
RINGMO-AGENT: A UNIFIED REMOTE SENSING FOUNDATION MODEL FOR MULTI-PLATFORM AND MULTI-MODAL REASONING

Huiyang Hu, Pejin Wang, Yingchao Feng, Kaiwen Wei, Wenxin Yin, Wenhui Diao, Mengyu Wang, Hanbo Bi, Kaiyue Kang, Tong Ling, Kun Fu, Xian Sun

Aerospace Information Research Institute, Chinese Academy of Sciences

School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences
University of Chinese Academy of Sciences

Key Laboratory of Target Cognition and Application Technology (TCAT)

{huhuiyang22, wangpeijin17}mails.ucas.ac.cn

ABSTRACT

Remote sensing (RS) images from multiple modalities and platforms exhibit diverse details due to differences in sensor characteristics and imaging perspectives. Existing vision-language research in RS largely relies on relatively homogeneous data sources. Moreover, they still remain limited to conventional visual perception tasks such as classification or captioning. As a result, these methods fail to serve as a unified and standalone framework capable of effectively handling RS imagery from diverse sources in real-world applications. To address these issues, we propose RingMo-Agent, a model designed to handle multi-modal and multi-platform data that performs perception and reasoning tasks based on user textual instructions. Compared with existing models, RingMo-Agent 1) is supported by a large-scale vision-language dataset named RS-VL3M, comprising over 3 million image-text pairs, spanning optical, SAR, and infrared (IR) modalities collected from both satellite and UAV platforms, covering perception and challenging reasoning tasks; 2) learns modality adaptive representations by incorporating separated embedding layers to construct isolated features for heterogeneous modalities and reduce cross-modal interference; 3) unifies task modeling by introducing task-specific tokens and employing a token-based high-dimensional hidden state decoding mechanism designed for long-horizon spatial tasks. Extensive experiments on various RS vision-language tasks demonstrate that RingMo-Agent not only proves effective in both visual understanding and sophisticated analytical tasks, but also exhibits strong generalizability across different platforms and sensing modalities.

Keywords Foundation model · Multi-modal · Multi-platform · Remote sensing multi-modal large language model · Instruction tuning

1 Introduction

With the remarkable advancements of large language models (LLMs) in semantic understanding and reasoning, vision-language models designed for open-world environments have experienced rapid development. Leveraging the capabilities of advanced LLMs such as DeepSeek [1–4], GPT [5, 6], and Llama [7–9], these models are now capable of perceiving complex semantics, conducting multi-turn interactions, and performing context-aware task planning. This evolution marks a significant shift from static perception toward dynamic understanding and autonomous decision-making, thereby enabling broader applications in intelligent interaction and high-level scene reasoning.

Driven by recent advances in these research for natural scenes, the paradigm of vision-language models has gradually extended to RS [10], introducing capabilities such as instruction tuning and joint vision-language modeling into RS image analysis. These efforts have led to the emergence of models that demonstrate promising performance

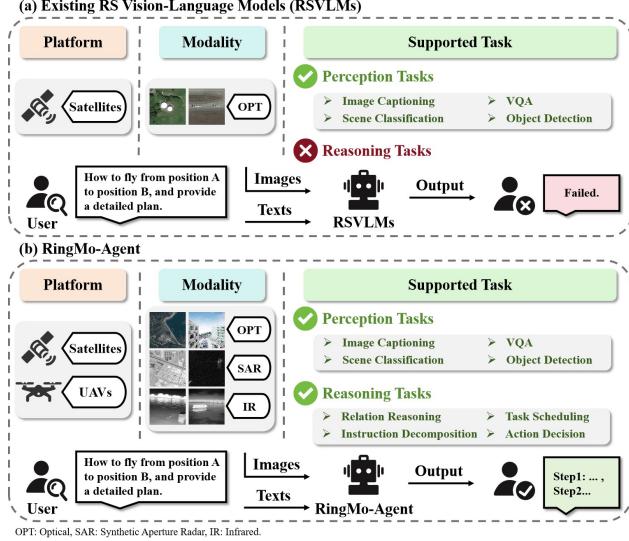


Figure 1: Comparison of RingMo-Agent and most existing RS vision-language models in terms of data platforms, modalities, and supported tasks.

in fundamental RS perception tasks such as object recognition and scene classification [11–21]. However, current vision-language models for RS still suffer from significant limitations:

- **Limitations of single-source modeling in diverse RS data.** Due to variations in data acquisition platforms (e.g., satellites and UAVs) and imaging modalities (e.g., optical, SAR, and infrared), RS data exhibit significant differences in physical characteristics such as spatial resolution, viewing geometry, and spectral response. These heterogeneous properties fundamentally affect the way information is represented in the imagery. However, most existing vision-language models for RS are trained on data from a single modality or platform, limiting their ability to handle the complexity of multi-source data in real-world scenarios. Consequently, these approaches often suffer from poor generalization and limited compatibility in cross-platform and cross-modal applications.
- **Limitations of existing reasoning task paradigms.** Existing RS vision-language research primarily focuses on specific visual perception tasks. However, these tasks remain confined to foundational perception and recognition, with models lacking training in advanced cognitive abilities such as relation reasoning among scene entities and action prediction. Therefore, these models are restricted to narrow applications and struggle to replicate the superior intelligence demonstrated by agent systems in natural scenes when faced with high-level tasks requiring decision-making, as illustrated in Figure 1. This limitation fundamentally stems from the lack of large-scale RS image-text datasets oriented toward complex semantic interactions and reasoning, which restricts the models' ability to generalize multi-modal knowledge.

To address the aforementioned limitations, we propose RingMo-Agent, a model designed to handle multi-modal and multi-platform data, capable of performing perception and reasoning tasks based on user instructions. **First**, to unleash the potential of vision-language models in intelligent agent applications, we construct a vision-language dataset named RS-VL3M, comprising 3 million image-text pairs across three modalities (optical, SAR, and infrared), two platforms (satellites and UAVs), and eight tasks, as shown in Figure 2. **Second**, we employ a high-dimensional hidden state decoding mechanism based on the special token, which models dynamic trajectory information by focusing on the final-layer hidden representations, enabling support for RS long-horizon reasoning tasks. **Third**, a modality-aware visual encoder with separated embeddings is incorporated to mitigate distribution shifts across different sensing platforms and modalities, supporting robust feature extraction and alignment. **Finally**, unlike previous methods that mainly focus on optical imagery and fundamental perception tasks, RingMo-Agent unifies both perception and reasoning across heterogeneous modalities and platforms, as summarized in Table 1. We further conduct comprehensive comparisons between our proposed model and existing advanced methods on public and self-constructed multi-source RS datasets.

The main contributions can be summarized as follows:

1. We propose RingMo-Agent, a model that supports three sensing modalities and eight task types across two platforms, enabling a unified framework spanning from basic visual perception to advanced reasoning.

Table 1: Comparison of model architectures and capabilities, including LLMs, visual encoders, supported modalities, and task coverage. SAR: Synthetic Aperture Radar, IR: Infrared, IC: Image Captioning, CL: Classification, RR: Relationship Reasoning, OD: Object Detection, ID: Instruction Decomposition, AD: Action Decision, TS: Task Scheduling.

Model	LLM	Visual Encoder	Size	Modalities			Perception Tasks				Reasoning Tasks			
				Optical	SAR	IR	IC	VQA	CI	OD	RR	ID	AD	TS
EarthGPT [11]	LLaMA-2	DINOv2 ViT-L/14, CLIP ConvNeXt-L	-	✓	✓	✓	✓	✓	✓	✓				
Popeye [12]	LLaMA-2 (7B)	DINOv2 ViT-L/14, CLIP ViT-L/14	-	✓	✓		✓	✓						
RS-CapRet [13]	LLaMA-2 (7B)	CLIP ViT-L/14	224	✓			✓							
LHRS-Bot [14]	LLaMA-2 (7B)	CLIP ViT-L/14	224	✓			✓	✓	✓	✓	✓			
SkyEyeGPT [15]	LLaMA-2-Chat (7B)	EVA ViT-G	448	✓			✓	✓	✓		✓			
SkySenseGPT [16]	Vicuna-v1.5	CLIP ViT-L/14	504	✓			✓	✓	✓	✓		✓		
GeoChat [17]	Vicuna-v1.5 (7B)	CLIP ViT-L/14	504	✓			✓	✓	✓	✓	✓			
H ² RSVLM [18]	Vicuna-v1.5 (7B)	CLIP ViT-L/14	336	✓			✓	✓	✓	✓	✓			
RingMoGPT [19]	Vicuna-v1.1 (13B)	EVA ViT-G	448	✓			✓	✓	✓	✓	✓			
RSGPT [20]	Vicuna (7B/13B)	EVA ViT-G	224	✓			✓	✓	✓	✓	✓			
RS-LLaVA [21]	Vicuna-v1.5 (7B/13B)	CLIP ViT-L/14	336	✓			✓	✓						
RingMo-Agent (ours)	DeepSeekMoE (3B)	SigLIP-SO400M-384	Any	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

2. We construct RS-VL3M, the first large-scale RS vision-language dataset integrating multi-platform and multi-modal data, with over 3 million image-text pairs. It also includes a high-quality shared multi-modal subset specifically designed for multi-turn dialogue evaluation.
3. RingMo-Agent employs separate embedding layers to mitigate modality-specific distribution shifts and enables joint processing of heterogeneous inputs. It further incorporates a high-dimensional hidden state decoder guided by long-horizon task-specific tokens to model dynamic trajectories through critical hidden representations.

Under the validation, RingMo-Agent shows strong performance across eight RS tasks, outperforming both expert models and generalist models, while exhibiting robust zero-shot generalization.

2 Related works

2.1 Vision-Language Research in Natural Scenes

In recent years, vision-language models have made significant advancements in the field of artificial intelligence, becoming an essential bridge between visual understanding and language understanding.

Early studies primarily focused on perception-level tasks such as image-text matching, image captioning, and closed-form question answering, emphasizing cross-modal alignment and representation learning between vision and language. For instance, BLIP-2 [22] introduces three distinct loss constraint strategies between visual and textual content to promote cross-modal alignment, achieving advanced performance on tasks such as VQA and image-text retrieval. InstructBLIP [23] explores a unified instruction-tuning paradigm, where instruction-aware visual feature extraction is employed to accomplish fundamental visual perception tasks. Subsequently, some studies began investigating whether LLMs can perform effectively on pixel-level tasks such as localization, detection, and segmentation. MiniGPTv2 [24], Kosmos-2 [25], and Shikra [26] have additionally explored the use of natural language to represent spatial coordinates. LISA [27] introduces an embedding-as-mask paradigm for reasoning-based segmentation, where the segmentation mask is decoded by predicting the hidden representation of a special token.

