.92	89.50	77.57	82.50	<b>69.20</b>	84.52	75.05	80.00	78.25	54.22	66.24
Qwen2VL-FT (CLIP-P30S60)	80.17	71.69	82.49	70.68	92.40	79.49	<b>82.60</b>	66.94	85.71	<b>92.67</b>	82.00	81.98	53.01	66.03
Qwen2VL-FT (ScoreRS-P30S60)	<b>85.66</b>	<b>74.86</b>	<b>89.49</b>	<b>73.33</b>	<b>92.80</b>	<b>83.23</b>	82.00	68.90	<b>89.68</b>	86.63	<b>87.00</b>	<b>82.84</b>	<b>55.58</b>	<b>66.58</b>

Table 3. Comparison of different data filtering methods and selection strategies. We report accuracy on the test sets across all datasets. CLIP and ScoreRS denote data selection using CLIP score and our ScoreRS, respectively. PX indicates the selection of top X% of pretraining data in the first stage, while SX represents the selection of top X% of vision instruction data in the second stage

	RS Classification						RSVQA						Grounding	General Knowledge
	AID	METERML	NWPU	SIRI-WHU	WHU-RS19	Avg.	HR-Comp.	HR-Pres.	LR-Comp.	LR-Pres.	LR-R-U	Avg.	VG-DOIR	LHRS-Bench
LLaVA-1.6-7B	52.83	44.78	44.70	59.08	69.30	54.14	68.60	64.40	64.32	56.84	61.00	63.03	41.59	64.78
InternVL-2.5-8B	64.50	57.17	59.17	57.66	80.90	63.88	75.50	65.80	71.16	66.21	72.00	70.13	15.39	65.86
Qwen2VL-7B	66.13	63.54	62.35	70.79	87.40	70.04	75.60	63.30	75.47	62.00	73.00	69.87	11.87	64.78
LHRS-Bot-Nova	83.06	72.74	83.97	72.21	96.20	81.64	<b>89.30</b>	<b>87.60</b>	88.11	83.89	79.00	85.58	31.51	52.46
GeoChat	73.47	34.87	89.37	53.04	85.30	67.21	83.30	59.10	90.52	<b>90.63</b>	<b>97.00</b>	84.11	19.77	36.23
VHM	<b>92.03</b>	74.33	<b>94.76</b>	70.62	<b>96.50</b>	<b>85.65</b>	83.30	68.30	90.11	89.89	87.00	83.72	55.99	33.04
SkysenseGPT	88.16	40.00	90.06	68.38	95.50	76.42	84.20	70.50	<b>92.11</b>	90.32	95.00	86.43	12.87	36.37
Qwen2VL-7B-RS	85.90	<b>74.42</b>	91.59	<b>74.75</b>	96.30	84.59	87.30	75.80	91.36	89.79	88.00	<b>86.45</b>	<b>58.34</b>	<b>67.08</b>

Table 4. Performance comparison between our finetuned model (Qwen2VL-7B-RS) and existing vision-language models. Our model is trained on quality-filtered data comprising the top 30% of pretraining samples and 60% of vision instruction samples from the original VHM datasets, selected using ScoreRS as the quality assessment model

less otherwise specified. Detailed experimental settings and hyperparameters are provided in Sec. 8.3.

**Main Result** We evaluate the finetuned model across multiple RS tasks, including image classification, visual grounding, and visual question answering. Additionally, we assess the model’s general RS knowledge using the more challenging LHRS-Bench benchmark [24]. The results are presented in Tab. 3. For the image-caption alignment dataset, utilizing ScoreRS to filter out low-quality data and retaining only the top 30% achieves the best performance, yielding a 1% improvement on the challenging LHRS-Bench. Given that the VHM image-caption dataset is synthetically generated by Gemini-Flash, this high ranking threshold further supports our assertion regarding the importance of data quality control for RS synthetic data. We further explore the application of ScoreRS for filtering vision instruction data. Since the VHM vision instruction dataset is primarily constructed from various standard RS benchmark datasets, we find that applying an overly aggressive filtering ratio (e.g., 30%) performs slightly inferior to using the complete vision instruction dataset. However, when adopting a more moderate filtering approach (e.g., 60%), we observe substantial improvements: 3% on classification tasks, 5% on vision question answering, and 0.36% on challenging grounding and LHRS-Bench benchmarks. Notably, our ScoreRS quality assessment model consistently demonstrates superior results compared to CLIP score-based selection methods across all evaluation scenarios.

Based on these empirical findings, we strategically select the highest-scoring 30% of pretraining data and 60% of vision instruction data as ranked by ScoreRS, while scaling

the LoRA rank size to 128. We then evaluate our finetuned model against leading RS-specific VLMs and state-of-the-art general-purpose vision-language models. As shown in Tab. 4, our model, trained on high-quality data selected by ScoreRS, not only achieves comparable or superior performance on classification, visual question answering, and visual grounding tasks compared to RS-specific VLMs, but also outperforms general-purpose models on the LHRS-Bench benchmark. These results further confirm that data quality in RS vision-language pairs represents the primary bottleneck limiting the full potential of VLMs for RS image understanding.

### 3.2. Reinforcement Learning

Building upon our fine-tuned Qwen2VL-7B-RS model, we explore the application of ScoreRS as a reward model in RL training. We randomly select 8K samples (4K open-ended and 4K close-ended) from the filtered VHM vision instruction dataset for the RL training process. Inspired by the success of Deepseek-R1 [11], we prompt Qwen2VL-7B-RS to engage in a two-step reasoning process: first thinking about the question, then providing an answer. This process is structured using special tokens: <think>, </think> for the reasoning step, and <answer>, </answer> for the response. For close-ended questions, we implement a binary reward function that assigns 1 for correct answers and 0 otherwise. For open-ended questions, we utilize ScoreRS for reward calculation as described in Sec. 2.2. Additionally, we implement a format reward to encourage proper prediction formatting. We evaluate performance on the more challenging vision grounding and LHRS-Bench tasks, as our manual analysis indicates that simpler classification and RSVQA tasks inadequately assess VLMs’ true capabilities

	VG-DOIR	LHRS-Bench
Qwen2VL-7B-RS	58.34	67.08
Qwen2VL-7B-RS-Zero (w/o ScoreRS)	58.66	66.05
Qwen2VL-7B-RS-Zero	59.64	67.21
Qwen2VL-7B-RS-SFT	59.21	66.34
Qwen2VL-7B-RS-R1 (w/o ScoreRS)	62.47	65.71
Qwen2VL-7B-RS-R1	<b>64.52</b>	<b>69.13</b>

Table 5. Performance comparison on VG-DOIR and LHRS-Bench across different RL-trained models. Qwen2VL-7B-RS-Zero denotes the model obtained by directly applying RL training to Qwen2VL-7B-RS. Qwen2VL-7B-RS-SFT refers to Qwen2VL-7B-RS fine-tuned with our manually generated reasoning data. Qwen2VL-7B-RS-R1 represents the model obtained by applying RL training to Qwen2VL-7B-RS-SFT. “W/O ScoreRS” indicates the variant trained using RL without ScoreRS-based reward calculation.

in RS image understanding. We establish two baselines: the non-RL-trained Qwen2VL-7B-RS and an RL-trained version using only close-ended answer rewards (without ScoreRS). Our detailed settings are provided in Sec. 8.4.

