

CoSpace: Benchmarking Continuous Space Perception Ability for Vision-Language Models

Supplementary Material

A. Details for Data Collection and Annotation

Image Collection. Our benchmark comprises 2,918 images, including 2,302 outdoor images from publicly available APIs and 616 indoor images from the simulator platform. This section reports details for image collection.

To collect required outdoor images, we utilize two different API interfaces from Baidu Map API¹: the Place API² and the Panorama API³. For Place API, we first provide some keywords, including 8 keywords for cities and 12 keywords for places as follows:

- **City:** Beijing, Shanghai, Guangzhou, Shenzhen, Hangzhou, Nanjing, Chengdu, Chongqing.
- **Place:** school, hospital, museum, park, library, mall, cinema, railway station, airport, stadium, supermarket.

The Place API is responsible for providing locations, in the form of latitude and longitude, e.g. `{"lat": 40.099567, "lng": 116.515935}`, of all the combinations of the given keywords, such as “schools in Beijing” and “parks in Chongqing”. Then, given the latitude and longitude of a location, the the Panorama API returns the image set (views of four orientations, north, east, south, and west) of the corresponding location following our defined continuous visual space format as elaborated.

We utilized Habitat-Sim platform [1–3] to collect images of indoor scenes from the simulator environment. This platform provides an interface to instruct a virtual robot to explore the surroundings and capture images of its views. Iteratively, we place the robot at random positions and capture views of four directions to the robot, including front, right, back and left, making them to form a continuous visual space as illustrated.

In total, the APIs and the platform generated 2,883 and 6,000 images respectively. We then manually filtered the images accessed from HM3D dataset and finally preserve 800 images. The filtering requirements are as follows: as a large number of images collected from HM3D are shot in the same scene and contain similar visual information and we require that there should exist notable difference between different sets of images to avoid duplication.

Data Annotation. After collecting and filtering the images, we follow a two-phase paradigm for annotation:

¹<https://lbsyun.baidu.com/>

²<https://lbsyun.baidu.com/faq/api?title=webapi/guide/web-service-placeapi>

³<https://lbsyun.baidu.com/faq/api?title=viewstatic>

firstly utilizing GPT-4o to generate questions, and then asking human annotators to provide the groundtruth answers for the generated questions. The prompt templates for question generation using GPT-4o are as following:

Prompt Templates Used for Data Annotation

System Prompt for Direction and Counting Category:

You are a data curation engineer and need to generate some `**question**-**answer**` pairs according to the requirements and given images.

You will be given an example, which includes a set of images and a perfect `**question**-**answer**` pair related to the images set.

You should follow the given example and generate `**question**-**answer**` pairs for another images set.

You can generate one or more `**question**-**answer**` pairs. These pairs should be practical, accurate according to the images.

Give your output in the following JSON format:

```
[
  {
    "question": "some text",
    "answer": "some text",
  }, // question-answer pair 1
  {
    "question": "some text",
    "answer": "some text",
  } // question-answer pair 2
]
```

System Prompt for Planning Category:

You are a data curation engineer and need to generate some `**question**-**answer**` pairs according to the requirements and given images.

You will be given an example, which includes a set of images and a perfect `**question**-**answer**` pair related to the image set.

I want you to follow the example and generate some similar `**question**-**answer**` pairs for another images set.

You can generate one `**question**-**answer**` pair or more, as long as you can ensure the quality and correctness of your output.

Give your output in the following JSON format:

```
[
```

```

{
  "question-qa": "some text",
  "answer-qa": "some text",
  "question-dec": "some text",
  "answer-dec": "some text",
}, // question-answer pair 1
{
  "question-qa": "some text",
  "answer-qa": "some text",
  "question-dec": "some text",
  "answer-dec": "some text",
} // question-answer pair 2
]

```

You should notice that there is a `**question**-**answer**` pair for qa and `**question**-**answer**` pair for dec in each output item. I want them to focus on the same object. The qa question should be about where a certain object is and the dec question should be about how to fetch that specific thing or something else related to that.

The question for the dec question can only be composed of the following actions: go ahead, turn left, turn right, turn back. So the accepted choices and answer are like "turn right and go ahead" or "go ahead and turn left".

