

SpatialBot: Precise Spatial Understanding with Vision Language Models

Wenxiao Cai^{1,2,3*}, Iaroslav Ponomarenko⁴, Jianhao Yuan⁵, Xiaoqi Li⁴,
Wankou Yang⁶, Hao Dong⁴, Bo Zhao^{1,3†}

¹School of Artificial Intelligence, Shanghai Jiao Tong University ²Stanford University
³BAAI ⁴Peking University ⁵University of Oxford ⁶Southeast University

Abstract

Vision Language Models (VLMs) have achieved impressive performance in 2D image understanding; however, they still struggle with spatial understanding, which is fundamental to embodied AI. In this paper, we propose SpatialBot, a model designed to enhance spatial understanding by utilizing both RGB and depth images. To train VLMs for depth perception, we introduce the SpatialQA and SpatialQA-E datasets, which include multi-level depth-related questions spanning various scenarios and embodiment tasks. SpatialBench is also developed to comprehensively evaluate VLMs' spatial understanding capabilities across different levels. Extensive experiments on our spatial-understanding benchmark, general VLM benchmarks, and embodied AI tasks demonstrate the remarkable improvements offered by SpatialBot. The model, code, and datasets are available at <https://github.com/BAAI-DCAI/SpatialBot>.

1. Introduction

Recently, Vision Language Models (VLM) [7, 28, 46, 52, 66, 76] have demonstrated notable capabilities in general 2D visual understanding and reasoning, based on vision encoder-based perception and language model-based reasoning. However, it is still challenging for VLMs to comprehend spatial information from 2D images merely, which is the key to implementing various real-world tasks [5, 22, 36, 43, 58], particularly those embodied AI related tasks such as manipulation [25, 48, 56, 64] and navigation [18, 44, 72, 79].

The main challenges for VLMs to have spatial understanding ability are in the following aspects: 1) Popular VLMs have limited capacity to understand depth informa-

*This work was done when Wenxiao Cai was a visiting student in SJTU, and when he was an intern in BAAI.

†Corresponding author: bo.zhao@sjtu.edu.cn .

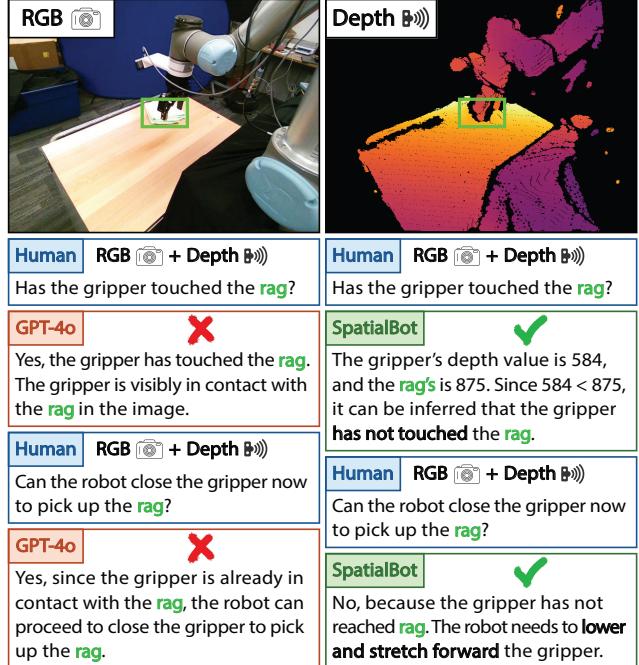


Figure 1. *SpatialBot* has better spatial understanding ability than GPT-4o. From a single RGB image, even a human can not judge whether the gripper has touched the rag. GPT-4o fails even if looking at both RGB and depth images. *SpatialBot* first obtains depth information of target objects from the depth map, and then judges that the gripper has not yet touched the rag.

tion as they are only trained on RGB images without seeing depth images. In addition, the training tasks need little depth information to solve. Consequently, directly inputting depth maps into VLMs results in poor performance. 2) A well-designed dataset for training VLMs to understand depth is absent. The popular VLM tuning datasets provide neither depth maps nor depth-related tasks. 3) The inconsistency of the scales between indoor and outdoor numerical depth is also an important problem preventing VLM from

uniformly processing depth in various tasks. For example, tasks such as indoor navigation and manipulation require millimeter-level precision, whereas outdoor tasks do not necessitate such high precision but demand a broader depth range.

To address these challenges, we propose *SpatialBot*, which can precisely comprehend spatial information through depth images and perform robotic manipulation. We design a progressive training approach to first improve the general spatial understanding capacity of VLMs with the proposed *SpatialQA* dataset, which contains general conversation tasks. We then leverage this spatial understanding capacity for embodied tasks using the collected robot manipulation task dataset, *SpatialQA-E*. We design various purpose-specific QA tasks that heavily rely on spatial understanding from low to high levels. These tasks include low-level depth estimation, middle-level object detection, referring QA, and depth comparison, high-level tasks that require depth reasoning, such as understanding spatial relationships in both general conversations and robot manipulation. To enable the model to accurately obtain depth information, we designed a depth API that allows the model to query the depth values of individual pixels or regions.

We validate the spatial comprehension capacity of VLMs with *SpatialBench* which consists of manually annotated question-answer pairs on spatial understanding and reasoning. We also deploy *SpatialBot* on robots to do manipulation tasks, for example, picking up the teacup in the middle and placing it on the closest board, as shown in Fig. 6. The experimental results verify that our *SpatialBot* can understand the depth in the three levels. Furthermore, it is also verified that the fine-tuning of VLMs in *SpatialQA* can improve their performance on general VLM benchmarks such as MME [23], MMBench [54], etc. Finally, robot manipulation abilities demonstrate the promising applications of *SpatialBot*. In summary, the main contributions of our work are as follows:

- We propose *SpatialBot* that shows promising performance in general visual recognition, spatial understanding, and robot manipulation.
- We curate a large-scale RGB-D VQA dataset, *SpatialQA*, for training *SpatialBot*, and *SpatialBench* for evaluating VLMs' spatial understanding performances. Three levels of tasks have been designed for a comprehensive analysis of depth.
- We finetune and deploy *SpatialBot* on embodiment tasks that involve spatial reasoning, and release the robot manipulation dataset focusing on spatial relationships, namely *SpatialQA-E*.

2. Related work

2.1. VLM and RGB datasets

In recent years, VLMs (or Multi-modal Large Language Models, MLLMs) have achieved significant advancements [35]. LLaVA [52] pioneered the visual instruction tuning, which is followed by subsequent works [7, 28, 50, 55, 78] with more extensive datasets [87] and different Large Language Models (LLM) backbones [1, 6, 16, 68]. These VLMs primarily tackle tasks related to perception [23], reasoning [54] and OCR [45, 81]. Additionally, some works have introduced an encoder-decoder structure beyond VLMs to perform pixel-level grounding tasks [42, 74, 77, 80, 84, 85]. However, their performances in counting and enumeration [20, 30] and spatial relationship understanding [33] are mediocre. We posit that comprehending the entire space from a monocular RGB image is overwhelming for VLMs. Integrating depth information could effectively enhance the spatial understanding capabilities of VLMs.

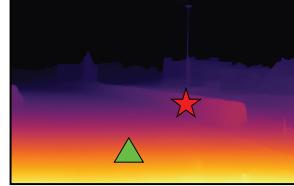
2.2. Spatial Understanding in General QA and Embodiment

Spatial understanding requires VLMs to understand scenes beyond 2D RGB images. This is particularly crucial in precision tasks such as robotic grasping [19]. Spatial understanding can be achieved through point clouds [19, 73] or depth maps [33]. Some studies have attempted to perform depth estimation [47] and 3D detection [17] directly from monocular RGB images, but the accuracy is limited regarding metric depth estimation. SpatialVLM [12] and SpatialRGPT [14] infer spatial relationships only from 2D images. However, in robotic tasks (see, e.g., Fig. 1), depth information from sensors is essential for spatial understanding. Recently, Monocular Depth Estimation (MDE) has seen rapid advancements. Using large amounts of unsupervised data [8, 75] and synthetic data [37], MDE can accurately estimate the depth in various scenarios [60]. Therefore, we improve the spatial understanding of VLMs by adding depth information to the RGB images they use, leveraging MDE. Despite the strength of monocular depth estimation models, training large models to estimate depth directly is not always feasible. In embodied AI scenarios, precise depth information is required from hardware devices, which depth estimation models cannot achieve. Additionally, enabling VLMs to precisely understand space from a single RGB image has proven to be extremely difficult [17, 47]. To extend spatial understanding abilities to embodiment, we propose *spatialQA-E*. To the best of our knowledge, it is the first manipulation dataset that focuses on spatial relationships. *SpatialBot* utilizes a similar model structure with state-of-the-art vision-language-action models like RT [10, 11], Octo [67] and OpenVLA [39], while acquires spatial knowl-

Low-Level

Point Depth: Depth of a Single Point, Directly Retrieved from a Depth Map

- Q: What is the depth value of the point (0.60, 0.58)?
A: The point (0.60, 0.58) corresponds to a depth value of ...
Q: What is the depth value of the point (0.31, 0.82)?
A: The point (0.31, 0.82) corresponds to a depth value of ...