The results are presented in Tab. 5. Direct application of RL training to Qwen2VL-7B-RS (i.e., Qwen2VL-7B-RS-Zero) yields performance improvements of 0.20% on LHRS-Bench and 1% on VG-DOIR. Moreover, the comparison experiment demonstrates that using ScoreRS for reward calculation outperforms the variant without ScoreRS-based rewards, supporting the effectiveness of ScoreRS as a reward model.

During our experiments, we observed that the reasoning patterns of Qwen2VL-7B-RS-Zero were overly simplistic. To further maximize the benefits of RL training, we implemented a multi-stage approach. First, we used Qwen2VL-7B-RS-Zero to answer our collected RS-specific questions introduced in Sec. 2.1.2. We then manually reviewed and refined the reasoning responses from the generated results, creating a curated dataset of 2K manually annotated reasoning-answer pairs. Finally, we fine-tuned Qwen2VL-7B-RS using this curated dataset before applying RL training with the same training data used for Qwen2VL-7B-Zero. The resulting model, Qwen2VL-7B-RS-R1, demonstrates significant improvements with this strategy, achieving gains of over 5% on VG-DOIR and 2% on LHRS-Bench, while outperforming the variant trained without ScoreRS-based reward calculation. Our analysis indicates that the reasoning patterns have become more reasonable. Representative conversation examples are provided in Fig. 7.

### 3.3. Best-of-N Selection

We employ our ScoreRS model as a BoN selector. Specifically, for each question, we generate multiple candidate answers and used ScoreRS to score these question-answer pairs, ultimately selecting the highest-scoring an-

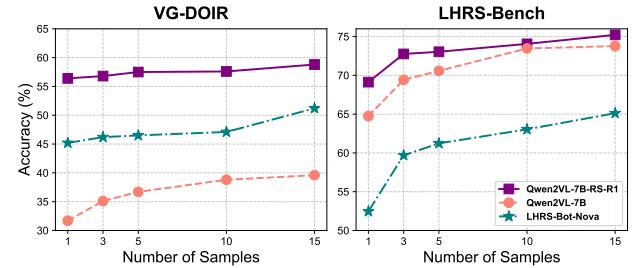


Figure 4. Performance improvements when using ScoreRS as a BoN selector. We employ ScoreRS to identify the optimal response from multiple generated candidates for each query.

swer for evaluation. To thoroughly evaluate the effectiveness of ScoreRS as a BoN selector, we utilize three models for answer generation: the RS-specific LHRs-Bot-Nova, the general-purpose Qwen2VL-7B, and our fine-tuned Qwen2VL-7B-RS-R1. We conduct evaluations on both VG-DOIR and LHRS-Bench datasets. Due to computational constraints and the large volume of the VG-DOIR dataset, we sample the first 1,000 instances for evaluation to reduce experimental costs. For all the answer generation, we set the sampling temperature and top-p parameters to 0.95 and 1, respectively.

The results are presented in Fig. 4, which clearly demonstrates that scaling computation at test time and employing our ScoreRS as a selector consistently improves evaluation performance across different models on both complex perception tasks (VG-DOIR) and holistic vision understanding tasks (LHRS-Bench). Notably, for the challenging LHRS-Bench dataset, using ScoreRS as the BoN selector enables accuracy to exceed 70% for the first time. This promising result further indicates the potential of ScoreRS not only for data selection but also for test-time scaling of base models in RS image understanding.

### 3.4. Ablation Analysis

To validate the effectiveness of our multi-stage training strategy and the quality of our preference dataset, we conduct ablation studies on a held-out evaluation set. The evaluation dataset comprises 6K samples, consisting of 2K randomly selected pairs from our image-caption preference dataset and 4K samples from the vision instruction preference dataset. We evaluate the performance using accuracy metrics, where success is defined as the scoring model assigning a higher score to the preferred text ( $T^+$ ) compared to its counterpart ( $T^-$ ).

We begin by validating the effectiveness of our preference dataset through joint training experiments. The joint training process consists of two stages: (1) an initial stage using pure text preferences to initialize the value head, which we find crucial for training stability, and (2) a subsequent stage where both the ViT and LLM are unfrozen. Our experiments demonstrate that training with our com-

	Accuracy @ ICPD	Accuracy @ VIPD	Accuracy
Jointly Training	83.13	81.07	82.09
✗ Image-Caption P.D.	59.44	76.30	67.87
✗ Vision Instruction P.D.	60.17	51.85	56.03
Multi-Stage Training	<b>92.91</b>	<b>93.03</b>	<b>92.97</b>

Table 6. Ablation study examining the impact of multi-stage training strategy and preference dataset composition. We first evaluate the contribution of each preference dataset component through joint training experiments, then demonstrate the benefits of our proposed multi-stage training approach compared to joint training alternatives. (P.D. = Preference Dataset, ICPD = Image-Caption Preference Dataset, VIPD = Vision Instruction Preference Dataset)

plete preference dataset yields the highest reward accuracy (Tab. 6). Notably, excluding any subset of our constructed dataset leads to decreased performance even when evaluating on the same domain (e.g., excluding the image-caption preference dataset results in inferior performance on the image-caption evaluation subset compared to joint training with all data). Furthermore, implementing our proposed multi-stage training strategy achieves best results, improving the reward accuracy by more than 10% compared to joint training. These results demonstrate both the effectiveness of our curated preference dataset and the clear benefits of our multi-stage training strategy.

## 4. Related Work

### 4.1. RS Vision-Language Data Curation

Unlike the general vision domain, where vision-language data can be readily crawled from abundant image-text interleaved webpages [4, 9, 30, 36, 44], the RS domain presents a unique challenge. While rich in open-source image data [23, 24, 35], RS lacks corresponding text descriptions and analytical conversations about these images. To harness the potential of VLMs in the RS domain, existing studies have primarily focused on two approaches for data curation: rule-based methods and synthetic data generation using established VLMs. For rule-based approaches, RS5M [51] explores the use of binary RS image classifiers and filtering methods to select RS image-text pairs from open-source general vision-language datasets. Similarly, RemoteCLIP [19], Skyscript [39], and SkysenseGPT [22] implement manually designed text templates and relation graphs for data construction. In parallel, studies such as VHM [27], LHRS-Bot [24], LHRS-Bot-Nova [18], and GeoChat [17] leverage flagship VLMs like GPT4 or Gemini for synthetic vision-language data curation. While these works have advanced the development of RS-specific VLMs, our practical applications reveal that data quality remains suboptimal, motivating the development of automated data selection methods.