User Prompt:

Task description

Now you need to generate some space and directions related `**question**-**answer**` pairs. You are given eight images. The first four belong to the example image set, while others belong to the test image set. The `**question**-**answer**` given in the example is based on the first four images and the `**question**-**answer**` pairs you generate should be based on the test images set. Also, you need to set this task as the form choice as is shown in the example.

Image explanation

The images in this task are arranged in the following sequence:

All the images in a set are shot in the same scene, but of four directions.

The `**first**` image is facing `**north**`, towards the `**front**` side.

`**Second**` facing `**east**`, towards the `**right**` side.

`**Third**` facing `**south**`, towards the `**back**` side.

`**Fourth**` facing `**west**`, towards the `**left**` side.

Also, there are some overlap between the adjacent images.

Example task

// A given question-answer pair as in-context example

to help GPT-4o better understand the requirements for question generation. Meanwhile, GPT-4o is required to generate a corresponding answer to the question. However, during manual review, we observed that although the generated questions were appropriate, GPT-4o often produced incorrect answers that did not align with the given images. To this end, we manually annotated all the answers for the generated questions. For each question-answer pairs, an average of two annotators are involved to ensure reliability. After annotation, 2,302 of 2,883 outdoor images and 616 of 800 images acquired from HM3D dataset are left, finally comprising our CoSpace.

B. Prompt Templates for Evaluation

We report our prompt templates used in the experiments for evaluation as follows:

Prompt Templates for Evaluation

Direction Category:

You are provided with four images shot in the same scene towards different direction. These images overlap in a certain manner, and are arranged in the following order:

`{order}`

Carefully analyze these images, and answer the following question from the given options.

Question: `{question}`. Options: `{options}`.

You should generate your answer from 'A, B, C or D'. Your answer:

Counting Category:

You are provided with four images shot in the same scene towards different direction. These images overlap in a certain manner, and are arranged in the following order:

`{order}`

Carefully analyze these images, and answer the following question.

Question: `{question}`.

You should generate a single number as your answer. Your answer:

Rotation-Angle Task:

You are provided with four images shot in the same scene. They are taken from the same position but towards different directions. For example, after taking the first image, the photographer turns a certain degree clockwise. We denote this degree as the turning degree between

We included a one-shot in-context example in the prompt

two adjacent images. These images are arranged in the following order: they are arranged clockwise and the turning angle between adjacent images are the same.

Also note that the image sequence does not always cover a full 360-degree scene. The covered degree of the image sequence can range from 90 to 360.

Carefully analyze these images, and answer the following question from the given options.

Question: **{question}**. Options: **{options}**.

You should generate your answer from 'A or B'.
Your answer:

Rotation-Difference Task:

You are provided with five images shot in the same scene at the same position towards different directions. In these five images, four are taken in the following way: after taking the first image, the photographer turns a certain degree clockwise to take the next one and the degree always remains the same.

These images are also arranged as the sequence they are taken. However, the rest one image is shot towards totally different direction and is randomly inserted into the image sequence.

Carefully analyze these images, and answer the following question.

Question: **{question}**

You should generate a single number as your answer, where 1 represents the first image and 5 represents the last image. Your answer:

Planning Category:

You are a human like robot. You can only go straight ahead. If you want to walk in the other direction, you need to first turn to the target direction and then move forward. The images are arranged in the following order:

{order}

There are two questions for you to answer at the same time. Please carefully analysis your surroundings and answer the following questions:

Question 1: **{question_qa}** Options: **{options_qa}**.

Question 2: **{question_dec}** Options: **{options_dec}**.

You should generate your answer in a JSON dict containing 2 fields:

```
{
  'Answer1': type str, answer to question 1, in the
  form of 'A', 'B', 'C' or 'D',
```

```
  'Answer2': type str, answer to question 2, in the
  form of 'A', 'B', 'C' or 'D'.
}
```

Your response:

For each single query, **{question}**, **{options}**, **{question_qa}**, **{question_qa}**, **{question_dec}** and **{options_dec}** are replaced by query-specific question and options. Also **{order}** are replaced with detailed explanation of the input image order, tailored to different tasks and settings.

