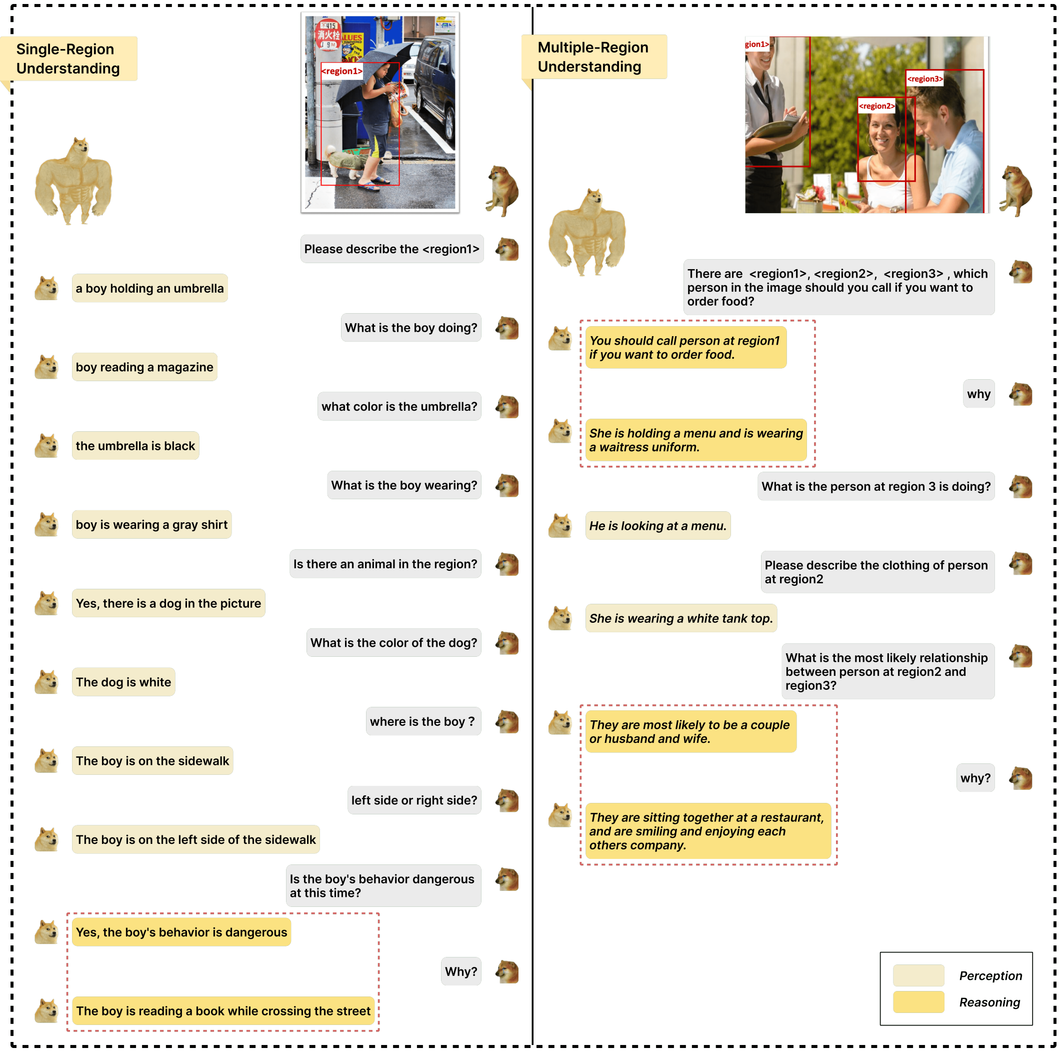
GPT4RoI: Instruction Tuning Large Language Model on Region-of-Interest

Shilong Zhang[1,2*] **Peize Sun**[1*] **Shoufa Chen**[1*]
Min Xiao[2] **Wenqi Shao**[2] **Wenwei Zhang**[2] **Kai Chen**[2] **Ping Luo**[1,2]

*equal contribution

[1]The University of Hong Kong [2]Shanghai AI Laboratory



Please describe the <region1>

a boy holding an umbrella

What is the boy doing?

boy reading a magazine

what color is the umbrella?

the umbrella is black

What is the boy wearing?

boy is wearing a gray shirt

Is there an animal in the region?

Yes, there is a dog in the picture

What is the color of the dog?

The dog is white

where is the boy ?

The boy is on the sidewalk

left side or right side?

The boy is on the left side of the sidewalk

Is the boy's behavior dangerous at this time?

Yes, the boy's behavior is dangerous

Why?

The boy is reading a book while crossing the street

There are <region1>, <region2>, <region3> , which person in the image should you call if you want to order food?

You should call person at region1 if you want to order food.

why

She is holding a menu and is wearing a waitress uniform.

What is the person at region 3 is doing?

He is looking at a menu.

Please describe the clothing of person at region2

She is wearing a white tank top.

What is the most likely relationship between person at region2 and region3?

They are most likely to be a couple or husband and wife.

why?

They are sitting together at a restaurant, and are smiling and enjoying each others company.

Figure 1: GPT4RoI is a vision-language model based on instruction tuning large language model (LLM) on region-text pairs. It is capable of processing the user instructions that contain interleaved sequences of language and spatial information `<region>`. It achieves fine-grained multimodal understanding tasks such as region caption and reasoning.

Preprint. Under review.

Abstract

Instruction tuning large language model (LLM) on image-text pairs has achieved unprecedented vision-language multimodal abilities. However, their vision-language alignments are only built on image-level, the lack of region-level alignment limits their advancements to fine-grained multimodal understanding. In this paper, we propose instruction tuning on region-of-interest. The key design is to reformulate the bounding box as the format of *spatial instruction*. The interleaved sequences of visual features extracted by the spatial instruction and the language embedding are input to LLM, and trained on the transformed region-text data in instruction tuning format. Our region-level vision-language model, termed as GPT4RoI, brings brand new conversational and interactive experience beyond image-level understanding. (1) *Controllability:* Users can interact with our model by both language and spatial instructions to flexibly adjust the detail level of the question. (2) *Capacities*: Our model supports not only single-region spatial instruction but also multi-region. This unlocks more region-level multimodal capacities such as detailed region caption and complex region reasoning. (3) *Composition*: Any off-the-shelf object detector can be a spatial instruction provider so as to mine informative object attributes from our model, like color, shape, material, action, relation to other objects, etc. The code, dataset, and demo can be found at https://github.com/jshilong/GPT4RoI.