Low-Level

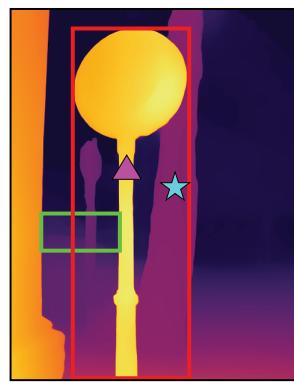
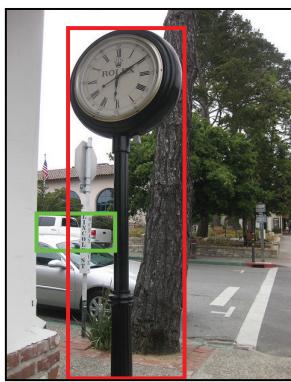
Depth Description and Analysis

- Q: Describe the depth map.
A: The depth map appears to depict an outdoor scene with a ground plane that extends into the distance. There are structures or objects in the middle to upper parts of the image that are further away. A vertical structure, can be seen extending upwards from the middle of the image.

Middle-Level

RGB-Depth Alignment: Depth of Object, Using Bounding Box or Mask

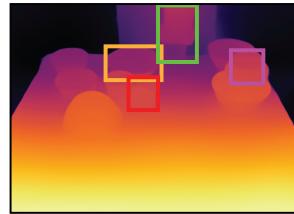
- Q: What is the metric depth of the object: ROLEX clock?
A: The clock translates to a depth value of 4909, with max depth 5051, min depth 4300, and mean depth 4512.
Q: How far is the pickup truck from the camera?
A: The pickup truck is 17802 mm away from the camera.



High-Level

Counting and Enumeration

- Q: How many orange bowls are there on the table?
A: There are 3 orange bowls.
Q: How many bowls that are upside-down are there on the table?
A: One.



Middle-Level

Proximity Relationships

- Q: Which is closer to the camera, the ROLEX clock, or the pickup truck?
A: The clock has a depth value of 4909, while the pickup truck corresponds to a metric depth of 17802, so the ROLEX clock is closer to the camera.
Q: Which appears to be more distant, the point (0.62, 0.72) or the point (0.39, 0.74)?
A: The point (0.62, 0.72) corresponds to a depth of 9312, and the point (0.39, 0.74) ...

High-Level

Spatial Relationship Understanding

- Q: Which bowl is closest to the gripper?
A: The orange bowl.
Q: If the red cup is to the right side of the gripper, what is the spatial relationship between the purple cup and the gripper?
A: The purple cup is to the left side of the gripper, and closer to the camera.

Figure 2. The proposed *SpatialQA* dataset consists of basic, middle, and high-level VQAs in general VLM tasks, aiming to (a) help VLMs understand depth images, (b) let VLMs learn to align RGB and depth images, (c) enable VLMs to do high-level tasks better by understanding both RGB and depth images, as depth images provide clear boundary information and spatial relationships.

edge necessary in manipulation tasks through training on *SpatialQA-E*.

3. SpatialBot

We use depth information to guide VLMs in understanding space [32, 57], because compared to point clouds, depth information is easier to collect and process. Since the RGB-D cameras are cheap, most of robots carry such cameras to capture RGB and depth images instantly. In addition, due to remarkable capacities of Monocular Depth Estimation (MDE), one can adapt large scale RGB datasets to RGB-D dataset in a fast and affordable way. Thus, we introduce depth images for spatial understanding and construct *SpatialQA* dataset with RGB-D images and depth-related QA pairs. In this section, we elaborate on the pipeline of dataset construction: collecting RGB and depth images, estimating depth images from RGB images using MDE, unifying the format of depth images, generating basic VQAs for VLM training, and generating depth related VQAs. This pipeline can be easily scaled up to construct larger datasets from available RGB datasets.

3.1. Depth Map Encoding

Our depth encoding aims to preserve all depth information for VLMs to use. A challenge is the indoor and outdoor consistency. Indoor scenes like robot manipulation [56] and indoor navigation [9, 27] may require millimeter-level precision, while outdoor scenes include a large range of depth values. Existing methods often adopt ordinal encoding [24, 75], which, however, cannot be subjected to basic mathematical operations. To address the issue, we use uint24 or three-channel uint8 to store depth values, measured in millimeters from 1mm to 131.071m. We directly save the raw depth values and leave subsequent computations to the powerful fitting capabilities of VLMs. For single-channel uint24, we use millimeter as unit directly. This way, VLMs can directly query the required values from the depth map. For three-channel uint8 images, we distribute the values across a broader range: the units for the three channels are 2^0 , 2^5 , and 2^{10} millimeters, respectively. Each channel has 2^5 , 2^5 , and 2^7 possible values. For an image of size (H, W) , to store depth value $d_{H,W}$ (in millimeters) in three-channel uint8 image $I_{H,W}^3$, we encode the

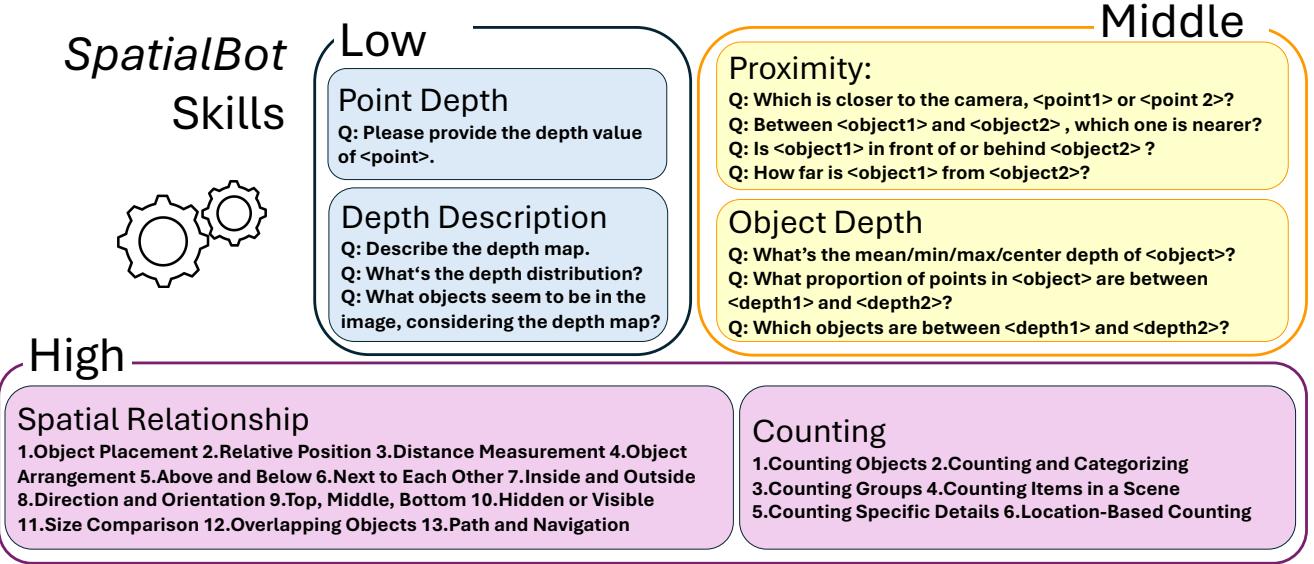


Figure 3. *SpatialBot* masters three levels of skills: (a) understanding depth images (b) aligning these RGB and depth inputs, and performing proximity comparison, (c) applying RGB-D for spatial relationship understanding and counting.

image I following:

$$I_{h,w}^0 = (d_{h,w} // 2^{10}) * 2^1, \quad (1)$$

$$I_{h,w}^1 = (d_{h,w} // 2^5) * 2^3, \quad (2)$$

$$I_{h,w}^2 = (d_{h,w} \% 2^5) * 2^3. \quad (3)$$

an image is d_{max} and d_{min} , the conversion should follow:

$$A = \frac{1}{d_{min}} - \frac{1}{d_{max}}, \quad (4)$$

$$B = \frac{1}{d_{max}}, \quad (5)$$

$$d = \frac{1}{(A * d_r + B)}. \quad (6)$$

It is incorrect to ask for the depth value but uses $\frac{1}{d_r}$ as the label in QAs [47]. While $\frac{1}{d_r}$ can reflect the relative size of depth, it does not maintain proportional relationships (e.g. $\frac{1}{d_r} = 0.4$ is not twice the depth of 0.2). Only when the max depth is infinite can $\frac{1}{d_r}$ be considered as true depth multiplies by a scale factor.

3.2. Depth Description of an Object

SpatialQA is a VQA dataset, and our model is a standard VLM (Fig. 4): it takes images and text as input and outputs text. To maintain generality, we do not use a separate image encoder, so *SpatialBot* cannot output pixel-level information. Intuitively, the center point of objects can simply represent their depth. However, for example, in the case of a cup, there is a significant difference between the depth of the inner and outer surfaces, so a single value cannot accurately represent the depth. Therefore, we use four depth values—max, min, mean, and center—to describe the object’s depth, if its mask is available. Considering that the mask and depth map cannot be perfectly precise, we use the 95th and 5th percentile values as the max and min depth

values. Bounding boxes in Visual Genome (VG) [41] are very inaccurate, and our experiments find that prompting SAM [40] with these bounding boxes will yield undesirable masks. In this case, to prevent incorrect depth from misleading the model, we fall back to using only the depth of the center point of the bounding box to describe depth.

