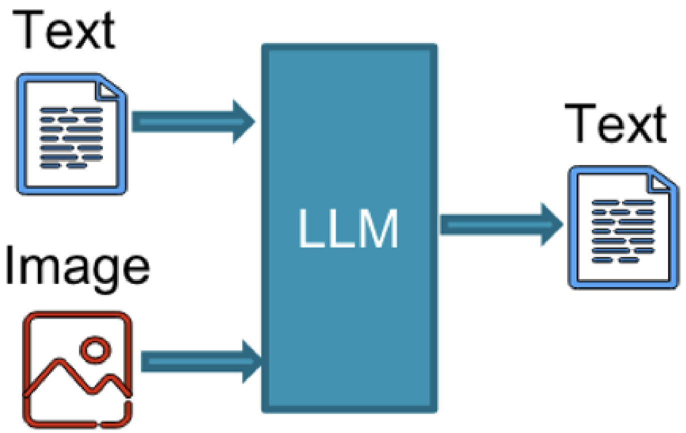
LMMs with Fine-Grained Grounding Capabilities

Haotian Zhang
Apple AI/ML
06/17/2023

Global Image-perceiving MLLMs

- Flamingo
- Kosmos-1
- BLIP-2
- Mini-GPT4, LLaVA
- InstructBLIP, Otter
- MM1, IDEFICS, SPHINX
- GPT4V, Gemini
- ...



☞ *Encode input images with external image encoders, generating LLM-understanding visual feature, which is then fed into the LLM. LLM then interprets the input images based on the input text instructions and produces a textual response.*

[1] Flamingo: a Visual Language Model for Few-Shot Learning. 2022
[2] Language Is Not All You Need: Aligning Perception with Language Models. 2023
[3] BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. 2023
[4] MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models. 2024

Global Image-perceiving MLLMs

What is this symbol about?

The image appears to be a promotional or informational graphic about an event or conference. It features a stylized skyline of Seattle, Washington, which includes the Space Needle and other notable buildings. The colors are a monochromatic blue palette, which gives it a professional and modern look.

The text "CVPR" stands for the Conference on Computer Vision and Pattern Recognition, which is a major international conference on computer vision and pattern recognition. It is organized by the IEEE Computer Society and held annually.

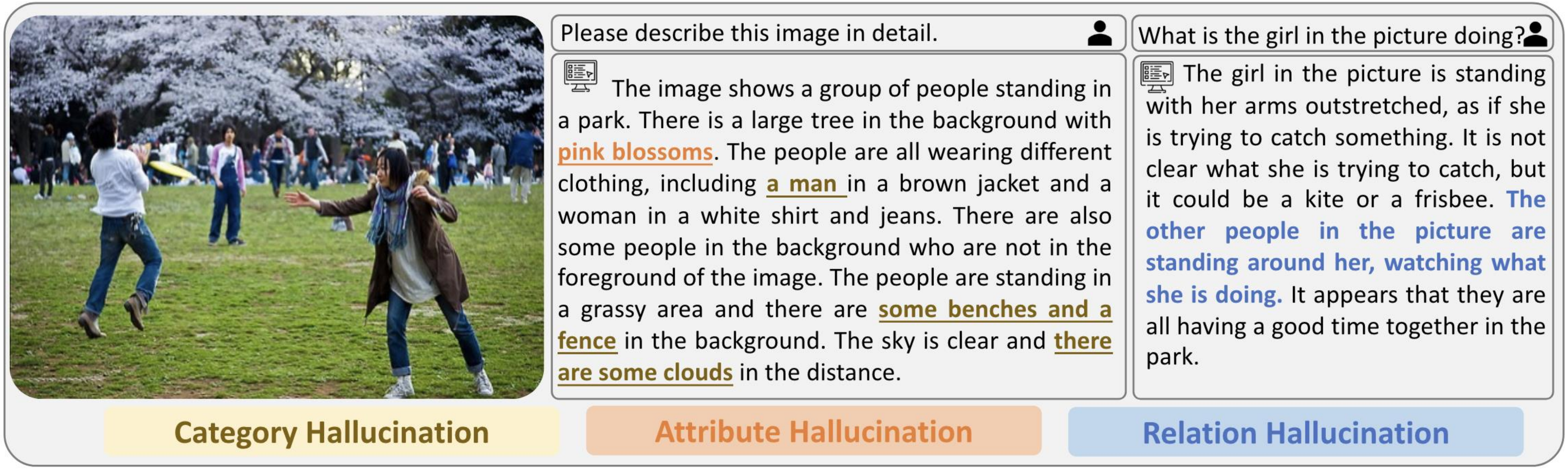
The dates "JUNE 17-21, 2024" indicate when the CVPR conference is scheduled for the upcoming year. The location "SEATTLE, WA" specifies where the event will take place, which is Seattle, Washington.

This graphic is likely used to advertise the CVPR 2024 conference to the computer vision and pattern recognition community, inviting them to submit papers, attend the conference, or exhibit their work.

[1] MM1: Methods, Analysis & Insights from Multimodal LLM Pre-training. 2024.

Are our current models perfect?

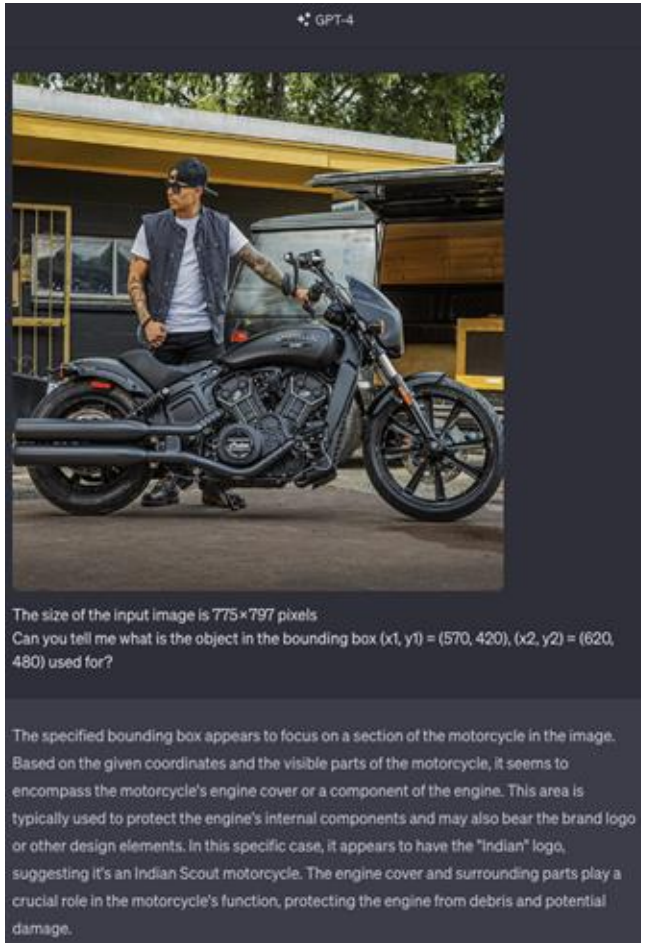
Case 1: The models sometimes produce “object hallucination”…

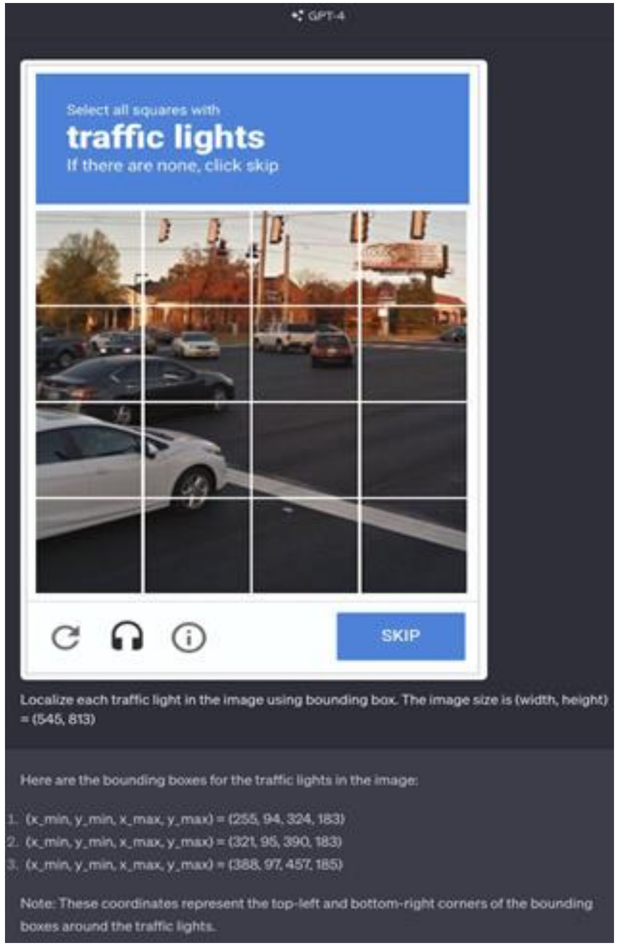


[1] Hallucination of Multimodal Large Language Models: A Survey. 2024.

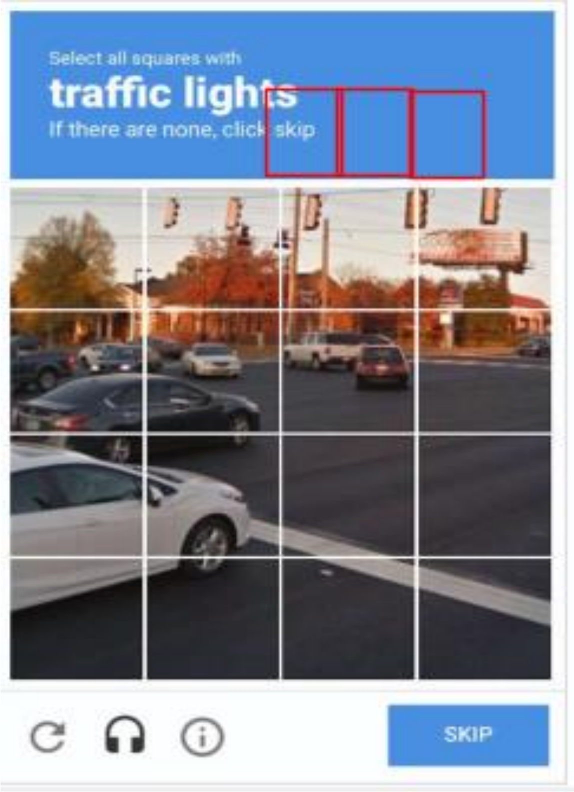
Are our current models perfect?