1 Introduction

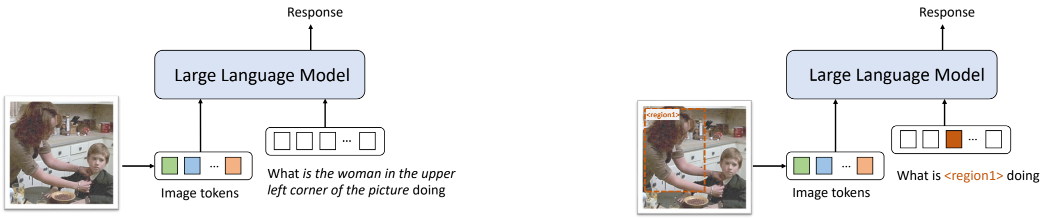
The recent advancements of large language models (LLM) have shown incredible performance in solving natural language processing tasks in a human-like conversational manner, for example, commercial product ChatGPT [31], Claude [2], Bard [18], text-only GPT-4 [32] and community open-source LLaMA [47], Alpaca [46], Vicuna [8], ChatGLM [14], MOSS [43], etc. Their unprecedented capabilities present a promising path towards general-purpose artificial intelligence models.

Witnessing the power of LLM, the field of multi-modal models [13, 16, 20, 54] is developing a new technology direction to leverage LLM as the universal interface to build general-purpose models, where the feature space of a specific task is tuned to be aligned with the feature space of pre-trained language models. As one of the representative tasks, vision-and-language models align the vision encoder to LLM by instruction tuning on image-text pairs, such as MiniGPT-4 [61], LLaVA [26], LLaMA-Adapter [58], InstructBLIP [11], etc. Under the design principle of instruction tuning, the capacities of vision-and-language models largely depend on the alignment quality.

Although these works achieve amazing multimodal abilities, their alignments are only on image-text pairs [6, 7, 33, 41, 42], the lack of region-level alignment limits their advancements to more fine-grained understanding tasks such as region caption [22] and reasoning [56]. To enable region-level understanding in vision-language models, some works attempt to leverage external vision models,

Model	Image Cap. & Rea.	Region Cap. & Rea.	Multi-Region Cap. & Rea.	Multi-Round Dialogue	End-to-End Model
Visual ChatGPT	✓	✗	✗	✓	✗
MiniGPT-4	✓	✗	✗	✓	✓
LLaVA	✓	✗	✗	✓	✓
InstructBLIP	✓	✗	✗	✓	✓
MM-REACT	✓	✓	✓	✓	✗
InternGPT	✓	✓	✓	✓	✗
VisionLLM	✓	✓	✗	✗	✓
CaptionAnything	✓	✗	✗	✗	✗
DetGPT	✓	✓	✗	✓	✗
GPT4RoI	✓	✓	✓	✓	✓

Table 1: Comparisons of vision-language models. Our GPT4RoI is an end-to-end model that supports region-level understanding and multi-round conversation.



(a) Visual Instruction Tuning

(b) Spatial Instruction Tuning

Figure 2: Comparison of visual instruction tuning on image-text pairs and spatial instruction tuning on region-text pairs. The bounding box and text description of each object are provided in region-text datasets. During training, spatial instruction is from annotations and in inference, spatial instruction is given by user input.

for example, MM-REACT [54], InternGPT [28] and DetGPT [35]. However, their non-end-to-end architecture is a sub-optimal choice for general-purpose multi-modal models.

In this paper, we are aimed to build up an end-to-end vision-language model that supports fine-grained understanding on region-of-interest. Since the model architecture in image-level vision-language models [11, 26, 58, 61] compresses the whole image as the image embedding without any operation to refer to specific regions, our key design is to introduce the object box as the format of *spatial instruction*, as shown in Figure 2. The visual features extracted by spatial instruction, along with language instruction, are sent to LLM to obtain the response. For example, when the question is the interleaved sequence of *"what is this <region> doing?"*, the model will replace the *<region>* with the region feature referred by spatial instruction. The spatial instruction can be flexibly implemented by RoIAlign [19] or Deformable attention [62].

To establish fine-grained alignment between region-text pairs, we upgrade the training data from image-text datasets to region-text datasets, where the bounding box and the text description of each object are provided. The datasets are consolidated from publicly available ones including COCO object detection [25], RefCOCO [55], RefCOCO+ [55], RefCOCOg [29], Flickr30K entities [36], Visual Genome(VG) [22] and Visual Commonsense Reasoning(VCR) [56]. These datasets are transformed to instruction tuning format. Furthermore, the spatial instruction can use image-text training data as well, such as LLaVA150K [26], by applying off-the-shelf object detectors to extract object boxes in the images and use them as spatial instruction. Learning from these image-text datasets curated for visual instruction tuning, our model is strengthened in conversational quality and produces more human-like responses.

The collected datasets are categorized into two types based on text length. First, short-text data contains object category and simple attribute information. It is used for pre-training the region feature extractor without impacting the LLM. Second, long-text data often contains complex concepts or requires common sense reasoning. To simulate flexible user instructions in actual use, we construct complex spatial instructions for these data to facilitate end-to-end fine-tuning of the region feature extractor and LLM.

Benefiting from spatial instruction tuning, our model brings a new interactive experience to the user of vision-language models, where the user can express the question to the model in both language form and spatial instruction form. This leads to new capacities beyond image-level understanding, such as region caption, and complex region reasoning, as shown in Figure 1.

In summary, our work makes the following contributions:

- We move forwards region-level vision-language models by instruction tuning LLM on region-text datasets. Compared to previous image-level models, our model is developed with new capabilities, such as region caption and reasoning.
- We introduce the spatial instruction to refer to region-of-interest, and the region features extracted from the visual encoder are input to LLM along with language instruction to obtain a response.
- We release the codebase, instruction tuning format of datasets and online demo in https://github.com/jshilong/GPT4RoI.

2 Related Work

2.1 Large Language Model

The field of natural language processing (NLP) has achieved significant development by the high-capability large language model (LLM). The potential of LLM is first demonstrated by pioneering works such as BERT [12] and GPT [38]. Then scaling up progress is started and leads to a series of excellent works, for example, T5 [39], GPT-3 [3], Flan-T5 [10], PaLM [9], etc. With the growth of training data and model parameters, this scaling up progress brings to a phenomenal product, ChatGPT [31]. By generative pre-trained LLM and instruction tuning [34] on human feedback, ChatGPT shows unprecedented performance on conversations with humans, reasoning and planning tasks [4, 30, 52], etc.