With the advancement of model capabilities, research exemplified by GPT-4 [28] has gradually expanded to encompass open-ended question answering, visual commonsense reasoning, and multi-turn dialogue tasks, incorporating contextual modeling and knowledge reasoning. Models such as MiniGPT-4 [29], mPLUG-Owl [30], Qwen-VL [31], DeepSeek-VL2 [32] tend to leverage extensive instruction tuning to enhance deep image understanding, reasoning, and generation capabilities, thereby achieving better alignment with user intent. Recently, emerging vision-language research [33–35] has increasingly centered around large language models, integrating environmental interaction to enable general-purpose agents capable of complex perception and reasoning.

Current models are evolving from perception-driven systems to generalist agents with reasoning capabilities, but remain largely focused on natural scenes and struggle with the complexity of RS imagery.

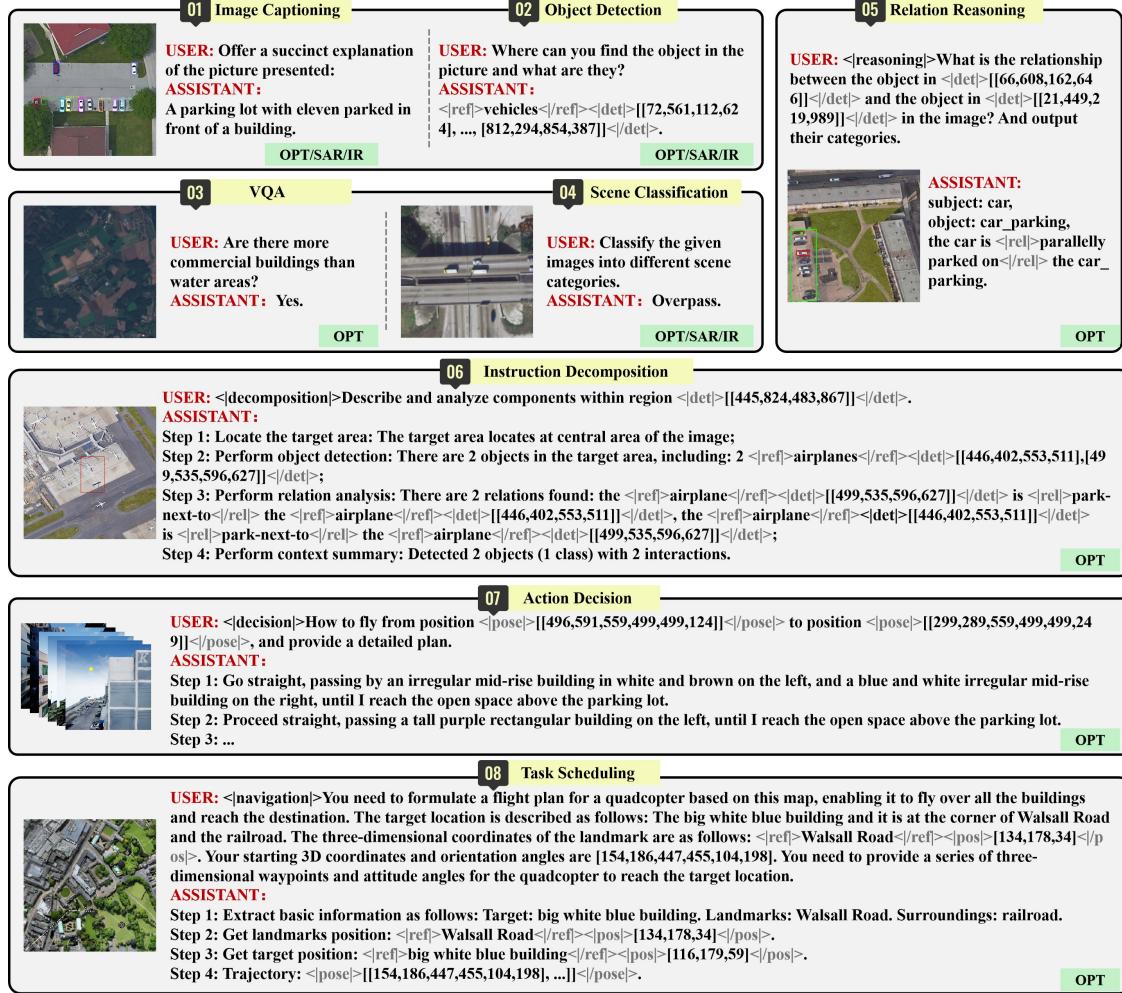


Figure 2: RingMo-Agent supports perception tasks such as image captioning, object detection, VQA, and scene classification, and reasoning tasks including relation reasoning, instruction decomposition, action decision, and task scheduling. Some tasks are enabled by optical (OPT), SAR, and infrared (IR) modalities.

2.2 Vision-Language Research in Remote Sensing

Due to the heterogeneity of RS data and the task-specific requirements, applying LLMs to RS remains largely exploratory. Recent methods adapt existing network architectures and leverage pre-trained weights through fine-tuning.

With works such as RSGPT [20] and RSVG [36] demonstrating that vision-language models can generalize effectively on RS images and achieve superior performance, research has increasingly focused on the design and training of these models. RemoteCLIP [37] establishes a unified RS vision-language model and validates its effectiveness across downstream tasks such as image classification, image-text retrieval, and object counting. Models such as GeoChat [17], RingMoGPT [19], and SkyEyeGPT [15] further extend the task scope to include multi-turn conversation, change detection, among others.

However, existing studies mainly focus on perception tasks using optical imagery, covering only a limited portion of RS data. Recently, EarthGPT [11] incorporates SAR and infrared data to enhance the multi-modal detection capabilities of vision-language models. SkySenseGPT [16] explored the model’s ability to reason about relationships between objects in images. AeroVerse [38] attempts to integrate UAV-based intelligent agent tasks into the vision-language model to explore capabilities in navigation exploration.

Despite recent progress, current RS research remains largely confined to basic perception representation, lacking autonomous decision-making capabilities. Moreover, they predominantly focus on single-platform and single-modal data, which limits their applicability in real-world RS scenarios.

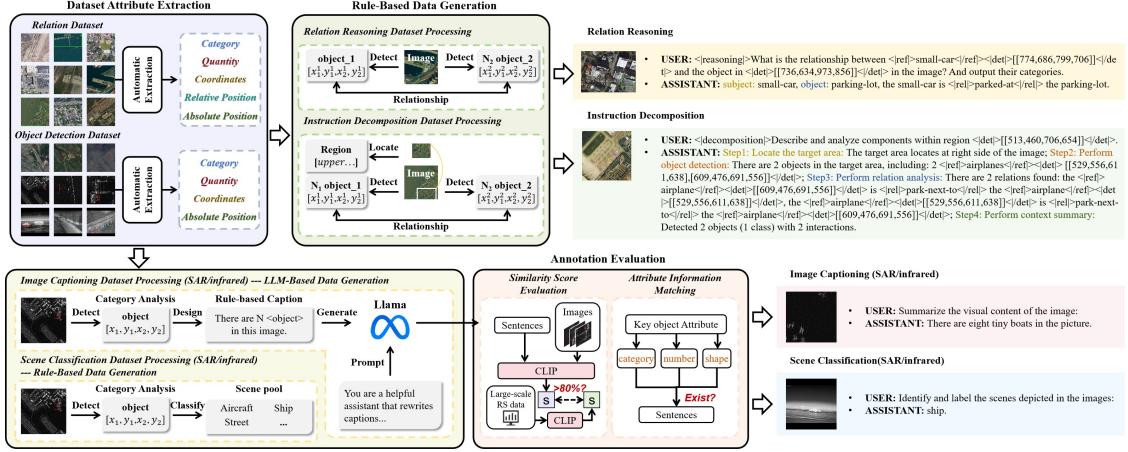


Figure 3: A dataset construction workflow that extracts relevant attributes from existing relation and detection datasets for instruction decomposition, relation reasoning, classification, and captioning tasks.

2.3 Tasks and Datasets of Remote Sensing Vision-Language Research

To support the training of RS intelligent systems, numerous vision-language datasets have been developed, most of which are extended from single-modality datasets originally designed for tasks such as object detection, semantic segmentation, image classification, and change detection, and have gradually evolved into large-scale and multi-modal benchmarks.

Early datasets were mostly based on manually annotated image descriptions, with limited scale semantic levels, mainly supporting basic tasks such as VQA and classification, as exemplified by RSVQA [39] and AID [40]. With the expansion of tasks, dataset types and scales have gradually grown. For instance, RemoteCLIP [37] constructs a large-scale caption dataset by applying rule-based methods to RS datasets designed for detection, segmentation, and retrieval, aligning visual and linguistic data. RSVG [36], built upon the optical object detection dataset DIOR [41], conducts attribute analysis on image objects and generates a visual grounding dataset through rule-based processing. RSGPT [20] extends the optical object detection dataset DOTA [42], leveraging expert annotations from RS specialists to develop a comprehensive image captioning dataset.

Recently, several studies have aimed to build large-scale, high-quality RS image-text datasets encompassing diverse task types. EarthGPT [11] collects a diverse range of RS data covering five task types and three image modalities, designing different instructions to construct a pretraining dataset. Additionally, models such as GeoChat [17], RingMoGPT [19], and SkySenseGPT [16] leverage constrained LLMs to generate responses for augmented tasks, serving as data for training.

Despite these advancements, there remains a lack of datasets capable of supporting diverse task types, multiple modalities, and various platforms, which is essential for advancing toward generalist intelligent agents.

3 Dataset Construction

3.1 Overall Data Analysis

We construct a large-scale, high-quality RS image-text dataset named RS-VL3M, encompassing images captured by various sensors mounted on different imaging platforms. The dataset features images with diverse resolutions, viewing angles, and imaging mechanisms, totaling over 3 million image-text pairs. It supports two major categories of tasks: reasoning-oriented tasks (task scheduling, action decision, instruction decomposition, and relation reasoning), and perception-oriented tasks (image captioning, object detection, image classification, and VQA), as detailed in Table 2. Furthermore, to enhance the multi-turn dialogue capability of the model under consistent data conditions, we establish a multi-modal dialogue subset based on a shared image pool. This subset enables interactive classification, detection, and captioning tasks on infrared and SAR modalities simultaneously for the same images.

Table 2: The overview of the dataset RS-VL3M, including task types, dataset size, modalities, and image size. <x, x> indicates variable image size, representing the range from minimum to maximum.