Case 2: The models are not good at “spatial understanding”…



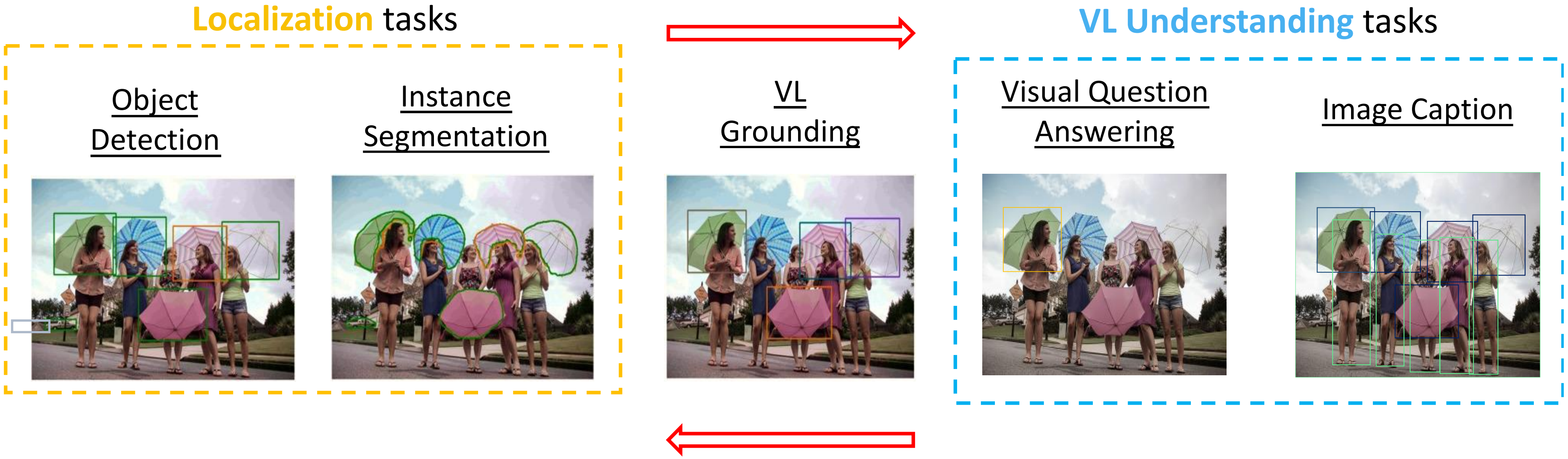


Visualized Output:



How “vision” play the role?

To reduce the previous issues, we think ”visual grounding” may be able to help



- General Purpose Vision Model: Localization & VL Understanding

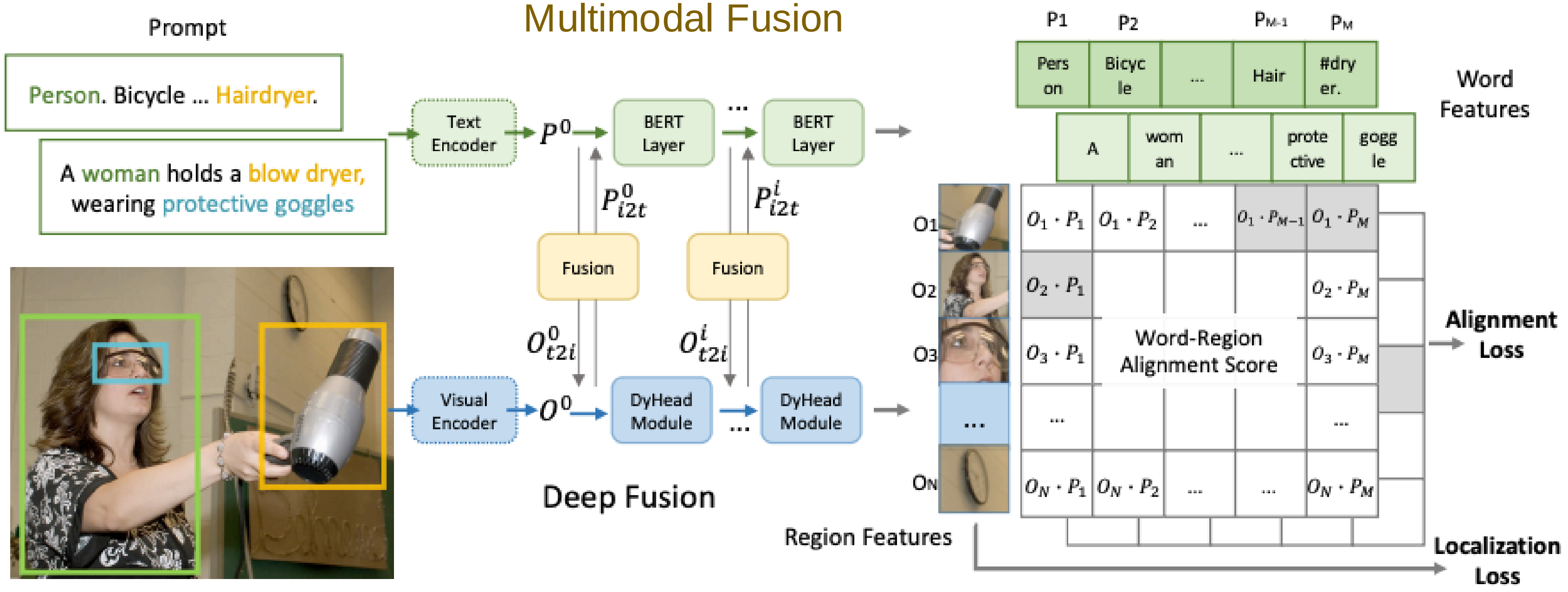
  1) Localization Tasks: Vision-only, fine-grained outputs.
  2) VL Understanding Tasks: Two modalities, high-level semantic outputs.

Traditional Visual Grounding Models

GLIP & GLIP v2: A Unified Framework for Detection and Grounding

Multimodal Fusion

Dual Encoders

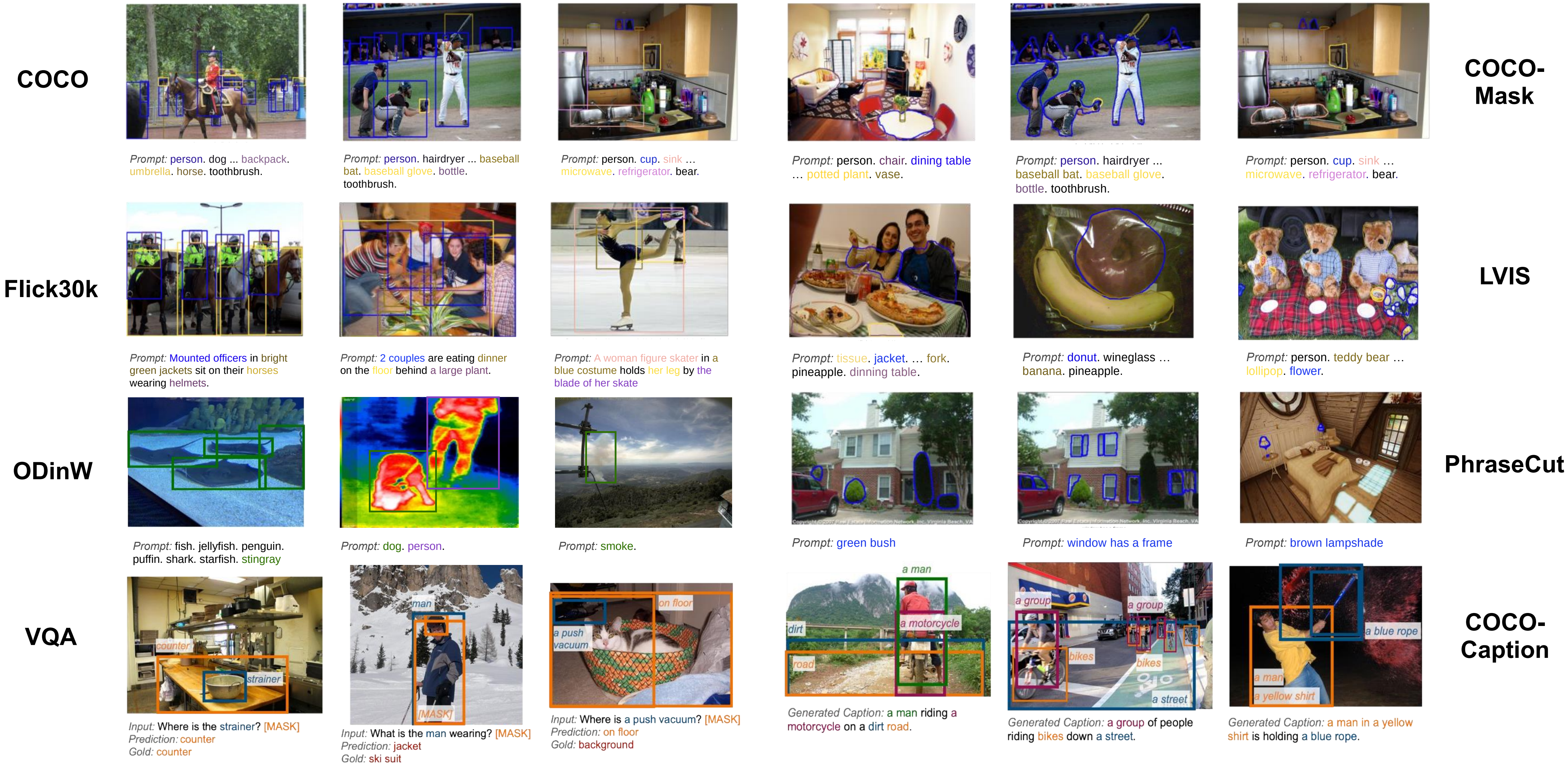


Contrastive Image-Text Alignment

- A model for learning object-level, language-aware, and semantic-rich visual representations.
- GLIP can learn from both detection and grounding data to improve both tasks.

[1] GLIP: Grounded Language-Image Pre-training. 2022.
[2] GLIPv2: Unifying Localization and Vision-Language Understanding. 2023