2.2 Vision-Language Model

To utilize high-performance LLM to build up vision-language models, LLM as task coordinator is proposed. Given the user instruction, LLM parses the instruction and calls various external vision models. Some representative works are Visual ChatGPT [48], ViperGPT [44], MM-REACT [54], InternGPT [28], VideoChat [23], etc. Although these models largely expand the scope of multimodal models, they depend on external vision models and these non-end-to-end architectures are not the optimal choice for multi-modal models. To obtain end-to-end vision-language models, instruction tuning LLM on image-text pairs is proposed to align visual features with LLM and accomplish multimodal tasks in a unified way, for example, Flamingo [1], MiniGPT-4 [61], LLaVA [26], LLaMa-Adapter [58], InstructBLIP [11], MM-GPT [17], etc. These models achieve amazing image-level multimodal abilities, while LVLM-eHub [50] finds that these models still have performance bottlenecks when need to be under specific region reference. Our GPT4RoI follows the research line of instruction tuning and moves forwards region-level multimodal understanding tasks such as region caption [22] and reasoning [56].

2.3 Region-Level Image Understanding

For region-level understanding, it is a common practice in computer vision to identify potential regions of interest first and then do the understanding. Object detection [5, 40, 62] tackles the search for potential regions, which are generally accompanied by a simple classification task to understand the region's content. Region captioning [21, 49, 53] provides more descriptive language descriptions in a generative way. Scene graph generation [24, 45, 51] analyzes the relationships between regions by the graph. The VCR [57] dataset presents many region-level reasoning cases in a visual question-answering format. However, traditional solutions have been largely limited to close-set recognition tasks. By harnessing the powerful reasoning capabilities of large language models [8, 47], our GPT4RoI adopts a generative approach to address these tasks and exhibits significant advantages over prior methods.

3 Method: GPT4RoI

The overall framework of GPT4RoI consists of a vision encoder, a projector for image-level features, a region feature extractor, and a large language model (LLM). Compared to previous works [26, 61], the characteristic of GPT4RoI lies in its capacity to generate region-level feature representations by leveraging spatial instructions, as shown in Figure 3.

3.1 Model Architecture

We adopt the ViT-H/14 architecture from CLIP [37] as the vision encoder. Following [26], we use the feature map before the last transformer layer as the representation of the entire image, and then map the image feature embedding to the language space using a single linear layer as projector. Finally, we employ the Vicuna-7B [60] to perform language processing.

To extract region-level features with spatial signal `<region>`, a multi-level image feature pyramid is constructed by selecting four layers from the clip vision encoder. These layers are located at the second-to-last, fifth-to-last, eighth-to-last, and eleventh-to-last positions, respectively. We then add

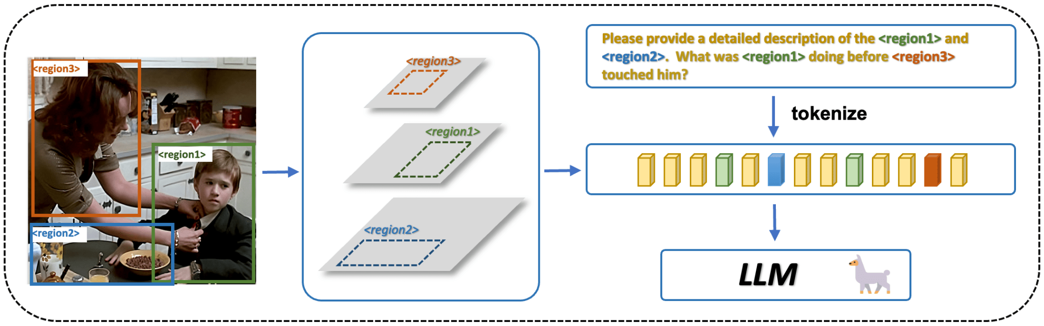

<region3>
<region1>
<region2>
<region3>
<region1>
<region2>
Please provide a detailed description of the <region1> and <region2>. What was <region1> doing before <region3> touched him?
tokenize
LLM

Figure 3: GPT4RoI is an end-to-end vision-language model for processing instructions that contain spatial information, such as `<region>`. During tokenization and conversion to embeddings, the embedding of `<region>` in the instruction is replaced with the RoIAlign results from multi-level image features. Subsequently, all embeddings can be sent to a large language model (LLM) for further processing, similar to pure text instructions. We also utilize the entire image feature to capture global information and omit it in the figure for brevity.

feature coordinates [27] for each level to maintain spatial information. Finally, we adopt lightweight scale shuffle modules [59] to obtain a stronger multi-level feature. We use RoIAlign [19] to extract region-level features with the output size of 14×14, which maintains sufficient detailed information for caption and reasoning. Moreover, all four level features are involved in the RoIAlign operation and fused into a single embedding as the representation of the spatial signal `<region>`.

3.2 Input to LLM

Regardless of the user instruction, we use a prefix prompt "`The <image> provides an overview of the picture.`" The "`<image>`" is a placeholder that will be replaced by a sequence of image features after tokenization and conversion to embeddings, providing the LLM with overall image information.

When the spatial instruction is present by `<region>` in the input text, we replace the corresponding `<region>` embedding with the RoIAlign results of the corresponding bounding box, during tokenization and conversion to embeddings. As shown in Figure 3, when a user asks "`What was <region1> doing before <region3> touched him?`", the instruction is converted to "`What was region1 <region1> doing before region2 <region2> touched him?`" This conversion allows the output to retain the normal string "`region1`" and "`region2`" as references, and replace the placeholder "`<region1>`" and "`<region2>`" to the corresponding RoIAlign results, which provide detailed region information for LLM.

4 Spatial Instruction Tuning

Our model is trained using a next-token prediction loss [11, 26, 58, 61], where the model predicts the next token in a given input text sequence. The loss calculation only involves the response and stop string ### in the sequence. The training details are in Section A.1.

To instruction tuning on region-text pairs, we first transform them into instruction tuning format. We divide the available region-text data into two types and use them in two training stages, separately. In the first stage, we attempt to align region features with word embeddings in language models using simple region-text pairs that contains color, position, or category information. The second stage is designed to handle more complex concepts, such as actions, relationships, and common sense reasoning. Furthermore, we provided diverse instructions for these datasets to stimulate chat-like input in this stage.

Object Detection

In the conversation below, you simply answer the category name based on what you see in the imagery inside a particular region. I will give you only one region each time. Categories containing person, bicycle, car ...

<region1> person

<region2> dog

Referring Expression Comprehension

I will provide you with only one region containing only one object, although there may be other objects present in the image. It is recommended that you describe the object's relative position with respect to other objects in the image and its basic attributes.