3.3. Image Sources

A RGB-D VLM dataset should include detailed QAs that help VLMs to understand the image, which may include reasoning, conversation, description and referring [87]. Specific object descriptions are required, e.g. in Fig. 4, woman is not a good description, but the woman in the middle or the woman standing tallest are good descriptions. Existing captioning, grounding and segmentation models [15, 53, 61, 88] can not generate detailed and specific descriptions. To this end, we base *SpatialQA* mainly on VLM data where detailed QAs are included [28].

In *SpatialQA*, we primarily include three data sources: COCO [51], VG [41], and Open X-Embodiment (RTX) [56]. Therefore, we base our dataset on Bunny_695k [28], which includes COCO and VG. Bunny_695k contains image QA covering reasoning, detailed descriptions, grounding, etc. On this basis, we added depth-related QA pairs. We use bounding boxes in Bunny_695k and prompt SAM [40] with bounding boxes and center points to get masks. We ensure that SAM masks do not exceed the bounding box limits, then select the mask with the highest confidence. RT-X integrates many robotics datasets. For datasets containing sensor depth data, we directly use the raw depth. For other datasets, we use model-estimated metric depth. We select 7.5k of these and manually annotated the bounding boxes, querying the depth information of the objects. For the remaining images, we only ask about the depth of certain pixels in the image. Also, we use GPT-4o to generate conversations based on RTX-7.5k, where we prompt GPT-4o to focus on: what robot are doing, how should the robot finish robot task, object count, object position, positional relationships and object appearance. In future versions of *SpatialQA*, we will include more images from a vast range of sources.

3.4. *SpatialQA* Pipeline

To help VLMs understand depth inputs, and use depth information to do high-level tasks like spatial relationship understanding, counting and enumeration, we design a three-step QA pipeline. We aim to make this pipeline effective and easy-to-follow: (a) This pipeline progressively let VLM learn to understand depth, align depth and RGB, and use depth for complex reasoning in high-level tasks. (b) existing RGB datasets can be easily converted to RGB-Depth datasets with our pipeline. *SpatialQA* pipeline is shown in Fig. 5, and the skills to learn are shown in Fig. 3.

Low level. To enable VLMs to understand depth images and learn to query information from them, we ask depth value of points. VLMs should learn to take the depth value directly from depth inputs, and relate point coordinates with pixels in image. In the meanwhile, since the visual encoder does not see depth images in pre-training, we also expect the encoder and projector to learn to encode depth images together with RGB images. We also let *SpatialBot* describe the depth map and infer what may be in the images, giving only a depth map.

Middle level. As VLMs have learnt to encode and query information from depth images, they should now learn to use depth information. Also, since image and depth inputs are given to VLMs, they should also know the relationships between them. First, we ask about proximity relationships, namely which point is closer or further away. Second, we let VLMs learn to describe the depth of objects or regions, by using center point depth, minimum, maximum and mean depth. VLMs should learn to locate an object in the RGB image and then find depth information from depth input. Third, we ask about proximity relationships between objects.

High level. Since VLMs can now understand depth input, align depth with RGB and have some knowledge about proximity relationships in the spatial world, we design tasks to help VLMs apply depth at a higher level. When the model sees the depth map, the boundaries of objects and their surroundings become clearer, so we believe that the depth map aids in grounding and counting tasks. Additionally, in *SpatialQA*, the model gains a clear understanding of the space, which helps the model determine spatial and positional relationships.

3.5. *SpatialQA-E*

We propose *SpatialQA-E* to extend spatial understanding and reasoning abilities to embodiment tasks. We use the 7-axis Franka Research 3 Robotic Arm to grasp objects on the table, avoid obstacles while moving, and place them on a cutting board on the table. We include spatial relationships in language instructions, so the model should learn spatial reasoning in manipulation. *SpatialQA-E* contains 2000 episodes in total. The dataset is composed of 4 steps, shown in Fig. 6 and Fig. 7:

- Learn to pick and place teacups, balls, bananas, etc.
- Find specific object and destination. The dataset includes spatial relationships in positive, comparative (-er), and superlative (-est) degrees from the perspective of the robot or the human (camera):
 - Positional: left/ right/ middle/ up/ down on/ in/ inside/ outside
 - Size: tall/ short/ large/ small/ wide/ thin/ big/ small
 - Illusion: we take photos of objects, print them

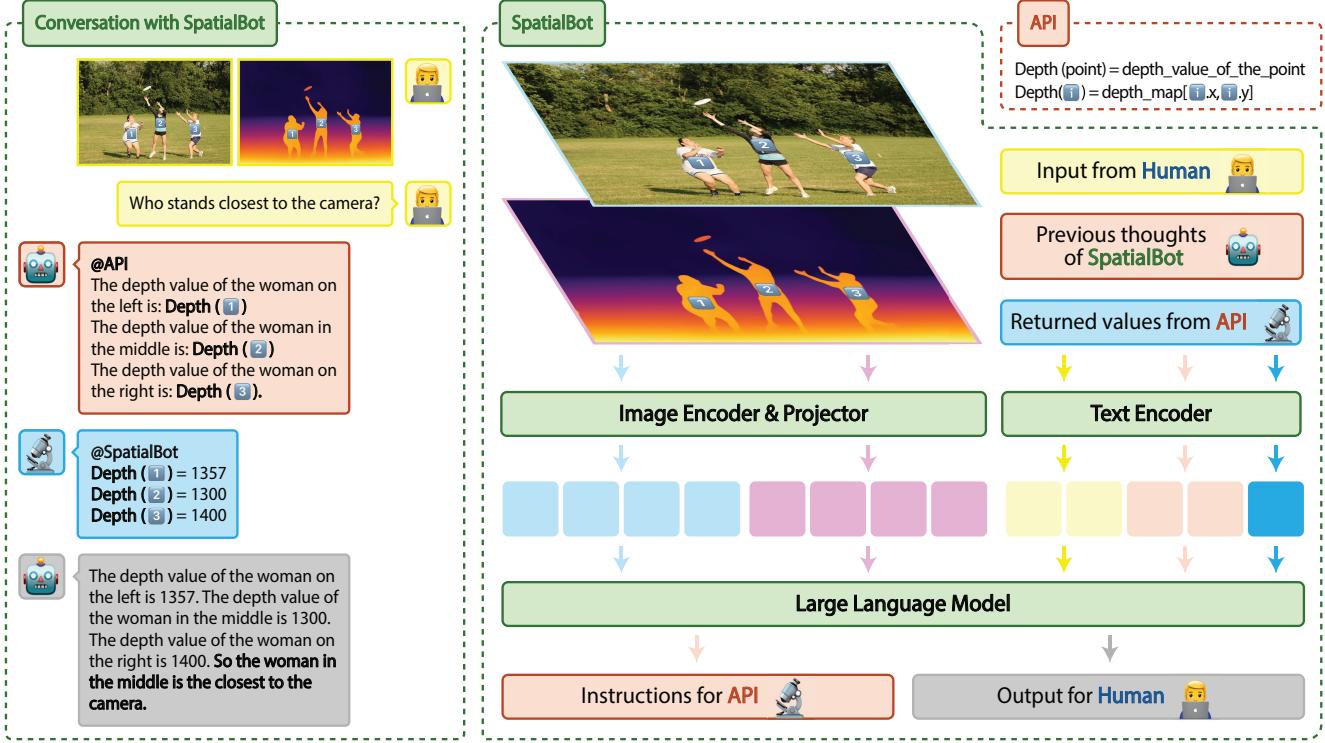


Figure 4. The architecture of *SpatialBot*. It takes a pair of RGB and depth images as input, where depth images are optional. *SpatialBot* can choose to call Depth API if it need accurate depth information.

out, and put the printed object on the table. It looks real, and the model needs to tell between printed and real objects through visual clues, e.g., depth information (printed objects are flat) and shadows.

- In moving objects, the robot needs to avoid obstacles.

3.6. SpatialBot in Embodiment Tasks

SpatialBot is finetuned on *SpatialQA-E* to work on embodiment tasks. In short, it is a Vision-Language-Action (VLA) model that supports multi-frame RGB or RGB-D inputs. Robot manipulation tasks are specified as: in current time stamp t , given history and current image observations $x_{j=0}^t$ (RGB or RGB-D), models should learn policy $\pi(|i, x_{j=0}^t|)$. Action a_t is sampled from π and applied to robots. For a robot of two-finger end effector, action space can be represented as 7 DoF vector: $(\Delta X, \Delta Y, \Delta Z, \Delta R, \Delta P, \Delta Yaw, C)$, indicating delta change in poses XYZ and rotation RPYa (roll, pitch, yaw), gripper closure C. The delta change in position and rotation of action space is encoded into 101 possible values, from 0, 0.01 to 1. The model output texts of 7 DoF actions directly. A sample conversation: 'User: What should the robot do to pick up the biggest teacup and move it to the left

cutting board? Answer with robot parameters. - *SpatialBot*: The robot should $<0.17, 0.51, 0.44, 0.62, 0.83, 0.07, 1>$ '. Then we decode the output to robot control signals to control the robot movement of each frame. If the model directly answers robotic parameters during the finetuning stage, we find that it can only respond to robot-specific questions. To enable multi-task training, we incorporate some natural language elements into the robot's responses, such as 'The robot should'. Then, we train the model on robotic data and general QA data, such as *SpatialQA-E* and *SpatialQA*. We have the model predict special tokens during robotic tasks to maintain the model's numerical reasoning abilities in general conversations. We predict each frame's delta pose instead of the target pose. This choice allows for more precise control of the robot by dividing each dimension of the action space into 100 bins. Additionally, we found that delta pose is harder to learn than target pose, as reflected in the slower decrease in loss. For some data, the delta pose loss doesn't decrease at all, and the model completely collapse and output the same value regardless of the input. A quick solution to this issue is to exponentially increase the amount of training data.