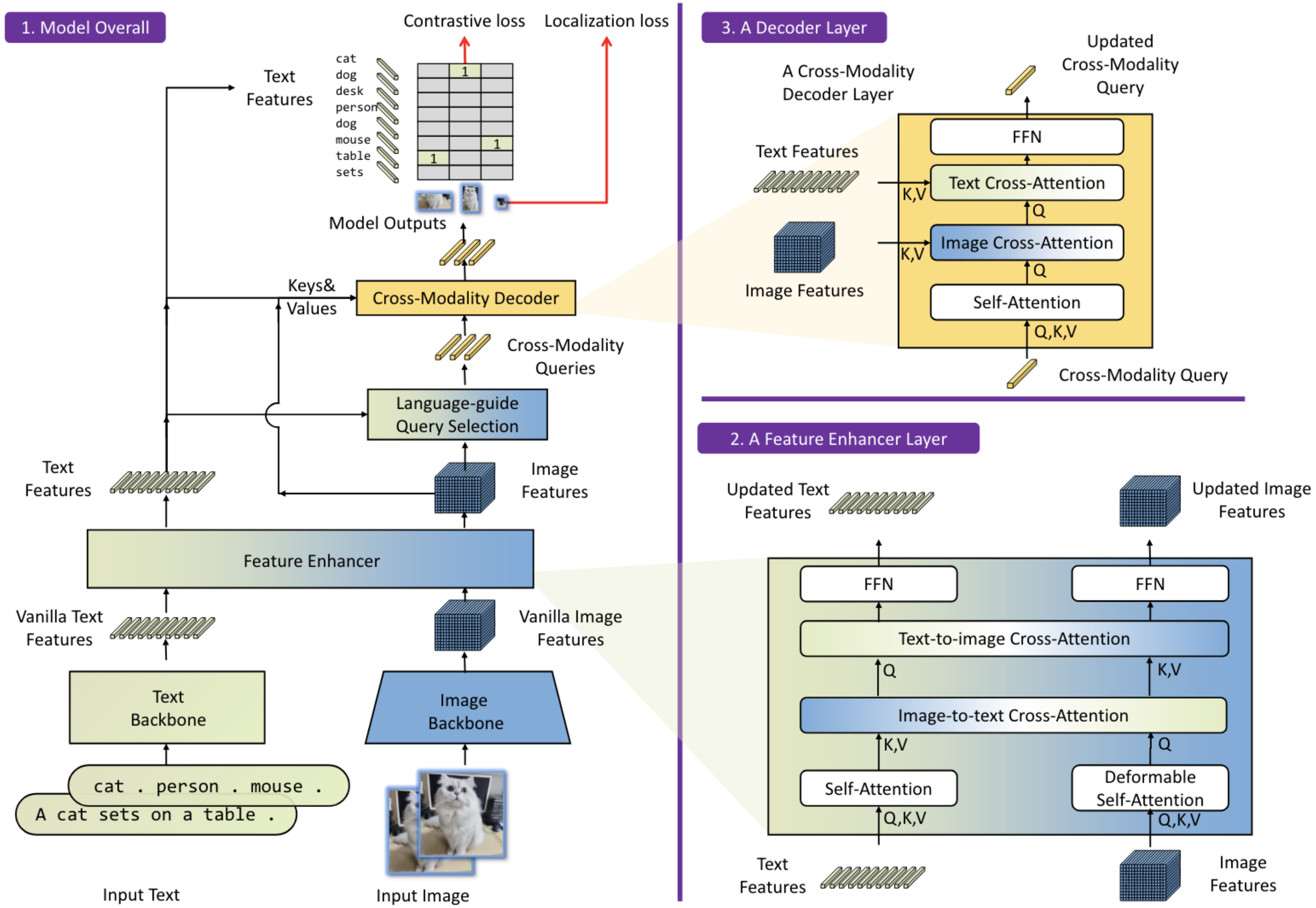
Traditional Visual Grounding Models



Traditional Visual Grounding Models

Grounding-DINO: An Improved Grounding Framework on Localization & Understanding

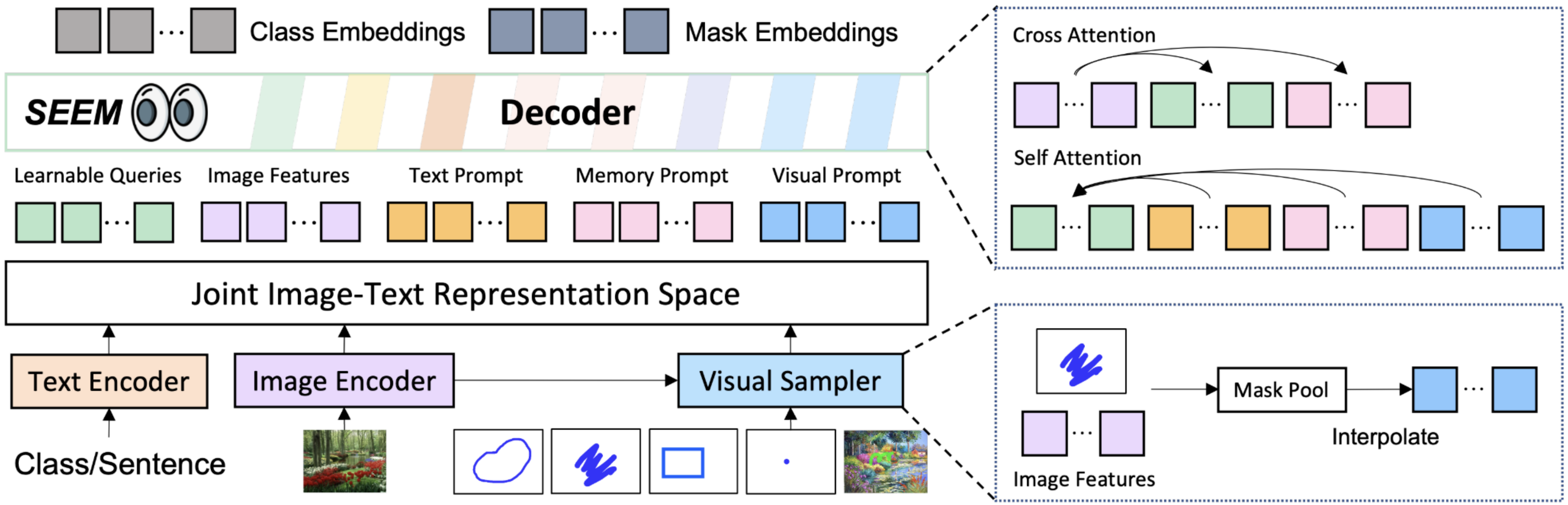
- Using DINO (transformer architecture) as a better detector.
- More and Better image-text feature fusion.
- Lead to superior performance.



[1] Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. 2023.

Traditional Visual Grounding Models

SEEM: A Single Generalist Approach for Pixel Level Understanding



- Introduce visual prompts to handle non-textual inputs, e.g., points/boxes/scribbles.
- Interact with user in multi-rounds, thanks to the memory prompt.
- Give a semantic label to any predicted segmentation masks (instead of boxes)

[1] *SEEM: Segment Everything Everywhere All at Once. 2023.*

Traditional Visual Grounding Models

Till 2023, trend became…

- An external detector
- Stronger encoders (esp., text encoder)
- Unified inputs / outputs

OpenSeeD

X-Decoder
DETR+CLIP+caption

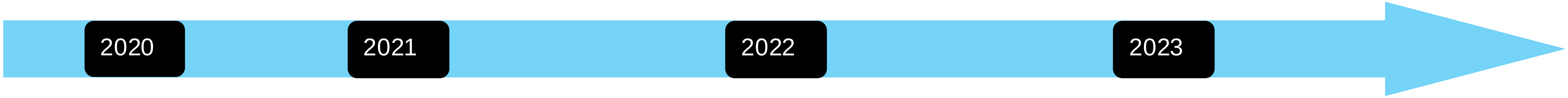
SEEM
Decoder +
Referring

ViLD
RCNN+CLIP

RegionCLIP
RCNN+finetune CLIP

OV-DETR
DETR+CLIP

DetCLIP
RCNN+CLIP+large data



MDETR
Grounding+DETR

UniTab
Grounding + Pix2Seq

GLIP
Grounding+ATSS
Extend to Detection

Grounding DINO
Grounding+DINO

GLIPv2
Unified Grounding+
VL Understanding

Time to Think

What we have covered so far…

- Typical VLMs/MLLMs suffer from “object hallucination” & “spatial understanding”
- Traditional “visual grounding” concept may alleviate the above issues.
- Several existing works & trends.

With the breakthrough of the LLMs…

- What other things may “grounding” LLMs can bring us?
- How to better integrate “grounding” into LLMs?

Why is MLLM + Spatial Understanding important?

New Functions:

1. Users to refer to specific regions/objects and ask model's help.
2. Model to localize/ground particular objects in response for better helping users.

Better Model:

1. Less Hallucination
2. More Trustworthy
3. Open-Vocabulary Concept Grounding

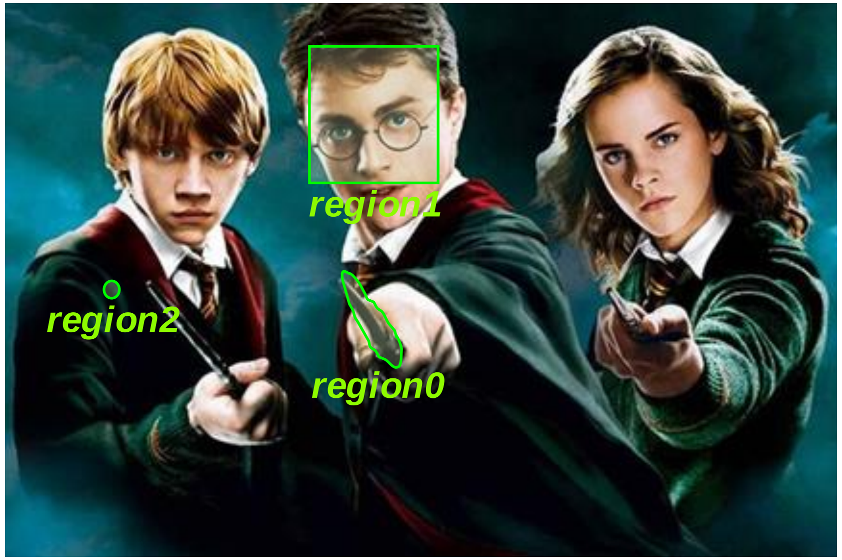
New Applications:

1. Phone/VR/AR Intelligence
2. 3D Embodiment
3. Medical Assistant
4. …

Problem Definition

Spatial Understanding can be reflected in two types of tasks:

1. Referring
   Input: Image + Text Instruction + Region
   Model is required to understand the referred regions and respond to the instruction.



*Q: What is in **region0**? What is it used for?*

*Q: Which movie characters are in **region1** and **region2**? And what is their relationship?*

Problem Definition

Spatial Understanding can be reflected in two types of tasks:

2. Grounding

Output: Text Response + Region

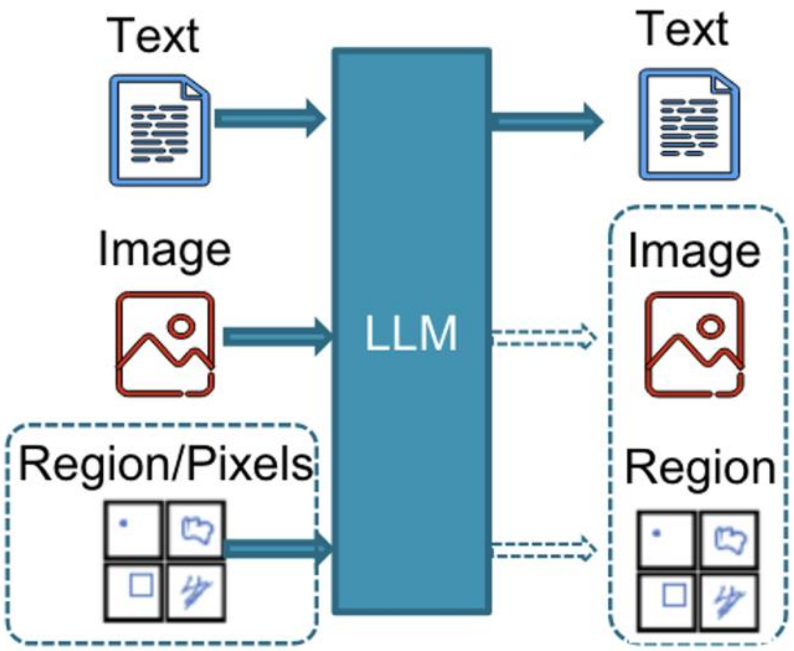
Model is required to localize the objects in image when mentioning them in response

Q: How to make a sandwich with available ingredients in the image? And where are they?

