When Large Vision-Language Model Meets Large Remote Sensing Imagery: Coarse-to-Fine Text-Guided Token Pruning

Supplementary Material

A. Detailed Construction Process of LRS-VQA

In this section, we describe the construction process of LRS-VQA in detail, including data collection and filtering, label creation, and quality assurance.

A.1. Unique Object Extraction

Given the challenge of precisely referring to a specific unique object in large Remote Sensing Images (RSIs) (e.g., identifying a particular ship among hundreds of ships parking on the same harbor), we perform rule-based unique object extraction using object detection labels from our collected RS datasets [35, 37, 62]. The process is detailed below:

(i) We first filter out small images and calculate the total number of instances for each category in the image and remove categories with more than 40 instances per image.

(ii) For the remaining categories, we extract attributes like *absolute position*, *absolute size*, *relative position* and *relative size* within the same category. Based on this information, we determine whether an object is unique and create **unique reference** (e.g., "the top-most airplane" or "the only storehouse in the bottom-left corner of the image"). Note that multiple thresholds are set during this process. For example, a target is labeled as the distinguishable "largest" only if its area exceeds that of the second-largest target in the same category by more than 20%. Similarly, a target is marked as "right-most" within its category only if it is located farthest to the right and its offset distance from the next closest target is greater than 20 pixels.

(iii) Based on the above results, we crop the region containing the unique targets from the large RSI and draw a red box around the object as the visual prompt. For small targets, if the longer side of the target is less than 400 pixels, the cropping area is expanded by 400 pixels. For larger targets, we apply appropriate scaling to ensure that the longer side does not exceed 1400 pixels.

Finally, for each large RSI, we obtain local image patches containing unique targets, along with their corresponding unique references.

A.2. Question-Answer Pair Generation

Based on the above information, we additionally filter out extremely small objects (smaller than 16×16 pixels). Then we design prompts as in Tab. 1 to instruct GPT-4V to generate a diverse set of question-answer pairs. We carefully design the prompt to avoid generating questions about the entire image (e.g., counting targets across the whole image). Questions involving whole-image counting for specific categories are separately generated based on object detection labels.

During the initial generation of the VQA corpus, we observed that the answers to VQA questions were predominantly "yes" or "no". This could lead to the LVLM achieving high accuracy even without visual input. To address this issue, we carefully refine the prompts to guide GPT-4V in generating diverse and informative responses, constrained to a length of 1 to 3 words, while ensuring the responses are provided in an open-ended format. For the final version filtered by Qwen2-VL, we then conduct expert spot-checking and correction.

Additionally, during the Qwen2-VL-based quality inspection process and manual check, we observed that GPT-4V exhibits limitations in handling certain types of questions specific to remote sensing scenarios, such as object orientation. This indicates that even state-of-the-art LVLMs still need improvement when interpreting RSI.

B. More Experiment Details

B.1. Training Data Construction

We first filter out excessively large images from three RS datasets, as the DIP has a fixed 2-layer structure during training. To ensure the GSD range during training covers the dynamic range in inference, we apply different downsampling factors to maintain varied image sizes and construct multi-GSD inputs.

Subsequently, we create two types of questions: count-type questions based on object detection labels and relation-based questions using scene graph annotations. For count-type questions, we avoid querying categories with excessively high counts and control the proportion of samples where the answer is "1". For relation-based questions, we ask whether a specific relationship exists between any two categories in the image, with answers provided as "yes" or "no".