Depth
Sensor
Data



RGB
Camera
Data

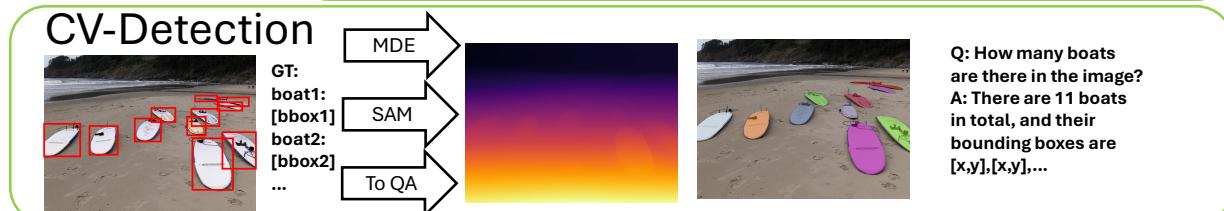
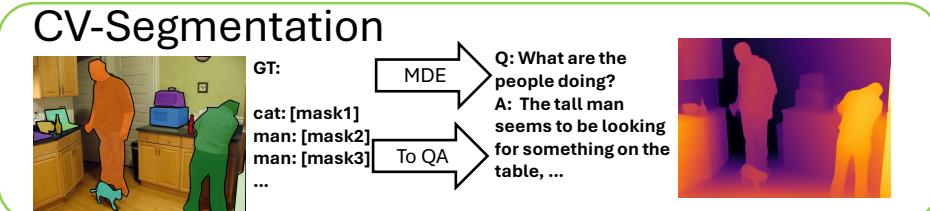
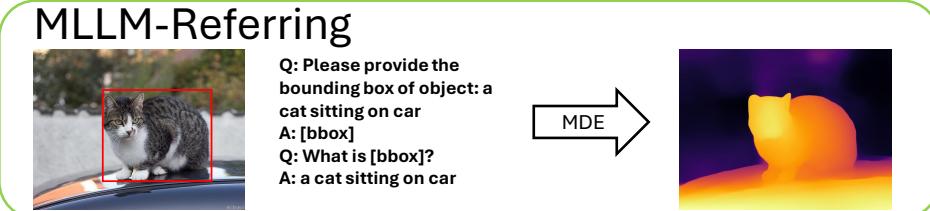
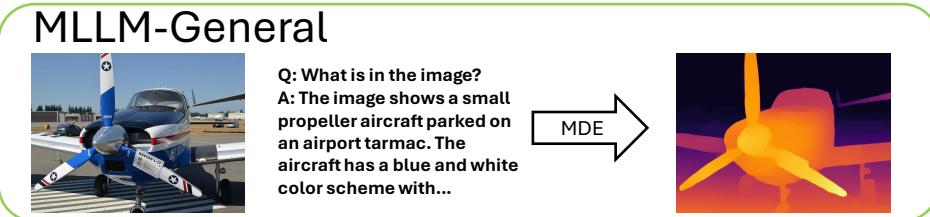
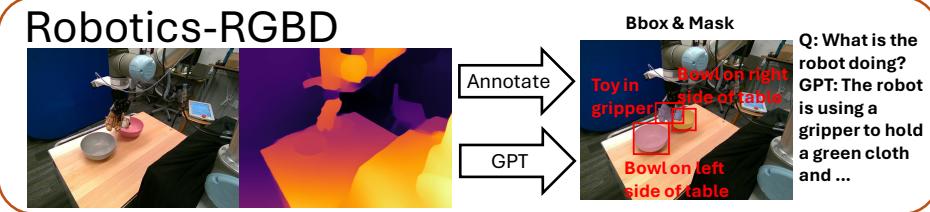
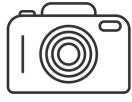


Figure 5. Image sources and RGB to RGB-Depth dataset conversion pipeline. RGB and depth information, captions or conversations about the images, bounding boxes or masks, and corresponding detailed descriptions of objects are required to make *SpatialQA* dataset.

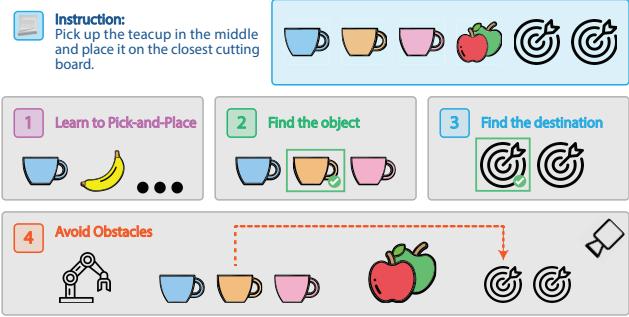


Figure 6. *SpatialQA-E* involves spatial relationships in robot manipulation.

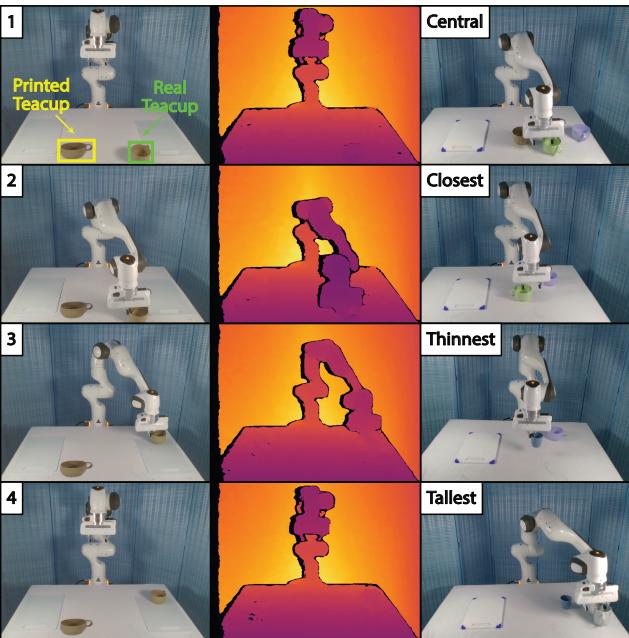


Figure 7. *SpatialQA-E* demonstration. Left: 4 steps in picking up the real teacup and putting it on the right cutting board relative to the camera. We print the teacup as a distraction. It's easier to tell between the real and printed teacup from the depth map. Right: 4 sample settings in *SpatialQA-E*, where we specify spatial relationships.

3.7. *SpatialBench*

To evaluate VLM’s performance on high-level tasks, we annotate the *SpatialBench*. On 120 images, we ask following questions:

- Has [Object 1] touched/reached [Object 2]?
- What is the spatial relationship between [Object]s?
- Counting and enumeration.
- Size comparison between objects.

All question are in Yes/No or multiple choice formats. Ad-

ditionally, *SpatialBench* includes depth and proximity questions on images from MME [23] dataset, and our manually annotated 80 images (3 bounding box per image).

3.8. *SpatialBot* Depth API Architecture

SpatialBot uses a VLM structure: Images are processed through an image encoder and a multi-modal projector, converted into tokens, and then sent along with text tokens into an LLM, which ultimately outputs responses. To enable the model to accurately obtain depth information, we designed **Depth API**. When the *SpatialBot*’s output contains text with a format of Depth(point), the API will query the depth value of that point in the corresponding depth map and then input this depth value back into *SpatialBot*. Combining the user’s question with the API’s return value, *SpatialBot* will provide the final answer. The model can call the API to get the precise depth value of a specific point. For example, when *SpatialBot* wants to know the depth information of an object, it first determines the bounding box of the object and then calls the Depth API using the center point of the bounding box. If the model wants to obtain the depth range of this object, it first observes which points in the image correspond to the maximum and minimum depth values and calls the Depth API using the coordinates of these points. However, to enhance the model’s understanding of the depth map itself, during training, we only allow *SpatialBot* to call the API on a subset of the data. For the remaining data, the model must directly use the depth map to answer the depth of the object.

4. Experiments

We start with validating *SpatialBot* has the ability to understand depth, extract information from depth maps, and perform high-level tasks. Then, we observe performance improvements in general VQA tasks, such as MME [23] and GQA [31], by introducing depth maps. This indicates that training on *SpatialQA* can help VLMs perform better on general tasks. Finally, experiments on RT-X [56] show that *SpatialBot* benefits from understanding depth in robot manipulation tasks.