Fine-grained Region-Level MLLMs

Besides the global Image-perceiving MLLMs,

- GPT4RoI
- Next-Chat
- Kosmos-2
- Shikra
- DetGPT
- Ferret
- MiniGPT-v2
- ...

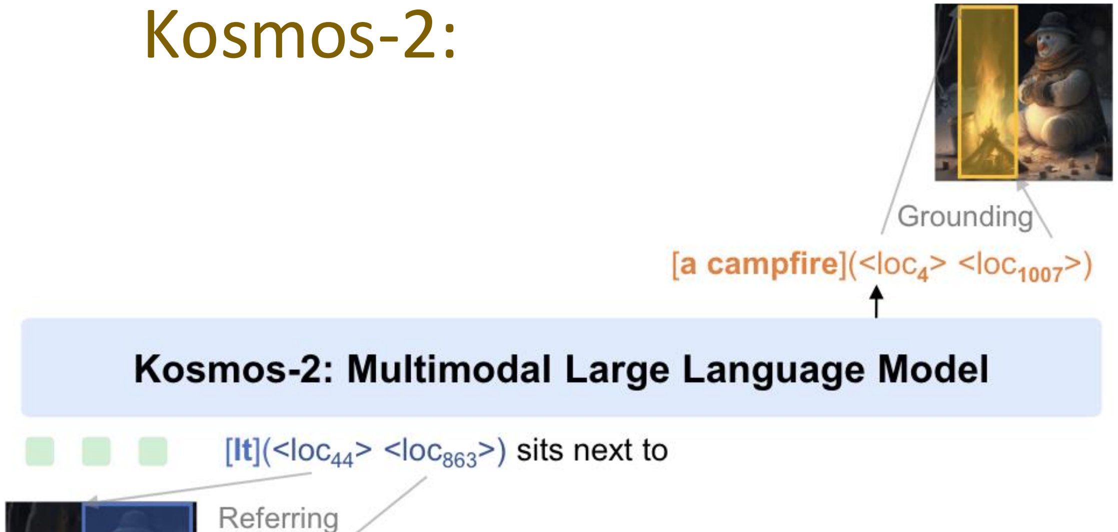


☞ *Users input an image (potentially specify a region), and the LLM outputs contents based on its understanding., grounding the visual contents to specific regions of the images.*

[1] GPT4RoI: Instruction Tuning Large Language Model on Region-of-Interest. 2023
[2] NExT-Chat: An LMM for Chat, Detection and Segmentation. 2023
[3] MiniGPT-v2: large language model as a unified interface for vision-language multi-task learning. 2023
[4] Osprey: Pixel Understanding with Visual Instruction Tuning. 2023
[5] GLaMM: Pixel Grounding Large Multimodal Model. 2023
[6] Kosmos-2: Grounding Multimodal Large Language Models to the World. 2023
[7] DetGPT: Detect What You Need via Reasoning. 2023
[8] PixelLM: Pixel Reasoning with Large Multimodal Model. 2023
[9] Lisa: Reasoning segmentation via large language model. 2023
[10] Shikra: Unleashing Multimodal LLM's Referential Dialogue Magic. 2023

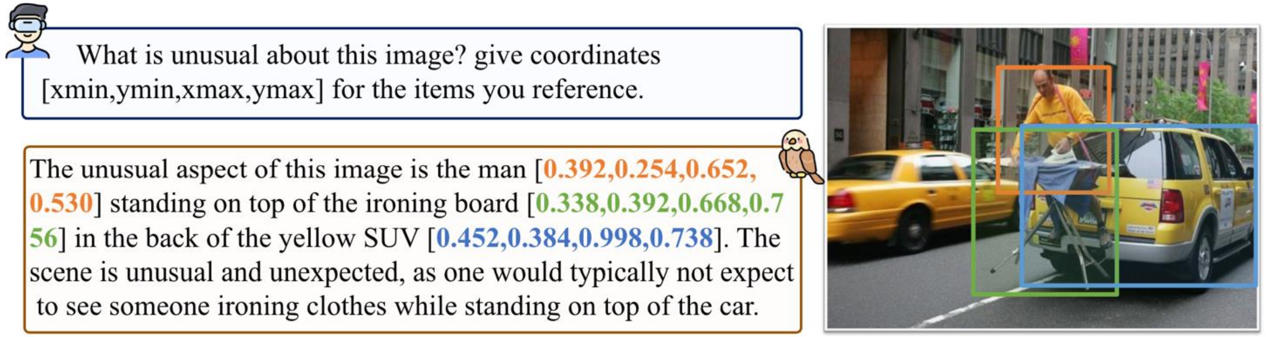
Fine-grained Region-Level MLLMs

Kosmos-2:



- Introduce a sequence of location tokens, in addition to the text tokens
- Construct a web-scale grounded image-text pairs (~90M)

Shikra:



Dataset	Split	Vocab.	Numerical
RefCOCO	val	81.03	81.47
	test-A	86.94	87.40
	test-B	70.91	73.25
RefCOCO+	val	72.32	74.30
	test-A	81.78	83.29
	test-B	59.95	63.08
RefCOCOg	val-u	72.81	75.69
	test-u	73.78	75.52

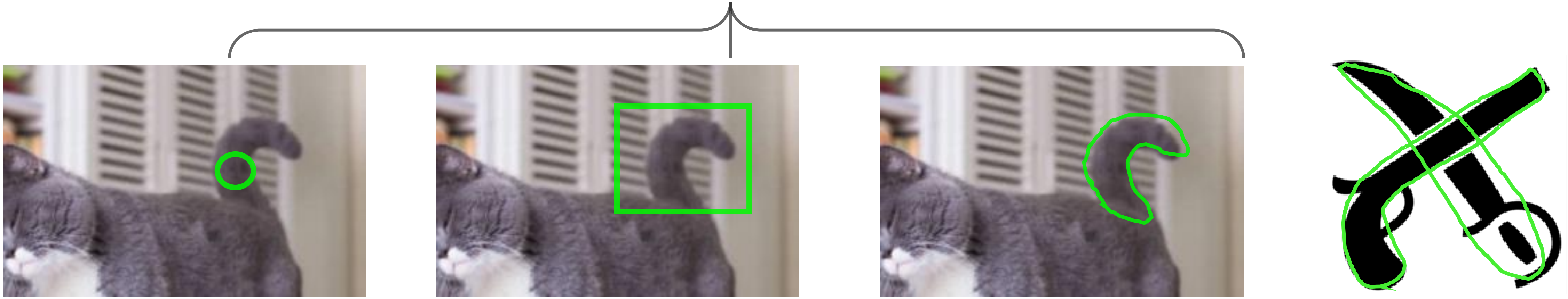
- Tend to use numerical numbers instead of location tokens.
- A more natural representation.

[1] Kosmos-2: Grounding Multimodal Large Language Models to the World. 2023.
[2] Shikra: Unleashing Multimodal LLM's Referential Dialogue Magic. 2023.

Ferret: a MLLM w/ Spatial Understanding

Hybrid Region Representation

Region Name + Discrete Coordinate + Continuous Feature



Point

Box

Free-form Shape
(Sketch, Scribble, polygons)

[1] Ferret: Refer and Ground Anything Anywhere at Any Granularity. 2023.

Hybrid Region Representation

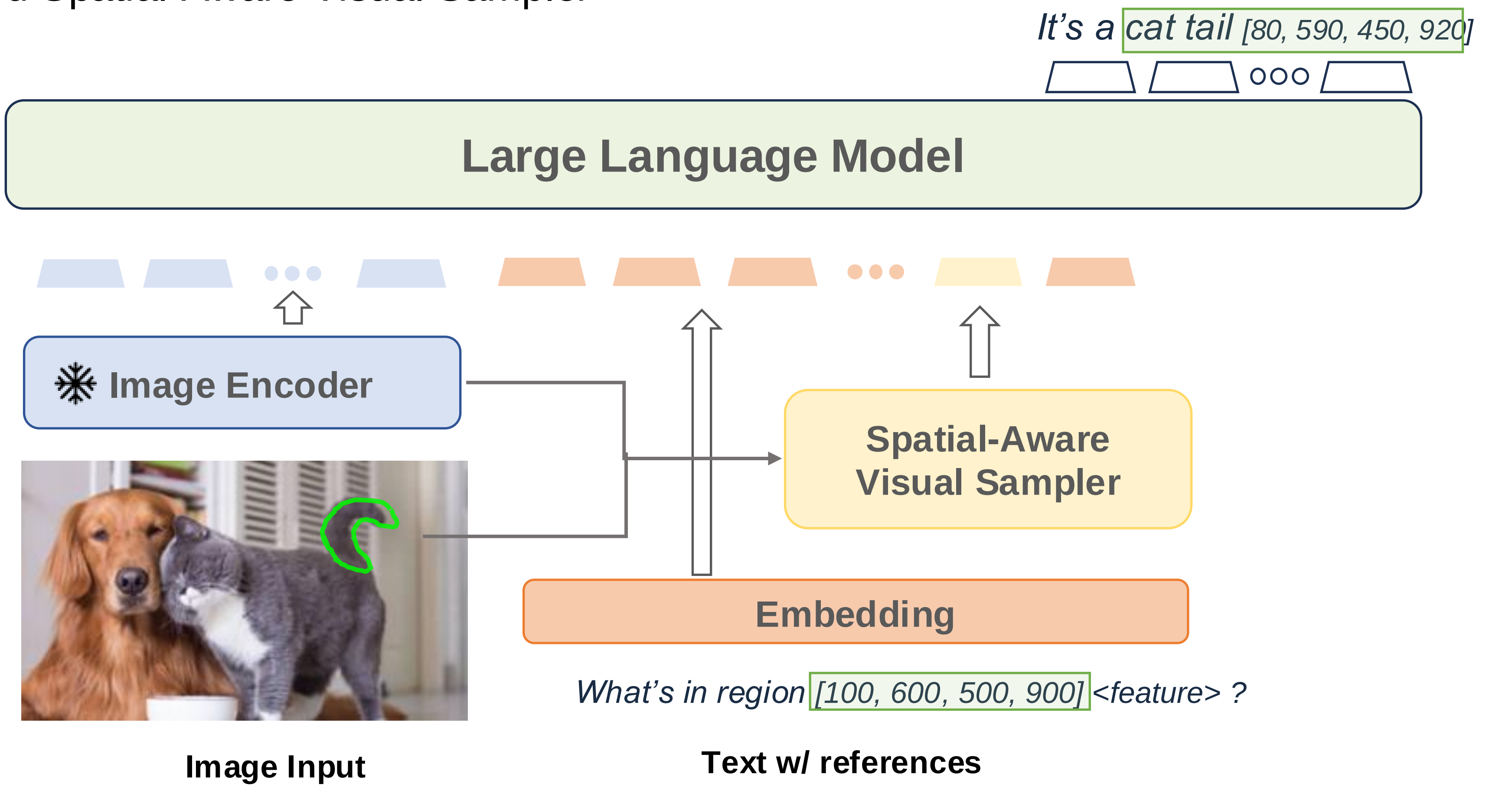
- **Discrete Coordinates**
  - Point: [x, y] (center point)
    Box and Free-form Shape: [x1, y1, x2, y2] (top-left and bottom-right points)
  - Tokenize them by LLM tokenizer.