B.2. Implementation Details

Training setting. During training, we follow the same experimental setup of the SFT stage as LLaVA-1.5 and LLaVA-Next, with a global batch size of 128 and a learning rate of 2e-5. Additionally, the maximum length of Vicuna-1.5 and Qwen2 are 2048 and 16384, respectively. The overall SFT process is largely consistent with that of LLaVA-1.5, as our RFM is a plug-and-play module that can be

```
messages = [ {"role":"system", "content": f“““You are an AI visual assistant tasked with analyzing remote sensing images. Given the visual input (a part of a large image) and corresponding object information, your job is to create a list of question-answer pairs around the target object and its surroundings. Each sentence should unambiguously refer to the object based on the ‘why unique’ information.
Finally, you need to return [‘qa-pairs’: [‘ques-id’: question id, ‘question’: question, ‘type’: question type, ‘answer’: answer]] in JSON format. Do not return any notes after the JSON. The target object is highlighted by a red rectangle in the given image patch, and the ‘why-unique’ provides how to refer it in the original large image, you need to rely on this information to ask questions.

1. Based on all visible elements and object information, ask 5-10 questions of various types, including object existence, object relation, complex reasoning, and object status. Avoid questions about color, object shape and object orientation. Additionally, questions requiring reasoning should involve multifaceted analytical thought processes (e.g., analyzing object distribution patterns) based on the target object and its surroundings. Possibly include objects that are not provided, such as houses, roads, water and trees if they are obvious and non-ambiguous.
2. Ensure each question has a definite answer without any ambiguity, and answer each question using a single word or phrase, no more than 3 words.
3. Only ask questions about clear answers; avoid uncertainties or unclear elements, such as unknown, uncertain, some, or several.
4.Avoid question formats that only allow for two options or overly simplistic responses (e.g., ‘Yes’ or ‘No’).
5. Do not mention the red highlight box or asking about the target object’s category—consider it known.
6. Use complete information from ‘why-unique’ to ensure unique reference.
Follow the above guidelines and ensure consistency with the provided category.”””}
]messages.append({"role":"user", "content":‘\n’.join(query)})
```

Table 1. The prompt to GPT-4V for generating question-answer pairs about the unique objects in the large-size RSIs.

seamlessly integrated into the SFT process of any modular LVLM.

More details about coarse-to-fine token pruning. Our method involves stopping at a specific layer p of the DIP during traversal, pruning based on the RFM output of that layer, and then concatenating the pruned vision tokens with the vision tokens from the thumbnail view, along with text instructions, to form the multimodal input for the LLM. Although only a subset of image tiles is selected when traversing the DIP, we pad the unselected image tiles after the vision encoder and then add the “image newline” delimiter along with global position embeddings. Subsequently, we extract the vision tokens corresponding to I_{key}^{p+1} and the “image newline” positions from the fully padded tensor, which serve as $T_{\text{vis}}^{p+1,hr}$. The definitions of these symbols are consistent with those in the main paper.

Maximum sequence length of LLM. For Vicuna-1.5, due to the limitations of its pre-trained weights, it is challenging to train its long-context processing capability from scratch. Additionally, its performance often degrades when extrapolating to longer sequences. As for Qwen2, we have not yet explored the impact of extending the maximum sequence length to cover all vision tokens from original resolution imagery, primarily due to the significant time and resource costs. Moreover, our method does not rely on enhancing long-sequence processing capabilities to handle large images but rather serves as a strategy to improve the perception performance of existing modular LVLMs.

More details of comparison methods.

During the comparison of token reduction methods, we strive to maintain consistency or make necessary adaptations due to the differing principles of each approach. Specifically, for PruneMerge++ [58], CLIP-based pruning, RemoteCLIP-based pruning, and our RFM-based pruning, we set the token retain ratio to 25%. For VisionZip [77], the number of retained tokens is set to 64. For PyramidDrop [72] under the Qwen2 backbone, the pruning layers are configured to [7, 14, 21]. For all other settings of FastV [8] and PyramidDrop, we adhere to their original implementations.

B.3. Flash Attention and Multi-Turn Support

Our method follows existing token pruning approaches [72, 88] and is compatible with the use of flash attention as well as multi-turn dialogue

For flash attention, following SparseVLM [88], we introduce an additional forward pass that incorporates a specially designed value matrix. This allows us to extract the mean value of the processed attention map without explicitly computing the full attention map.

For multi-turn conversations, our method offers advantages over existing grid-based dynamic high-resolution approaches. These methods typically rely on pre-defined grids to partition and store all corresponding image tiles, which can be memory-intensive. In contrast, our approach only caches the features from the first two layers of the DIP (i.e., the vision tokens from the thumbnail view and the first group of image tiles). For higher-resolution image tiles, we maintain a dynamic selection strategy, extracting features only for text-related key tiles. This significantly reduces memory overhead while preserving efficiency in multi-turn scenarios.