<region1> red shirt girl

<region2> guy in black

<region3> right most person blurred

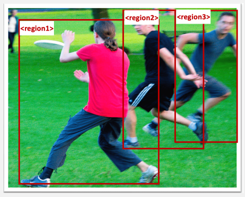


Table 2: The instruction template for Stage 1 training data: For both tasks, we begin by providing a description of the task definition and the expected answer. Then, we concatenate all region-text pairs into a sequence. For detection data, the format is "`<regionx> category_name`". For referring expression comprehension, the format is "`<regionx> description of region`". Only the responses highlighted in red are used to calculate the loss.

4.1 Stage 1: Pre-training

In this stage, we first load the weights of LLaVA [26] after its initial stage of training, which includes a pre-trained vision encoder, a projector for image-level feature, and a LLM. We keep only the region feature extractor trainable and aim to align region features with language embedding by collecting short text and bounding box pairs. These pairs can include normal detection datasets and datasets for referring expression detection with short expressions. The aim is to enable the model to recognize the categories and simple attributes of the region in an image, where the text annotations for the regions are typically short (usually within 5 words). To achieve this goal, we utilize the COCO [25], RefCOCO [55], and RefCOCO+ [55] datasets in this stage.

As shown in Table 2, for COCO detection data, we first explain the task in the prompt and then convert the annotations to a single-word region caption task. For RefCOCO and RefCOCO+, we also give task definitions first and train the model to generate descriptions containing basic attributes of the region. Only the description of the region (in red color) will be used to calculate loss.

After this training stage, GPT4RoI can recognize categories, simple attributes, and positions of regions in images, as shown in Fig. 4.

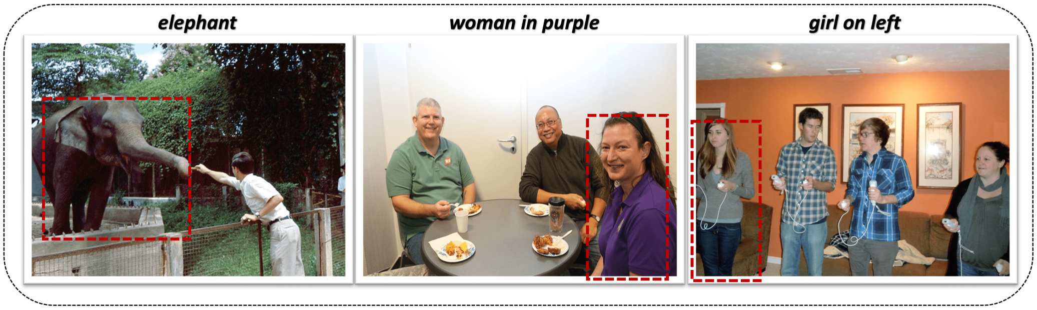


Figure 4: After stage 1 training, GPT4RoI is capable of identifying the category of the region (elephant), simple attributes such as color (purple), and the position of the region (left).

4.2 Stage 2: End-to-end Fine-tuning

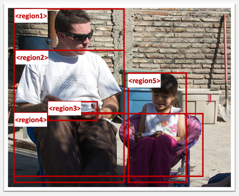
In this stage, we only keep the vision encoder weights fixed and train the region feature extractor, image feature projector, and LLM weights. Our aim is to develop GPT4RoI's capability for complex region-level captions and reasoning based on user instructions, including single region caption, multiple region caption, and reasoning.

We tailor specific instructions for different tasks. For single region caption, we construct from Visual Genome (VG) region caption portion [22] and RefCOCOg [29]. For multiple region caption, Flicker30k [36] is converted to a multiple region caption task where the caption should include all visual elements emphasized by bounding boxes. To simulate user instruction, we create 20 questions for each caption task, separately, as shown in Table 4 and Table 5. The Visual Commonsense Reasoning(VCR) dataset [56] includes naturally diverse instructions. To meet the input format requirements of GPT4RoI, we conducted extensive preprocessing to convert them to three formats, which is detailed in Section A.4 in the supplementary material.

To utilize the LLaVA150k [26] visual instructions, we employ the powerful LVIS detector EVA-02 [15] to extract up to 100 detection boxes per image. These boxes are then concatenated with the user instructions in the format "`<regionx> may feature a class_name`". This data enhancement significantly improves the capability of GPT4RoI for multi-round conversation and reasoning.

Single Region Caption
Question: Can you provide me with a detailed description of the region in the picture marked by <region1>
Answer: A man wearing a light blue t-shirt and jeans with his arms extend

Multiple Region Caption
Question: Can you provide a thorough description of the regions <region1>, <region5> in this image?
Answer: A man is watching a girl draw as he eats
Question: Could you please give me a detailed description of areas <region1>, <region2>, <region3>, <region4>, <region5> ?
Answer: A man in a white shirt with a plate of food sits outside in a folding chair with a little girl who is writing



Single Region Reasoning
Question: How is <region1> feeling ?
Answer: Person at region1 is feeling nervous
Question: Why do you think so?
Answer: His hands are tried behind his back and he is surrounded by nazi guards. He is looking around to escape.

Multi Region Reasoning
Question: Is <region1> happy to be speaking with <region2> and <region3> ?
Answer: No, person at region1 is bothered by the conversation.
Question: What factors influenced your perspective?
Answer: Person at region1 is standing with his hand on his hip in a defensive way.

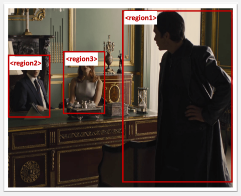


Table 3: Instruction template for Stage 2 training data: During training, we randomly select one question for both single and multiple region caption tasks. For reasoning tasks, we modify the original questions to include a reference for each region so that GPT4RoI can mention them in its answers. Only the response in red color will be used to calculate the loss.