### 4.2. RS VLMs

The remarkable success of VLMs in understanding images and engaging in complex tasks [21, 26] has inspired the development of specialized VLMs for RS image interpretation. Several approaches have focused on adapting CLIP [29] for the RS domain. RemoteCLIP [19], Skyscript [39], and RS5M [51] fine-tuned CLIP using carefully curated RS image-caption pairs, enabling robust RS zero-shot classification and image-text retrieval capabilities. For larger VLMs, GeoChat [17] pioneered the finetuning of LLaVA [20] with RS-specific vision instruction datasets. Subsequent works have expanded this paradigm through various strategies: incorporating volunteer geographic information (VGI) [24], integrating high-quality vision-language data [18, 27], modeling relational graphs [22], implementing ensemble vision encoders for vision-centric designs [50], and improving temporal understanding of RS images [14]. The versatile and successful applications of RS VLMs have directed our attention to the fundamental building blocks underlying these models: data quality and utilization. Rather than proposing new architectural designs, our work focuses on addressing how to improve data quality and enhance VLM performance within existing frameworks.

### 4.3. Parameter-Wise Data Selection

As data volume continues to expand and data complexity evolves, quality control has become increasingly challenging. Beyond traditional rule-based data selection methods [34, 40], researchers have shifted focus toward parameter-wise selection approaches for more sophisticated data curation. In the language domain, methods such as QaRater [41] and LESS [43] explore training dedicated scoring models or leveraging model gradients to select high-quality data for LLM pretraining and instruction tuning. Similarly, for vision-language data, Zhang et al. [49] investigated scoring models for curating image-text pairs. The model selection approach has been widely adopted in recent state-of-the-art VLMs, including Llama-3.2-Vision [10], DeepseekVL2 [42], and Qwen2VL [38], all of which implement scoring models for data filtering. Despite these advances, there is a notable absence of open-source scoring models specifically designed to evaluate the quality of RS vision-language data. In this work, we develop and release the first open-source scoring model tailored for RS vision-language data quality assessment.

## 5. Conclusion

Vision-language data serves as the fundamental building block for training VLMs. However, the quality control and data curation for RS-specific VLMs have not been fully addressed. In this study, we explore the development of parameter-wise scoring models for high-quality data selection. Through careful construction of RS preference

datasets and the training of our ScoreRS model, we demonstrate that current vision-language datasets are far from optimal. Our findings show that using just 30% of quality-filtered data achieves superior performance compared to the complete dataset in both CLIP training and large VLM fintuning. Furthermore, we investigate the application of ScoreRS in RL training and BoN selections. Both applications demonstrate that ScoreRS can enhance VLM capabilities in solving complex and challenging tasks. We anticipate that this study will encourage the RS community to place more emphasis on vision-language data quality control and the strategic utilization of high-quality data.

## References

- [1] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016. 1
- [2] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. 2
- [3] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation, 2023. 3
- [4] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. ShareGPT4V: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pages 370–387. Springer, 2024. 8
- [5] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024. 1, 3
- [6] Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, et al. Ultrafeedback: Boosting language models with scaled ai feedback. In *Forty-first International Conference on Machine Learning*, 2024. 4
- [7] Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. RLHF workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*, 2024. 4
- [8] Hongyuan Dong, Zijian Kang, Weijie Yin, Xiao Liang, Chao Feng, and Jiao Ran. Scalable vision language model training via high quality data curation. *arXiv preprint arXiv:2501.05952*, 2025. 1
- [9] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datadcomp: In search of the next generation of multimodal datasets. *Advances in Neural Information Processing Systems*, 36:27092–27112, 2023. 8
- [10] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024. 3, 8
- [11] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-R1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 4, 6
- [12] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022. 1
- [13] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 5
- [14] Jeremy Andrew Irvin, Emily Ruoyu Liu, Joyce Chui Chen, Ines Dormoy, Jinyoung Kim, Samar Khanna, Zhuo Zheng, and Stefano Ermon. Teochat: A large vision-language assistant for temporal earth observation data. *arXiv preprint arXiv:2410.06234*, 2024. 8
- [15] Aditi Jha, Sam Havens, Jeremy Dohmann, Alex Trott, and Jacob Portes. Limit: Less is more for instruction tuning across evaluation paradigms. *arXiv preprint arXiv:2311.13133*, 2023. 4
- [16] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 1
- [17] Kartik Kuckreja, Muhammad Sohail Danish, Muzammal Naseer, Abhijit Das, Salman Khan, and Fahad Shahbaz Khan. Geochat: Grounded large vision-language model for remote sensing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27831–27840, 2024. 2, 3, 8
- [18] Zhenshi Li, Dilxat Muhtar, Feng Gu, Xueliang Zhang, Pengfeng Xiao, Guangjun He, and Xiaoxiang Zhu. LHRSS-Bot-Nova: Improved multimodal large language model for remote sensing vision-language interpretation. *arXiv preprint arXiv:2411.09301*, 2024. 2, 3, 8
- [19] Fan Liu, Delong Chen, Zhangqinyun Guan, Xiaocong Zhou, Jiale Zhu, Qiaolin Ye, Liyong Fu, and Jun Zhou. Remoteclip: A vision language foundation model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 2024. 2, 5, 8
- [20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024. 1, 2, 3, 4, 8
- [21] Yadong Lu, Jianwei Yang, Yelong Shen, and Ahmed Awadallah. Omniparser for pure vision based gui agent. *arXiv preprint arXiv:2408.00203*, 2024. 8
- [22] Junwei Luo, Zhen Pang, Yongjun Zhang, Tingzhu Wang, Linlin Wang, Bo Dang, Jiangwei Lao, Jian Wang, Jingdong Chen, Yihua Tan, et al. Skysensegt: A fine-grained instruction tuning dataset and model for remote sensing vision-