C. Detailed Efficiency Calculation

Detailed calculation of vision tokens. For vision tokens number calculation, a 4,000×4,000 image generates thumbnail view and 144 image tiles after processing with anyres-p144, resulting in $(144+1)\times 576 = 83,520$ vision tokens. Among these, the tokens from the image tiles are downsampled using bilinear interpolation in the anyres strategy, ultimately yielding $144 \times 144 + 576 = 21,312$ that are fed into the LLM. This accounts for the number of vision tokens reported for anyres-p144, FastV, and PyramidDrop. For PrunMerge++ and VisionZip, we calculate the number of vision tokens based on their respective compression strategies.

For our method, we fix a 4-layer pyramid for the 4,000×4,000 input. Assuming that the thumbnail view and image tiles of the first three layers are fully utilized, this generates $(1+9+36)\times 576 = 26,496$ vision tokens. For the fourth layer of the DIP, as our strategy dynamically selects image tiles based on the output of the RFM, we computed the average number of image tiles generated in the fourth layer for all images close to 4,000×4,000 in the datasets. This average is 50, resulting in $50 \times 576 = 28,800$ vision tokens. Therefore, the total number of vision tokens processed by the vision encoder in our method is $26,496 + 28,800 = 55,296$. After token pruning with the ratio 0.25, the token number to LLM is $50 \times 144 \times 0.25 + 576 = 2,376$.

Detailed calculation of TFLOPs. We follow PyramidDrop to calculate the TFLOPs during inference. For PyramidDrop and FastV, we compute the TFLOPs by adapting the formulas provided in their respective papers, adjusting the number of vision tokens accordingly.

For our method, we account for the 4 layers in RFM and the 28 layers in Qwen2-7B. The FLOPs of the multi-head attention and the feedforward network modules are represented as $4nd^2 + 2n^2d + 2ndm$, where n is the number of vision tokens fed into the LLM, and d, m are 3584 and 18944 in Qwen2-7B, respectively. The number of vision tokens input to RFM in DIP is calculated as $9 \times 144 + 576 = 1,872$ (layers 1-2), $36 \times 144 + 576 = 5,760$ (layer 3), and $50 \times 144 + 576 = 7,776$ (layer 4), respectively. Based on these values, the FLOPs $\mathcal{F}$ of our method is computed as follows:

$$\begin{aligned}
\mathcal{F} = 4 \cdot \Big(&4 \cdot 1872 \cdot d^2 + 2 \cdot 1872^2 \cdot d + 3 \cdot 1872 \cdot d \cdot m\Big) \\
&+ 4 \cdot \Big(4 \cdot 5760 \cdot d^2 + 2 \cdot 5760^2 \cdot d + 3 \cdot 5760 \cdot d \cdot m\Big) \\
&+ 4 \cdot \Big(4 \cdot 7776 \cdot d^2 + 2 \cdot 7776^2 \cdot d + 3 \cdot 7776 \cdot d \cdot m\Big) \\
&+ 28 \cdot \Big(4 \cdot 2376 \cdot d^2 + 2 \cdot 2376^2 \cdot d + 3 \cdot 2376 \cdot d \cdot m\Big) \\
= 36.61\,T \qquad (1)
\end{aligned}$$

D. More Experimental Results

Different distillation losses. We explored different combinations of distillation losses, as shown in Tab. 2. It can be observed that the KL loss plays a crucial role, while the MSE loss applied to the high-resolution vision tokens also contributes to performance improvement.

KL	MSE	Color	Count	Pos	Acc
✓		44.70	29.04	49.56	41.20
	✓	42.07	30.75	46.14	39.73
✓	✓	44.70	31.00	49.72	**41.89**

Table 2. Ablation study on different losses used in attention distillation, under LLaVA-Next-Qwen2.

Different pruning ratios. As shown in Tab. 3, we employ multiple higher-resolution grids under the anyres strategy: anyres-p36 and anyres-p49. Then we conduct experiments with different pruning rates based on RFM-based pruning. The results indicate that the performance of the anyres baseline degrades as the supported image size increases, whereas our pruning method achieves consistent improvements. We attribute this to the fact that larger image sizes introduce more irrelevant background information, while our method effectively drops unnecessary vision tokens.