Implementation. We design *SpatialBot* based on Bunny [28], a family of VLMs. Phi-2-3B, Phi-3-4B [1], QWen-1.5-4B [6] and Llama-3-8B [68] are used as the base LLM. *SpatialBot* model architecture is shown in Fig. 4. The image encoder is SigLIP [83] with 384x384 image resolution. QWen-1.5-0.5B and CLIP [59] with 336x336 image resolution are adopted in robot manipulation tasks. We pre-train models on two million image-text pairs from LAION-2B [62] (Bunny-pretrain-LAION-2M [28]) and finetune them on Bunny_695k [28]. The learning rate is kept $2e - 4$, and learning rate for multi-modal projector is $2e - 5$, except for Llama-3-8B, where we halve both learning rates. For manipulation tasks, we also halve the learning rate. The

Table 1. Results on *SpatialBench*. The best results of models with the same base LLMs are marked with **bold** text. LLM-RGB and LLM-RGBD are trained on RGB images only and tested with RGB and RGBD inputs, respectively. *SpatialBot* with RGB input in depth estimation is the same as the MDE task.

Model		Depth ↑	Position ↑	Existence ↑	Counting ↑	Reaching ↑	Size ↑
GPT-4o-RGB	-	70.6	85.0	84.5	51.7	43.3	
GPT-4o-RGBD	-	61.8	90.0	85.2	51.7	40.0	
Bunny-Phi2-3B-RGB	70.6	50.0	75.0	89.4	51.7	26.7	
<i>SpatialBot</i> -Phi2-3B-RGB	84.1	64.7	80.0	88.0	61.7	28.3	
Bunny-Phi2-3B-RGBD	85.8	50.0	75.0	90.4	43.3	28.3	
<i>SpatialBot</i> -Phi2-3B-RGBD	>99	61.8	80.0	91.7	55.0	26.7	
Bunny-Phi3-4B-RGB	32.3	58.8	75.0	91.0	31.7	16.7	
<i>SpatialBot</i> -Phi3-4B-RGB	83.2	64.7	75.0	91.0	40	23.3	
Bunny-Phi3-4B-RGBD	63.3	52.9	60.0	85.4	31.7	18.3	
<i>SpatialBot</i> -Phi3-4B-RGBD	>99	67.7	70.0	91.7	35.0	21.7	
Bunny-QWen-1.5-4B-RGB	42.2	50.0	75.0	91.6	26.7	15.0	
<i>SpatialBot</i> -QWen1.5-4B-RGB	89.9	52.9	75.0	88.6	46.8	18.3	
Bunny-QWen-1.5-4B-RGBD	74.6	44.1	70.0	90.7	25.0	15.0	
<i>SpatialBot</i> -QWen1.5-4B-RGBD	>99	52.9	60.0	90.5	41.7	26.7	
Bunny-Llama3-8B-RGB	58.1	50.0	75.0	91.7	38.3	23.3	
<i>SpatialBot</i> -Llama3-8B-RGB	85.6	55.9	80.0	91.2	40.0	20.0	
Bunny-Llama3-8B-RGBD	64.0	50.0	75.0	90.4	38.3	25.0	
<i>SpatialBot</i> -Llama3-8B-RGBD	>99	53.0	75.0	90.4	45.0	20.0	

Table 2. Results on general VLM Benchmarks. For the same base LLM models, better results are marked with **bold** text. RGB-D inputs are only used in MME. We report the results of Bunny trained with RGB and tested with RGB/RGB-D in it, split with slash. *SpatialBot* is trained on RGBD and tested on RGB/RGB-D on MME.

Model	MME ^P ↑	MME ^C ↑	MMB ^T ↑	MMB ^D ↑	SEED-I ↑	VQA ^{v2} ↑	GQA ↑	POPE ↑
Bunny-Phi2-3B	1472/1474	286/285	67.90	68.90	69.91	78.98	61.52	86.21
<i>SpatialBot</i> -Phi2-3B	1483/ 1487	310/312	70.12	68.56	70.85	79.80	62.28	87.04
Bunny-Phi3-4B	1417/1364	308/319	70.15	70.74	71.04	80.57	61.18	84.60
<i>SpatialBot</i> -Phi3-4B	1431 /1433	337 /329	73.49	73.11	71.64	80.01	62.16	85.47
Bunny-QWen1.5-4B	1340/1364	251/254	69.56	68.56	70.05	80.63	61.55	85.10
<i>SpatialBot</i> -QWen1.5-4B	1378/ 1406	266/285	70.91	69.67	70.36	79.69	62.77	86.09
Bunny-Llama3-8B	1574/1542	342/318	73.67	74.15	72.32	80.50	62.18	85.22
<i>SpatialBot</i> -LLama3-8B	1577 /1576	352 /333	75.78	74.83	72.40	80.94	62.90	85.33

multi-modal projector is trainable in both pretrain and finetune stage, and we add a LoRA [29] module in finetuning. We use 8 A100 for training. On *SpatialQA*, it takes about 15 hours for Phi-2 [1].

4.1. Spatial Understanding

We first validate that *SpatialBot* can get accurate metric depth value from depth images or Depth API, and decide proximity relationships, which are low-level and middle level tasks in *SpatialQA*. We use bounding box and metric depth from *SpatialBench*. We then ask about depth of random points and objects in them. We tell VLMs the names w/

and w/o bounding boxes of target object. For ground truth depth value d_{gt} , estimated depth value d_{est} from VLMs, we estimate depth accuracy by $\frac{d_{gt} - d_{est}}{d_{gt}} * 100\%$. Results by answering with Depth API are shown in Depth and Proximity column in Table. 1. Also, we ask the proximity relationships. Sample conversations on depth map understanding and high-level tasks are shown in Fig. 12.

4.2. *SpatialBench*

We compare model performance on our *SpatialBench*, which composes on positional relationship, object existence, reaching and size comparison tasks. GPT-4o is com-

pared with models trained on *SpatialQA*. 3B, 4B and 8B models trained on *SpatialQA* reaches comparable results with GPT-4o. Results are reported in Table 1.

4.3. General VLM Benchmarks

We report results on general benchmarks: MME perception [23] (MME^P), MME cognition (MME^C), MM-Bench [54] test and dev set (MMB^T and MMB^D), SEED Bench Image [45] (SEED-(I)), VQA [26] test-dev split (VQA^{v2}), GQA [31], and POPE [49] (the averaged F1-score of three categories on the validation set of COCO). In most of these benchmarks, RGB information along is enough. We only use RGB-Depth input on MME^P and GQA since they contain counting, existence and position questions, where we expect depth information can benefit such cases.

4.4. SpatialBot in Embodiment Tasks

We finetune *SpatialBot* on *SpatialQA-E* to do manipulation tasks on real robots. It can be seen as a VLA model supporting multi-frame RGB or RGBD inputs. We use QWen-1.5-0.5B [6] as the base LLM and CLIP [59] as the vision encoder. The pretrain dataset is Bunny-pretrain-LAION-2M [28], and *SpatialQA-E* is used in finetuning. Four frames in history are used to predict the end-effector delta position of the current frame. The model runs locally or connects through an ssh/sftp connection to run on RTX 4090 GPU. It is validated through experiments that *SpatialBot* can do manipulation tasks with spatial instructions. Fig. 8 shows the success rate of *SpatialBot* RGB and RGBD variants. With depth information, *SpatialBot* can pick and place more accurately.

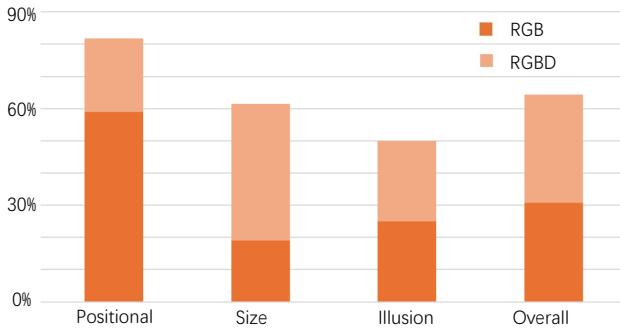


Figure 8. *SpatialBot* success rate in pick-and-place of RGB and RGBD variants.

5. Monocular Depth Estimation by VLM

We let VLMs to understand depth input so as to get accurate depth information. Readers may wonder that, is MDE for a single point or object difficult? To verify this, we ask

SpatialBot to predict the depth of points or objects. The results are reported in Table. 1 of main paper (Depth scores of *SpatialBot-RGB*). It seems that VLMs has not been fully prepared for MDE in a text-only output fashion. As reported in [3, 4, 82, 86], an extra decoder may be needed to generate accurate and pixel-level depth. We are not doing so since this new structure limits the generality of *SpatialQA*.

6. SpatialBench Metadata

In SpatialBench, GPT-4o generate 34 multiple choice questions regarding positional relationships, and we use the correct selection ratio as accuracy. GPT-4o generated 20 positive and negative question pairs on another 20 image. Only when the model answers both the positive and negative questions of a problem correctly is it considered correct. On 20 image, human experts choose a category of object and annotate its quantity. We ask the model to count the objects. Human experts think of positive and negative multiple choice questions regarding whether object A has reached or touched object B, and size comparisons on 20 images respectively. We first calculate the rate of correct choices from models. When it answers a pair of positive and negative questions correctly, we give it a bonus score.