However, some of the assessed models do not follow instructions properly, and slightly adapt the templates for them to obtain valid responses. For instance, Mantis-8B and VILA1.5-8B output the single "A" as the answer for all queries regardless of the question and options. Moreover, models like Mono-InternVL-2B and Brote-IM-XXL can not follow the instruction to output the required JSON dict in the Planning category. For the evaluation of models that output the same answer for all questions, we use different prompts. For instance, we only maintain question, options and the explanation of order for the evaluation of Mantis-8B, which greatly simplifies the prompt template and leads to the better response. For the models that cannot follow instructions for the Planning category, we adopt the strategy of asking models to response to the PLA-QA task and PLA-Dec task separately.

C. Discussion on Open-ended Evaluation

As mentioned, we adopted the form of multiple choices for all tasks except for the ROT-Dif task. Our benchmark typically features tasks in real-world scenarios. In practical applications, models are often required to handle fully open-ended questions without being constrained to a set of pre-defined choices. However, we chose not to evaluate these tasks in an open-ended setting for the following concerns:

- For DIR-Rec and PLA-QA tasks, the answers universally contain fixed directions (e.g. "south") or trajectory directions (e.g. "east to west"). These answers fall in a certain range, meaning that in an open-ended evaluation, models are actually choosing from a fixed and implicit set of options, with the number of options being more than four. Therefore, we conclude these tasks as semi-open-ended, reducing the necessity of conducting fully open-ended evaluations.
- Similarly, the PLA-Dec task also features semi-open-ended answers, because the action space for this task is limited to "go ahead", "turn right", "turn left" and "turn back". Any final decision needs to be composed of these atomic actions. Consequently, open-ended evaluation can be replaced by providing multiple options and we argue

that providing four options is sufficient for the evaluation of the continuous space perception ability.

- Open-ended evaluation is not suitable for the DIR-Obj task, which requires models to identify existing objects regarding the given direction. There usually exist more than one objects in a given direction and all these objects should be noted as potential answers, significantly increasing the difficulty and ambiguity of evaluation. Thus, open-ended evaluation is not employed for this task.

To summarize, the current evaluation setting can provide us with a comprehensive understanding of the assessed continuous space perception ability and we chose not to implemented open-ended setting for further evaluation.

D. Human Evaluation

To assess the difficulty and reasonability of our benchmark, we conducted an extensive human evaluation with each sample tested two times. We provide results for human evaluation in Table 1. Concluded from this table, humans achieve significantly higher accuracy compared to the best scores from models for all the tasks except for ROT-Dif, where the best model performance only lags behind by 2.09%. The superiority of Claude-3.7-sonnet in the ROT-Dif task (93.50%) lies in the sensitivity inconsistencies within a series of continuous images. In this task, human might overlook subtle inconsistencies, especially when the differences are as small as for ROT-Dif task.

	DIR-Rec	DIR-Obj	CNT	ROT-Ang	ROT-Dif	PLA-QA	PLA-Dec
Random	24.82	24.91	10.01	49.37	19.88	25.11	24.30
Models	44.40	54.40	51.25	64.33	93.50	54.73	69.34
Human	82.40	80.20	78.25	96.17	95.59	88.26	82.87

Table 1. Results for human evaluation. We report the average accuracy of human annotators, and take the highest accuracy among all assessed models for each task as comparison.

E. Details for Single Image Pipeline

For single-image models, we convert images into captions to enable the evaluation. In this section we provide the used prompt template and cases for the single image pipeline.


Prompt Template Used for Single Image Pipeline

Prompts for Generating Captions:


You are a helpful assistant and is now faced with a task. There are a series of images and several questions related to it. However, you can access to only one image at the same time. Therefore, you need to write down some captions about the image when you view it to help you answer the question. Finally, you will be provided with only the captions you write down when tasked with

Instance of question-option-image triplet


Which direction is the alleyway going according to the images?
A. north to south **B. east to west**
C. northeast to southwest D. northwest to southeast




Prompt for generating captions: *(described in Section E of Supplementary)*
Generated captions:




Caption: A street scene with multi-story buildings ... with an alleyway alone the buildings. The perspective suggests the alleyway runs from **north to south**.



Caption: The alleyway in the image appears to be aligned with the direction of the street ... Therefore, the correct answer is: **east to west**.



Caption: Given that the first image indicates the alleyway runs ... , aligning with the second image's perspective ... , the correct answer is: **northeast to southwest**.