- language understanding. *arXiv preprint arXiv:2406.10100*, 2024. 2, 3, 8
- [23] Dilxat Muhtar, Xueliang Zhang, Pengfeng Xiao, Zhenshi Li, and Feng Gu. CMID: A unified self-supervised learning framework for remote sensing image understanding. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–17, 2023. 8
- [24] Dilxat Muhtar, Zhenshi Li, Feng Gu, Xueliang Zhang, and Pengfeng Xiao. LHR-S-Bot: Empowering remote sensing with vgi-enhanced large multimodal language model. In *European Conference on Computer Vision*, pages 440–457. Springer, 2024. 2, 3, 6, 8
- [25] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. 2
- [26] Chenbin Pan, Burhaneddin Yaman, Tommaso Nesti, Abhirup Mallik, Alessandro G Allievi, Senem Velipasalar, and Liu Ren. Vlp: Vision language planning for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14760–14769, 2024. 1, 8
- [27] Chao Pang, Xingxing Weng, Jiang Wu, Jiayu Li, Yi Liu, Jiaxing Sun, Weijia Li, Shuai Wang, Litong Feng, Gui-Song Xia, and Conghui He. Vhm: Versatile and honest vision language model for remote sensing image analysis, 2024. 2, 5, 8
- [28] Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14532–14542, 2022. 3
- [29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 5, 8
- [30] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022. 8
- [31] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015. 1
- [32] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 1
- [33] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 2, 4
- [34] Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Author, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. Dolma: An open corpus of three trillion tokens for language model pre-training research. *arXiv preprint arXiv:2402.00159*, 2024. 8
- [35] Haifa Tamiminia, Bahram Salehi, Masoud Mahdianpari, Lindi Quackenbush, Sarina Adeli, and Brian Brisco. Google earth engine for geo-big data applications: A meta-analysis and systematic review. *ISPRS journal of photogrammetry and remote sensing*, 164:152–170, 2020. 8
- [36] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024. 8
- [37] Chenglong Wang, Yang Gan, Yifu Huo, Yongyu Mu, Muren Yang, Qiaozhi He, Tong Xiao, Chunliang Zhang, Tongran Liu, Quan Du, et al. Rovrm: A robust visual reward model optimized via auxiliary textual preference data. *arXiv preprint arXiv:2408.12109*, 2024. 2, 4
- [38] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 1, 2, 3, 4, 5, 8
- [39] Zhecheng Wang, Rajanie 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, 8, 1
- [40] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. Cnet: Extracting high quality monolingual datasets from web crawl data. *arXiv preprint arXiv:1911.00359*, 2019. 8
- [41] Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. Qurating: Selecting high-quality data for training language models. *arXiv preprint arXiv:2402.09739*, 2024. 8
- [42] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, et al. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024. 8
- [43] Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024. 8
- [44] Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. *arXiv preprint arXiv:2309.16671*, 2023. 5, 8
- [45] Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for reasoning. *arXiv preprint arXiv:2502.03387*, 2025. 4, 5
- [46] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng,

- Maosong Sun, et al. RLHF-V: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13807–13816, 2024. [4](#)
- [47] Yang Zhan, Zhitong Xiong, and Yuan Yuan. Rsvg: Exploring data and models for visual grounding on remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–13, 2023. [2](#)
- [48] Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-CLIP: Unlocking the long-text capability of clip. In *European Conference on Computer Vision*, pages 310–325. Springer, 2024. [5](#)
- [49] Lei Zhang, Fangxun Shu, Tianyang Liu, Sucheng Ren, Hao Jiang, and Cihang Xie. Filter & align: Leveraging human knowledge to curate image-text data. *arXiv preprint arXiv:2312.06726*, 2023. [8](#)
- [50] Wei Zhang, Miaoxin Cai, Tong Zhang, Yin Zhuang, and Xuerui Mao. Earthgpt: A universal multi-modal large language model for multi-sensor image comprehension in remote sensing domain. *IEEE Transactions on Geoscience and Remote Sensing*, 2024. [8](#)
- [51] Zilun Zhang, Tiancheng Zhao, Yulong Guo, and Jianwei Yin. Rs5m: A large scale vision-language dataset for remote sensing vision-language foundation model. *arXiv preprint arXiv:2306.11300*, 2023. [2](#), [8](#)
- [52] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v (ision) is a generalist web agent, if grounded. *arXiv preprint arXiv:2401.01614*, 2024. [1](#), [4](#)
- [53] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023. [4](#)

## Appendix

### 6. Additional Result

#### 6.1. CLIP Training

We extend our investigation by applying ScoreRS to select data from the larger-scale RS image-caption dataset Skyscript [39]. Except for the dataset used, all experimental details remain identical to our RemoteCLIP fine-tuning setup.

The results, presented in Tab. 7, further demonstrate that filtering with our ScoreRS yields superior performance compared to both using the complete dataset and applying CLIP-score filtering. We observe that directly fine-tuning CLIP with the entire Skyscript dataset actually resulted in performance inferior to the original CLIP model. We attribute this to the rule-based caption construction method used in Skyscript, which introduces significant redundancy and thus necessitates more careful data selection for effective fine-tuning.

#### 6.2. PPO Training

We explored using our ScoreRS directly as a reward model for RL through proximal policy optimization (PPO) [32]. We employ our finetuned Qwen2VL-7B-RS as the policy model, while the critic model is initialized with the same Qwen2VL-7B-RS architecture but with its language head replaced by a learnable linear layer. We utilize the same dataset described in Sec. 3.2 for this training process. LoRA is applied to all linear layers in the policy model with rank and  $\alpha$  parameters both set to 64. For text generation, we configure a sampling temperature of 0.95 and top-p of 0.9, with maximum new token generation limited to 512 tokens. The implementation uses a rollout batch size of 16, with PPO buffer size of 8 and 4 PPO epochs. We initialize the KL penalty coefficient at 2 and gradually increase it to 6 throughout the training process. The  $\lambda$  parameter for Generalized Advantage Estimation (GAE) [31] is set to 0.95. For optimization, we use a learning rate of  $1 \times 10^{-5}$  with a 0.1 warmup ratio and cosine learning rate decay schedule.

The results presented in Tab. 8 reveal that the model after PPO training performs inferiorly to its base variant across almost all evaluation datasets. We suspect this underperformance stems from suboptimal hyperparameter selection and reward calculation. Since we directly use our ScoreRS as reward model without any normalization or reference-based approach, this may be caused by reward hacking [1]. Moreover, given the considerable number of parameters associated with the PPO algorithm, we expect that developing comprehensive reward calculation strategies and conducting thorough hyperparameter searches remain important directions for future work.

### 7. Preference Data Generation

#### 7.1. Image-Caption Preference Dataset

**Caption Generation Setting** We employ three state-of-the-art VLMs—LHRS-Bot-Nova, Qwen2VL-7B, and InternVL-2.5-8B—to generate captions for the given images. For all models, we use the prompt: *Provide a factual description highlighting important details in this picture.* The generation temperature parameter for LHRS-Bot-Nova was set to 0.7, while for Qwen2VL-7B and InternVL-2.5-8B, we maintain their standard generation configuration parameters.

**Caption Judgment Setting** For the selection of captions in our preference pairwise data, we employ GPT-4o (GPT-4o-2024-05-13). The complete prompt used for this judgment is provided in Tab. 10.

**Data Samples** We provide several representative samples from our image-caption preference dataset in Fig. 5 to illustrate the quality of the collected data.

#### 7.2. Vision Instruction Preference Dataset

**Judgment Setting** We provide the prompt in Tab. 11 for using Qwen2VL-72B as judge to select the better answer for the input question.