7. Data Generation in SpatialQA

We prompt GPT on about 50k images for depthmap understanding, spatial understanding and robot scene understanding in *SpatialQA*. The data source and data selection protocol are shown in Table. 3. GPT prompts we use for three separate tasks are shown in Table. 4. Sample images and generated QAs are shown in Fig. 9, 10, 11. Additionally, human experts annotated 3 bounding boxes per image in RTX, and are asked to annotate gripper if it is clearly visible in the image.

8. Dataset, Model and Benchmark Usage

The official repository of *SpatialBot* is <https://github.com/BAAI-DCAI/SpatialBot?tab=readme-ov-file>, where we provide metadata, codes, scripts, checkpoints, licenses and links to resources.

SpatialQA can be downloaded on Hugging Face: <https://huggingface.co/datasets/RussRobin/SpatialQA>.

SpatialQA-E can be available on Hugging Face: <https://huggingface.co/datasets/RussRobin/SpatialQA-E>.

SpatialBench can be accessed through <https://huggingface.co/datasets/RussRobin/SpatialBench>.

Checkpoint for *SpatialBot-3B*, which is based on Phi-2 [1] and SigLip [83], can be downloaded at <https://huggingface.co/RussRobin/SpatialBot-3B>.

The pretrained checkpoints of SpatialBot can be accessed through Bunny [28] model zoo: <https://github.com/BAAI-DCAI/Bunny>.

9. Conclusion

We propose *SpatialBot*, a family of state-of-the-art VLMs, for effective depth understanding and thus precise robot manipulating in embodied AI by training on our constructed *SpatialQA* and *SpatialQA-E* datasets. *SpatialBot* can understand depth inputs and use depth information to do spatial understanding and reasoning tasks in Visual QA and Embodiment. *SpatialBench* is also designed to evaluate the model performance of spatial knowledge in multiple aspects. Experimental results on our benchmark, general VLM benchmarks, and robot manipulation deployment verify the effectiveness and superiority of *SpatialBot* comparing to competitors.

Acknowledgement

This work was supported by National Natural Science Foundation of China under No. 62306046 and the National Youth Talent Support Program under No. 8200800081.

Table 3. RGB image and depth sources in SpatialQA. Depthmap is either from sensors, as included in the original datasets, or MDE depth by ZoeDepth [8].

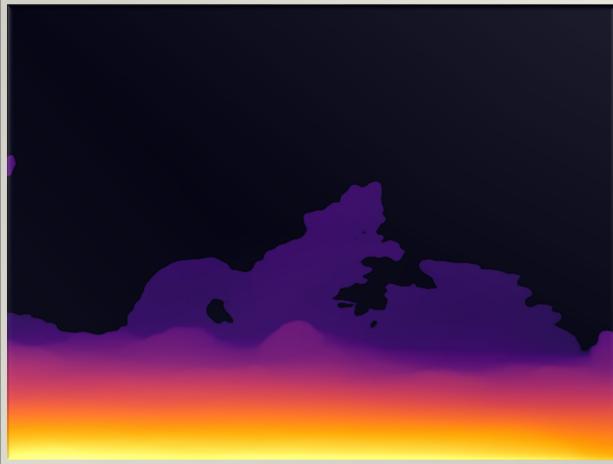
Data	Aim	Img. Num.	Image Selection Protocol and Depthmap Source
Bunny695k [28]	General MLLM abilities	695k	-
VG, COCO	depthmap understanding	20k	Random selection. GPT is prompted to first infer from the depth color map and then verify its inference against the RGB data to ensure its correctness. The inference of GPT is stored in QA format. MDE.
KITTI [69]	spatial understanding	1.75k	Randomly select images from each sequence of the KITTI dataset. Since some scenes are captured when the car is stationary or temporarily stopped (e.g., at a traffic light or due to a stopped vehicle ahead), there are many repetitive images. We then Manually filter out those with a particularly high repetition rate. Since depth information in KITTI only include sparse points, and ZoeDepth has been finetuned on KITTI, we use MDE.
NYU Depth v2 [65]	spatial understanding	1.5k	All images and sensor depth images are adopted.
RT-X [56]	robot scene understanding	7.5k	We annotate 3 bounding boxes per image, and annotate the gripper if it is visible. We also prompt GPT to generate general QAs. We use depthmap from subsets if available. Otherwise, we use MDE to estimate depth.
SA-1B [40]	spatial understanding	15k	We randomly select real-world images from SA-1B and prompt GPT-4o to generate conversations regarding spatial relationships. MDE.
2D-3D-S [2]	spatial understanding	2.9k	We randomly select images from 2d3ds and manually exclude images with no more than 3 objects in them. Sensor depth.

Table 4. GPT prompts used in SpatialQA.

Aim	Prompts for GPT
depthmap understanding	Design a conversion between you and a human talking about the depth map. The human asks you to describe the depth map. You should focus on depth value predictions. The colors just represent depth values. Do not directly mention colors on the image in your response, instead, mention the depth distribution they stand for. Looking at the depth map, you should also infer what may be in the image. If something really exists in the rgb image, and can be inferred from the depth map, you can mention they in your response. If possible, pay attention to spatial relationships. When referring to spatial relationships, such as left and right, you should use the real-world left and right, rather than those in the image coordinate system.
spatial understanding	Design a conversation, consisting of no more than 3 Question-Answer pairs, between you and a person asking about this image. The content within the conversation should be logically connected. You should think of what are spatial relationships of objects in the image. Then generate the conversation according regarding the spatial relationships. Spatial relationships can be about, but not limited to these categories: positional (left/right, below/above, behind/front), distance (further/closer to the camera, further/closer to something), size(big/small, tall/short, wide/thin), reach (has A touched/reached B physically). When describing spatial relationships, always use the real-world orientation as if you are standing in the real scene. e.g. when using right side of object, it should talk about what is on the right side of object in the real world, not on the right side of image'. Only describe the things that you are sure about.
robot scene understanding	Design a conversation, consisting of no more than 3 Question-Answer pairs, between you and a person asking about this image. The content within the conversation should be logically connected. You should first think of robot task: what may the robot want to do with the objects. And then generate the conversation according to robot task. The conversations can include what robot are doing, how should the robot finish robot task, object count, object position, positional relationships, object appearance, etc. Only describe the things that you are sure about. Please note that you are talking to a person about the image and robot. You are not the robot, and you are not talking to the robot.

Q: Can you describe the depth map for me?

A: The depth map shows varying depth values across the scene. The bottom portion of the image represents the foreground with the closest depth values, indicating the presence of grass or a field. Towards the middle of the image, there are structures with intermediate depth values suggesting the presence of multiple trees. The upper section depicts the background with the farthest depth values, likely representing the sky. The trees are positioned such that the one to the left is closer than the rightmost one in the scene.

**Q: Can you describe the depth map for me?**

A: The depth map indicates variations in distance within the scene. The foreground, from the ground plane closest to us, shows the shallowest depth, hinting at the ground level area immediately in front. Further back, shapes that likely represent standing people and possibly objects like posts or bollards can be inferred, with mid-range depth values. Farther back still, we see deeper values, representing a building facade and a vertically elongated structure, probably a lamp post or pole situated in front and towards the middle. It can be inferred that this scene is of a streetscape with people, a sidewalk, and storefronts.

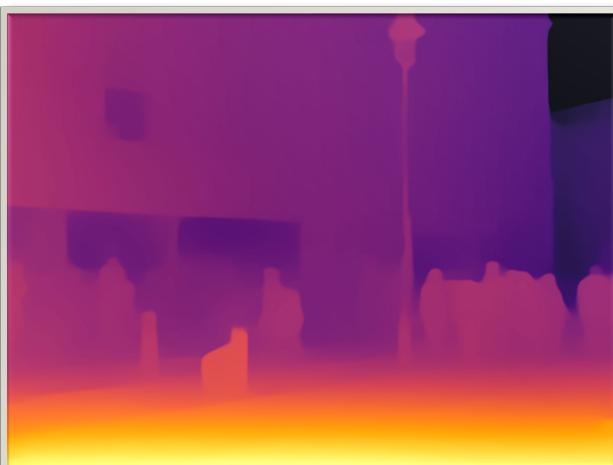


Figure 9. Sample data of depthmap understanding in *SpatialQA*.

Q1: What can you see on the left side of the canal?

A1: On the left side of the canal, there are several wooden boats docked, a large tree, and some buildings, including garages with closed doors.

Q2: What is the woman in the pink coat walking towards?

A2: The woman in the pink coat is walking along the path that runs parallel to the canal, heading towards the background where more buildings and structures are visible.

Q3: Can you describe the scene on the right side of the canal further up?

A3: Further up on the right side of the canal, there are more boats docked and some trees and buildings in the background. There is also another person walking along the path.



Q1: Which penguin seems to be the tallest?

A1: The penguin sitting on top of the igloo seems to be the tallest.

Q2: Which penguin is the smallest in size?

A2: The penguin on the left, with the red hat, seems to be the smallest in size.

Q3: In the real world, which character is bigger, the igloo or the penguin on top of it?

A3: In the real world, the igloo is bigger than the penguin on top of it.

Q1: Has the yellow car reached the motorcycles?

A1: No, the yellow car has not reached the motorcycles. There is a small gap between them.

Q2: Has anyone reached the area with the red stalls?