Caption: Given that the first image indicates the alleyway runs ... , aligning with the third image's perspective ... , the correct answer is: **northwest to southeast**.

Prompt for generating the final answer:
(described in Section E of Supplementary, no images are provided in this stage, model sees only the captions generated above)

Answer: Based on the provided images and their directions, the alleyway is aligned with the street which runs from **north to south**. The first image shows indicates the alleyway runs from **north to south**, and assuming a consistent orientation across the images, the fourth image should show the alleyway **extending westward, aligning with the third image's perspective**. Therefore, the correct answer is: **D. northwest to southeast**.

Figure 1. Case for single image pipeline. For illustration, we showcase all the images in the figure, but models can only see one image at the same time. The responses in this case are all generated by MiniCPM-V 2.6.

answering the question.

{query} // Same as the query for regular evaluation, containing task descriptions, explanations of input images and the question.

This is the {first, second, third, fourth} image. Now you can write down the caption of this image to help you finally answer the question. You should notice that you will not be provided with the images when generating final answer, so the caption should be as detailed as possible. The captions of the past images are listed below:

{captions} // The captions for the past images. Captions are generated as the order of input images.

Prompts for Generating the Final Answer:

You are a helpful assistant and is now faced with a

task. There are a series of images and several questions related to it. However, you can access to only one image at the same time. Therefore, you need to write down some captions about the image when you view it to help you answer the question. Finally, you will be provided with only the captions you write down when tasked with answering the question.

Your captions of all the images are listed below:

{captions} // Generated captions for all images.

Now answer the given question and you should output in the required format.

{query} // Query containing the task description, explanation for the input images and the question, same as the prompt of the regular setting.

As shown in Figure 1, MiniCPM-V 2.6 captured the existence of alleyway in the first image which is facing towards north but mistakenly captioned that the alleyway run from north to south. Actually, the alleyway is running parallel in the first image, standing for the direction of east to west. Though the model correctly identify the direction through the second image, when generating the captions for other two images, it was mistaken by the caption of the first image and finally generate the wrong answer.

F. Impact of Rationales on the ROT-Ang Task

We notice that most of the evaluated models fail to response properly for the Rotation-Angle (ROT-Ang) task. To further investigate into this phenomenon, we provide two examples of the rationales respectively generated by InternVL-2 and Claude-3.5-sonnet in the ROT-Ang task. As shown in Figure 2, InternVL-2 outputs brief and generic captions for each image, and mistakenly perceives the turning angle between images as 90 degrees, even if 90 is not included in the options. In contrast, Claude-3.5-sonnet correctly identify the total scene coverage as 180 degrees and successfully recognizes the three equal intervals, which helps it accurately derive the answer of 60 degrees.

G. Case Study

In this section, we provide examples of cases generated by different models on our proposed CoSpace through Figure 3 to Figure 8. As shown in Figure 7, for the left case, three of the four assessed proprietary models selected “*B. backleft*” as the answer. In order to correctly answer this question, we should notice that the dining table appears in the third image, which is facing back, and therefore choose from “*B*” and “*D*”. Actually, these models successfully recognize the appearance of the dining table in the third image, but as