- **Continuous Visual Features**.
  - Introduce a Visual Sampler module to extract and summarize visual features of **referred regions (point -> circle)** into a single feature vector.

- Examples of data:
  Input:
  What is in region [100, 600, 500, 900] <feature>?
  Output:
  It's a box of egg [100, 600, 500, 900].

Ferret Model Structure

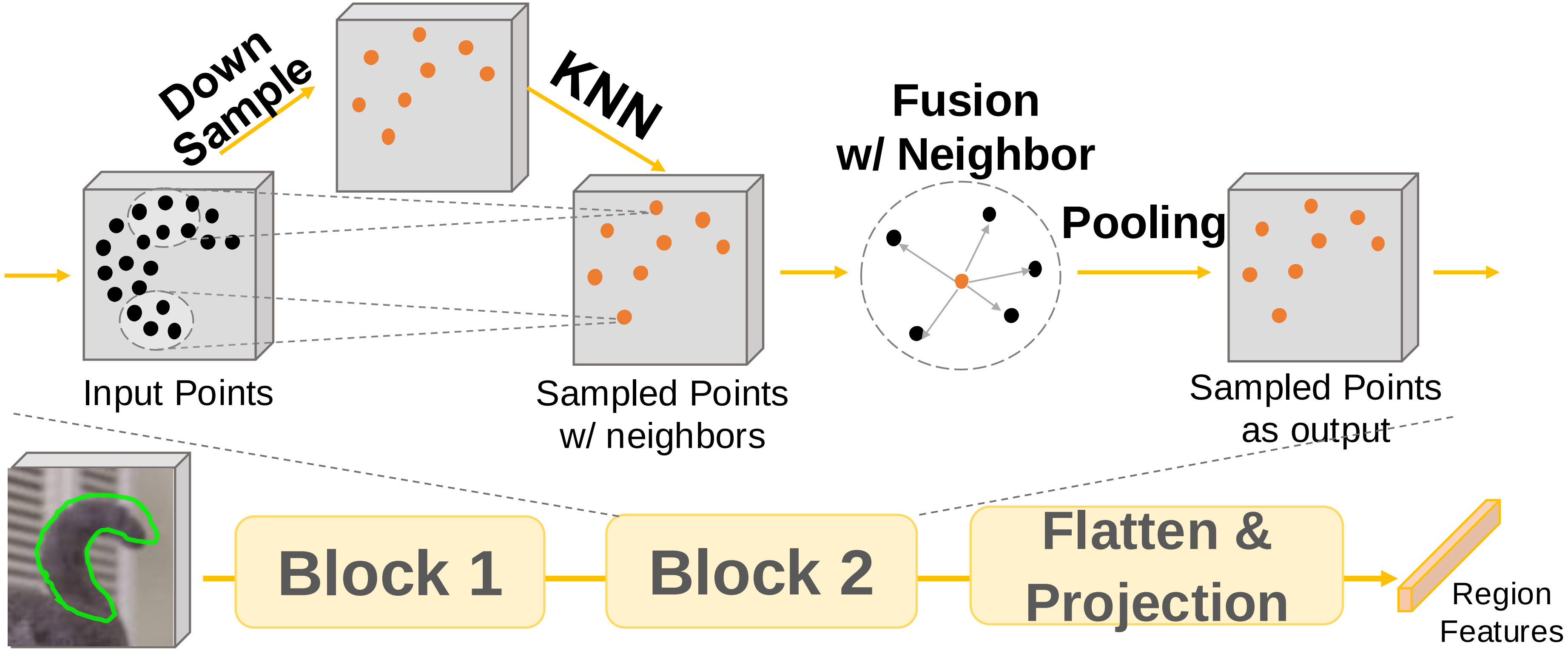
- Model:
  - Image Encoder: CLIP-ViT-L/14
  - LLM: Vicuna-V1.3
  - Proposed Spatial-Aware Visual Sampler
- Optimization:
  - Next Token Prediction.
  - Fix Image Encoder, Update Others.



Ferret: a MLLM w/ Spatial Understanding

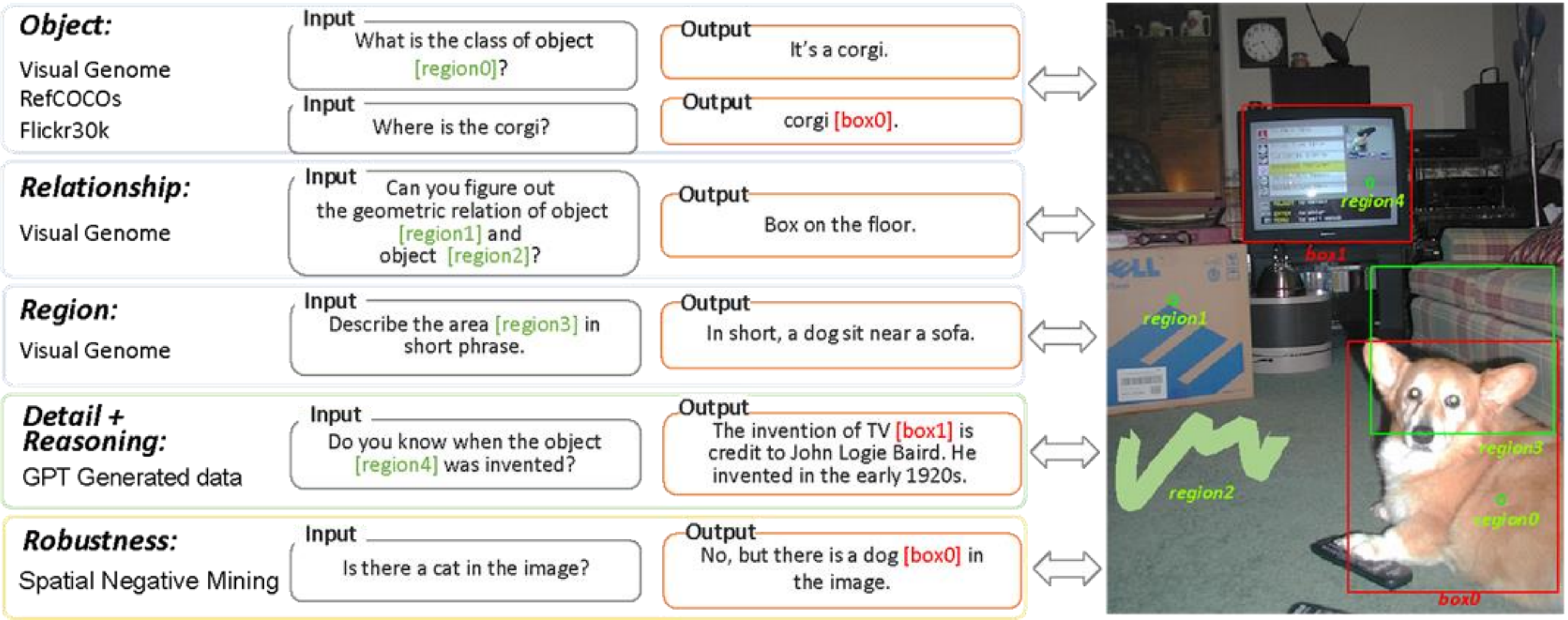
Spatial-aware Visual Sampler

- Sample 512 points inside the region from feature maps.
- Go through 2 blocks. Inside each one,
  - Down-sample the number of points.
  - Find K-Nearest Neighbors
  - Fuse neighbor features and then pooling
- Flatten final 32 points and linearly project their features to LLM's embedding space.



GRIT: A Ground-and-Refer Instruction-Tuning Dataset

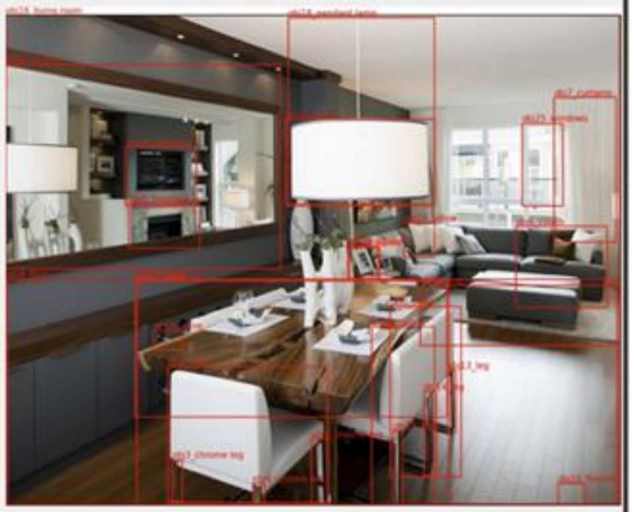
- Hierarchical; Unified Format, Instruction-Following; Robust.
- Dataset Size: 1.1M



Data Collection – GPT-Assisted Instructions

A few-shot Example:

Objects
Object 0: chair at [0.596, 0.637, 0.698, 0.997].
...
Relationships
Object 11 : frame → with → Object 9 : photo
...
Region Descriptions
Region Description at [0.560, 0.466, 0.600, 0.529] :
a white picture frame with a black and white photo on it.
...
Global Caption
White chairs sit around a polished wood dining table while
a sectional soft sits in the background.
...



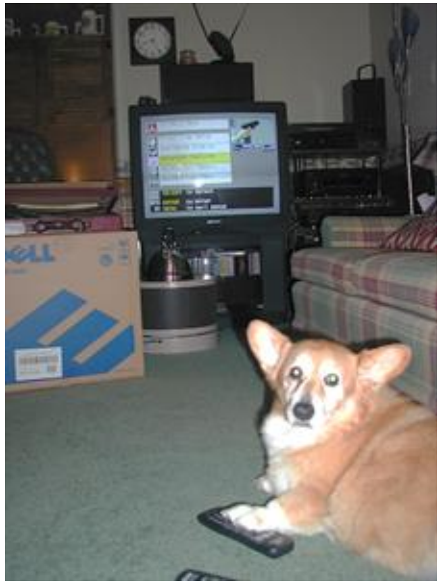
Conversation
Question:
What is the object [0.212, 0.539, 0.718, 0.820] made of?
===
Answer:
The dining table over there is crafted from a tree trunk.
===
Question:
Where would be a good place to have my dinner? Point out the locations in image.
===
Answer:
I suggest having dinner at the table [0.214, 0.541, 0.720, 0.997], possibly sitting on the white chair [0.264, 0.717, 0.524, 0.995]. Also, enjoy the wine [0.242, 0.644, 0.288, 0.682].
===
Question:
In terms of comfort for resting, which object is better - the one [0.830, 0.431, 0.978, 0.596] or the one [0.264, 0.717, 0.524, 0.995]?
===
Answer:
Among the two, the couch located [0.830, 0.431, 0.978, 0.596] would be more comfortable for resting. The object [0.264, 0.717, 0.524, 0.995] is a dining chair, which is typically less comfortable for extended relaxation.
....