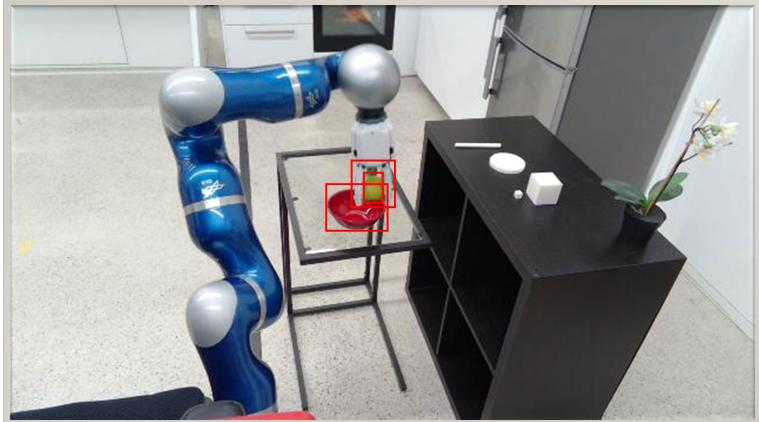
A2: Yes, there are several people around and inside the area with the red stalls.

Q3: Has the white structure near the motorcycles touched any buildings?

A3: No, the white structure near the motorcycles has not touched any buildings. It is standing independently.



Figure 10. Sample data of spatial understanding in *SpatialQA*.

**Q1: What is the robot doing in this image?**

A1: The robot appears to be reaching towards two cans placed on a table.

Q2: How many objects are on the table?

A2: There are two cans on the table.

Q3: What is the robot likely trying to do with the cans?

A3: The robot is likely attempting to pick up or move one or both of the cans.

Bounding boxes

Gripper: [0.3, 0.117, 0.42, 0.235],
Can_1: [0.325, 0.147, 0.403, 0.289],
Can_2: [0.232, 0.131, 0.319, 0.3]

Q1: What is the robot doing in this image?

A1: The robot appears to be holding a green object with its gripper arm, placing it on or picking it up from a table.

Q2: What objects are present on the black shelf beside the robot?

A2: There are three objects on the black shelf: a white cylindrical object, a white cube, and a potted plant.

Q3: How should the robot approach picking up the white cube?

A3: The robot should position its gripper arm directly above the white cube, then lower the arm carefully to grip the cube securely before lifting it.

Bounding boxes

Gripper: [0.462, 0.389, 0.514, 0.487],
Bowl: [0.425, 0.443, 0.505, 0.54],
Apple: [0.471, 0.412, 0.499, 0.475]

Figure 11. Sample data of robot scenes in *SpatialQA*.



1. What is the depth value of the woman?

SpatialBot-Phi2-3B: 1241 (GT: 1240)

2. What is the depth value of the cat?

SpatialBot-Phi2-3B : 990 (GT: 987)

3. Which is closer to the camera, cat or woman?

SpatialBot-Phi2-3B : cat corresponds to a depth level of 990, and woman corresponds to a depth level of 1241. Since $990 < 1241$, it can be inferred that: cat is closer to the camera, so the answer is: cat. ✓

Bunny-Phi2-3B : The cat. ✓

Bunny-QWen1.5-4B: They are equally close. ✗

LLaVA-34B: The woman. ✗

GPT-4o: The woman. ✗

4. Where is the cat? Answer with the option's letter from the given choices directly.

- (A) The cat is in the woman's bag.
- (B) The cat is in the woman's pocket.
- (C) The cat is out of the woman's bag.
- (D) The cat is not with the woman.

SpatialBot-Phi2-3B: (A) ✓

Bunny-Phi2-3B : (C) ✗

Bunny-QWen1.5-4B : (C) ✗

LLaVA-34B: (C) ✗

GPT-4o: (A) ✓



1. What is the depth value of the sign?

SpatialBot-Phi2-3B: 4137 (GT: 4137)

2. What is the depth value of the white car?

SpatialBot-Phi2-3B : 28482 (GT: 28479)

3. Which is closer to the camera, sign or white car?

SpatialBot-Phi2-3B : The sign corresponds to a depth measurement of 4137, and the white car corresponds to a depth measurement of 28482. Since $4137 < 28482$, it can be inferred that: the sign is closer to the camera. ✓

Bunny-Phi2-3B : The sign. ✓

Bunny-QWen1.5-4B : The sign. ✓

LLaVA-34B: The sign. ✓

GPT-4o: The sign. ✓

4. What is the arrow of sign pointing to? Answer with the option's letter from the given choices directly.

- (A) The car.
- (B) The grass.
- (C) The sky.
- (D) The building.

SpatialBot-Phi2-3B: (B) ✓

Bunny-Phi2-3B : (A) ✗

Bunny-QWen1.5-4B : (D) ✗

LLaVA-34B: (C) ✗

GPT-4o: (A) ✗

Figure 12. Sample conversations of *SpatialBot* and baseline models. It first asks about depth value of objects, then lets models compare depth between objects. Finally, spatial relationship questions are asked.

References

- [1] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024. 2, 8, 9, 10
- [2] Iro Armeni, Sasha Sax, Amir Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *ArXiv*, abs/1702.01105, 2017. 12
- [3] Dylan Auty and Krystian Mikolajczyk. Learning to prompt clip for monocular depth estimation: Exploring the limits of human language. *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 2031–2049, 2023. 10
- [4] Dylan Auty and Krystian Mikolajczyk. Language-based depth hints for monocular depth estimation. *ArXiv*, abs/2403.15551, 2024. 10
- [5] Daich Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19107–19117, 2021. 1
- [6] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 2, 8, 10
- [7] 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. 1, 2
- [8] S. Bhat, Reiner Birk, Diana Wofk, Peter Wonka, and Matthias Muller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *ArXiv*, abs/2302.12288, 2023. 2, 4, 12
- [9] Guillaume Le Blanc, Youcef Mezouar, and Philippe Martinet. Indoor navigation of a wheeled mobile robot along visual routes. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 3354–3359, 2005. 3
- [10] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J. Joshi, Ryan C. Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Anand Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Ho Vuong, F. Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale. *ArXiv*, abs/2212.06817, 2022. 2
- [11] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Krzysztof Choromanski, Tianli Ding, Danny Driess, Kumar Avinava Dubey, Chelsea Finn, Peter R. Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil J. Joshi, Ryan C. Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Sergey Levine, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Pierre Sermanet, Jaspiar Singh, Anika Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Ho Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Ted Xiao, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *ArXiv*, abs/2307.15818, 2023. 2
- [12] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvilm: Endowing vision-language models with spatial reasoning capabilities. *ArXiv*, abs/2401.12168, 2024. 2
- [13] Lawrence Yunliang Chen, Simeon Adebola, and Ken Goldberg. Berkeley UR5 demonstration dataset. <https://sites.google.com/view/berkeley-ur5/home>. 4
- [14] An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatialrgpt: Grounded spatial reasoning in vision language model. 2024. 2
- [15] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. *ArXiv*, abs/2401.17270, 2024. 5
- [16] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. See <https://vicuna.lmsys.org> (accessed 14 April 2023), 2(3):6, 2023. 2
- [17] Jang Hyun Cho, B. Ivanovic, Yulong Cao, Edward Schmerling, Yue Wang, Xinshuo Weng, Boyi Li, Yurong You, Philipp Krahenbuhl, Yan Wang, and Marco Pavone. Language-image models with 3d understanding. 2024. 2
- [18] Jonathan Crespo, José Carlos Castillo, Óscar Martínez Mozo, and Ramón Barber. Semantic information for robot navigation: A survey. *Applied Sciences*, 2020. 1
- [19] Haoshu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11441–11450, 2020. 2
- [20] Zhen fei Yin, Jiong Wang, Jianjian Cao, Zhelun Shi, Dingning Liu, Mukai Li, Lu Sheng, Lei Bai, Xiaoshui Huang, Zhiyong Wang, Wanli Ouyang, and Jing Shao. Lamm: Language-assisted multi-modal instruction-tuning dataset, framework, and benchmark. *ArXiv*, abs/2306.06687, 2023. 2
- [21] Yunhai Feng, Nicklas Hansen, Ziyan Xiong, Chandramouli