Task	Dataset	Dataset Size	Modality	Image Size
Task Scheduling	CityNav [43]	32,637	optical	<617, 4001>
Action Decision	SkyAgent-Plan3k [38]	3,000	optical	512×512
Instruction Decomposition	ReCon1M-DEC (ours)	27,819	optical	<135, 1000>
Relation Reasoning	FIT-RS [16] ReCon1M-REL (ours)	97,843 125,000	optical optical	512×512 <135, 1000>
Object Detection	DIOR [41]	23,463	optical	800×800
	DOTA [42]	108,047	optical	800×800
	NWPU VHR-10 [44]	800	optical	<500, 1100>
	SARDet-100k [45]	116,598	SAR	<190, 1000>
Image Captioning	IR-DET (ours)	56,353	infrared	<92, 6912>
	UCM-Captions [46]	10,500	optical	256×256
	Sydney-Captions [46]	3,065	optical	500×500
	RSICD [47]	54,605	optical	224×224
	NWPU-Captions [48]	157,500	optical	256×256
	SAR-CAP (ours)	582,990	SAR	<190, 1000>
Scene Classification	IR-CAP (ours)	281,765	infrared	<92, 6912>
	AID [40]	10,000	optical	600×600
	NWPU-RESISC45 [49]	31,500	optical	256×256
	WHU-RS19 [50]	1,005	optical	600×600
	UCMerced-LandUse [51]	2,100	optical	256×256
	SAR-CLA (ours)	116,598	SAR	<190, 1000>
VQA	IR-CLA (ours)	56,353	infrared	<92, 6912>
	RSVQA-HR [39]	1,066,316	optical	512×512
	RSVQA-LR [39]	77,232	optical	256×256

3.2 Dataset Construction for Tasks

In addition to utilizing publicly available datasets directly, we also constructed datasets for complex tasks by processing open-source multi-modal data. The overall workflow is illustrated in Figure 3. We extracted object-level attribute information from optical, SAR, and infrared images, and formatted the data according to the design requirements of each task. For data usage, to ensure consistent spatial representation, all coordinates are normalized to the range [0, 999]. Users are required to specify the current modality using the token [label], where the possible values are opt, sar, and ir. Details of the data for each task category are provided below.

Task Scheduling. This task focuses on trajectory planning for autonomous aerial navigation in complex urban environments based on RS imagery. Given instructions that describe the spatial context surrounding the target location, the model must identify the 3D position of the target based on references to nearby buildings or landmarks, and, given the initial pose of the agent, produce a sequence of 3D waypoints along with corresponding orientation angles. Our task scheduling dataset is derived from CityNav [43] and structured as follows:

- Prompt: <|navigation|> You need to formulate a flight plan for a quadcopter based on this map, enabling it to fly over all the buildings and reach the destination. The target location is described as follows: <description>. The 3D coordinates of the landmark are as follows: < |ref| >< landmark >< |ref| >< |pos| > [x₁, y₁, z₁] < |pos| >. Your starting 3D coordinates and orientation angles are < |pose| > [[x₀, y₀, z₀, φ₀, θ₀, ψ₀]] < |pose| >. You need to provide a series of 3D waypoints and attitude angles for the quadcopter to reach the target location.
- Response: Step 1: Extract basic information as follows: Target: < target >. Landmarks: < landmark >. Surroundings: < surrounding >. Step 2: Get landmarks position: < |ref| >< landmark >< |ref| >< |pos| > [x₁, y₁, z₁] < |pos| >. Step 3: Get target position: < |ref| >< target >< |ref| >< |pos| > [x₂, y₂, z₂] < |pos| >. Step 4: Trajectory: < |pose| > [[x₀, y₀, z₀, φ₀, θ₀, ψ₀], ..., [x_n, y_n, z_n, φ_n, θ_n, ψ_n]] < |pose| >.

A set of special tokens is embedded in the prompt: < |navigation| > defines the task type. < |ref| > and < |ref| > denote the name of a landmark, while < |pos| > and < |pos| > indicate its corresponding coordinates. The trajectory is structured using < |pose| > and < |pose| > tags, where each pose entry consists of six elements: spatial position (x, y, z) and orientation angles (ϕ, θ, ψ), representing roll, pitch, and yaw respectively. Besides, placeholders such as < landmark >, < surrounding >, and < target > are used to mark category information (e.g., “Wellington Road”), while < description > is a placeholder sentence used to describe the spatial surroundings of the target (e.g., “The row of grayish brown houses on Leslie Road to the left of the gray house at the intersection with Wellington Road”).

Action Decision. This task focuses on fine-grained action decision planning for aerial agents navigating urban environments, bridging visual-language reasoning with control execution. Given the initial and target 3D poses, along with urban map imagery, the model is required to generate a detailed sequence of instructions. Each instruction corresponds to an interpretable motion step (e.g., “go straight,” “slightly turn right”) and is grounded in observable visual cues such as nearby buildings, colors, shapes, or spatial layouts. Our action decision dataset uses data from SkyAgent-Plan3k [38], as exemplified below:

- Prompt: <|decision|>How to fly from position<|pose|>[[$x_0, y_0, z_0, \phi_0, \theta_0, \psi_0$]]</|pose|> to position<|pose|>[[$x_n, y_n, z_n, \phi_n, \theta_n, \psi_n$]]</|pose|>, and provide a detailed plan.
- Response: Step1: <description>.
Step2: <description>.
...
Stepn: <description>.

We introduce a special token <|decision|> to specify task semantics. The usage of <|pose|> and </|pose|> is consistent with the task scheduling dataset. The starting point appears in the first image, and the destination in the last. The output format requires each scene transition to be annotated with a <description> tag, with each step written from the agent perspective in natural language to support human understanding and decision-making.

Instruction Decomposition. This task aims to improve the capability of the model to execute complex, high-level RS commands by breaking them down into a sequence of fine-grained, interpretable sub-tasks. In our setting, the model receives an image alongside a natural language command that may reference specific regions or semantic goals. The model is expected to decompose this input into a structured multi-step reasoning process, including region localization, object detection, relation analysis, and contextual summarization. Our instruction decomposition dataset builds upon the RS relation comprehension dataset ReCon1M [52].

- Prompt: <|decomposition|>Analyze the region <|det|>[[x_1, y_1, x_2, y_2]]</|det|> of the image.
- Response: Step1: Locate the target area: The target area locates at the <direction> of the image.
Step2: Perform object detection: There are N entities in the target area, including: $N_1 <|\text{ref}|><|\text{object}|>$ </|ref|><|det|>[[$x_1^1, y_1^1, x_2^1, y_2^1$]]</|det|>,..., $N_n <|\text{ref}|><|\text{object}|>$ </|ref|><|det|>[[$x_1^n, y_1^n, x_2^n, y_2^n$]]</|det|>.
Step3: Perform relation analysis: There are M relations found: <|ref|><|object|></|ref|><|det|>[[$x_1^1, y_1^1, x_2^1, y_2^1$]]</|det|> is <|rel|><|relation|></|rel|> the <|ref|><|object|></|ref|><|det|>[[$x_1^m, y_1^m, x_2^m, y_2^m$]]</|det|>;
Step4: Perform context summary: N object types with M interactions.

As shown above, <direction> has 9 directions, upper, lower, left, right, upper left, upper right, lower left, lower right, and center. N is the total number of detected objects, with N_1 to N_n representing category-wise counts. <|ref|><|object|></|ref|><|det|>[[$x_1^1, y_1^1, x_2^1, y_2^1$]]</|det|> denotes the specific class of objects with spatial location information, where $[x_1^1, y_1^1, x_2^1, y_2^1]$ denote the top-left and bottom-right corners, respectively. <|rel|><|relation|></|rel|> denotes the relationship between the two objects.

Relation Reasoning. This task requires the model to infer the categories of the subject and object and, and to describe the relationship between them, thereby enabling the model to recognize objects in the image and reason about their interactions. Our dataset is derived from the RS instruction-tuning dataset FIT-RS [16] and the RS relation comprehension dataset ReCon1M [52].

- Prompt: <|reasoning|>What is the relationship between <|ref|><|object₁|></|ref|><|det|>[[$x_1^1, y_1^1, x_2^1, y_2^1$]]</|det|> and the object in <|det|>[[$x_1^2, y_1^2, x_2^2, y_2^2$]]</|det|> in the image? And output their categories.
- Response: subject: <|object₁|>, object: <|object₂|>, the <|object₁|> is <|rel|><|relation|></|rel|> the <|object₂|>.

Consistent with other tasks, we define a set of special tokens to explicitly mark different semantic components in both the prompt and response.

Object Detection. The optical images are from DIOR [41], DOTA [42], and NWPU VHR-10 [44], SAR images are from SARDet-100k [45], and infrared images are from HIT-UAV [53], Sea shipping [54], Infrared-security [55], Aerial-mancer [56], Double-light-vehicle [57], Oceanic ship [58].

- Prompt: Detect all objects shown in the remote sensing image and describe using HBBs.
- Response: There is/are $M < |\text{ref}| > < \text{object} > < |\text{ref}| > < |\text{det}| > [[x_1^1, y_1^1, x_2^1, y_2^1], \dots, [x_1^m, y_1^m, x_2^m, y_2^m]] < |\text{det}| >$ in the image.

A placeholder $< \text{object} >$ is used to represent category names (e.g., “ship”). The use of coordinates follows the same approach as described previously.

Image Captioning. We construct a multi-modal image captioning benchmark using SAR-CAP, IR-CAP, and various optical captioning datasets. The SAR and infrared data are sourced from the same datasets as the object detection task, while optical data is sourced from public datasets [46–48].

Optical captions are often richer due to easier manual annotation, typically describing simple scenes with buildings and vegetation. In contrast, SAR and infrared data demand expert analysis. To address this issue, as illustrated in Figure 3, we first generate rule-based captions for objects in the images using standard templates. These captions are then fed as text inputs to Llama3.1 [9], accompanied by carefully designed prompts that constrain the model to enrich the language while preserving three key attributes: the number, shape, and category of objects. The model produces five distinct textual descriptions for each image. We perform automated validation on the generated texts, which consists of similarity score evaluation and attribute information matching. The similarity scores between the generated texts and RS images are computed using RemoteCLIP [37] and compared against benchmark scores from open-source image captioning datasets. Only texts with similarity scores exceeding 80% of the benchmark are retained. For attribute matching, we verify the number, shape, and category of objects mentioned in the texts, allowing for synonyms, and retain only those texts that meet all these criteria. This approach effectively ensures the accuracy and consistency of the generated descriptions.

- Prompt: Please provide a short depiction of the picture:
- Response: There are 2 aircrafts in the image, which are small in size.

Image Classification. We construct SAR and infrared classification datasets using the same sources as before. For SAR, we adopt the original category definitions, while for infrared data, inconsistent annotations across datasets are addressed by grouping labels into two coarse categories: street scenes and ships. Optical data is sourced from publicly available datasets.

- Prompt: Please output the scene corresponding to the image:
- Response: aircraft.