**RS Specific Category and Question** We manually design RS specific questions across 12 categories: *basic visual recognition, spatial analysis, environmental assessment, urban analysis, agricultural analysis, disaster assessment, geological features, infrastructure analysis, temporal understanding, advanced reasoning, quantitative assessment, and color spectral analysis.* These domain-specific questions are presented in Tab. 12 and Tab. 13.

**Generation Prompt** We utilized LHRS-Bot-Nova and Qwen2VL-7B to answer the RS-specific questions. The prompt template employed for this question-answering task is presented in Tab. 14.

**Rephrase Prompt** We employ Qwen2.5-13B to rephrase the manually designed questions to enhance question diversity. The prompt used for this rephrasing process is presented in Tab. 15.

**Data Samples** We provide several samples from our vision instruction preference dataset in Fig. 6.

	NWPU@1	NWPU@5	EuroSAT@1	EuroSAT@5	fMoW@1	fMoW@5	AID@1	AID@5	SIRI-WHU@1	SIRI-WHU@5	WHU-RS19@1	WHU-RS@5	Avg.@1	Avg.@5
CLIP	<b>65.31</b>	<b>93.23</b>	42.14	89.20	<b>29.40</b>	<b>60.21</b>	64.11	91.21	<b>58.11</b>	85.17	<b>86.24</b>	<b>99.21</b>	57.55	86.37
Skyscript (ALL)	48.11	76.36	50.60	85.13	19.07	45.76	51.81	77.44	43.29	82.29	62.49	93.83	45.90	76.80
Skyscript (30% w. CLIP-Score)	60.53	90.34	58.60	93.44	26.36	52.28	59.67	84.40	47.75	84.50	80.60	98.01	55.59	83.83
Skyscript (30% w. ScoreRS)	63.43	92.54	<b>60.79</b>	<b>96.99</b>	29.32	59.69	<b>64.59</b>	<b>91.56</b>	55.02	<b>85.91</b>	84.27	98.96	<b>59.57</b>	<b>87.61</b>

Table 7. Comparison of different finetuned CLIP models on classification tasks

	RS Classification						RSVQA					Grounding	General Knowledge	
	AID	METERML	NWPU	SIRI-WHU	WHU-RS19	Avg.	HR-Comp.	HR-Pres.	LR-Comp.	LR-Pres.	LR-R-U	Avg.		
Qwen2VL-7B-RS	<b>85.90</b>	<b>74.42</b>	<b>91.59</b>	<b>74.75</b>	<b>96.30</b>	<b>84.59</b>	<b>87.30</b>	<b>75.80</b>	<b>91.36</b>	<b>89.79</b>	88.00	<b>86.45</b>	<b>58.34</b>	<b>67.08</b>
Qwen2VL-7B-RS-PPO	84.76	69.03	90.68	70.79	92.60	81.57	85.70	72.80	86.32	86.85	<b>92.00</b>	84.73	57.10	66.04

Table 8. Comparison between Qwen2VL-7B-RS and its PPO training variant

	Stage 1	Stage 2	Stage 3
Batch Size	64	16	
Weight Decay	0	0.1	
Learning Rate	$2 \times 10^{-5}$	$1 \times 10^{-6}$	
WarmUp Iter	500		
Epoch	1		
Gradient Accumulation	1	2	4

Table 9. Hyperparameter for training our ScoreRS

## 8. Experimental Setting

### 8.1. ScoreRS Training

We implement a three-stage training approach for our ScoreRS model. Throughout all stages, we employ the AdamW optimizer with cosine learning rate decay and set  $(\beta_1, \beta_2)$  to  $(0.9, 0.95)$ . For computational efficiency, we implement ZeRO-2 optimization with bfloat16 precision across all training stages. Additional hyperparameters are detailed in Tab. 9.

### 8.2. CLIP Finetuning

We utilize CLIP-ViT-L/14 from Hugging Face<sup>1</sup> as our base model. Across all fine-tuning experiments, we employ ZeRO-2 optimization with bfloat16 precision and the AdamW optimizer.

For fine-tuning on RemoteCLIP, we configure a batch size of 1,024, learning rate of  $1 \times 10^{-5}$ , weight decay of 1.0, and warmup iterations of 200. The model is fine-tuned for 5 epochs using cosine learning rate decay.

For fine-tuning on Skyscript, we use a larger batch size of 2,048, learning rate of  $1 \times 10^{-5}$ , weight decay of 0.01, and warmup iterations of 2,000. The model is fine-tuned for 20 epochs with a fixed learning rate schedule, reducing the rate by a factor of 0.316 at 80% and 90% of the training process.

**Evaluation Setting** We evaluate the fine-tuned CLIP model on both RS classification and RS image-text retrieval tasks. For the classification tasks, we employ text prompts in the format of “*a satellite photo of {class name}*” and “*a*

<sup>1</sup><https://huggingface.co/openai/clip-vit-large-patch14>

*satellite image of {class name}*” to perform the classification.

### 8.3. Large VLMs Finetuning

We select Qwen2VL-7B-Instruct as our base model and utilize the RS image-caption and vision instruction data from VHM as our finetuning dataset. Due to special token format differences between the VHM dataset and Qwen2VL-7B, we implement rule-based conversion methods to align the special tokens with Qwen2VL-7B requirements.

Our fine-tuning approach consists of two sequential stages: 1. In the first stage, we finetune the model using image-caption data while only unfreezing the vision-language connector. 2. In the second stage, we finetune using vision instruction data, applying LoRA adaptation to the base LLM while maintaining the unfrozen vision-language connector.

For both stages, we employ the AdamW optimizer with ZeRO-2 optimization strategy, bfloat16 precision, a maximum context length of 8,192, and cosine learning rate scheduling. Each stage is trained for 1 epoch.

Stage-specific hyperparameters were as follows:

- First stage: learning rate of  $8 \times 10^{-5}$ , batch size of 64, weight decay of 0.01, warmup ratio of 0.1, and maximum image resolution of 768.
- Second stage: learning rate of  $2 \times 10^{-4}$ , batch size of 64, weight decay of 0.01, warmup ratio of 0.03, and increased maximum image resolution of 1,024.

**Evaluation Setting** We adhere to the evaluation dataset split established in the VHM and utilize their task-specific prompts for each evaluation task. To ensure fair comparison, we augment each prompt with the appropriate task identifier when evaluating models that support task identifier. For response generation across all models and tasks, we maintain consistent hyperparameters with a temperature of 1.0 and top-p of 1.0.

### 8.4. GRPO Training

We implement the GRPO algorithm with our ScoreRS model for rewarding open-ended question responses. We select GRPO due to its simplicity and computational efficiency.