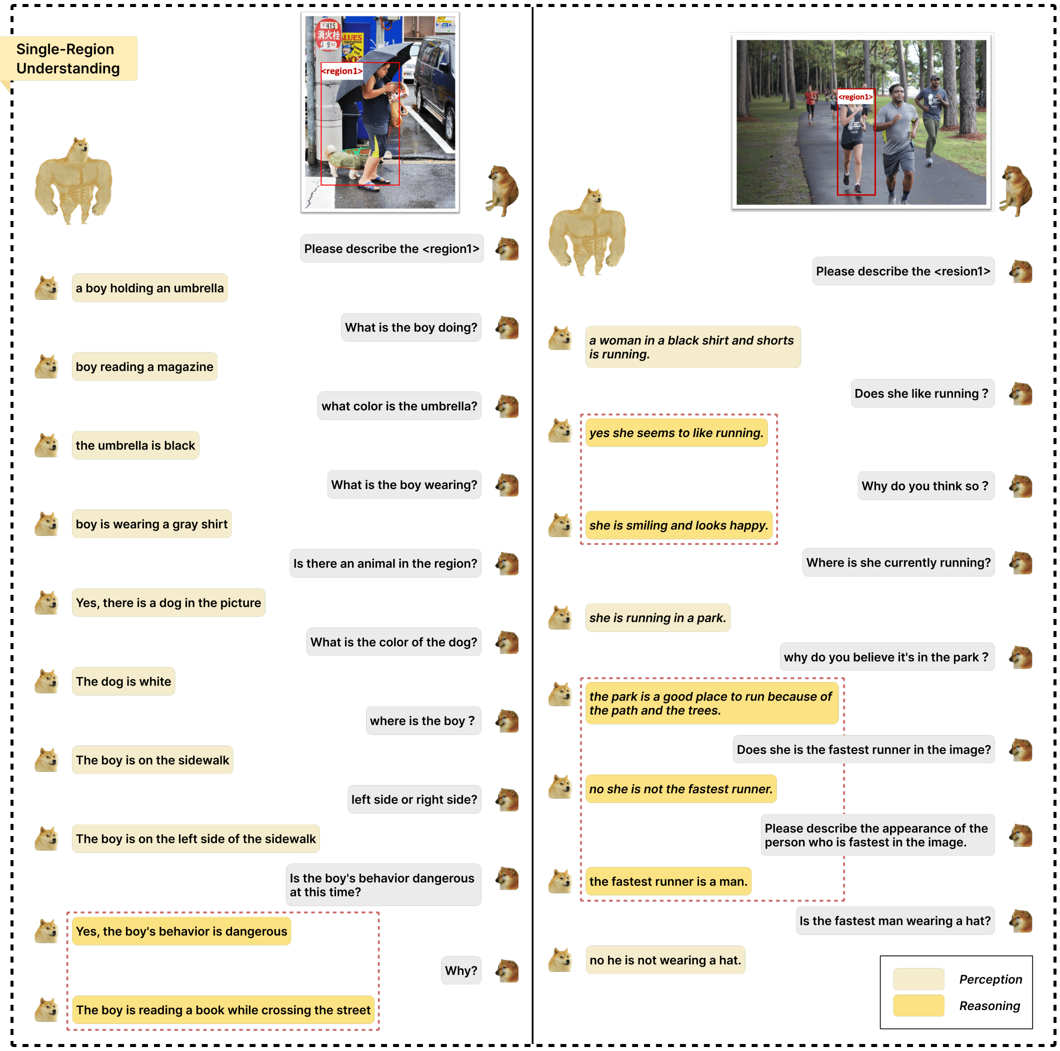
After completing this training stage, GPT4RoI is capable of performing complex region understanding tasks based on user instructions, including single-region and multiple-region captioning and reasoning, as demonstrated in Section 5.

5 Demostrations

5.1 Single Region Understanding

As shown in the first example (Figure 1 left or Figure 5 left), GPT4RoI demonstrates its ability of following user instructions to identify existing objects and their attributes *"a boy holding an umbrella"* within the region, as well as recognizing the action *"reading a magazine"*. Moreover, GPT4RoI has the reasoning ability and gets the conclusion that *"the boy's behavior is dangerous"* based on visual information.

The second example (Figure 1 right) shows that GPT4RoI can recognize the action and facial expression of the runner. Besides, it can also infer that the runner is in a park by analyzing the surrounding trees. More impressively, GPT4RoI can analyze the fastest runner by analyzing the relative positions among runners in the image and identify that the runner within the referred region is not the fastest, while the fastest runner is the man without a hat.



Please describe the <region1>

a boy holding an umbrella

What is the boy doing?

boy reading a magazine

what color is the umbrella?

the umbrella is black

What is the boy wearing?

boy is wearing a gray shirt

Is there an animal in the region?

Yes, there is a dog in the picture

What is the color of the dog?

The dog is white

where is the boy ?

The boy is on the sidewalk

left side or right side?

The boy is on the left side of the sidewalk

Is the boy's behavior dangerous at this time?

Yes, the boy's behavior is dangerous

Why?

The boy is reading a book while crossing the street

Please describe the <resion1>

a woman in a black shirt and shorts is running.

Does she like running ?

yes she seems to like running.

Why do you think so ?

she is smiling and looks happy.

Where is she currently running?

she is running in a park.

why do you believe it's in the park ?

the park is a good place to run because of the path and the trees.

Does she is the fastest runner in the image?

no she is not the fastest runner.

Please describe the appearance of the person who is fastest in the image.

the fastest runner is a man.

Is the fastest man wearing a hat?