We construct a shared image subset for infrared and SAR data across detection, captioning, and classification tasks, where each image is annotated to support all three tasks. This design enables multi-turn dialogue evaluation under consistent visual input conditions.

VQA. We use public datasets [39] as our training and evaluation benchmarks. These datasets are widely adopted as standard benchmarks for evaluating vision-language models.

- Prompt: Is a small road present? The answer to this question is
- Response: yes.

4 Method

4.1 Architecture

The proposed RingMo-Agent comprises four main components: a visual encoder for extracting visual features, a LLM for processing language information, an MLP projector serving as an intermediate adapter that projects visual features into the semantic space of the language model, and a trajectory decoder designed for spatiotemporal modeling, as illustrated in Figure 4. Our approach employs a two-stage training strategy that unifies multi-source data such as optical, infrared, and SAR imagery, while also extending conventional perception tasks to more challenging reasoning tasks, such as complex urban scene understanding from UAV perspectives.

Frozen Visual Encoder. The visual encoder is based on the SigLIP [59] method, dynamically rescaling input images to multiples of 384 while accommodating RS images with varying aspect ratios. In particular, given an input image of size $(H \times W \times C) \in \mathbb{R}$, the image is rescaled to dimensions $m \times 384, n \times 384, c$, where $m, n \in \mathbb{N}, 1 \leq m, n, mn \leq 9$. The values of m and n are determined by selecting the smallest multiples of 384 that are greater than or equal to

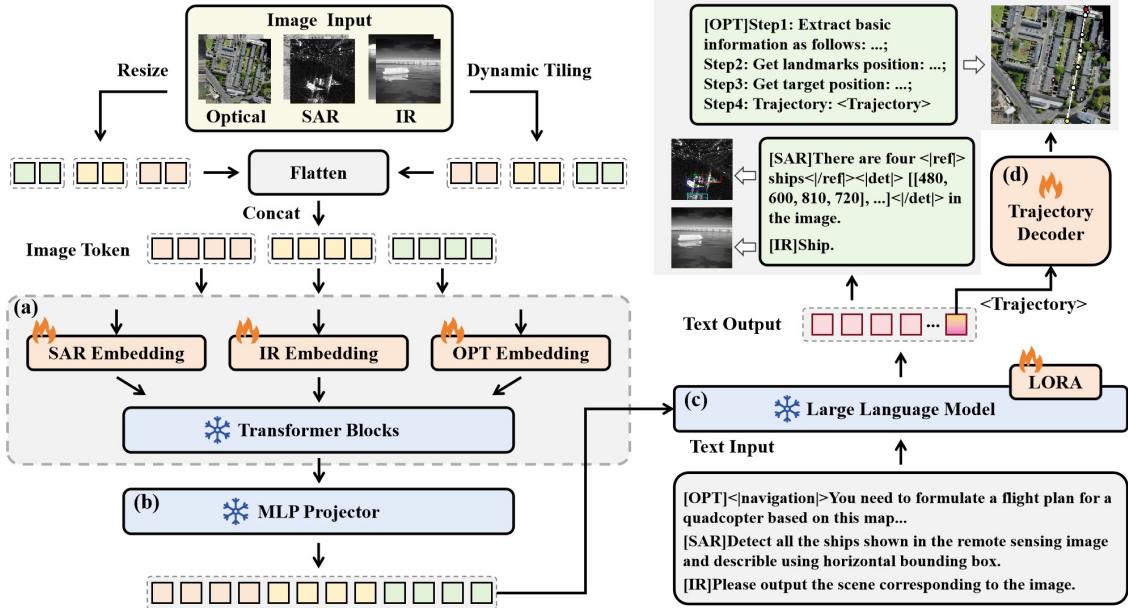


Figure 4: The overall architecture of RingMo-Agent, which supports image inputs from multiple platforms and modalities, enabling both perception and reasoning capabilities. The image features are fed into (a) a frozen visual encoder. After passing through (b) a frozen MLP and (c) a LLM fine-tuned via LoRA, the model directly generates textual content, with (d) additional decoding applied for tasks involving trajectory outputs.

the original image height H and width W , respectively. Following this resizing step, the image is partitioned into independent patches of size 384×384 . Additionally, a global thumbnail is generated by directly resizing the image to 384×384 to provide a coarse global representation.

Most existing research primarily focus on single optical modality learning. However, when handling multi-modal data, distribution differences across modalities hinder a unified encoder from extracting discriminative features. To mitigate this, we introduce separate embedding layers for isolated feature extraction of infrared, SAR, and optical modalities, reducing cross-modal interference. To formalize the modality-specific visual embedding process, let $x_{\text{opt}}, x_{\text{ir}}, x_{\text{sar}} \in \mathbb{R}^{H \times W \times C}$ represent the input images from the optical, infrared, and SAR modalities, respectively. The visual embeddings for each modality are obtained as:

$$z_{\text{opt}} = E_{\text{opt}}(x_{\text{opt}}), \quad z_{\text{ir}} = E_{\text{ir}}(x_{\text{ir}}), \quad z_{\text{sar}} = E_{\text{sar}}(x_{\text{sar}}) \quad (1)$$

where E_{opt} , E_{ir} , and E_{sar} are the embedding layers for optical, infrared, and SAR data, respectively. These embeddings are then forwarded to the subsequent image encoder for further processing. The frozen visual encoder, denoted as F_V , receives an input image z and outputs the corresponding visual features:

$$V = F_V(z), \quad z = \{z_{\text{opt}}, z_{\text{ir}}, z_{\text{sar}}\} \quad (2)$$

where V represents the extracted visual features.

Frozen MLP Projector. The adapter follows a MLP architecture composed of linear layers. Following the setup of DeepSeek-VL2 [32], the mapping performed by the adapter can be represented by the following computation:

$$Q = F_A(V) \quad (3)$$

Where, F_A denotes the adapter composed of MLP layers, and Q represents the mapped features. In F_A , a 2×2 shuffling operation is first applied to transform each tile's visual tokens from a 27×27 layout to 14×14 . Subsequently, three types of special tokens are inserted into the visual features to serve as positional indicators: one is appended to the end of each row in the global thumbnail tile, another to the end of the last column in the local tiles, and the third is placed between the global and local tiles. Then, as illustrated in Figure 4, the processed embeddings are fed into the LLM as input.

LoRA-Fine-Tuned Large Language Model. The LLM is based on the DeepSeekMoE architecture [2, 3], which introduces multi-head latent attention by compressing the Key-Value cache into a latent representation, thereby reducing training costs. DeepSeek-VL2 [32] provides models with 1.0B, 2.8B, and 4.5B activated parameters, among which we adopt the 1.0B version for implementation. We adopt the LoRA approach [60], which inserts a small number of

Algorithm 1 Trajectory Decoding from Predicted Tokens

Input: Embedding vector of the `< trajectory >` token h_{tra} , maximum termination step T , termination threshold p , finished flag P_{flag} .

Output: Updated trajectory S .

```

1:  $h_0 \leftarrow \text{LatentProjection}(h_{tra})$ 
2: while  $t \leq T \&& P_{flag} \text{ do:}$ 
3:    $f_t \leftarrow \text{StateProjection}(h_{t-1})$ 
4:    $h_t \leftarrow \text{GRUCell}(f_t, h_{t-1})$ 
5:    $S_t \leftarrow \text{OutputProjection}(h_t)$ 
6:    $S_t \leftarrow \text{Sigmoid}(S_t)$ 
7:    $S \leftarrow \text{Update}(S_t)$ 
8:    $P_{flag} \leftarrow \text{TerminationCondition}(S_t, p)$ 
9:    $t \leftarrow t + 1$ 
10: end while
```

trainable low-rank matrices into the LLM. Fine-tuning is then performed only on these additional parameters. The process can be formulated as follows:

$$\hat{y} = F_L(Q, x_{txt}) \quad (4)$$

Where, F_L denotes the LLM fine-tuned with LoRA, x_{txt} represents the textual input, and \hat{y} represents the output results.

Trainable Trajectory Decoder. Conventional models generate text or 2D coordinates via sequential token decoding, limiting spatial motion modeling. Inspired by LISA [27], we propose a high-dimensional hidden state decoding mechanism using task-specific tokens to explicitly capture dynamic trajectories from the final transformer layer, improving temporal action continuity. The detailed procedure of the decoding mechanism is described in Algorithm 1. The special token `< trajectory >` is predicted only in the task scheduling scenario, where coordinate outputs are required, enabling end-to-end training within a unified framework.

Specifically, the multi-modal input and corresponding ground truth are uniformly denoted as $x = \{x_{opt}, x_{ir}, x_{sar}\}$ and y , respectively, while the model generates the prediction \hat{y} . Let h_{tra} denote the embedding vector of the `< trajectory >` token in the last layer. We define the internal computations of the trajectory decoder as follows.

$$h_0 = \phi_{latent}(h_{tra}) \quad (5)$$

where ϕ_{latent} is a linear projection. The latent vector is projected into the hidden state space to obtain the initial hidden state h_0 , encoding the contextual features of the current environment. At each time step $t = 1, \dots, T$, the decoder updates the hidden state and predicts the next trajectory state as:

$$f_t = \phi_{state}(h_{t-1}) \quad (6)$$

where ϕ_{state} is a linear projection of the previous state. Then, we use a GRU network to update the current state:

$$h_t = \text{GRUCell}(f_t, h_{t-1}) \quad (7)$$

Where h_t represents the updated hidden state. Then, the next trajectory state is predicted from the hidden state via an output projection layer:

$$S_t = \sigma(\phi_{output}(h_t)) \quad (8)$$

where ϕ_{output} is a linear projection outputting the next trajectory point. The predicted trajectory state at each step is appended to the corresponding trajectory sequence S . We set the termination distance threshold to 1e-3, and define the maximum number of steps based on the longest trajectory in the scenario. The loop terminates when either condition is met.

The constraint loss consists of a cross-entropy loss L_{txt} for text prediction and a mean squared error loss L_{mse} for trajectory regression. Each loss is weighted by its respective coefficient λ_{txt} and λ_{mse} , and the final loss is computed as their weighted sum. The loss can be expressed as follows:

$$L = \lambda_{txt} L_{txt} + \lambda_{mse} L_{mse} \quad (9)$$

4.2 Training Methodology

RingMo-Agent employs a two-stage training approach, consisting of the vision-language generation stage and the instruction-tuning stage.