Different high-resolution vision token process. Additionally, for the DIP layers that have already been traversed (i.e., the layers before the final pruning layer), we attempt to prune their vision tokens as well. Specifically, we concatenate the pruned vision tokens from all traversed DIP layers, as shown in Tab. 4, the performance slightly decreases, possibly due to interference caused by the mixed resolutions of vision tokens across multiple hierarchical DIP levels.

Training with pruning. We further explore the effects of directly applying pruning during the SFT stage, as shown in Tab. 5. During training, the pruning ratio is set to 0.75, and the LLM observes the pruned vision tokens. Additionally, we experiment with feeding the text portion (i.e., fusion text) of the hidden states output by the RFM to the

Method	Drop Ratio	Color	Count	Pos	Acc
anyres-p36	0%	42.31	29.77	44.71	39.00
w/ prune (Ours)	25%	42.71	30.10	45.19	39.41
	50%	42.79	31.40	47.49	**40.64**
	75%	41.75	30.91	48.45	40.45
anyres-p49	0%	40.00	30.34	44.95	38.50
w/ prune (Ours)	25%	40.72	30.91	46.38	39.41
	50%	40.64	31.65	48.61	**40.37**
	75%	41.75	30.51	48.37	40.29

Table 3. Ablation study on different prune ratios with LLaVA-Next-Qwen2 with larger resolutions, when pruning vision tokens based on RFM results without DIP. "pX" indicates the max number is X for the pre-defined grids.

Setting	Color	Count	Pos	Acc
concat	44.22	31.48	48.21	41.39
select	44.70	31.00	49.72	41.89

Table 4. Ablation study on different high-resolution vision token processing, under LLaVA-Next-Qwen2. "concat" means concatenating all pruned vision tokens from all traversed DIP layers as high-resolution part. "select" means selecting the pruned tokens of the final pruning layer as high-resolution part, which is employed in the main paper.

Prune	Fus. Text	Color	Count	Pos	Acc
✓	×	41.91	26.02	49.01	39.09
✓	✓	40.96	31.97	44.63	39.25
×	×	44.70	31.00	49.72	41.89

Table 5. Ablation study on training with vision token pruning and text fusion. "Fus. Text" refers to replacing the original text tokens with the text portion (i.e., fusion text) output by the RFM when feeding into the LLM.

LLM, represented as "Fus. Text" in Tab. 5. The results indicate that introducing token pruning during training leads to a performance drop. We think this is because the model lacks access to complete image information, which impairs its ability to accurately perform text-aware region localization, resulting in inferior performance compared to standard SFT.

It is important to note that our method cannot utilize the fusion text setting in Tab. 5. Because under such a setting, during training, the LLM receives fusion text along with full vision tokens as input, whereas during inference, the LLM receives fusion text along with pruned vision tokens. This creates an inconsistency between training and inference.

Different layer-pairs in distillation. We explore the impact of different RFM-LLM layer-pairs used for distillation, as shown in Tab. 6. Specifically, for the 28-layer Qwen2-7B, we observed that the text-related attention localization is most accurate in its deep layers (approximately layers 14–24) when answering questions about both the global content and local details of the large RSIs.

RFM Layers	LLM Layers	Color	Count	Pos	Acc
3	[1,8,14]	41.35	28.30	47.89	39.27
4	[1,5,11,14]	**44.70**	31.00	**49.72**	**41.89**
4	[1,7,13,18]	40.88	**32.63**	49.48	41.06
4	[1,7,14,20]	41.83	28.22	47.89	39.41
5	[1,5,10,12,14]	43.98	29.36	48.13	40.58
6	[1,3,6,9,12,14]	42.31	31.08	47.18	40.26
6	[1,4,8,12,16,20]	43.75	30.51	48.13	40.88
6	[1,5,10,15,20,24]	42.31	31.08	47.18	40.26

Table 6. Ablation study on different RFM-LLM layer-pair configurations in MME-RealWorld-RS, with LLaVA-Next-Qwen2. "RFM Layers" means the number of layers in RFM.