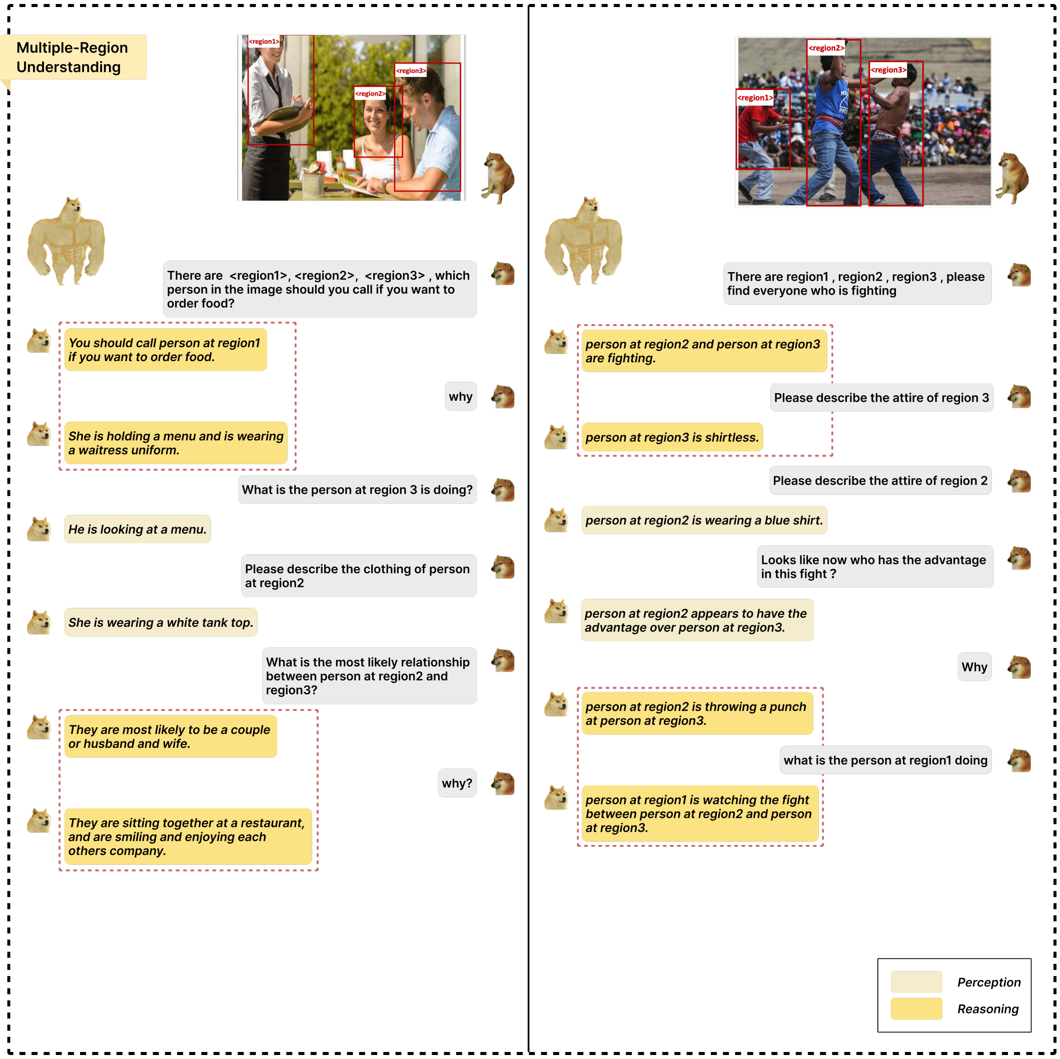
no he is not wearing a hat.

Figure 5: **GPT4RoI for single-region understanding.**

5.2 Multi-Region Understanding

For the first example (Figure 1 right or Figure 6 right), GPT4RoI retains the ability to recognize the objects and their actions within each region, such as the *"white tank top"* in region2 and *"looking at the menu"* in region3. Moreover, GPT4RoI can understand the user's requirements *"the person to call when ordering food"* and identify that *"person at region1"* meets the requirement. Most importantly, GPT4RoI can recognize the relationships between the given regions based on what it sees, such as the likely relationship between region2 and region3 is *"a couple"*.

For the second example (Figure 6 right), GPT4RoI can recognize *"people are region2 and region3 are fighting"* among three people. Surprisingly, GPT4RoI is able to analyze *"who has the advantage in the fight"* by providing reasonable evidence of *"person at region2 is throwing a punch at person at region3"*.



There are <region1>, <region2>, <region3> , which person in the image should you call if you want to order food?

You should call person at region1 if you want to order food.

why

She is holding a menu and is wearing a waitress uniform.

What is the person at region 3 is doing?

He is looking at a menu.

Please describe the clothing of person at region2

She is wearing a white tank top.

What is the most likely relationship between person at region2 and region3?

They are most likely to be a couple or husband and wife.

why?

They are sitting together at a restaurant, and are smiling and enjoying each others company.

There are region1 , region2 , region3 , please find everyone who is fighting

person at region2 and person at region3 are fighting.

Please describe the attire of region 3

person at region3 is shirtless.

Please describe the attire of region 2

person at region2 is wearing a blue shirt.

Looks like now who has the advantage in this fight ?

person at region2 appears to have the advantage over person at region3.

Why

person at region2 is throwing a punch at person at region3.

what is the person at region1 doing

person at region1 is watching the fight between person at region2 and person at region3.

Figure 6: **GPT4RoI for multiple-region understanding.**

6 Discussion

In this paper, we present GPT4RoI, an end-to-end vision-language model that can execute user instructions to achieve region-level image understanding. Our approach employs spatial instruction tuning for the large language model (LLM), where we convert spatial signals (bounding boxes in this study) from user instructions into region features. These region features are then combined with standard language embeddings to create an interleaving sequence to input into the large language model. By utilizing existing open-source region-text pair datasets, we demonstrate that GPT4RoI achieves strong performance in region-level image understanding tasks.

In our exploration, we find GPT4RoI produces failure cases as shown in Section A.5. To further improve the performance, we identify the following potential directions:

- Model architecture. We find that 224 ×224 input image resolution struggles with understanding smaller regions. However, if we switch to a larger resolution, we must consider the potential burden on inference speed from global attention ViT architecture, while the more efficient CNN architecture or sliding window attention has no available pre-trained large-scale vision encoder like CLIP ViT-H/14.
- More region-text pair data. The amount of available region-text pairs is notably smaller than that of image-text pairs, which makes it challenging to sufficiently align region-level features with language models. To tackle this issue, we may try to generate region-level pseudo labels by leveraging off-the-shelf detectors to generate bounding boxes for image-text data.
- Region-level instructions. Although we have generated instructions for each task from existing open-source datasets, users in practical applications may ask various questions about an arbitrary number of regions, and the existing data may not contain satisfactory answers. To tackle this issue, we suggest generating a new batch of spatial instructions through manual labeling or by leveraging ChatGPT or GPT4.
- Interaction mode. Currently, GPT4RoI only supports natural language and bounding box interaction. Incorporating more open-ended interaction modes such as point, scribble, or image-based search could further improve the user interaction experience.

A Supplementary Material

A.1 Training Details

The model is trained on 8 GPUs, each with 80G of memory. During the first training stage, a learning rate of 4e-5 is used with a cosine learning schedule. The batch size is 16 for 2 epochs, with a warm-up iteration set to 3000 and a warm-up ratio of 0.003. The weight decay for all modules was set to 0. During the second training stage, the learning rate is reduced to 2e-5 and the model is trained for 2 epochs. To enable end-to-end fine-tuning of the model, which includes a 7B Vicuna, Fully Sharded Data Parallel (FSDP) is enabled in PyTorch to save memory.

A.2 Instrucion of Single-Region Caption

The instructions for single-region caption are provided in Table 4. We randomly select one as the question in training.

A.3 Instrucion of Multi-Region Caption

The instructions for multi-region caption are provided in Table 5. We randomly select one as the question in training.