In the vision-language generation stage, the primary objective is to adapt the model from natural scene understanding to the RS image domain. The original pretrained weights struggle to handle complex RS scenes, often misidentifying multiple distinct objects as a single entity, making it insufficient for fine-grained recognition tasks. To address this issue, we utilize paired RS image-text data for generative training. In this process, we adopt image-text paired datasets to enable optimization via the generative loss of LLMs. The data is sourced from publicly available classification, captioning, and detection datasets [41, 42, 46, 47, 49–51, 61–65]. To enhance linguistic diversity while preserving semantic accuracy, we use GPT-4 [28] to expand the original annotations, applying constraints such as requiring the generated sentence to contain the original ground truth. Additionally, we employ rule-based extraction methods to verify whether the generated descriptions accurately reflect the original labels. In the instruction-tuning stage, the model is optimized to generate outputs that align with the given instructions, ensuring accurate task execution.

5 Experiments

This section details the training procedure and evaluates the instruction-tuned model on both perception tasks (detection, VQA, classification, captioning) and reasoning tasks (relation reasoning, task scheduling, instruction decomposition, action decision), using both qualitative and quantitative analyses.

In the reported results, FT denotes the fine-tuned results, while ZS refers to the zero-shot results. **Bold** indicates the best performance, and underline denotes the second-best.

5.1 Experimental Settings

During vision-language generation stage, LoRA [60] is applied to fine-tune the LLM. The LoRA configuration is defined with a rank of 64 and a scaling factor of 16, which balances parameter efficiency and representational capacity. We train our model on 8 NVIDIA A100 GPUs (80 GB) using the AdamW optimizer. The learning rate follows a linear warmup followed by a cosine decay schedule, with an initial learning rate of 1e-4, a warmup learning rate of 1e-6, and a minimum learning rate of 1e-5. We apply a weight decay of 0.05 for regularization. We train the model on over 500,000 samples for 10 epochs, with the image size fixed at 384.

During instruction-tuning stage, RingMo-Agent adopts a linear warmup followed by a cosine decay learning rate schedule. The training utilizes the AdamW optimizer, with an initial learning rate of 1e-6, a warmup starting from 1e-8, and a minimum learning rate decaying to 0. To further enhance generalization and mitigate overfitting, a weight decay of 0.05 is applied throughout the optimization process. In this stage, we impose no restriction on image resolution and retain the original dimensions of each sample. The model is also trained for 10 epochs. The combined training, validation, and test sets contain over 3 million samples, collectively referred to as RS-VL3M. For the evaluation datasets used to report accuracy, we conduct separate fine-tuning in addition to the zero-shot setting.

5.2 Task Scheduling

Dataset. We use the CityNav dataset [43], which comprises 34 diverse urban scene environments. The dataset is partitioned into a training set, a seen validation set, an unseen validation set, and an unseen test set. The dataset encompasses a wide range of target object categories, such as buildings, vehicles, ground, and parking lots.

Metrics. We evaluate on all three test splits and compare its performance against existing specialist models. Following CityNav [43], four metrics are used to assess the accuracy of the generated trajectories: Navigation Error (NE), Success Rate (SR), Oracle Success Rate (OSR), and Success weighted by Path Length (SPL).

Table 3: Evaluation results on the CityNav dataset.

Model	Validation Seen				Validation Unseen				Test Unseen			
	NE ↓	SR ↑	OSR ↑	SPL ↑	NE ↓	SR ↑	OSR ↑	SPL ↑	NE ↓	SR ↑	OSR ↑	SPL ↑
Specialist models												
Seq2Seq (FT) [66]	257.1	1.81	7.89	1.58	317.4	0.79	8.82	0.61	245.3	1.50	8.34	1.30
CMA (FT) [67]	240.8	0.95	9.42	0.92	268.8	0.65	7.86	0.63	252.6	0.82	9.70	0.79
MGP (FT) [43]	59.70	8.69	35.51	8.28	75.10	5.84	22.19	5.56	93.80	6.38	26.04	6.08
RSVLMs												
RingMo-Agent (FT)	<u>132.0</u>	<u>5.89</u>	<u>21.39</u>	<u>5.24</u>	<u>156.0</u>	<u>4.91</u>	<u>17.22</u>	<u>4.33</u>	<u>149.6</u>	<u>4.74</u>	<u>18.94</u>	<u>4.17</u>

Results. As shown in Table 3, RingMo-Agent outperforms the fine-tuned Seq2Seq [66] and CMA [67], and approaches the performance of MGP [43], which incorporates map encoding and multi-view features. Unlike MGP, we achieve

competitive results without relying on external map priors, with only a 0.93% SR gap on the seen validation set. Notably, current general-purpose vision-language models (VLMs) and remote sensing vision-language models (RSVLMs) still struggle to complete this complex spatial navigation task.

5.3 Action Decision

Dataset. We evaluate this task on the SkyAgent-Plan3k [38] dataset, which encompasses four distinct urban scene types: Shanghai, Shenzhen, Campus, and Residence.

Metrics. To assess the quality of the generated step-by-step action plans, we adopt BLEU and SPICE as evaluation metrics.

Table 4: Evaluation results on the SkyAgent-Plan3k dataset.

Model	ShangHai		ShenZhen		Campus		Residence	
	BLEU-1 ↑	SPICE ↑	BLEU-1 ↑	SPICE ↑	BLEU-1↑	SPICE ↑	BLEU-1 ↑	SPICE ↑
VLMs								
MiniGPT-4 (ZS) [29]	12.88	4.03	13.96	4.51	16.89	4.25	15.85	4.51
GPT-4o (ZS) [28]	10.64	5.12	11.52	5.20	11.67	5.03	11.83	4.83
LLaVA-v1.5-Vicuna-7B (ZS) [68]	10.54	12.69	11.90	15.87	4.22	4.12	23.39	30.33
LLaVA-v1.5-Vicuna-13B (ZS) [68]	12.35	13.86	12.05	18.37	5.09	4.73	21.59	27.79
RSVLMs								
RingMo-Agent (FT)	37.33	31.68	37.76	31.85	42.79	28.45	42.78	37.68

Results. As shown in Table 4, we significantly outperform all baseline methods across all scene categories, achieving the highest BLEU-1 and SPICE in every case. Existing RSVLMs have yet to explore their reasoning capabilities in 3D space guided by multi-image inputs. These results validate the effectiveness of our model in bridging the gap between visual-language reasoning and actionable control.

5.4 Relation Reasoning

Dataset. We conduct evaluation experiments on two relation reasoning datasets: the FIT-RS dataset [16] and our self-constructed ReCon1M-REL dataset. The FIT-RS dataset includes 54 relation categories, with all images cropped to a fixed resolution of 512×512 . In contrast, ReCon1M-REL retains 59 relation categories and adopts variable resolutions.

Metrics. We use F1-score to assess this task.

Table 5: Evaluation results on the FIT-RS and ReCon1M-REL datasets.

Model	FIT-RS		ReCon1M-REL	
	F1-score ↑	F1-score ↑	F1-score ↑	F1-score ↑
VLMs				
MiniGPT-v2 (ZS) [29]	0.00	0.00	-	-
DeepSeek-VL2 (ZS) [32]	0.06	0.30	-	-
Kosmos-2 (ZS) [25]	0.00	0.15	-	-
Shikra (ZS) [26]	0.52	0.65	-	-
RSVLMs				
SkySenseGPT (FT) [16]	74.33	-	-	-
RingMo-Agent (FT)	75.34	90.23	-	-

Results. As presented in Table 5, RingMo-Agent demonstrates superior performance, achieving an accuracy of 75.34%, which slightly surpasses the 74.33% obtained by SkySenseGPT [16]. On ReCon1M-REL dataset, we achieve the F1-score of 90.23%. In contrast, other VLMs such as MiniGPT-v2 [29], as well as other RSVLMs [17], are unable to perform this type of task effectively, as they have not been trained on the corresponding domain-specific corpora.

5.5 Instruction Decomposition

Dataset. We evaluate our model on the self-constructed ReCon1M-DEC dataset.

Metrics. To comprehensively assess performance, we evaluate performance using mAP@50 and F1-score.

Table 6: Evaluation results on the ReCon1M-DEC dataset.

Model	mAP@50 ↑	F1-Score ↑
MiniGPT-v2 (FT) [24]	11.50	15.19
DeepSeek-VL2 (FT) [32]	19.80	10.32
RingMo-Agent (FT)	24.20	32.85

Results. We report the precision of object identification within the region, as well as the accuracy of relation reasoning, in Table 6. Existing RSVLMs are not capable of handling this type of task. For comparison, we fine-tuned MiniGPT-v2 [24] and DeepSeek-VL2 [32] using its default LoRA configuration, aligning with our own training settings. The results are reported as baselines. Specifically, MiniGPT-v2 achieved 11.50% in mAP@50 and 15.19% in F1 score, while DeepSeek-VL2 achieved 19.80% and 10.32%, respectively. In contrast, our model achieves significantly higher performance, with an mAP@50 of 24.20% and an F1 score of 32.85%. This improvement stems from domain-specific pre-training and fine-tuning, which enhance the model’s ability to capture characteristics of small objects in RS imagery.

5.6 Image Captioning

Dataset. We conduct evaluation separately on optical, SAR, and infrared modalities. The optical datasets utilize UCM-Captions [46] and NWPU-Captions [48], while the SAR and infrared modalities are tested on our self-constructed datasets.

Metrics. The performance is reported using BLEU, METEOR, ROUGE-L, and CIDEr metrics.

Table 7: Evaluation results on the SAR-CAP and IR-CAP datasets.

Model	SAR-CAP						IR-CAP					
	BLEU-1 ↑	BLEU-2 ↑	BLEU-3 ↑	BLEU-4 ↑	METEOR ↑	ROUGE-L ↑	BLEU-1 ↑	BLEU-2 ↑	BLEU-3 ↑	BLEU-4 ↑	METEOR ↑	ROUGE-L ↑
MiniGPT-v2 (ZS) [24]	7.00	3.64	1.59	0.60	7.67	9.25	5.65	3.35	1.92	0.98	7.62	8.67
DeepSeek-VL2 (ZS) [32]	12.52	5.88	1.88	0.60	10.65	14.10	13.95	7.57	3.47	1.43	12.48	15.12
RingMo-Agent (FT)	55.93	44.49	33.57	23.94	25.06	51.12	56.84	40.45	29.17	21.50	26.15	43.13

Table 8: Evaluation results on the UCM Dataset.