- Rajagopalan, and Xiaolong Wang. Finetuning offline world models in the real world. *ArXiv*, abs/2310.16029, 2023. 4
- [22] Wendy Flores-Fuentes, Gabriel Trujillo-Hernández, Iván Y. Alba-Corpus, Julio César Rodríguez-Quiñonez, Jesús E. Mirada-Vega, Daniel Hernández-Balbuena, Fabian Natanael Murrieta-Rico, and Oleg Yu. Sergiyenko. 3d spatial measurement for model reconstruction: A review. *Measurement*, 2022. 1
- [23] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, and Rongrong Ji. Mme: A comprehensive evaluation benchmark for multimodal large language models. *ArXiv*, abs/2306.13394, 2023. 2, 8, 10
- [24] Huan Fu, Mingming Gong, Chaohui Wang, K. Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2002–2011, 2018. 3
- [25] Jensen Gao, Bidipta Sarkar, Fei Xia, Ted Xiao, Jiajun Wu, Brian Ichter, Anirudha Majumdar, and Dorsa Sadigh. Physically grounded vision-language models for robotic manipulation. *arXiv preprint arXiv:2309.02561*, 2023. 1
- [26] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. *International Journal of Computer Vision*, 127:398 – 414, 2016. 10
- [27] Faiza Gul, Wan Rahiman, and Syed Sahal Nazli Alhadhy. A comprehensive study for robot navigation techniques. *Cogent Engineering*, 6, 2019. 3
- [28] Muyang He, Yixin Liu, Boya Wu, Jianhao Yuan, Yueze Wang, Tiejun Huang, and Bo Zhao. Efficient multimodal learning from data-centric perspective. *ArXiv*, abs/2402.11530, 2024. 1, 2, 5, 8, 10, 11, 12
- [29] J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685, 2021. 9
- [30] Wen Huang, Hongbin Liu, Minxin Guo, and Neil Zhenqiang Gong. Visual hallucinations of multi-modal large language models. *ArXiv*, abs/2402.14683, 2024. 2
- [31] Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6693–6702, 2019. 8, 10
- [32] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard A. Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew J. Davison, and Andrew William Fitzgibbon. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 2011. 3
- [33] Jitesh Jain, Jianwei Yang, and Humphrey Shi. Vcoder: Versatile vision encoders for multimodal large language models. *ArXiv*, abs/2312.14233, 2023. 2
- [34] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi (Jim) Fan. Vima: General robot manipulation with multimodal prompts. *ArXiv*, abs/2210.03094, 2022. 4
- [35] Yizhang Jin, Jian Li, Yixin Liu, Tianjun Gu, Kai Wu, Zhengkai Jiang, Muyang He, Bo Zhao, Xin Tan, Zhenye Gan, Yabiao Wang, Chengjie Wang, and Lizhuang Ma. Efficient multimodal large language models: A survey. *ArXiv*, abs/2405.10739, 2024. 2
- [36] Zhizhong Kang, Juntao Yang, Zhou Yang, and Sai Cheng. A review of techniques for 3d reconstruction of indoor environments. *ISPRS Int. J. Geo Inf.*, 9:330, 2020. 1
- [37] Bing Wen Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. *ArXiv*, abs/2312.02145, 2023. 2
- [38] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024. 4
- [39] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 2
- [40] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003, 2023. 5, 12
- [41] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32 – 73, 2016. 5
- [42] Xin Lai, Zhiotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. *ArXiv*, abs/2308.00692, 2023. 2
- [43] Edmund Y. Lam. Computational photography with plenoptic camera and light field capture: tutorial. *Journal of the Optical Society of America. A, Optics, image science, and vision*, 32 11:2021–32, 2015. 1
- [44] Sergey Levine and Dhruv Shah. Learning robotic navigation from experience: principles, methods and recent results. *Philosophical Transactions of the Royal Society B*, 378, 2022. 1
- [45] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *ArXiv*, abs/2307.16125, 2023. 2, 10
- [46] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. Blip-2: Bootstrapping language-image pre-training

- with frozen image encoders and large language models. In *International Conference on Machine Learning*, 2023. 1
- [47] Jianing Li, Xi Nan, Ming Lu, Li Du, and Shanghang Zhang. Proximity qa: Unleashing the power of multi-modal large language models for spatial proximity analysis. *ArXiv*, abs/2401.17862, 2024. 2, 4
- [48] Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. Manipllm: Embodied multimodal large language model for object-centric robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18061–18070, 2024. 1
- [49] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji rong Wen. Evaluating object hallucination in large vision-language models. In *Conference on Empirical Methods in Natural Language Processing*, 2023. 10
- [50] Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-gemini: Mining the potential of multi-modality vision language models. *ArXiv*, abs/2403.18814, 2024. 2
- [51] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. 5
- [52] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *ArXiv*, abs/2304.08485, 2023. 1, 2
- [53] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chun yue Li, Jianwei Yang, Hang Su, Jun-Juan Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *ArXiv*, abs/2303.05499, 2023. 5
- [54] Yuanzhan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? *ArXiv*, abs/2307.06281, 2023. 2, 10
- [55] Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024. 2
- [56] Abhishek Padalkar, Acorn Pooley, Ajinkya Jain, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anikait Singh, Anthony Brohan, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023. 1, 3, 5, 8, 12
- [57] Hailong Pan, Tao Guan, Yawei Luo, Liya Duan, Yuan Tian, Liu Yi, Yizhu Zhao, and Junqing Yu. Dense 3d reconstruction combining depth and rgb information. *Neurocomputing*, 175:644–651, 2016. 3
- [58] Xingang Pan, Jianping Shi, Ping Luo, Xiaogang Wang, and Xiaou Tang. Spatial as deep: Spatial cnn for traffic scene understanding. In *AAAI Conference on Artificial Intelligence*, 2017. 1
- [59] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. 8, 10
- [60] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44:1623–1637, 2019. 2, 4
- [61] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kun-chang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks. *ArXiv*, abs/2401.14159, 2024. 5
- [62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. *ArXiv*, abs/2210.08402, 2022. 8
- [63] Haochen Shi, Huazhe Xu, Samuel Clarke, Yunzhu Li, and Jiajun Wu. Robocook: Long-horizon elasto-plastic object manipulation with diverse tools. *ArXiv*, abs/2306.14447, 2023. 4
- [64] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. *ArXiv*, abs/2109.12098, 2021. 1
- [65] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, 2012. 12
- [66] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1
- [67] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024. 2
- [68] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971, 2023. 2, 8
- [69] Jonas Uhrig, N. Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. *2017 International Conference on 3D Vision (3DV)*, pages 11–20, 2017. 12
- [70] Homer Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Maximilian Du, Chongyi Zheng, Tony Zhao, Philippe Hansen-Estruch, Quan Ho Vuong, Andre Wang He, Vivek

- Myers, Kuan Fang, Chelsea Finn, and Sergey Levine. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, 2023. 4
- [71] Yixuan Wang, Zhuoran Li, Mingtong Zhang, K. Driggs-Campbell, Jiajun Wu, Li Fei-Fei, and Yunzhu Li. D3fields: Dynamic 3d descriptor fields for zero-shot generalizable robotic manipulation. *ArXiv*, abs/2309.16118, 2023. 4
- [72] Steffen Werner, Bernd Krieg-Brückner, Hanspeter A. Mallot, Karin Schweizer, and Christian Freksa. Spatial cognition: The role of landmark, route, and survey knowledge in human and robot navigation. In *GI Jahrestagung*, 1997. 1
- [73] Zhixuan Xu, Chongkai Gao, Zixuan Liu, Gang Yang, Chenrui Tie, Haozhuo Zheng, Haoyu Zhou, Weikun Peng, Debang Wang, Tianyi Chen, Zhouliang Yu, and Lin Shao. Manifoundation model for general-purpose robotic manipulation of contact synthesis with arbitrary objects and robots. 2024. 2
- [74] Jihan Yang, Runyu Ding, Ellis L Brown, Xiaojuan Qi, and Saining Xie. V-irl: Grounding virtual intelligence in real life. *ArXiv*, abs/2402.03310, 2024. 2
- [75] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiasi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. *ArXiv*, abs/2401.10891, 2024. 2, 3
- [76] Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. The dawn of lmms: Preliminary explorations with gpt-4vision. *ArXiv*, abs/2309.17421, 2023. 1
- [77] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *ArXiv*, abs/2310.07704, 2023. 2
- [78] Alex Young, Bei Chen, Chao Li, Chengan Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, et al. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*, 2024. 2
- [79] Jianhao Yuan, Shuyang Sun, Daniel Omeiza, Bo Zhao, Paul Newman, Lars Kunze, and Matthew Gadd. Rag-driver: Generalisable driving explanations with retrieval-augmented in-context learning in multi-modal large language model. *arXiv preprint arXiv:2402.10828*, 2024. 1
- [80] Yuqian Yuan, Wentong Li, Jian Liu, Dongqi Tang, Xinjie Luo, Chi Qin, Lei Zhang, and Jianke Zhu. Osprey: Pixel understanding with visual instruction tuning. *ArXiv*, abs/2312.10032, 2023. 2
- [81] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhua Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *ArXiv*, abs/2311.16502, 2023. 2
- [82] Ziyao Zeng, Daniel Wang, Fengyu Yang, Hyoungseob Park, Yangchao Wu, Stefano Soatto, Byung-Woo Hong, Dong Lao, and Alex Wong. Wordepth: Variational language prior for monocular depth estimation. *ArXiv*, abs/2404.03635, 2024. 10
- [83] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11941–11952, 2023. 8, 10
- [84] Ao Zhang, Wei Ji, and Tat-Seng Chua. Next-chat: An lmm for chat, detection and segmentation. *ArXiv*, abs/2311.04498, 2023. 2
- [85] Hao Zhang, Hongyang Li, Feng Li, Tianhe Ren, Xueyan Zou, Shilong Liu, Shijia Huang, Jianfeng Gao, Lei Zhang, Chun yue Li, and Jianwei Yang. Llava-grounding: Grounded visual chat with large multimodal models. *ArXiv*, abs/2312.02949, 2023. 2
- [86] Renrui Zhang, Ziyao Zeng, and Ziyu Guo. Can language understand depth? *Proceedings of the 30th ACM International Conference on Multimedia*, 2022. 10
- [87] Bo Zhao, Boya Wu, Muyang He, and Tiejun Huang. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*, 2023. 2, 5
- [88] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *ArXiv*, abs/2304.06718, 2023. 5