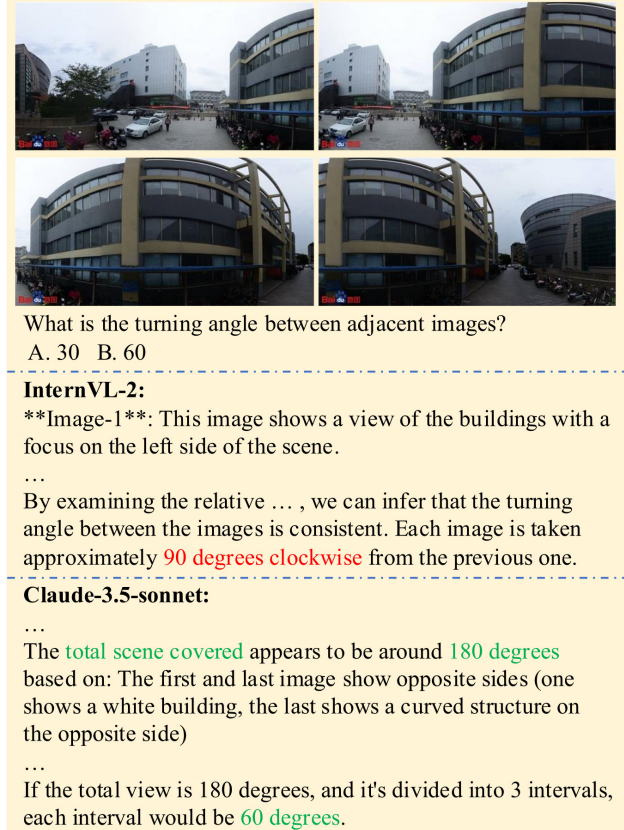


Figure 2. Cases of generated rationales in the Rotation-Angle task.

is located on the left side of the image, they consequently identify the answer as “*B. backleft*”. However, the left side of the third image represents the “*backright*” relative to the standing position in the real space. These models were deceived by the raw visual clues and failed to fill the gap between given images and the original continuous space.

References

- [1] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander Clegg, Michal Hlavac, So Yeon Min, Vladimír Vondruš, Theophile Gervet, Vincent-Pierre Berges, John M Turner, Oleksandr Maksymets, Zsolt Kira, Mrinal Kalakrishnan, Jitendra Malik, Devendra Singh Chait, Unnat Jain, Dhruv Batra, Akshara Rai, and Roozbeh Mottaghi. Habitat 3.0: A co-habitat for humans, avatars, and robots. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR 2024)*, 2024. 1
- [2] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.
- [3] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans,

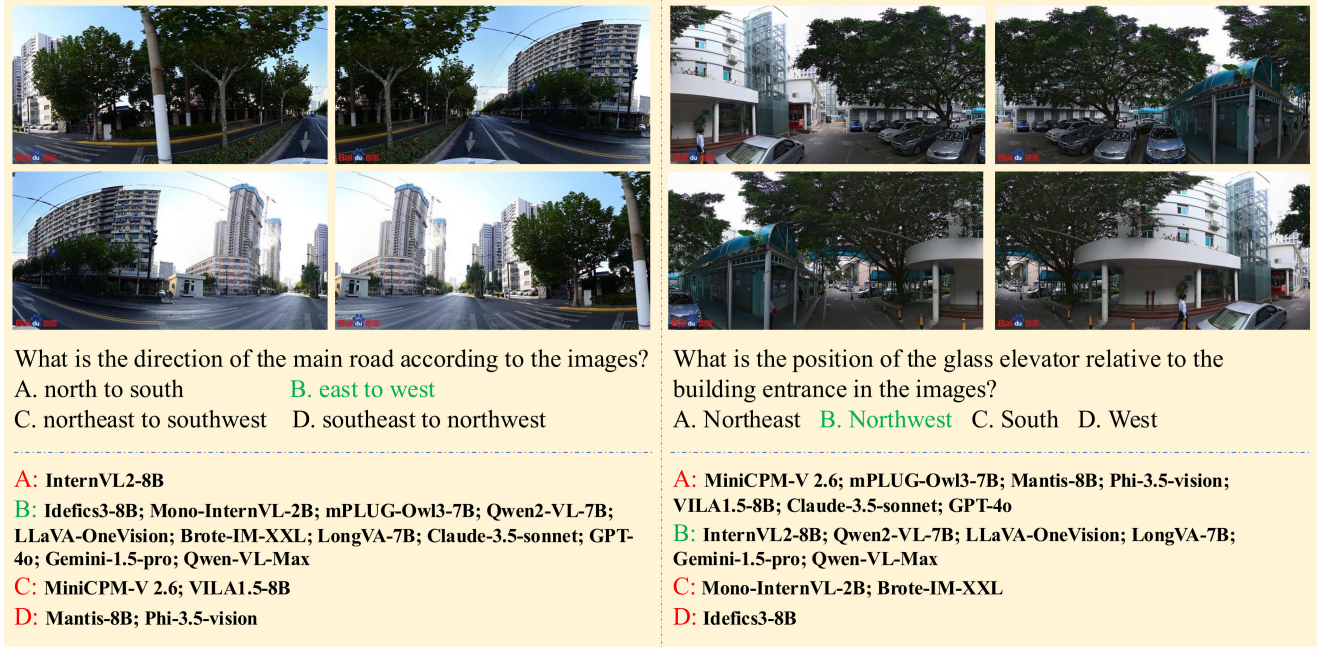


Figure 3. Cases for Direction Recognition task. Images are arranged as the following order: the first image is facing towards north, second facing east, third facing south and fourth facing west.