Model	BLEU-4 ↑	METEOR ↑	ROUGE-L ↑	CIDEr ↑
<i>Specialist models</i>				
SAA (FT) [69]	64.77	38.59	69.42	294.51
Post-processing (FT) [70]	62.62	40.80	74.06	309.64
<i>VLMs</i>				
MiniGPT-4 (ZS) [29]	18.10	33.36	41.37	0.03
Shikra (ZS) [26]	33.98	32.56	56.73	56.69
MiniGPT-v2 (ZS) [24]	36.16	32.41	56.57	60.66
DeepSeek-VL2 (ZS) [32]	1.87	13.91	19.06	8.29
<i>RSVLMs</i>				
RSGPT (FT) [20]	65.74	42.21	78.34	333.23
RingMoGPT (FT) [19]	-	49.90	83.29	359.32
RS-CapRet (FT) [13]	67.00	47.20	81.70	354.80
RS-LLaVA (FT) [21]	72.84	47.98	85.17	349.43
SkyEyeGPT (FT) [15]	78.41	46.24	79.49	236.75
RingMo-Agent (FT)	77.63	51.79	85.51	373.68

Results. As shown in Table 8, RingMo-Agent outperforms specialized models like SAA [69] across all metrics on the UCM dataset, and surpasses SkyEyeGPT [15] by 5.55% and 6.02% in METEOR and ROUGE-L, respectively. Zero-shot results on NWPU-Captions (Table 9) further demonstrate its strong generalization despite multi-task fine-tuning. Additionally, on SAR and infrared datasets (Table 7), RingMo-Agent significantly outperforms unfinetuned VLMs, indicating robust cross-modal captioning capability.

5.7 VQA

Dataset. We report the fine-tuned result on RSVQA-LR [39] test set and zero-shot result on RSVQA-HR [39]. These benchmarks assess the model’s ability to understand object types, quantities, and spatial locations.

Metrics. We computed the accuracy for each question type as well as the overall average accuracy.

Results. Table 10 presents the results on RSVQA-LR, focusing on two question types: Presence and Comparison. Compared with other methods, we achieve 93.10% accuracy on presence questions, surpassing the second-best

Table 9: Zero-shot results on the NWPU-Captions dataset.

Model	METEOR \uparrow	ROUGE-L \uparrow	CIDEr \uparrow
VLMs			
Qwen-VL [31]	12.60	26.24	24.64
MiniGPT-v2 [24]	13.70	26.60	19.70
LLaVA [71]	13.70	28.80	32.60
DeepSeek-VL2 [32]	14.88	19.56	3.18
RSVLMs			
RingMoGPT [19]	20.90	37.50	74.70
RingMo-Agent	18.81	34.11	48.39

Table 10: Evaluation results on the RSVQA-LR Dataset.

Model	Pre. Acc \uparrow	Comp. Acc \uparrow	Avg. Acc \uparrow
Specialist models			
EasyToHard (FT) [72]	90.66	87.49	89.08
Bi-Modal (FT) [73]	91.06	91.16	91.11
SHRNet (FT) [74]	91.03	90.48	90.76
VLMs			
MiniGPT-4 (ZS) [29]	43.86	57.55	50.71
Shikra (ZS) [26]	46.47	60.31	53.39
Qwen-vl (ZS) [31]	38.57	67.59	53.08
MiniGPT-v2 (ZS) [24]	49.85	63.09	56.47
InstructBLIP (ZS) [23]	48.83	65.92	57.38
mPLUG-Owl2 (ZS) [75]	74.04	63.69	68.87
LLaVA-1.5 (ZS) [68]	55.46	68.20	61.83
RSVLMs			
RSGPT (FT) [20]	91.17	91.70	<u>91.44</u>
Geochat (FT) [17]	91.09	90.33	90.71
LHRS-Bot (FT) [14]	89.07	88.51	88.79
H ² RSVLM (FT) [18]	89.58	89.79	89.69
RS-LLaVA (FT) [21]	<u>92.80</u>	<u>91.33</u>	92.07
SkyEyeGPT (FT) [15]	88.93	88.63	88.78
RingMo-Agent (FT)	93.10	87.50	90.30

Table 11: Zero-shot results on the RSVQA-HR dataset.

Model	Pre. Acc \uparrow	Comp. Acc \uparrow	Avg. Acc \uparrow
VLMs			
LLaVA-v1.5 [71]	66.44	60.41	63.06
MiniGPT-v2 [24]	40.79	50.91	46.46
RSVLMs			
EarthGPT [11]	62.77	79.53	72.06
Geochat [17]	58.45	83.19	70.82
H ² RSVLM [18]	65.00	83.70	74.35
SkySenseGPT [16]	<u>69.14</u>	84.14	76.64
RingMo-Agent	75.24	<u>83.92</u>	79.58

RS-LLaVA [21] by 0.3%. Table 11 shows that we achieve accuracies of 75.24% on presence questions, 83.92% on comparison questions, and an overall average accuracy of 79.58%, demonstrating a clear advantage over other RSVLMs.

5.8 Classification

Dataset. We report results on the AID [40] and NWPU-RESISC45 [49] datasets, along with our self-constructed IR-CLA and SAR-CLA datasets, and provide zero-shot results on UCMerced-LandUse [51] and WHU-RS19 [50].

Metrics. We report classification accuracy as the evaluation metric. For optical datasets, each question queries the image category with five candidate options provided.

Results. As shown in Table 12, RingMo-Agent achieves strong performance on optical tasks, reaching 94.72% accuracy on NWPU-RESISC45. For SAR and infrared classification, we achieve 92.67% and 99.45%, respectively, as shown in Table 14. On the zero-shot set UCMerced-LandUse, we achieve 88%, surpassing other RSVLMs, as shown in Table 13. This improvement is attributed to the robust generalization ability of the model.

5.9 Object Detection

Dataset. RingMo-Agent supports object detection across three modalities: optical, SAR, and infrared. SAR results are reported on SARDet-100k [45], and infrared results on our IR-DET dataset.

Table 12: Evaluation results on the AID and NWPU-RESISC45 datasets.

Model	AID (Acc ↑)	NWPU-RESISC45 (Acc ↑)
Specialist models		
LSENet (FT) [76]	94.41	93.34
SeCo-ResNet-50 (FT) [77]	93.47	92.91
VLMs		
MiniGPT-4 (ZS) [29]	43.86	57.55
MiniGPT-v2 (ZS) [24]	77.90	83.34
DeepSeek-VL2 (ZS) [32]	81.89	84.69
RSVLMs		
EarthGPT (FT) [11]	-	93.84
RingMo-Agent (FT)	91.67	94.72

Table 13: Zero-shot results on the WHU-RS19 and UCMerced-LandUse datasets.

Model	WHU-RS19 (Acc ↑)	UCMerced-LandUse (Acc ↑)
VLMs		
Qwen-VL [31]	83.51	62.90
MiniGPT-v2 [24]	62.08	4.76
LLaVA [71]	74.52	68.00
CLIP [78]	87.50	-
RSVLMs		
RemoteCLIP [37]	94.66	-
GeoChat [17]	86.47	84.43
RingMoGPT [19]	97.71	86.48
SkySenseGPT [16]	97.02	-
RingMo-Agent	96.81	88.00

Table 14: Evaluation results on the SAR-CLA and IR-CLA datasets.

Model	SAR-CLA		IR-CLA	
	Acc ↑	Acc ↑	Acc ↑	Acc ↑
MiniGPT-v2 (ZS) [24]	5.87		60.52	
DeepSeek-VL2 (ZS) [32]	4.40		45.15	
RingMo-Agent (FT)	92.67		99.45	

Metrics. We report both per-class and overall detection performance using mAP@50.

Results. As shown in Table 16 and Table 15, RingMo-Agent achieves superior performance on SAR and infrared object detection compared to existing models. Compared to VLMs with strong localization capabilities, our model, having undergone multiple rounds of fine-tuning, demonstrates significantly improved understanding across different modalities.

5.10 Ablation Study

To assess the effectiveness of key design components within RingMo-Agent, we conduct ablation experiments from three perspectives: (1) the impact of the two-stage training paradigm, (2) the role of modality-specific embedding layers, and (3) the function of the trajectory decoder.

Two-Stage Training. We pretrain RingMo-Agent on a large-scale RS image-text dataset to enhance image-text generation. To assess the impact, we compare it with a baseline initialized from DeepSeek-VL2 without pretraining, using identical datasets and training strategies. As shown in Table 17 and Table 18, the two-stage training consistently improves performance on both reasoning and perception tasks. This gain stems from the ability to capture RS-specific features such as resolution, viewpoint, and spectral differences, thereby boosting generalization.

Modality-Specific Embedding Layers. We fine-tune separate embedding layers for optical, SAR, and infrared modalities, guided by modality labels during training and inference. Only the embedding layers are updated, while the rest of the visual encoder remains frozen. As shown in Table 19, freezing these layers and removing modality-specific design degrades performance, highlighting the importance of dedicated extractors. Despite sharing identical structures, separate embeddings better capture modality-specific characteristics, avoiding feature degradation caused by forcing heterogeneous data through shared filters.

Trajectory Decoder. To address challenges such as loss imbalance and difficulty in sequential constraints when LLMs generate point-by-point trajectories, we additionally designed a trajectory decoder specifically for task scheduling. In Table 20, we additionally compare the case without using the trajectory decoder, where the model predicts the next token sequentially to output the six-dimensional coordinates for each point. The results on the CityNav dataset [43]

Table 15: Evaluation results on the IR-DET dataset.

Model	Fine-Grained Categories												Coarse-Grained Categories				mAP@50 ↑
	Bulk Carrier	Bus	Canoe	Container Ship	Fishing Boat	Liner	Nonmotor Vehicle	Other Vehicle	Sailboat	Truck	Warship	Bike	Cyclist	People	Ship	Vehicle	
MiniGPT-v2 (ZS) [24]	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
Kosmos-2 (ZS) [25]	0.00	3.08	0.00	0.00	0.00	0.00	0.00	4.29	7.48	0.00	0.00	0.00	2.91	39.20	30.42	12.48	
Shirka (ZS) [26]	0.00	2.66	0.00	0.00	0.00	0.00	0.00	4.24	3.85	0.00	0.00	0.00	1.43	23.83	0.92	5.27	
RingMo-Agent (FT)	70.63	21.25	74.43	59.77	91.47	87.03	78.11	62.71	70.80	39.61	89.66	35.95	16.61	53.67	42.52	63.83	59.88

Table 16: Evaluation results on the SARDet-100k dataset.

Model	Ship	Aircraft	Bridge	Harbor	Car	Tank	mAP@50 ↑
MiniGPT-v2 (ZS) [24]	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Kosmos-2 (ZS) [25]	7.70	19.82	0.00	0.00	0.00	0.00	6.88
Shirka (ZS) [26]	0.03	0.07	0.00	0.00	0.00	0.00	0.02
RingMo-Agent (FT)	74.43	52.93	49.90	62.30	63.37	20.10	53.84

Table 17: Ablation results on the impact of two-stage training for reasoning tasks.

Model	Relation Reasoning				Instruction Decomposition			
	FIT-RS (F1-score ↑)	ReCon1M-REL (F1-score ↑)	ReCon1M-DEC (mAP@50 ↑)	ReCon1M-DEC (F1-score ↑)	ReCon1M-DEC (mAP@50 ↑)	ReCon1M-DEC (F1-score ↑)	ReCon1M-DEC (mAP@50 ↑)	ReCon1M-DEC (F1-score ↑)
RingMo-Agent (one stage)	72.82		87.11		16.72		20.32	
RingMo-Agent (two stage)	75.34		90.23		24.20		32.85	

Table 18: Ablation results on the impact of two-stage training for perception tasks.