Figure 5. Representative examples from our image-caption preference dataset

For open-ended vision question answering tasks, we employ a binary accuracy reward: 1 if the prediction matched the ground truth, and 0 otherwise. For bounding box prediction tasks, we implement a graduated reward system based on IoU: 0.5 for  $\text{IoU} > 0.5$ , 0.6 for  $\text{IoU} > 0.6$ , 0.7 for  $\text{IoU} > 0.7$ , and 1.0 for  $\text{IoU} > 0.8$ , with 0 reward otherwise. The application of ScoreRS for close-ended questions has been previously described in Section 2.2, and the hyperparameter  $\beta$  in Equation 3 is set to 0.2.

We employ LoRA on all linear layers of the base LLM with a rank of 128 and  $\alpha$  of 128. During the generation phase of GRPO, we configure the sampling temperature and top-p parameter to 0.9, with a maximum token generation limit of 256. We set the KL penalty coefficient to 0.04 and use a batch size of 8, with each sample generating 5 candidate answers for reward calculation.

We optimize the model using AdamW with a cosine learning rate schedule and a base learning rate of  $8 \times 10^{-6}$ . To improve computational efficiency, we utilized bfloat16 precision, gradient accumulation with a factor of 4, and a maximum image resolution of 1024 pixels. Memory consumption is reduced by implementing Flash Attention in conjunction with ZeRO-2 optimization. All experiments are conducted for a single epoch.

For the supervised finetuning of Qwen2VL-7B-RS with our manually collected reasoning data, we employ identical hyperparameter settings as described in Sec. 8.3, with the LoRA rank set to 128.

**Evaluation Setting** The evaluation metrics, datasets, and framework for assessing the trained reasoning model remained consistent with those detailed in Sec. 8.3. The key distinction is the implementation of an answer extraction process from the reasoning model’s output for evaluation purpose. Importantly, if this parsing process failed to extract a valid answer from a reasoning model, we classify the response as incorrect.

## 9. Qualitative Example

We provide representative conversation examples generated by our finetuned model, Qwen2VL-7B-RS-R1, in Fig. 7.

## 10. Reasoning for Reference Based Reward Calculation

In Sec. 2.2, we introduce the reference-based reward mechanism using our ScoreRS for utilizing open-ended questions. Before arriving at this approach, we explored mul-

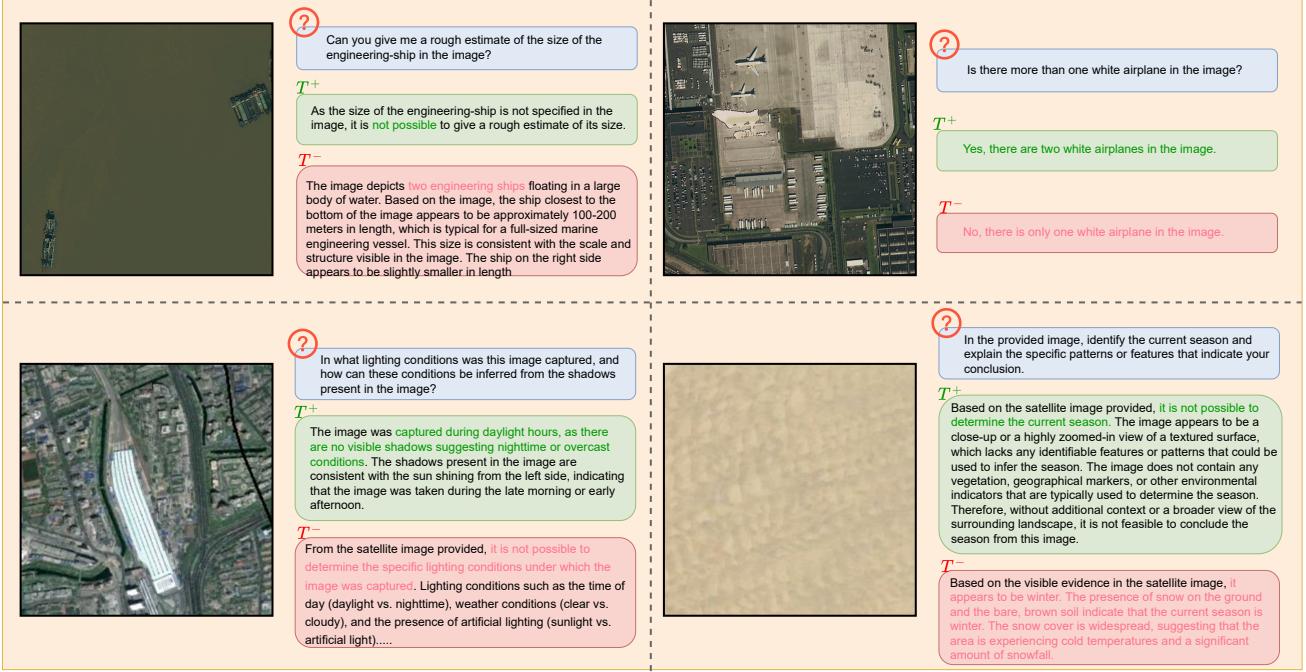


Figure 6. Representative examples from our vision instruction preference dataset

tiple strategies. The following describes our unsuccessful attempts and our reasoning:

**Direct Reward** We initially attempted to directly use the output of ScoreRS for rewarding. However, since the output of ScoreRS can be significantly larger or smaller than the 0-1 range, and we do not know the possible range of the ScoreRS output (therefore, could not normalize them), the training proved very unstable. Additionally, the rule-based reward for close-ended questions was greatly overshadowed.

**Function Normalization Reward** We then tried normalizing the ScoreRS output with sigmoid or other similar normalization functions ( $\tanh$ ). This approach was problematic because the range of our ScoreRS output likely falls in the convergence part of these functions (for example, in the  $> 10$  or  $< -10$  regions of the sigmoid function). Although we found ways to sketch the domain of the function, learning remained too slow and did not yield satisfying results.

With these unsuccessful attempts behind us, we opted for the reference-based reward calculation. Although the requirement for reference datasets may reduce the possible data size for RL training, we conclude that data volume is not the critical part of RL training—better answer sampling and reward strategies are far more important. Of course, we do not consider this the optimal solution, but rather our

current finding, and we hope future work will explore more effective approaches.

## 11. Limitation and Discussion

In this work, we demonstrate the efficacy of parameter-wise learned scoring models for RS vision-language data selection. Our investigation reveals that current RS vision-language datasets fall considerably short of optimal quality, underscoring the need for more rigorous curation efforts in this domain-specific context.

As the machine learning community increasingly validates the "less is more" principle [15, 45, 53], the RS community should prioritize data quality improvement through systematic quality control mechanisms or scoring model implementations. While our work highlights the suboptimal quality of existing RS vision-language data, we have not exhaustively explored what constitutes "better" or "more valuable" data for finetuning VLMs in the RS domain. Future research should investigate various dimensions including data difficulty levels, category distributions, and optimal data combinations to fully harness the potential of RS-specialized VLMs.