A.4 Preprocess of VCR

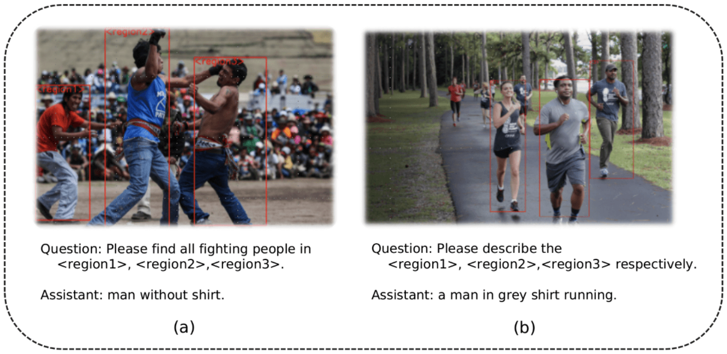
The Visual Commonsense Reasoning(VCR) dataset [57], comprises 290,000 multiple-choice questions obtained from 110,000 movie scenes. Each image in the dataset is annotated with a question that requires common-sense reasoning, along with its corresponding answer and the explanation for the answer. To construct a sequence of questions, we convert the explanation to a follow-up question and format them into a two-round conversation. Table 6 shows an example of the follow-up question that asks for the reasoning behind the answer.

The VCR dataset is valued for its diverse question-answer pairs that require referencing from prior question-answers to perform reasoning. Therefore, it's crucial to assign a reference to each region in the dataset. We accomplish this by starting each conversation with a reference to all regions, e.g., "There are <region1> and <region2> in the image." This approach explicitly references every region, avoiding confusion in future analyses. Additionally, we substitute the corresponding "<regionx>" in the answer with "category_name at regionx" to ensure a plain text output sequence.

A.5 Failure Case Analysis

Due to limited data and instructions, GPT4RoI may fail in several landmark scenarios. We have conducted a thorough analysis and look forward to improving these limitations in future versions.

Instruction obfuscation As shown in Fig. 7.(a), our multiple-region reasoning capability mainly relies on VCR, where we often use sentences that declare <region1>, <region2>, etc. at the beginning of the question. However, when users adopt the less common sentence structure to refer to regions, it can often be confused with region captions that have the highest proportion in the dataset. As shown in Fig. 7.(b), because our data and instructions are mainly generated by rules, our training data does not include content with the "respectively" instruction in multi-region scenarios. This can be resolved by adding specific instructions. In future versions, we aim to develop more diverse instructions, while ensuring data balance.



Question: Please find all fighting people in <region1>, <region2>,<region3>.

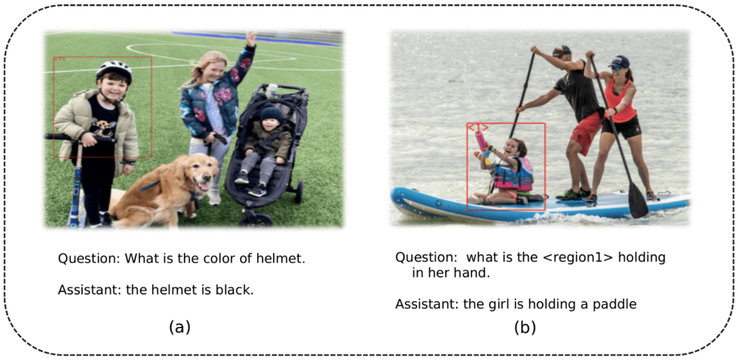
Assistant: man without shirt.

Question: Please describe the <region1>, <region2>,<region3> respectively.

Assistant: a man in grey shirt running.

Figure 7: **GPT4RoI on instruction obfuscation.**

Misidentification of fine-grained information Although GPT4RoI has improved the fine-grained perception ability of images compared to image-level vision language models, the limited amount of region-level data results in insufficient fine-grained alignment within regions. For example, in Fig 8.(a), the model incorrectly identifies the color of the helmet, and in Fig 8.(b), it misidentifies the object in the girl's hand. Both cases generate the corresponding answers based on the most prominent feature within the region. To address this issue, we plan to use semi-supervised methods to create more region-level data.



Question: What is the color of helmet.

Assistant: the helmet is black.

Question: what is the <region1> holding in her hand.

Assistant: the girl is holding a paddle

Figure 8: **GPT4RoI on Misidentification of fine-grained information.**

1. Can you provide me with a detailed description of the region in the picture marked by <region>?
2. I'm curious about the region represented by <region> in the picture. Could you describe it in detail?
3. What can you tell me about the region indicated by <region> in the image?
4. I'd like to know more about the area in the photo labeled <region>. Can you give me a detailed description?
5. Could you describe the region shown as <region> in the picture in great detail?
6. What details can you give me about the region outlined by <region> in the photo?
7. Please provide me with a comprehensive description of the region marked with <region> in the image.
8. Can you give me a detailed account of the region labeled as <region> in the picture?
9. I'm interested in learning more about the region represented by <region> in the photo. Can you describe it in detail?
10. What is the region outlined by <region> in the picture like? Could you give me a detailed description, please?
11. Can you provide me with a detailed description of the region in the picture marked by <region>, please?
12. I'm curious about the region represented by <region> in the picture. Could you describe it in detail, please?
13. What can you tell me about the region indicated by <region> in the image, exactly?
14. I'd like to know more about the area in the photo labeled <region>, please. Can you give me a detailed description?
15. Could you describe the region shown as <region> in the picture in great detail, please?
16. What details can you give me about the region outlined by <region> in the photo, please?
17. Please provide me with a comprehensive description of the region marked with <region> in the image, please.
18. Can you give me a detailed account of the region labeled as <region> in the picture, please?
19. I'm interested in learning more about the region represented by <region> in the photo. Can you describe it in detail, please?
20. What is the region outlined by <region> in the picture like, please? Could you give me a detailed description?

Table 4: A list of instructions for single-region caption.

1. Could you please give me a detailed description of these areas [<region1>, <region2>, ...]?
2. Can you provide a thorough description of the regions [<region1>, <region2>, ...] in this image?
3. Please describe in detail the contents of the boxed areas [<region1>, <region2>, ...].
4. Could you give a comprehensive explanation of what can be found within [<region1>, <region2>, ...] in the picture?
5. Could you give me an elaborate explanation of the [<region1>, <region2>, ...] regions in this picture?
6. Can you provide a comprehensive description of the areas identified by [<region1>, <region2>, ...] in this photo?
7. Help me understand the specific locations labeled [<region1>, <region2>, ...] in this picture in detail, please.
8. What is the detailed information about the areas marked by [<region1>, <region2>, ...] in this image?
9. Could you provide me with a detailed analysis of the regions designated [<region1>, <region2>, ...] in this photo?
10. What are the specific features of the areas marked [<region1>, <region2>, ...] in this picture that you can describe in detail?
11. Could you elaborate on the regions identified by [<region1>, <region2>, ...] in this image?
12. What can you tell me about the areas labeled [<region1>, <region2>, ...] in this picture?
13. Can you provide a thorough analysis of the specific locations designated [<region1>, <region2>, ...] in this photo?
14. I am interested in learning more about the regions marked [<region1>, <region2>, ...] in this image. Can you provide me with more information?
15. Could you please provide a detailed description of the areas identified by [<region1>, <region2>, ...] in this photo?
16. What is the significance of the regions labeled [<region1>, <region2>, ...] in this picture?
17. I would like to know more about the specific locations designated [<region1>, <region2>, ...] in this image. Can you provide me with more information?
18. Can you provide a detailed breakdown of the regions marked [<region1>, <region2>, ...] in this photo?
19. What specific features can you tell me about the areas identified by [<region1>, <region2>, ...] in this picture?
20. Could you please provide a comprehensive explanation of the locations labeled [<region1>, <region2>, ...] in this image?