Model	Object Detection			Image Captioning			VQA	
	SARDet-100k (mAP@50 ↑)	IR-Det (mAP@50 ↑)	UCM (BLEU-4 ↑)	SAR-CAP (BLEU-4 ↑)	IR-CAP (BLEU-4 ↑)	RSVQA-LR (Avg. Acc ↑)	RSVQA-LR (Avg. Acc ↑)	RSVQA-LR (Avg. Acc ↑)
RingMo-Agent (one stage)	50.12	51.56	76.12	50.12	49.16	89.96		
RingMo-Agent (two stage)	53.84	59.88	77.63	55.93	56.84	90.30		

Table 19: Ablation results on the impact of modality-specific embedding layers, where MSEL denotes modality-specific embedding layers.

Model	MSEL	Object Detection			Image Captioning			Image Classification	
		SARDet-100k (mAP@50 ↑)	IR-Det (mAP@50 ↑)	UCM (BLEU-4 ↑)	SAR-CAP (BLEU-4 ↑)	IR-CAP (BLEU-4 ↑)	SAR-CLA (Acc ↑)	IR-CLA (Acc ↑)	
RingMo-Agent	✗	48.10	51.23	74.11	46.32	51.08	88.17	96.60	
RingMo-Agent	✓	53.84	67.36	77.63	55.93	56.84	92.67	99.45	

Table 20: Ablation results on the impact of trajectory decoder.

Model	Trajectory Decoder	Validation Seen				Validation Unseen				Test Unseen			
		NE ↓	SR ↑	OSR ↑	SPL ↑	NE ↓	SR ↑	OSR ↑	SPL ↑	NE ↓	SR ↑	OSR ↑	SPL ↑
RingMo-Agent	✗	196.4	2.65	14.23	2.16	220.1	1.94	12.36	1.91	187.9	1.57	13.16	2.02
RingMo-Agent	✓	132.0	5.89	21.39	5.24	156.0	4.91	17.22	4.33	149.6	4.74	18.94	4.17

demonstrate the effectiveness of the trajectory decoder, showing improved performance in both seen and unseen scenarios.

5.11 Visualization

To provide a clearer understanding of the model behavior, we present visualizations of representative results in this work, as shown in Figure 5. These include task scheduling (a), action decision (b), instruction decomposition (c), relation reasoning (d), as well as classification, captioning, and detection on SAR and infrared data (e, f). Additionally, Figure 6 provides visualized examples to demonstrate the dialogue interactions.

6 Conclusion

In this work, we propose RingMo-Agent, a unified foundation model tailored for RS data across multiple platforms and modalities. It is capable of handling images with diverse viewing angles and sensing characteristics, and performing complex reasoning tasks. The model benefits from several key supports: a large-scale dataset RS-VL3M comprising over 3 million image-text pairs; the construction of separated embedding layers to address distribution differences across modalities and isolate the feature extraction process; and the introduction of a token-based high-dimensional hidden

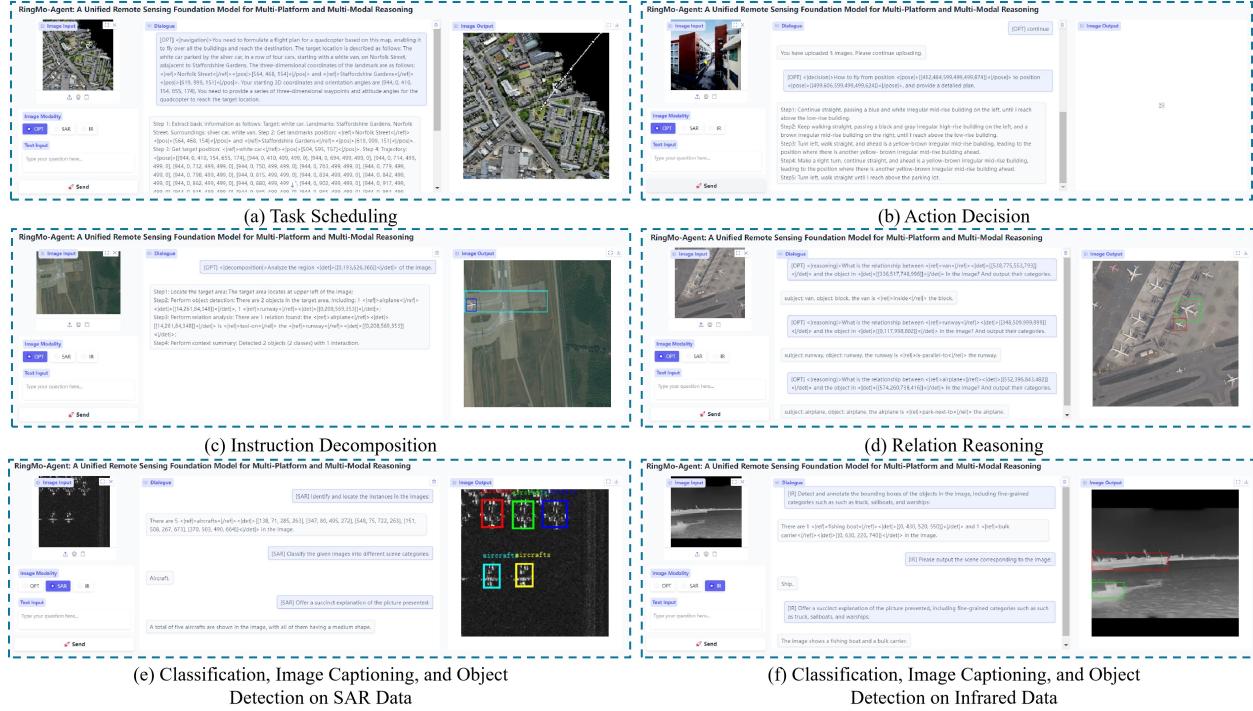


Figure 5: The visual results presented by RS-Agent in response to user questions about image content.

state decoding mechanism for long-horizon spatial tasks, which significantly expands the task coverage. We envision that future RS agents will further advance toward deeper logical reasoning abilities, thereby supporting real-world applications such as damage assessment and decision-making in emergency scenarios like disaster response.

References

- [1] 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.
- [2] Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. [arXiv preprint arXiv:2401.06066](#), 2024.
- [3] Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. [arXiv preprint arXiv:2405.04434](#), 2024.
- [4] DeepSeek-AI. Deepseek-v3 technical report, 2025.
- [5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. [Advances in neural information processing systems](#), 33:1877–1901, 2020.
- [6] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. [arXiv preprint arXiv:2303.08774](#), 2023.
- [7] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. [arXiv preprint arXiv:2302.13971](#), 2023.

Examples			
Image	Task	Question	Answer
	Image Captioning	[SAR] Offer a succinct explanation of the picture presented:	This image shows five aircrafts on a runway.
	Image Classification	[IR] What is the scene category of the given image:	Ship.
	Object Detection	[OPT] Detect and annotate the bounding boxes of the objects in the image:	There are 4 <ref>aircrafts</ref><det>[[107, 149, 384, 533], [117, 544, 428, 922], [680, 213, 964, 604], [698, 615, 985, 989]]</det> in the image.
	Visual Question Answer	[OPT] Question how can people improve their strength in this image? short answer:	Aerobic.
	Relation Reasoning	[OPT] <reasoning>What is the relationship between <ref>solar-panel</ref><det>[[170,546,242,646]]</det> and the object in <det>[[0,236,301,872]]</det> in the image? And output their categories.	Subject: solar-panel, object: building, the solar-panel is <rel>power</rel> the building.
	Instruction Decomposition	[OPT] <decomposition>Break down visual elements in region <det>[[343,641,586,739]]</det>.	Step1: Locate the target area: The target area locates at lower area of the image: Step2: Perform object detection: There are 2 objects in the target area, including: 2 <ref>airplanes</ref><det>[[365,651,446,732], [490,649,566,731]]</det>; Step3: Perform relation analysis: There are 2 relations found: the <ref>airplane</ref><det>[[490,649,566,731]]</det> is <rel>park-next-to</rel> the <ref>airplane</ref><det>[[365,651,446,732]]</det>, ...; Step4: Perform context summary: Detected 2 objects (1 class) with 2 interactions.
	Action Decision	[OPT] <decision> How to fly from position <det>[[719,749,439,499,874]]</det> to position <det>[[352,470,439,499,499,998]]</det>, and provide a detailed plan.	Step1: Go straight, passing a brown irregular mid-rise building on the right and a red irregular mid-rise building on the left, until I reach above the trees. Step2: Continue straight ahead towards a blue-brown irregular mid-rise building, reaching above a parking lot.
	Task Scheduling	[OPT] <navigation> You need to formulate a flight plan for a quadcopter based on this map, enabling it to fly over all the buildings and reach the destination. The target location is described as follows: The row of gray and brown houses on Trinity Street with the Blue Boar A to H building behind them. The three-dimensional coordinates of the landmark are as follows: <ref>Trinity Street</ref><pos>[866, 933, 143]</pos> and Blue Boar A to H building does not have landmark positions. Your starting 3D coordinates and orientation angles are [0, 187, 575, 612, 321, 46]. You need to provide a series of three-dimensional waypoints and attitude angles for the quadcopter to reach the target location.	Step1: Extract basic information as follows: Target: row of gray and brown houses. Landmarks: Blue Boar A to H building, Trinity Street. Surroundings: No surroundings available. Step2: Get landmarks position: <ref>Trinity Street</ref><pos>[866, 933, 143]</pos> and Blue Boar A to H building does not have landmark positions. Step3: Get target position: <ref>row of gray and brown houses</ref><pos>[999, 999, 178]</pos>. Step4: Trajectory: <pose>[[0, 187, 575, 612, 321, 46], [0, 187, 575, 765, 828, 233], ..., [972, 985, 146, 572, 803, 109]]</pose>.

Figure 6: Examples of supported tasks.