From Tab. 6, it can be observed that as the number of LLM layers for distillation increases, it becomes more challenging for the RFM to learn precise text-aware localization capabilities. Although increasing the number of layers in the RFM can enhance its learning ability, it also raises the cost of training and inference. Moreover, the RFM doesn't need to possess highly accurate localization capabilities in the shallow layers of DIP, it only needs to provide rough positions to index image tiles of the next DIP layer. Additionally, when used for pruning, the vision tokens from key image tiles already narrow down the scope, making it sufficient to recognize general background information. Furthermore, retaining a certain number of context tokens can actually be beneficial for certain types of questions.

Detailed comparison with baselines. We provide a more comprehensive comparison with two simple but vital baselines: CLIP-L14 and RemoteCLIP-L14. Specifically, under identical anyres or DIP settings, we partition the image using a sliding-window approach (336×336 for CLIP and 224×224 for RemoteCLIP). Then we compute the similarity map between the input text and the image features for each image tile, which are subsequently stitched together to form a complete heatmap that guides the token pruning.

Tab. 7 presents a comparison of localization accuracy against these baselines across three datasets. While conceptually simpler, these baselines struggle due to their limited capacity to understand complex referring expressions and the lack of global perception inherent in the sliding-window mechanism. This highlights the importance of distilling knowledge from the LLM's attention, which enables our RFM to grasp complex semantics.

Furthermore, Tab. 8 shows the performance and FPS comparison under the same pruning setting (LLaVA-Next-p25). The accuracy trends observed here are largely con-

Pruning Guidance	LRS-FAIR	LRS-Bridge	LRS-STAR	DIOR-RSVG	RRSIS-D
Teacher LLM Attn.	53.03	58.66	56.73	74.81	73.88
CLIP Sim.	23.87	16.73	19.82	29.10	27.53
RemoteCLIP Sim.	37.04	22.99	32.21	43.12	41.81
RFM Attn. (Ours)	**47.89**	**49.08**	**47.01**	**64.76**	**61.17**

Table 7. Localisation recall (%) of different pruning guidance.

Setting	MME-RW-RS	FPS	LRS-FAIR	LRS-Bridge	LRS-STAR	FPS
LLaVA-Next-p25	39.65	0.188	20.99	36.38	26.18	0.176
w/ CLIP Sim.	36.86	0.171	18.12	32.30	24.46	0.162
w/ RemoteCLIP Sim.	38.77	0.148	20.36	34.24	25.32	0.139
w/ RFM (Ours)	**41.28**	0.165	**21.65**	**37.55**	**26.83**	0.157

Table 8. VQA accuracy (%) and FPS with different token pruning guidance methods. FPS on LRS-VQA averaged across 3 datasets.

sistent with the localization accuracy results. It is worth noting that although RemoteCLIP enhances performance on remote sensing imagery, its smaller input size of 224×224 necessitates partitioning the image into more tiles, which adversely affects inference speed on large images.

Detailed results on LRS-VQA. The complete leaderboards on the three parts of LRS-VQA are shown in Tab. 9, Tab. 10 and Tab. 11, respectively. Notably, on questions that require more global-scale perception capabilities, such as rural/urban classification, high-resolution LVLMs do not necessarily outperform low-resolution LVLMs. Additionally, the language preference inherent in the LVLM itself can influence its performance when answering open-ended questions, as it must precisely describe the corresponding vocabulary or its synonyms. Overall, LVLMs like Qwen2-VL, LLaVA-OV, and IXC-2.5, which are trained on large datasets and utilize higher resolutions, also demonstrate strong performance in large RSI perception tasks.