Throughout our investigation, we gained significant insights regarding the misalignment between current data characteristics and the capabilities expected from advanced VLMs. Modern large-scale VLMs excel not only in perception tasks but also in reasoning and planning for complex problem-solving [52]. For the RS community, these mod-

els should ideally support sophisticated applications such as disaster analysis, urban planning, and transportation system assessment. However, current datasets predominantly focus on basic perception capabilities—classification, object localization, and color detection—tasks that could potentially be addressed by more lightweight models.

We advocate for the development of VLMs that transcend basic perception tasks and deliver advanced analytical capabilities specific to RS applications. Furthermore, we hypothesize that exposure to more challenging, application-oriented RS analysis tasks could paradoxically enhance these models’ fundamental perception capabilities [45]. The creation of high-quality, application-specific datasets represents a critical direction for future work in this domain.

**Figure 7. Representative conversation examples demonstrating the capabilities of our finetuned Qwen2VL-7B-RS-R1 model**

The figure displays three vertical columns of conversational examples between a user and a AI model about satellite images. Each column contains a small image of a scene (rural landscape, port, and urban area) followed by a series of blue-bordered boxes representing the conversation.

- Column 1 (Top): Rural Landscape**
  - User: What seasonal indicators can you identify in this image? What time of year might this image have been captured?
  - Model: To identify seasonal indicators, I'll look for any visual cues such as vegetation, weather conditions, or animal activity. The image shows green fields and trees, which suggests a warmer season. However, the absence of snow or specific seasonal decorations makes it difficult to pinpoint a specific time of year. Therefore, I can only speculate that the image was captured during a warmer season.
  - User: Oops, I see. So, what type of land use or agricultural activity is primarily depicted in this image, and what features support this observation?
  - Model: To determine the type of land use or agricultural activity depicted, I'll consider the overall layout, the presence of buildings, and the surrounding fields. The image shows a large area of land with various buildings, silos, and feed bunkers, which are typical of livestock farming. The open fields and the absence of crops further support this observation. Therefore, I can conclude that the image primarily depicts livestock farming.
  - User: If this image were part of a time-series analysis, what changes in land use or infrastructure might you expect to observe over a 5-year period, and why?
  - Model: To predict changes in land use or infrastructure over a 5-year period, I'll consider factors such as economic trends, technological advancements, and environmental regulations. Over a 5-year period, I might expect to see changes in the size and layout of the facility, as well as the adoption of new technologies or practices. However, the image does not provide enough information to make a definitive prediction. Therefore, I can only speculate on potential changes based on general trends and considerations.
- Column 2 (Middle): Port**
  - User: Can you identify the number of ships docked in the image and describe their positions relative to the cranes and dock?
  - Model: To answer this question, I'll start by identifying the ships in the image. There are two distinct ships visible: one is docked near the upper left side of the image, and the other is further away on the bottom. The cranes are positioned over the dock, with one crane near the upper left ship and another near the bottom ship. The cranes are directly over the dock area, indicating they are used for loading and unloading cargo from the ships. The ships are positioned parallel to the dock, with the cranes strategically placed to facilitate the transfer of containers between the ships and the dock.
  - User: If the port appears busy with cranes actively engaged, what might be the time of day or economic activity level, and why might this be inferred from the image?
  - Model: To infer the time of day or economic activity level, I'll consider the context of the image. The presence of multiple cranes actively engaged in loading and unloading suggests a high level of activity. This could indicate a peak period for import/export operations, possibly during the morning or evening when shipping schedules are more frequent. The overall busy atmosphere of the port, with cranes in operation and containers being handled, supports the idea of a high economic activity level.
  - User: What is the spatial relationship between the water body and the container storage areas? How does the interface between land and water appear in this image?
  - Model: To analyze the spatial relationship between the water body and the container storage areas, I'll examine the layout of the image. The water body is located on the left side of the image, while the container storage areas are on the right side. The interface between land and water appears to be a dock area, where the cranes are positioned to handle cargo from the ships. The proximity of the water body to the dock suggests that the port is well-suited for maritime trade.
- Column 3 (Bottom): Urban Area**
  - User: Does this image is panchromatic or multispectral images? Why? And how would the counterpart image type of same area enhance our ability to distinguish between different urban features?
  - Model: To answer this question, I need to understand the difference between panchromatic and multispectral images. Panchromatic images capture light across a wide range of wavelengths, typically in grayscale. This type of image is useful for identifying patterns and shapes, but it lacks the color information necessary to distinguish between different urban features. A multispectral image, on the other hand, captures light at specific wavelengths, allowing for the identification of different materials and objects based on their unique spectral signatures. This type of image would enhance our ability to distinguish between different urban features, such as buildings, roads, and vegetation, by providing more detailed and accurate information about their appearance and composition.
  - User: So this is a panchromatic image, could you estimate the resolution of this image (give me an exact numeric resolution range, 0.5m-1m, 10m-20m, or other) and describe the spatial pattern of the buildings relative to the vegetation. What geometric arrangement do the buildings follow?
  - Model: To estimate the resolution of the image, I need to consider the scale of the objects visible. The image shows a bird's-eye view of a residential area with buildings and trees. The buildings are arranged in somewhat regular shapes, and the trees are also quite small in comparison. Given the level of detail and the size of the objects, it seems reasonable to estimate the resolution as being in the range of 0.5m to 1m per pixel. The spatial pattern of the buildings relative to the vegetation is quite clear: the buildings are arranged in a grid-like pattern, with trees interspersed between them. The buildings are aligned in straight lines, forming a geometric arrangement that is typical of planned urban or suburban areas.
  - User: Analyze the shadow patterns in this image. What can they tell us about the rough estimation about the height of buildings and the time of day when this image was captured?
  - Model: To analyze the shadow patterns, I'll start by observing the direction and length of the shadows. The shadows appear to be relatively short, indicating that the sun is high in the sky. This suggests that the image was likely taken during the middle of the day. The shadows are also consistent in length and direction across the image, which is typical for a clear day with no significant cloud cover. The shadows do not appear to be very long, which would be expected if the image was taken early in the morning or late in the afternoon. Considering the shadows and the angle of the sun, it seems reasonable to estimate that the image was captured around noon. The height of the buildings can be inferred from the length of the shadows relative to the buildings. The shadows are relatively short, which suggests that the buildings are not very tall. However, without specific measurements, it's difficult to provide an exact height estimate.
  - User: Analyze the shadow patterns in this image. What can they tell us about the rough estimation about the height of buildings and the time of day when this image was captured?
  - Model: The shadow patterns in this image indicate that the image was likely captured during the middle of the day, as the shadows are relatively short and consistent in length. The shadows suggest that the sun is high in the sky, which is typical for a clear day with no significant cloud cover. The shadows do not appear to be very long, which would be expected if the image was taken early in the morning or late in the afternoon. The height of the buildings can be inferred from the length of the shadows relative to the buildings. The shadows are relatively short, which suggests that the buildings are not very tall. However, without specific measurements, it's difficult to provide an exact height estimate.