Data Collection – Spatial Negative Mining

Two types of data:

- Image-conditioned Category Localization:
    - Ask model to localize an object class that are in common vocabulary but not in image.

- Semantics-conditioned Category Localization.
    - Ask model to localize an object class that is semantically close to an existing object in image.



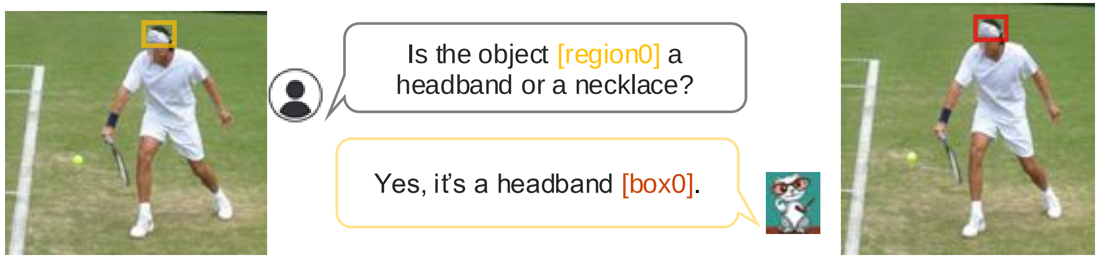
- *Image-conditioned Category Localization:*
  Is there a car in the image?

- *Semantics-conditioned Category Localization.*
  Is there a Hasky in the image?

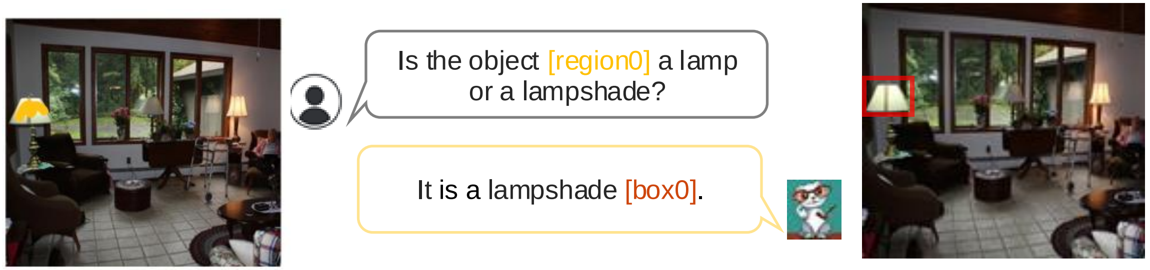
Evaluation - Referring

Introduced **Referring Object Classification**, a basic task to validate referring in object-level.

- Data Source:
  - LVIS
- Format:
  - A binary-choice question in the format of “Is the object ⟨location⟩ a ⟨class A⟩ or a ⟨class B⟩?”
  - Two objects are near each other.
  - Three formats: Point, Box, Free-from Shape.



<u>Referring Object Classification</u> (LVIS) – box



<u>Referring Object Classification</u> (LVIS) – free form shape

Evaluation - Referring

Table 3: Results of referring object classification on three different referring types, including point, box, and free-form shape. '✕' means no such capability.

Models	LVIS (%)		
	Point	Box	Free-form
Random Guess	50	50	50
LLaVA	50.1	50.3	✕
Kosmos-2 (Peng et al., 2023)	✕	60.25	✕
Shikra-7B (Chen et al., 2023b)	57.82	67.71	✕
GPT4-ROI (Zhang et al., 2023)	✕	61.76	✕
Ferret-7B	67.94	79.42	69.77
Ferret-13B	**68.35**	**80.46**	**70.98**

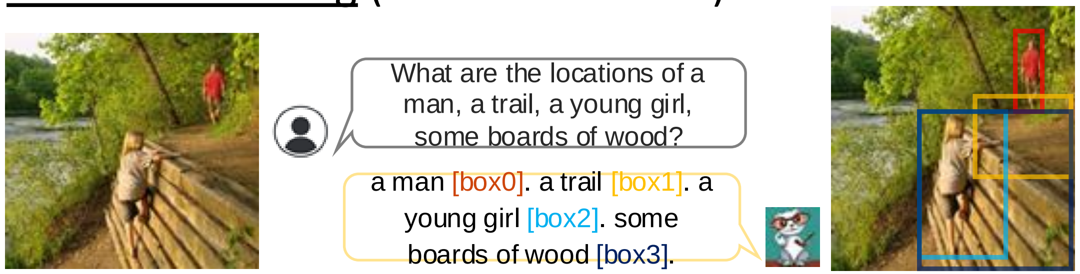
Ferret: a MLLM w/ Spatial Understanding

Evaluation - Grounding

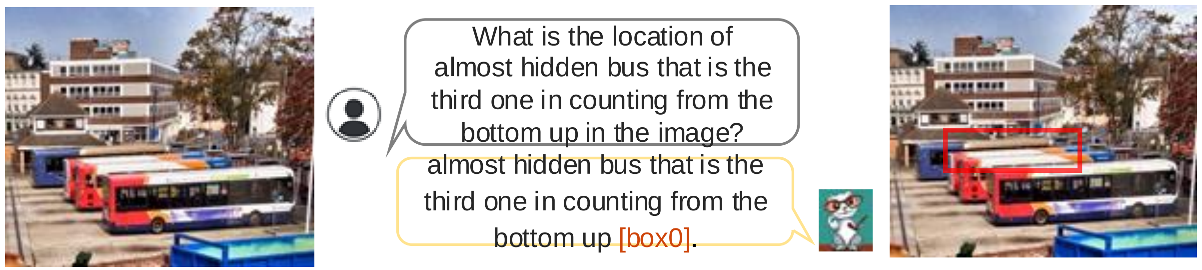
We evaluate conventional object/phrase grounding.

- Dataset:
  - RefCOCO
  - RefCOCO+
  - RefCOCOg
  - Flickr30k

Phrase Grounding (Flickr30k Entities)



Referring Expression Comprehension (RefCOCOg)



Evaluation - Grounding

Performance comparison (Acc@0.5) on the referring expression comprehension (RefCOCO, RefCOCO+, RefCOCOg) and phrase grounding (Flickr30k Entities) tasks. ∗ indicates that the method is specifically fine-tuned in the second stage.

Models	RefCOCO			RefCOCO+			RefCOCOg		Flickr30k Entities	
	val	testA	testB	val	testA	testB	val	test	val	test
MAttNet (Yu et al., 2018)	76.40	80.43	69.28	64.93	70.26	56.00	66.67	67.01	–	–
OFA-L (Wang et al., 2022b)	79.96	83.67	76.39	68.29	76.00	61.75	67.57	67.58	–	–
TransVG (Deng et al., 2021)	81.02	82.72	78.35	64.82	70.70	56.94	68.67	67.73	–	79.10
UNITER (Chen et al., 2020)	81.41	87.04	74.17	75.90	81.45	66.70	74.02	68.67	–	–
VILLA (Gan et al., 2020)	82.39	87.48	74.84	76.17	81.54	66.84	76.18	76.71	–	–
UniTAB (Yang et al., 2022)	86.32	88.84	80.61	78.70	83.22	69.48	79.96	79.97	78.76	79.58
MDETR (Kamath et al., 2021)	86.75	89.58	81.41	79.52	84.09	70.62	81.64	80.89	82.3*	83.8*
Shikra-7B (Chen et al., 2023b)	87.01	90.61	80.24	**81.60**	87.36	72.12	82.27	82.19	75.84	76.54
Ferret-7B	**87.49**	**91.35**	**82.45**	80.78	**87.38**	**73.14**	**83.93**	**84.76**	**80.39**	**82.21**
Shikra-13B (Chen et al., 2023b)	87.83	91.11	81.81	**82.89**	87.79	74.41	82.64	83.16	77.41	78.44
Ferret-13B	**89.48**	**92.41**	**84.36**	82.81	**88.14**	**75.17**	**85.83**	**86.34**	**81.13**	**84.76**

Evaluation – Ferret-Bench

A new benchmark to evaluate tasks jointly requiring referring/grounding, semantics, knowledge, and reasoning.

- Three New Tasks:
  - **Referring Description**: Describe a referred region based on its interaction with surrounding objects.
  - **Referring Reasoning**: Reason on top of one or more referred regions correctly
  - **Grounding in Conversation**: Reason correctly and accurately ground/localize the objects/regions necessary for the reasoning.

- Each task have 40 annotations, generated by GPT-4 in a similar way to preparing GPT-Assisted Data in GRIT, verified by human.
- Evaluated by GPT-4.

Evaluation – Ferret-Bench

An Example:

Quantitative Comparison:

	Ferret-Bench			
	Referring Description	Referring Reasoning	Grounding in Conversation	Avg.
LLaVA[8]	41.4	31.7	28.8	34.0
Kosmos-2	51.8	33.7	48.4	44.6
Shikra-7B	46.0	41.6	50.1	45.9
Ferret-7B	68.7	67.3	57.5	64.5
Ferret-13B	**70.6**	**68.7**	**59.7**	**66.3**

Question:
What is the purpose of the object [region0]?

Ground Truth:
The object is a bottle. Bottles are typically used to hold and store different types of liquids. In this context, it might be used to store a beverage for the meal.

LLaVA:
A knife and a fork, indicating that it is ready for a meal.