Method	**Max Res.**	count	category	shape	status	reasoning	rural/urban	**OverAll**
LLaVA1.5	336×336	10.50	13.44	7.37	7.75	25.25	48.25	18.76
SLiME	672×1,008	14.25	9.82	8.07	9.25	24.50	36.75	17.11
SEAL	-	7.00	12.50	3.50	20.50	28.75	55.50	21.29
LLaVA-FlexAttn	1,008×1,008	9.50	10.59	6.32	20.75	24.00	46.25	19.57
MGM-HD	1,536×1,536	15.50	12.66	9.47	6.00	23.75	40.00	17.90
LLaVA-UHD-v2	672×1,008	17.50	11.63	9.04	23.00	26.25	49.50	22.82
IXC-2.5	4,096×4,096	22.75	15.25	15.50	22.00	26.50	49.50	25.25
LLaVA-OV	2,304×2,304	16.25	19.90	8.77	12.75	27.25	38.75	20.61
Qwen2-VL	3,333×3,333	22.50	15.25	12.28	10.00	24.25	58.50	23.80
Geochat	504×504	13.50	8.01	14.04	3.50	19.75	62.25	20.18
RSUniVLM	336×336	21.00	11.37	15.98	2.00	25.00	50.75	21.02
Claude-3.5-Sonnet	-	11.75	4.12	16.84	1.50	15.00	28.50	12.95
Gpt-4o-mini	-	12.75	11.37	11.37	12.50	19.75	44.25	18.67
Gpt-4o	-	16.00	13.44	14.98	18.00	24.00	46.50	22.15

Table 9. Detailed results on LRS-FAIR.

Method	**Max Res.**	count	background	color	rural/urban	**OverAll**
LLaVA1.5	336×336	6.50	18.37	38.79	59.13	30.70
SLiME	672×1,008	13.50	18.78	34.55	61.51	32.09
SEAL	-	0.00	31.50	36.00	71.50	34.75
LLaVA-FlexAttn	1,008×1,008	4.50	17.96	37.58	59.92	29.99
MGM-HD	1,536×1,536	12.25	18.78	52.73	59.92	35.92
LLaVA-UHD-v2	672×1,008	5.00	17.55	44.24	63.49	32.57
IXC-2.5	4,096×4,096	14.50	20.30	55.34	63.49	38.41
LLaVA-OV	2,304×2,304	4.50	20.82	57.58	57.54	35.11
Qwen2-VL	3,333×3,333	15.50	20.41	57.58	59.00	38.12
Geochat	504×504	8.75	11.84	22.42	55.16	24.54
RSUniVLM	336×336	10.25	13.06	43.64	63.49	32.61
Claude-3.5-Sonnet	-	5.25	11.43	33.33	56.75	26.69
Gpt-4o-mini	-	4.75	17.96	50.97	54.29	31.99
Gpt-4o	-	14.75	18.37	43.03	51.19	31.84

Table 10. Detailed results on LRS-Bridge.

Method	**Max Res.**	count	category	color	shape	status	reasoning	rural/urban	**OverAll**
LLaVA1.5	336×336	10.75	18.67	36.50	10.00	11.33	15.67	55.50	22.63
SLiME	672×1,008	12.75	18.33	41.50	10.50	12.67	16.00	49.17	22.99
SEAL	-	0.25	20.50	33.50	10.00	10.50	17.75	56.50	21.29
LLaVA-FlexAttn	1,008×1,008	11.50	17.00	36.17	8.67	13.33	14.83	57.83	22.76
MGM-HD	1,536×1,536	15.50	12.66	49.50	18.75	10.30	10.18	24.00	20.13
LLaVA-UHD-v2	672×1,008	16.25	18.67	43.50	15.67	14.83	16.17	57.50	26.08
IXC-2.5	4,096×4,096	15.75	23.50	48.00	16.50	13.33	17.50	56.50	27.30
LLaVA-OV	2,304×2,304	9.75	25.33	51.00	14.00	10.17	18.83	53.50	26.08
Qwen2-VL	3,333×3,333	19.25	22.83	46.50	11.17	13.00	18.33	64.00	27.87
Geochat	504×504	13.50	8.01	24.75	10.75	5.42	14.04	19.75	13.75
RSUniVLM	336×336	8.00	13.50	51.67	25.17	4.50	14.00	56.17	24.72
Claude-3.5-Sonnet	-	6.34	6.00	41.67	1.67	12.33	3.00	22.00	13.29
Gpt-4o-mini	-	10.78	20.67	40.67	15.17	14.50	20.33	58.83	25.85
Gpt-4o	-	11.78	21.50	48.17	23.83	12.50	20.50	53.50	27.40

Table 11. Detailed results on LRS-STAR.