**System Message:**

You are a highly precise and analytical image description evaluator. Your task is to select the most accurate caption from given options or provide your own if none are satisfactory. You should approach this task systematically and provide detailed justification while maintaining objectivity.

**Prompt :**

Please analyze the given image and evaluate these three captions:

Caption 1: {caption1}

Caption 2: {caption2}

Caption 3: {caption3}

Follow these steps:

1. First, provide your own detailed description of the image, focusing on:

- Observable objects and their characteristics
- Spatial relationships and positioning
- Key visual patterns and structures
- Colors and textures

Avoid making assumptions about image type or source.

2. Evaluate each provided caption using the following criteria (score 1-10):

- Accuracy: How well does it match the visible elements?
- Completeness: Does it capture all significant features?
- Conciseness: Is it appropriately detailed without unnecessary information?
- Objectivity: Does it avoid assumptions or hallucinated details?
- Spatial Clarity: How well does it describe object relationships and positions?

3. Provide your analysis in the following format:

[description]

your independent description

[end-description]

[evaluation]

1: [score],[score],[score],[score],[score],[justification]

2: [score],[score],[score],[score],[score],[justification]

3: [score],[score],[score],[score],[score],[justification]

[end-evaluation]

[selection]

best-id: [id]

best-reason: [reason]

[end-selection]

Table 10. Prompts for selecting image-caption preference pair

**Prompt :**

You are an expert evaluator specialized in determining the quality of answers. Your task is to systematically analyze and compare answers to determine which one is better. You must:

1. Always analyze the core requirements of the question first
2. Break down both the given answers into key components
3. Compare them systematically and objectively
4. Provide conclusions in a strictly formatted output and output your confidence

Your evaluation must be based on:

- Completeness: All key points covered
- Accuracy: Correctness of information
- Relevance: Direct address of the question
- Clarity: Clear communication of ideas
- Hallucination: Whether the answer is hallucinated

Now, please evaluate the following answer pair:

QUESTION:{question}

ANSWER: {answer1}

{answer2}

{answer3}

Provide your evaluation with the following structure and the result as json object:

```
““json
{ “answer1”:
{
“Completeness”: [0-5]
“Accuracy”: [0-5]
“Relevance”: [0-5]
“Clarity”: [0-5]
“Hallucination”: [0-5]
},
“answer2”: ...
“answer3”: ...
“confidence”: [0-5]
} ““
```

Table 11. Prompts for selecting vision instruction preference pair

#### *Basic Visual Recognition*

- What is the dominant land cover type in this image?
- How many distinct types of land cover can you identify in this image?
- What percentage of the image is covered by water bodies?
- Are there any clouds present in the image? If yes, approximately what percentage of the image is cloud-covered?
- What season does this image appear to be taken in? What visual cues support your answer?

#### *Spatial Analysis*

- What is the approximate scale of this image? (city-scale/regional/continental)
- Describe the spatial distribution of urban areas in relation to natural features.
- What patterns of human settlement can you observe? (clustered/dispersed/linear)
- How does the terrain influence the distribution of vegetation?
- Can you identify any transportation networks? How do they relate to urban development?

#### *Environmental Assessment*

- Are there any visible signs of environmental degradation?
- Can you identify potential areas of soil erosion?
- What evidence of water pollution can you observe?
- How healthy does the vegetation appear? What indicators are you using?
- Are there any visible impacts of climate change or extreme weather events?

#### *Urban Analysis*

- What is the predominant urban development pattern?
- Can you identify different types of urban land use (residential/commercial/industrial)?
- How well-connected is the transportation infrastructure?
- Are there clear boundaries between urban and rural areas?
- Can you identify any informal settlements or rapid urbanization patterns?

#### *Agricultural Analysis*

- What types of agricultural practices are visible in the image?
- How does field size and shape vary across the image?
- Can you identify any irrigation systems or water management features?
- What is the current stage of crop growth in the visible fields?
- Are there any visible patterns of crop rotation or fallow land?

#### *Disaster Assessment*

- What evidence of natural disasters can you observe?
- How has infrastructure been affected by the disaster?
- Can you identify areas at risk of future disasters?
- What emergency response activities are visible?
- How has the landscape changed post-disaster?

#### *Geological Features*

- What major geological formations are visible?
- Can you identify any fault lines or tectonic features?
- What types of erosional patterns are present?
- Are there any visible mining or extraction activities?
- How does geology influence vegetation patterns?

#### *Infrastructure Analysis*

- What types of energy infrastructure can you identify?
- How well-developed is the transportation network?
- Can you locate major water management facilities?
- What patterns of industrial development are visible?
- How does infrastructure density vary across the image?

#### *Temporal Understanding*

- What time of day was this image captured? What shadows or lighting conditions support your answer?
- Which season is represented in this image? What evidence supports this?
- What indications of recent urban development can you identify?
- What stage of vegetation growth is visible in different areas?
- What evidence of water level fluctuations can be observed from shoreline features?

Table 12. Manually designed questions for RS specific applications (Part 1)

#### **Advanced Reasoning**

Based on the visible patterns, what are the main economic activities in this region?

How do natural features constrain or enable human development?

What ecosystem services are visible in this image?

How sustainable are the visible land use practices?

What future development challenges might this area face based on current patterns?

#### **Quantitative Assessment**

What is the approximate area covered by different land use types?

How dense is the urban development in different parts of the image?

What is the ratio of built-up area to green space?

How fragmented are the natural habitats?

What is the distribution pattern of settlement sizes?

#### **Color Spectral Analysis**

What does the variation in vegetation color indicate about plant health?

Can you identify areas of bare soil based on color signatures?

What do the color patterns in urban areas suggest about building materials?

How do seasonal changes affect the spectral signatures of different features?

Table 13. Manually designed questions for RS specific applications (Part 2)

You are an expert remote sensing analyst. Examine the provided satellite image and answer the given question.

Remember to:

1. Start with direct observations
2. Base all conclusions on visible evidence only

If you cannot answer the question with the available information, please explicitly state what cannot be determined and explain specifically what prevents you from doing so.

Table 14. Instruction for prompting models to answer the RS specific questions

You are an expert in remote sensing and satellite image analysis. Transform the following question into a new, more diverse version. The new question should:

1. Test the same core concept but approach it differently
2. Either increase or decrease the complexity
3. Change the context or application domain
4. Use a different response format or analytical approach

Also please task care that there is just single image. So do not output any temporal question. Moreover, do not explicitly mention the image is satellite image.

Original question: "{question}"

Generate ONE new question that maintains the spirit of the original but differs in at least 2 of the above aspects. Do not explain your choices - only output the new question.

Output here:

Table 15. Instruction for prompting models to rephrase the questions