- [8] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- [9] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024.
- [10] Xiang Li, Congcong Wen, Yuan Hu, Zhenghang Yuan, and Xiao Xiang Zhu. Vision-language models in remote sensing: Current progress and future trends. *IEEE Geoscience and Remote Sensing Magazine*, 2024.
- [11] 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.
- [12] Wei Zhang, Miaoxin Cai, Tong Zhang, Guoqiang Lei, Yin Zhuang, and Xuerui Mao. Popeye: A unified visual-language model for multi-source ship detection from remote sensing imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2024.
- [13] João Daniel Silva, João Magalhães, Devis Tuia, and Bruno Martins. Large language models for captioning and retrieving remote sensing images. *arXiv preprint arXiv:2402.06475*, 2024.
- [14] Dilxat Muhtar, Zhenshi Li, Feng Gu, Xueliang Zhang, and Pengfeng Xiao. Lhrs-bot: Empowering remote sensing with vgi-enhanced large multimodal language model. In *European Conference on Computer Vision*, pages 440–457. Springer, 2024.
- [15] Yang Zhan, Zhitong Xiong, and Yuan Yuan. Skyeyegpt: Unifying remote sensing vision-language tasks via instruction tuning with large language model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 221:64–77, 2025.
- [16] Junwei Luo, Zhen Pang, Yongjun Zhang, Tingzhu Wang, Linlin Wang, Bo Dang, Jiangwei Lao, Jian Wang, Jingdong Chen, Yihua Tan, et al. Skysensegpt: A fine-grained instruction tuning dataset and model for remote sensing vision-language understanding. *arXiv preprint arXiv:2406.10100*, 2024.
- [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.
- [18] Chao Pang, Jiang Wu, Jiayu Li, Yi Liu, Jiaxing Sun, Weijia Li, Xingxing Weng, Shuai Wang, Litong Feng, Gui-Song Xia, et al. H2rsvlm: Towards helpful and honest remote sensing large vision language model. *arXiv e-prints*, pages arXiv–2403, 2024.

- [19] Peijin Wang, Huiyang Hu, Boyuan Tong, Ziqi Zhang, Fanglong Yao, Yingchao Feng, Zining Zhu, Hao Chang, Wenhui Diao, Qixiang Ye, et al. Ringmogpt: A unified remote sensing foundation model for vision, language, and grounded tasks. *IEEE Transactions on Geoscience and Remote Sensing*, 2024.
- [20] Yuan Hu, Jianlong Yuan, Congcong Wen, Xiaonan Lu, and Xiang Li. Rsgpt: A remote sensing vision language model and benchmark. *arXiv preprint arXiv:2307.15266*, 2023.
- [21] Yakoub Bazi, Laila Bashmal, Mohamad Mahmoud Al Rahhal, Riccardo Ricci, and Farid Melgani. Rs-llava: A large vision-language model for joint captioning and question answering in remote sensing imagery. *Remote Sensing*, 16(9):1477, 2024.
- [22] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.
- [23] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023.
- [24] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*, 2023.
- [25] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.
- [26] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm’s referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.
- [27] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9579–9589, 2024.
- [28] OpenAI. Gpt-4 technical report. 2023.
- [29] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.
- [30] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023.
- [31] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023.
- [32] 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.
- [33] Xuefeng Li, Haoyang Zou, and Pengfei Liu. Torl: Scaling tool-integrated rl. *arXiv preprint arXiv:2503.23383*, 2025.
- [34] Georg Wölflein, Dyke Ferber, Daniel Truhn, Ognjen Arandjelović, and Jakob Nikolas Kather. Llm agents making agent tools. *arXiv preprint arXiv:2502.11705*, 2025.
- [35] Ziyu Wan, Yunxiang Li, Xiaoyu Wen, Yan Song, Hanjing Wang, Linyi Yang, Mark Schmidt, Jun Wang, Weinan Zhang, Shuyue Hu, et al. Rema: Learning to meta-think for llms with multi-agent reinforcement learning. *arXiv preprint arXiv:2503.09501*, 2025.
- [36] 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.
- [37] Fan Liu, Delong Chen, Zhangqingyun 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.
- [38] Fanglong Yao, Yuanchang Yue, Youzhi Liu, Xian Sun, and Kun Fu. Aeroverse: Uav-agent benchmark suite for simulating, pre-training, finetuning, and evaluating aerospace embodied world models. *arXiv preprint arXiv:2408.15511*, 2024.
- [39] Sylvain Lobry, Diego Marcos, Jesse Murray, and Devis Tuia. Rsvqa: Visual question answering for remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 58(12):8555–8566, 2020.

- [40] Gui-Song Xia, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, Liangpei Zhang, and Xiaoqiang Lu. Aid: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3965–3981, 2017.
- [41] Ke Li, Gang Wan, Gong Cheng, Lijiu Meng, and Junwei Han. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS journal of photogrammetry and remote sensing*, 159:296–307, 2020.
- [42] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3974–3983, 2018.
- [43] Jungdae Lee, Taiki Miyanishi, Shuhei Kurita, Koya Sakamoto, Daichi Azuma, Yutaka Matsuo, and Nakamura Inoue. Citynav: Language-goal aerial navigation dataset with geographic information. *arXiv preprint arXiv:2406.14240*, 2024.
- [44] Gong Cheng, Junwei Han, Peicheng Zhou, and Lei Guo. Multi-class geospatial object detection and geographic image classification based on collection of part detectors. *ISPRS Journal of Photogrammetry and Remote Sensing*, 98:119–132, 2014.
- [45] Yuxuan Li, Xiang Li, Weijie Li, Qibin Hou, Li Liu, Ming-Ming Cheng, and Jian Yang. Sardet-100k: Towards open-source benchmark and toolkit for large-scale sar object detection. *arXiv preprint arXiv:2403.06534*, 2024.
- [46] Bo Qu, Xuelong Li, Dacheng Tao, and Xiaoqiang Lu. Deep semantic understanding of high resolution remote sensing image. In *2016 International conference on computer, information and telecommunication systems (Cits)*, pages 1–5. IEEE, 2016.
- [47] Xiaoqiang Lu, Binqiang Wang, Xiangtao Zheng, and Xuelong Li. Exploring models and data for remote sensing image caption generation. *IEEE Transactions on Geoscience and Remote Sensing*, 56(4):2183–2195, 2017.
- [48] Qimin Cheng, Haiyan Huang, Yuan Xu, Yuzhuo Zhou, Huanying Li, and Zhongyuan Wang. Nwpucaptions dataset and mlca-net for remote sensing image captioning. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–19, 2022.
- [49] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.
- [50] Gui-Song Xia, Wen Yang, Julie Delon, Yann Gousseau, Hong Sun, and Henri Maître. Structural high-resolution satellite image indexing. In *ISPRS TC VII Symposium-100 Years ISPRS*, volume 38, pages 298–303, 2010.
- [51] Yi Yang and Shawn Newsam. Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, pages 270–279, 2010.
- [52] Xian Sun, Qiwei Yan, Chubo Deng, Chenglong Liu, Yi Jiang, Zhongyan Hou, Wanxuan Lu, Fanglong Yao, Xiaoyu Liu, Lingxiang Hao, et al. Recon1m: A large-scale benchmark dataset for relation comprehension in remote sensing imagery. *arXiv preprint arXiv:2406.06028*, 2024.
- [53] Jiashun Suo, Tianyi Wang, Xingzhou Zhang, Haiyang Chen, Wei Zhou, and Weisong Shi. Hit-uav: A high-altitude infrared thermal dataset for unmanned aerial vehicle-based object detection. *Scientific Data*, 10(1):227, 2023.
- [54] InfiRay. Sea-shipping. Online, 2021. Available: http://openai.iraytek.com/apply/Sea_shipping.html/.
- [55] InfiRay. Infrared-security. Online, 2021. Available: http://openai.iraytek.com/apply/Infrared_security.html/.
- [56] InfiRay. Aerial-mancar. Online, 2021. Available: http://openai.raytrontek.com/apply/Aerial_mancar.html/.
- [57] InfiRay. Double-light-vehicle. Online, 2021. Available: http://openai.raytrontek.com/apply/Double_light_vehicle.html/.
- [58] Center for Optics Research and Engineering of Shandong University. Oceanic-ship. Online, 2020. Available: <http://www.gxzx.sdu.edu.cn/info/1133/2174.htm/>.
- [59] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.
- [60] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

- [61] Haifeng Li, Xin Dou, Chao Tao, Zhixiang Wu, Jie Chen, Jian Peng, Min Deng, and Ling Zhao. Rsi-cb: A large-scale remote sensing image classification benchmark using crowdsourced data. *Sensors*, 20(6), 2020.
- [62] Qin Zou, Lihao Ni, Tong Zhang, and Qian Wang. Deep learning based feature selection for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 12(11):2321–2325, 2015.
- [63] Pei Zhang, Yunpeng Bai, Dong Wang, Bendu Bai, and Ying Li. Few-shot classification of aerial scene images via meta-learning. *Remote Sensing*, 13(1), 2021.
- [64] Weixun Zhou, Shawn Newsam, Congmin Li, and Zhenfeng Shao. Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS journal of photogrammetry and remote sensing*, 145:197–209, 2018.
- [65] Dongyang Hou, Zelang Miao, Huaqiao Xing, and Hao Wu. Two novel benchmark datasets from arcgis and bing world imagery for remote sensing image retrieval. *International Journal of Remote Sensing*, 42(1):240–258, 2021.
- [66] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018.
- [67] Shubo Liu, Hongsheng Zhang, Yuankai Qi, Peng Wang, Yanning Zhang, and Qi Wu. Aerialvln: Vision-and-language navigation for uavs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15384–15394, 2023.
- [68] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024.
- [69] Xiaoqiang Lu, Binqiang Wang, and Xiangtao Zheng. Sound active attention framework for remote sensing image captioning. *IEEE Transactions on Geoscience and Remote Sensing*, 58(3):1985–2000, 2019.
- [70] Genc Hoxha, Giacomo Scuccato, and Farid Melgani. Improving image captioning systems with postprocessing strategies. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–13, 2023.
- [71] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.
- [72] Zhenghang Yuan, Lichao Mou, Qi Wang, and Xiao Xiang Zhu. From easy to hard: Learning language-guided curriculum for visual question answering on remote sensing data. *IEEE transactions on geoscience and remote sensing*, 60:1–11, 2022.
- [73] Yakoub Bazi, Mohamad Mahmoud Al Rahhal, Mohamed Lamine Mekhalfi, Mansour Abdulaziz Al Zuair, and Farid Melgani. Bi-modal transformer-based approach for visual question answering in remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2022.
- [74] Zixiao Zhang, Licheng Jiao, Lingling Li, Xu Liu, Puhua Chen, Fang Liu, Yuxuan Li, and Zhicheng Guo. A spatial hierarchical reasoning network for remote sensing visual question answering. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–15, 2023.
- [75] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 13040–13051, 2024.
- [76] Qi Bi, Kun Qin, Han Zhang, and Gui-Song Xia. Local semantic enhanced convnet for aerial scene recognition. *IEEE Transactions on Image Processing*, 30:6498–6511, 2021.
- [77] Oscar Manas, Alexandre Lacoste, Xavier Giró-i Nieto, David Vazquez, and Pau Rodriguez. Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9414–9423, 2021.
- [78] 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.