Table 5: A list of instructions for multiple-region caption.

1. Why?
2. What’s the rationale for your decision
3. What led you to that conclusion?
4. What’s the reasoning behind your opinion?
5. Can you explain the basis for your thinking?
6. What factors influenced your perspective?
7. How did you arrive at that perspective?
8. What evidence supports your viewpoint?
9. What’s the logic behind your argument?
10. Can you provide some context for your opinion?
11. What’s the basis for your assertion?
12. What experiences have shaped your perspective?
13. What assumptions underlie your reasoning?
14. What’s the foundation of your assertion?
15. What’s the source of your reasoning?
16. What’s the motivation behind your decision?
17. What’s the impetus for your belief?
18. What’s the driving force behind your conclusion?
19. What’s your reasoning?
20. What makes you say that?
21. What’s the story behind that?
22. What’s your thought process?
23. What’s the deal with that?
24. What’s the logic behind it?
25. What’s the real deal here?
26. What’s the reason behind it?
27. What’s the rationale for your opinion?
28. What’s the background to that?
29. What’s the evidence that supports your view?
30. What’s the explanation for that?

Table 6: A list of instructions for the second round chat in VCR.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 4

[2] Anthropic. Claude. `https://www.anthropic.com/index/introducing-claude`, 2023. 2

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 4

[4] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023. 4

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 4

[6] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021. 2

[7] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 2

[8] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. 2, 4

[9] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022. 4

[10] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 4

[11] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 2, 3, 4, 5

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 4

[13] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023. 2

[14] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335, 2022. 2

[15] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *arXiv preprint arXiv:2303.11331*, 2023. 7

[16] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023. 2

[17] Tao Gong, Chengqi Lyu, Shilong Zhang, Yudong Wang, Miao Zheng, Qian Zhao, Kuikun Liu, Wenwei Zhang, Ping Luo, and Kai Chen. Multimodal-gpt: A vision and language model for dialogue with humans, 2023. 4

[18] Google. Bard. `https://bard.google.com/`, 2023. 2

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 3, 5

[20] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023. 2

[21] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning, 2015. 4

[22] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017. 2, 3, 4, 7

[23] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 4

[24] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions, 2017. 4

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 3, 6

[26] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2, 3, 4, 5, 6, 7

[27] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution, 2018. 5

[28] Zhaoyang Liu, Yinan He, Wenhai Wang, Weiyun Wang, Yi Wang, Shoufa Chen, Qinglong Zhang, Yang Yang, Qingyun Li, Jiashuo Yu, et al. Internchat: Solving vision-centric tasks by interacting with chatbots beyond language. *arXiv preprint arXiv:2305.05662*, 2023. 3, 4

[29] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016. 3, 7

[30] Yao Mu, Qinglong Zhang, Mengkang Hu, Wenhai Wang, Mingyu Ding, Jun Jin, Bin Wang, Jifeng Dai, Yu Qiao, and Ping Luo. Embodiedgpt: Vision-language pre-training via embodied chain of thought. *arXiv preprint arXiv:2305.15021*, 2023. 4

[31] OpenAI. Chatgpt. `https://openai.com/blog/chatgpt`, 2022. 2, 4

[32] OpenAI. Gpt-4 technical report, 2023. 2

[33] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011. 2

[34] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022. 4

[35] Renjie Pi, Jiahui Gao, Shizhe Diao, Rui Pan, Hanze Dong, Jipeng Zhang, Lewei Yao, Jianhua Han, Hang Xu, and Lingpeng Kong Tong Zhang. Detgpt: Detect what you need via reasoning. *arXiv preprint arXiv:2305.14167*, 2023. 3

[36] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 3, 7

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 4

[38] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI*, 2018. 4

[39] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020. 4

[40] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015. 4

[41] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 2

[42] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 2

[43] Tianxiang Sun and Qiu Xipeng. Moss. `https://github.com/OpenLMLab/MOSS`, 2022. 2

[44] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. *arXiv preprint arXiv:2303.08128*, 2023. 4

[45] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts, 2018. 4

[46] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023. 2

[47] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2, 4

[48] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023. 4

[49] Jialian Wu, Jianfeng Wang, Zhengyuan Yang, Zhe Gan, Zicheng Liu, Junsong Yuan, and Lijuan Wang. Grit: A generative region-to-text transformer for object understanding, 2022. 4

[50] Peng Xu, Wenqi Shao, Kaipeng Zhang, Peng Gao, Shuo Liu, Meng Lei, Fanqing Meng, Siyuan Huang, Yu Qiao, and Ping Luo. Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models, 2023. 4

[51] Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation, 2022. 4

[52] Jiange Yang, Wenhui Tan, Chuhao Jin, Bei Liu, Jianlong Fu, Ruihua Song, and Limin Wang. Pave the way to grasp anything: Transferring foundation models for universal pick-place robots. *arXiv preprint arXiv:2306.05716*, 2023. 4

[53] Linjie Yang, Kevin Tang, Jianchao Yang, and Li-Jia Li. Dense captioning with joint inference and visual context. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. 4

[54] Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023. 2, 3, 4

[55] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 69–85. Springer, 2016. 3, 6

[56] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6720–6731, 2019. 2, 3, 4, 7

[57] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 4, 10

[58] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 2, 3, 4, 5

[59] Shilong Zhang, Xinjiang Wang, Jiaqi Wang, Jiangmiao Pang, Chengqi Lyu, Wenwei Zhang, Ping Luo, and Kai Chen. Dense distinct query for end-to-end object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7329–7338, June 2023. 5

[60] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. 4

[61] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2, 3, 4, 5

[62] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 3, 4