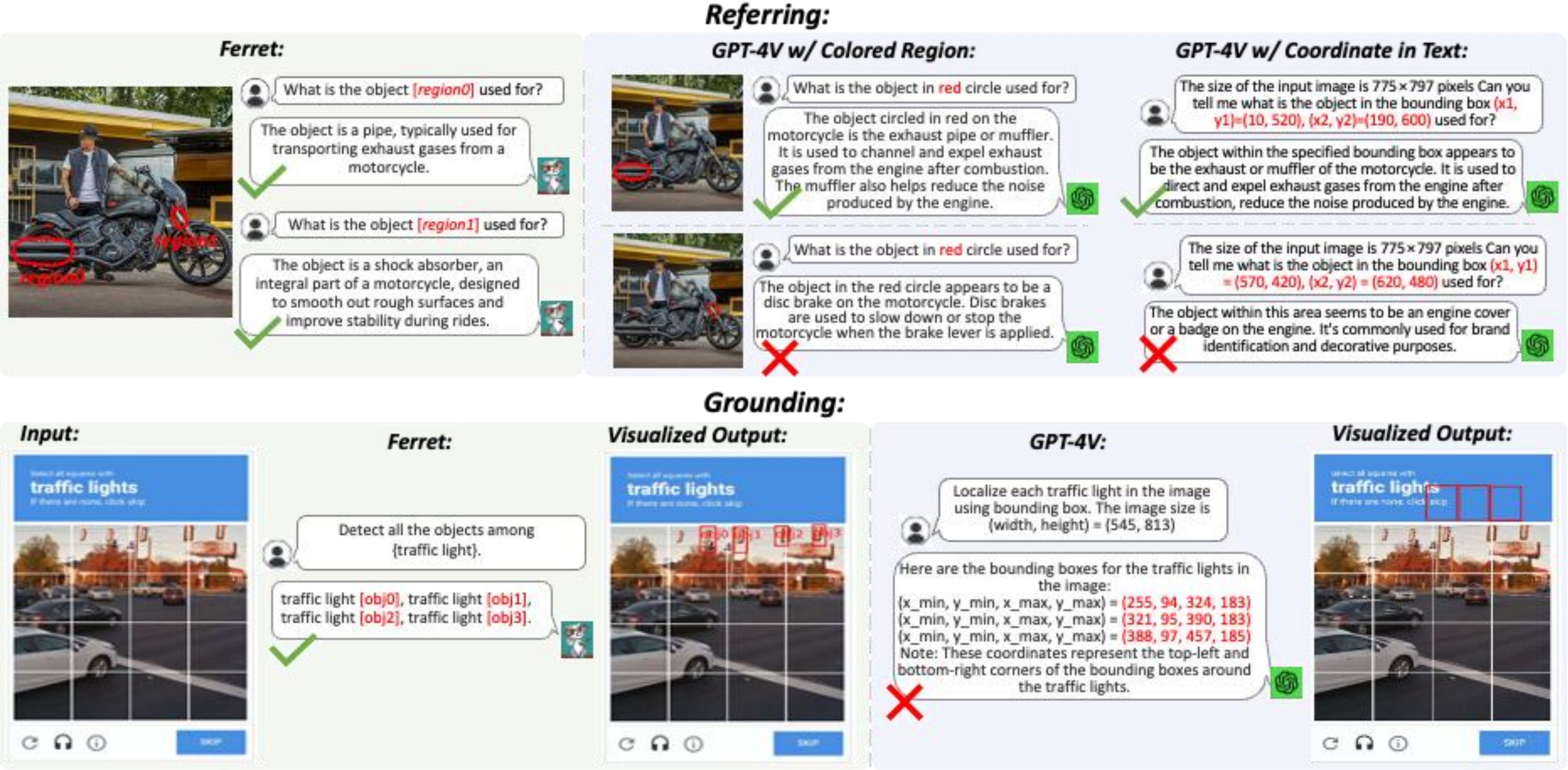
Kosmos-2 :
The purpose of the object is to attract the birds to the table.

Shikra :
The answer is keep birds away.

Ferret (Ours) :
The object is a bottle, generally used to store and dispense liquids like water, juice, or other beverages.

Ferret: a MLLM w/ Spatial Understanding

Comparison w/ GPT4-V



Ferret-v2: an improved baseline for referring and grounding with LLMs

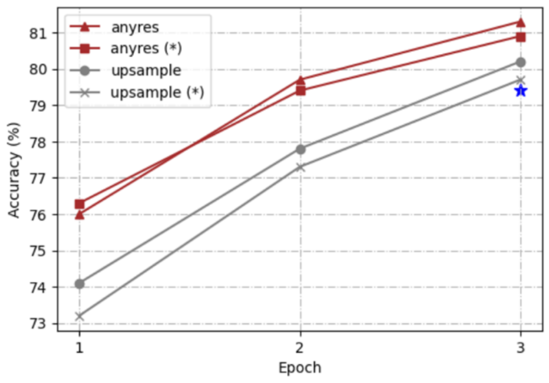
Substantial Improvements over both fine-grained **region-level** and global **image-level** tasks

(a). Qualitative Example

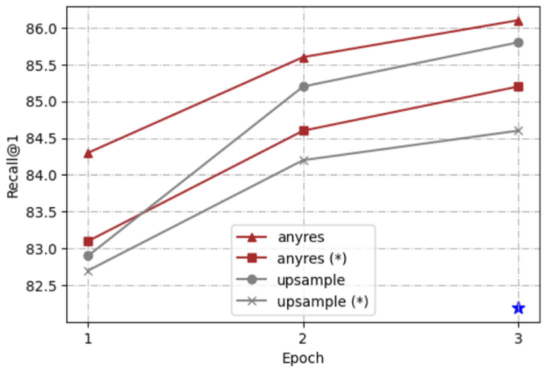
(b). Quantitative Result

[1] *Ferret-v2: An Improved Baseline for Referring and Grounding with Large Language Models. 2024.*

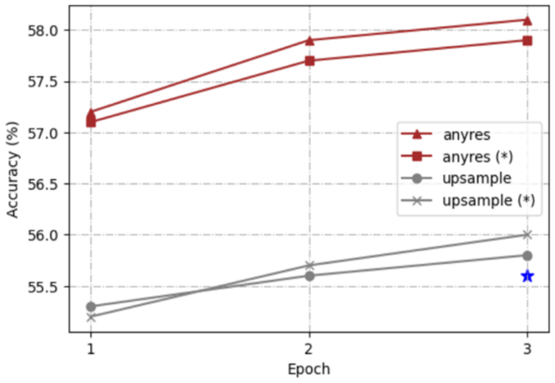
Any Resolution Referring & Grounding



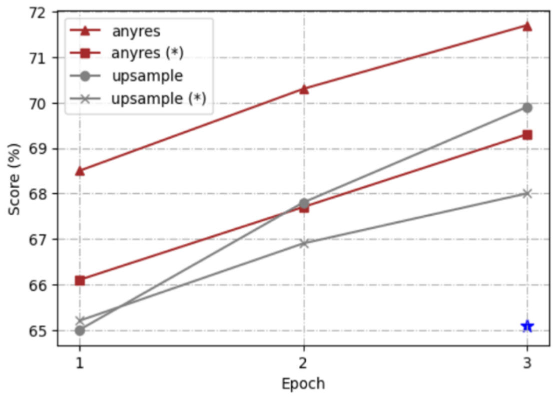
(a) ROC (LVIS-box).



(b) REC (RefCOCOg).



(c) TextVQA.



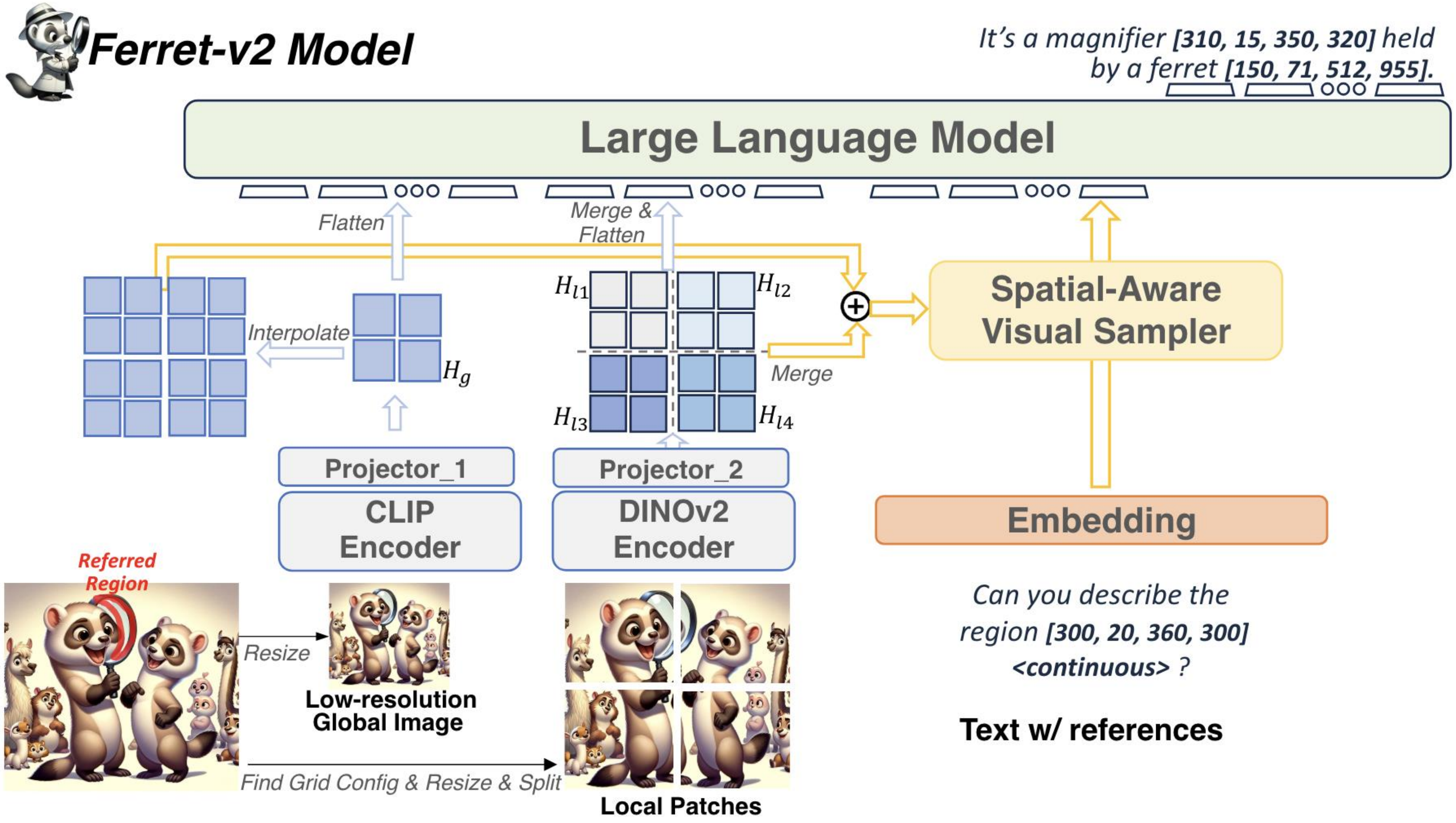
(d) Ferret-Bench.

- Resolution is critical for fine-grained tasks.
- Typically, with LLMs, there are two ways:
  1. Direct Up-sampling (interpolate)
  2. Any Resolution (split)
- Generally, 2 >> 1
  1. Preserving valuable pre-training knowledge
  2. Effortlessly combined with Ferret's grounding.

Ferret-v2: an improved baseline for referring and grounding with LLMs

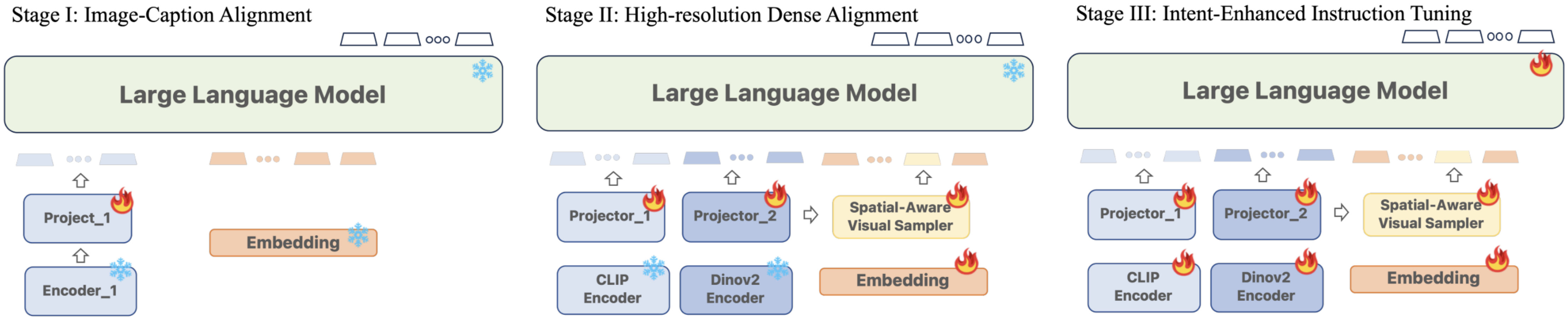
Multi-Granularity Visual Encoding

- CLIP is used to encode the global patch; DINOv2 is used to encoder the local patches.
- Integrate the best of both semantics and local details for precise referring.