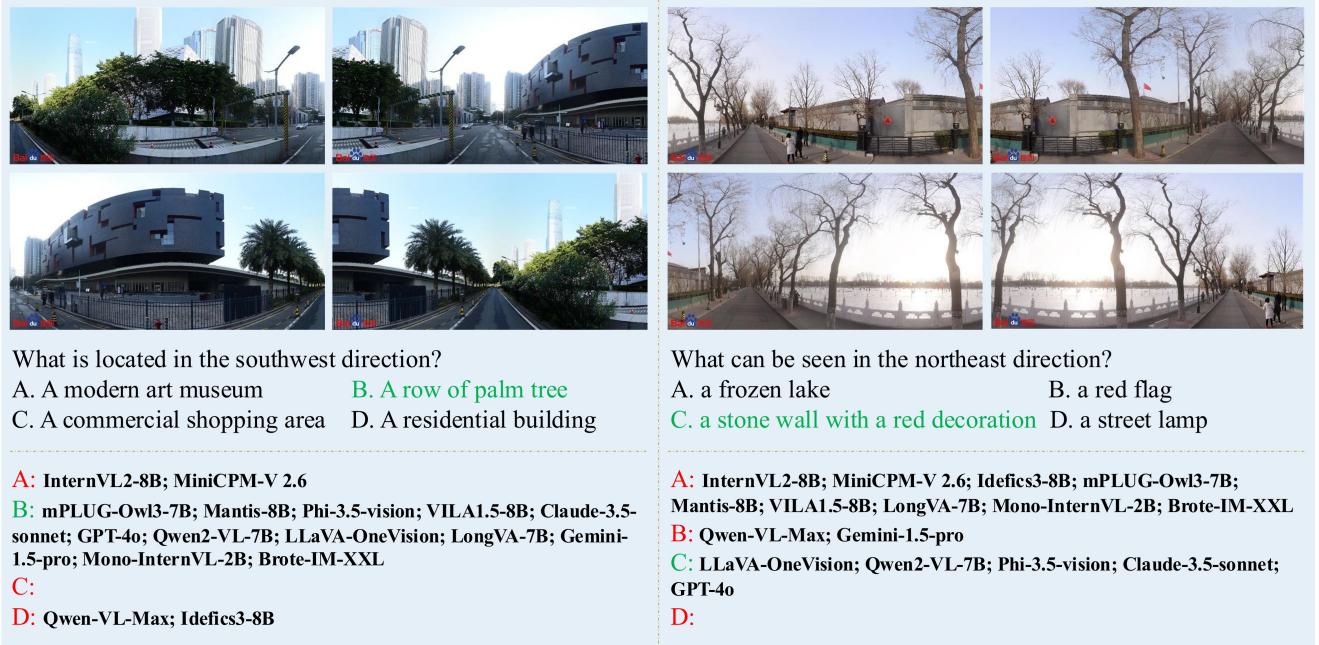


Figure 4. Cases for Direction Object Perception task. Images are arranged as the following order: the first image is facing towards north, second facing east, third facing south and fourth facing west.

Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Ji-

tendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 1

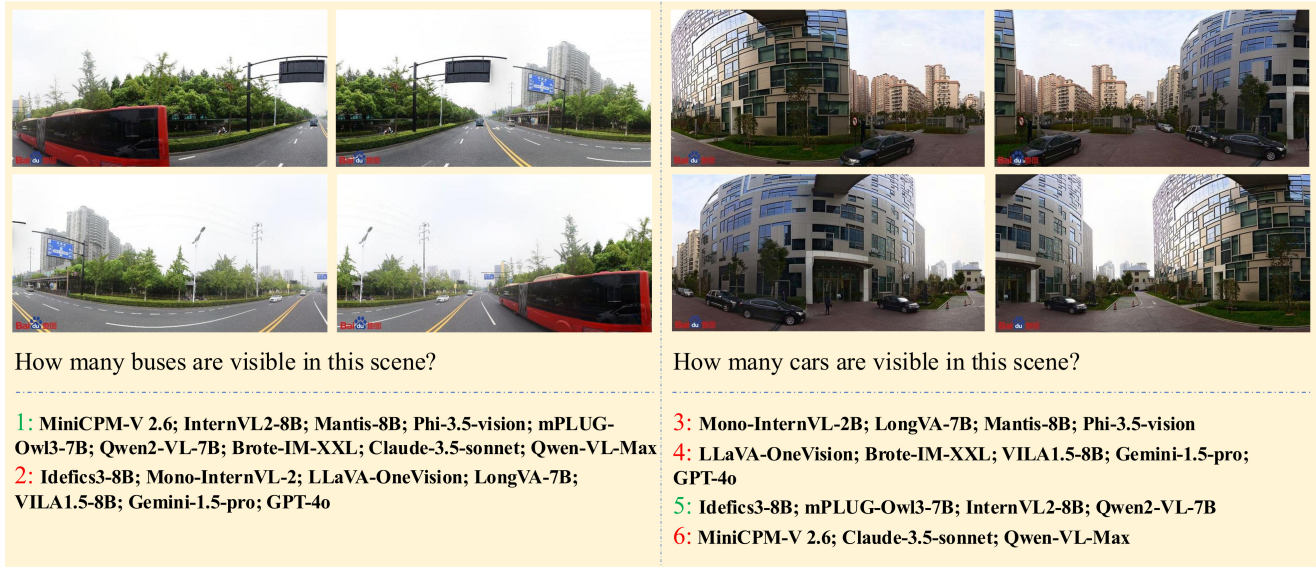


Figure 5. Cases for Counting task. Images are arranged as the following order: the first image is facing towards north, second facing east, third facing south and fourth facing west.

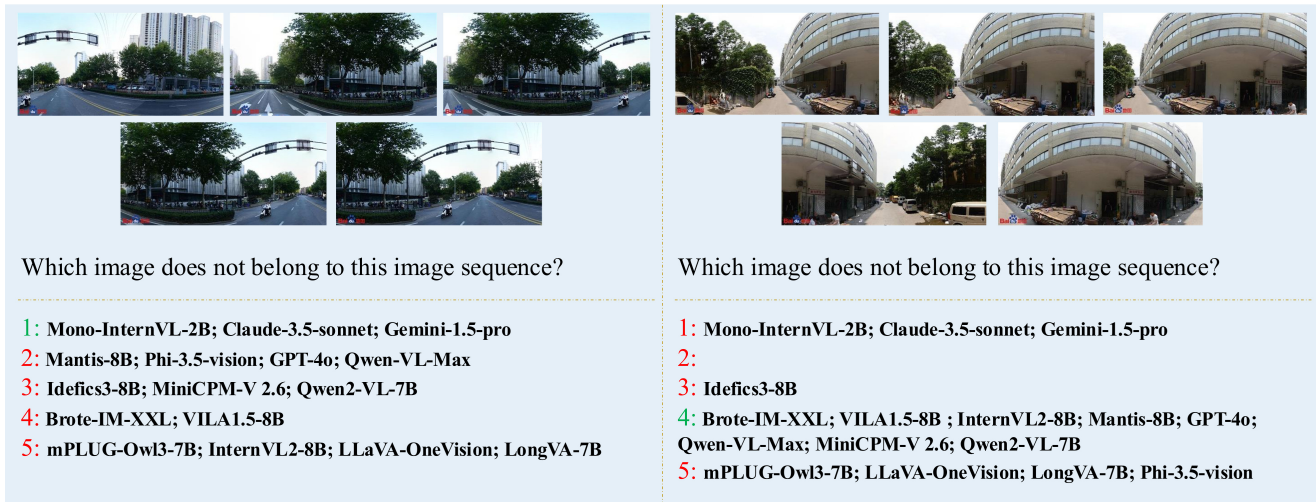


Figure 6. Cases for Direction Rotation Difference task.



Figure 7. Cases for Planning Question Answering task. Images are arranged as the following order: the first image is facing towards front, second facing right, third facing back and fourth facing left.



Figure 8. Cases for Planning Decision task. Images are arranged as the following order: the first image is facing towards front, second facing right, third facing back and fourth facing left.