Ferret-v2: an improved baseline for referring and grounding with LLMs

Multi-stage Training



- 2nd Stage: High-Resolution Dense Alignment

Dense Referring –
"**Question**: Please classify the objects in the following locations. *1: ⟨region_1⟩, 2: ⟨region_2⟩, ...*
Answer: Here are the categories: *1: cat, 2: dog, ...".*

Dense Grounding –
Question: Please localize visible objects in the image in a raster scan order.
Answer: The objects are: *1: cat ⟨coordinate_1⟩, 2: dog ⟨coordinate_2⟩, ...".*

Ferret-v2: an improved baseline for referring and grounding with LLMs

Performance

Models	LVIS (%)			SA-refer (%)		
	Point	Box	Free-form	Point	Box	Free-form
Random Guess	50	50	50	50	50	50
Kosmos-2	✕	60.25	✕	✕	53.97	✕
Shikra-7B	57.82	67.71	✕	54.15	56.82	✕
GPT4-ROI	✕	61.76	✕	✕	55.02	✕
CogVLM-17B	✕	79.62	✕	✕	61.77	✕
SPHINX-2k	72.83	82.97	✕	61.21	63.39	✕
Ferret-7B	67.94	79.42	69.77	61.91	62.99	57.74
Ferret-v2-7B (Ours)	**74.55**	**86.59**	**76.13**	**68.38**	**68.83**	**62.07**
Ferret-13B	68.35	80.46	70.98	63.16	63.35	58.02
Ferret-v2-13B (Ours)	**75.09**	**87.74**	**76.35**	**67.38**	**69.49**	**62.58**

Models	RefCOCO			RefCOCO+			RefCOCOg		Flickr30k Entities	
	val	testA	testB	val	testA	testB	val	test	val	test
MAttNet (Yu et al., 2018)	76.40	80.43	69.28	64.93	70.26	56.00	66.67	67.01	–	–
OFA-L (Wang et al., 2022)	79.96	83.67	76.39	68.29	76.00	61.75	67.57	67.58	–	–
UNITER (Chen et al., 2020)	81.41	87.04	74.17	75.90	81.45	66.70	74.02	68.67	–	–
VILLA (Gan et al., 2020)	82.39	87.48	74.84	76.17	81.54	66.84	76.18	76.71	–	–
UniTAB (Yang et al., 2022)	86.32	88.84	80.61	78.70	83.22	69.48	79.96	79.97	78.76	79.58
MDETR (Kamath et al., 2021)	86.75	89.58	81.41	79.52	84.09	70.62	81.64	80.89	82.3*	83.8*
G-DINO-L (Liu et al., 2023c)	90.56*	93.19*	88.24*	82.75*	88.95*	75.92*	86.13*	87.02*	–	–
Shikra-7B (Chen et al., 2023b)	87.01	90.61	80.24	81.60	87.36	72.12	82.27	82.19	75.84	76.54
MiniGPT-v2-7B (Chen et al., 2023a)	88.06	91.29	84.30	79.58	85.52	73.32	84.19	84.31	–	–
Qwen-VL-7B (Bai et al., 2023)	88.55	92.27	84.51	82.82	88.59	76.79	85.96	86.32	–	–
SPHINX-2k (Lin et al., 2023)	91.10	92.88	87.07	85.51	90.62	80.45	88.07	88.65	–	–
LLaVA-G (Zhang et al., 2023a)	89.16	–	–	81.68	–	–	84.82	–	83.03	83.62
VistaLLM (Pramanick et al., 2023)	88.1	91.5	83.0	82.9	89.8	74.8	83.6	84.4	–	–
Ferret-7B (You et al., 2023)	87.49	91.35	82.45	80.78	87.38	73.14	83.93	84.76	80.39	82.21
Ferret-v2-7B (Ours)	**92.79**	**94.68**	**88.69**	**87.35**	**92.75**	**79.3**	**89.42**	**89.27**	**85.52**	**85.83**
Shikra-13B (Chen et al., 2023b)	87.83	91.11	81.81	82.89	87.79	74.41	82.64	83.16	77.41	78.44
Griffon v2 (Zhan et al., 2024)	89.6	91.8	86.5	81.9	85.5	76.2	85.9	86.0	–	84.8
CogVLM-Grounding-17B (Wang et al., 2023a)	92.76	94.75	88.99	88.68	92.91	83.39	89.75	90.79	–	–
Ferret-13B (You et al., 2023)	89.48	92.41	84.36	82.81	88.14	75.17	85.83	86.34	81.13	84.76
Ferret-v2-13B (Ours)	**92.64**	**94.95**	**88.86**	**87.39**	**92.05**	**81.36**	**89.43**	**89.99**	**85.33**	**86.25**

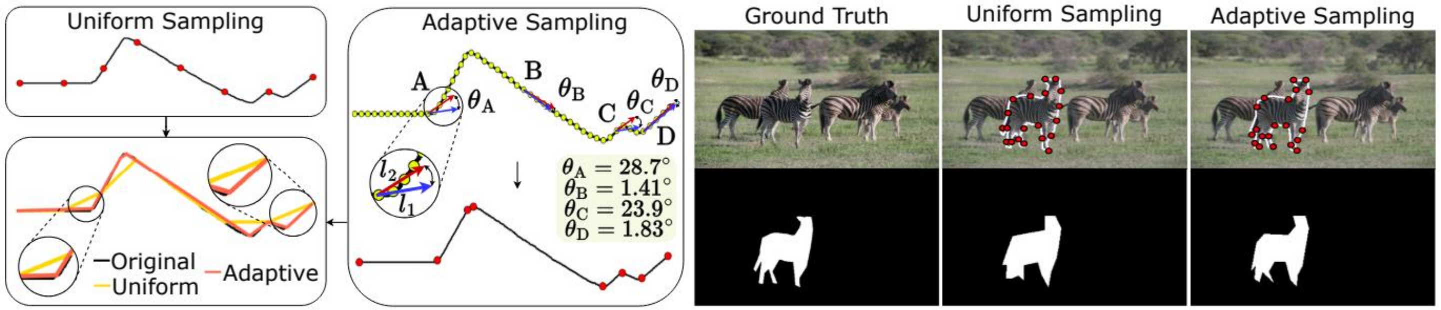
Models	Ferret-Bench			
	Referring Description	Referring Reasoning	Grounding in Conversation	Avg.
LLaVA	41.4	31.7	28.8	34.0
Kosmos-2	51.8	33.7	48.4	44.6
Shikra-7B	46.0	41.6	50.1	45.9
CogVLM-17B	67.1	67.6	51.7	62.1
Osprey-7B	72.2	67.8	–	–
SPHINX-2k	55.6	70.2	**66.4**	64.0
Ferret-7B	68.7	67.3	57.5	64.5
Ferret-v2-7B (Ours)	**79.9**	**81.7**	**65.2**	**75.6**
Ferret-13B	70.6	68.7	59.7	66.3
Ferret-v2-13B (Ours)	**79.6**	**79.4**	**65.7**	**74.9**

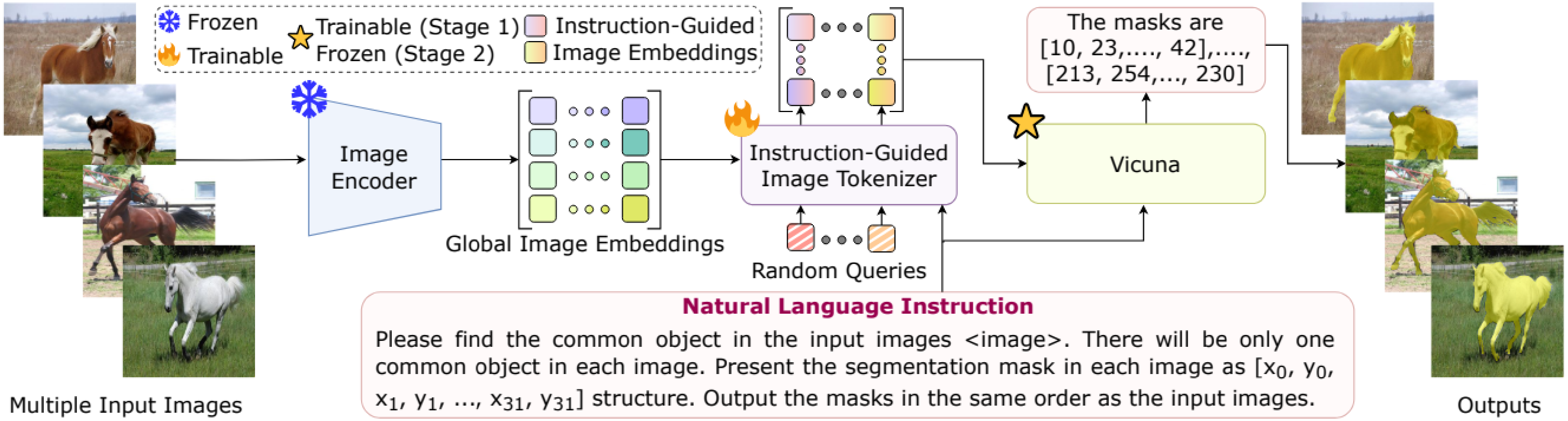
Method	VQA^{v2}	GQA	VQA^{T}	POPE	MME^{P}	SEED	LLaVA^{C}	LLaVA^{W}	MM-Vet	Obj-Hal ↓
BLIP-2-13B	41.0	41	42.5	85.3	1293.8	46.4	–	38.1	22.4	–
InstructBLIP-7B	–	49.2	50.1	–	–	53.4	–	60.9	26.2	–
IDEFICS-9B	50.9	38.4	25.9	–	–	–	–	–	–	–
Qwen-VL-7B	78.8*	59.3*	63.8	–	–	56.3	–	–	–	–
Qwen-VL-Chat-7B	78.2*	57.5*	61.5	–	1487.5	58.2	–	–	–	43.8/23.0
LLaVA-1.5-7B	78.5*	62.0*	58.2	85.9	1510.7	58.6	82.7	63.4	30.5	46.3/22.6
Ferret-v2-7B (Ours)	**81.5***	**64.7***	**61.7**	**87.8**	**1510.3**	**58.7**	**89.1**	**67.7**	**34.9**	**23.8/14.7**
InstructBLIP-13B	–	49.5	50.7	78.9	1212.8	–	–	58.2	25.6	–
Shikra-13B	77.4*	–	–	–	–	–	–	–	–	–
IDEFICS-80B	60.0	45.2	30.9	–	–	–	–	–	–	–
LLaVA-1.5-13B	80.0*	63.3*	61.3	85.9	1531.3	61.6	83.4	70.7	35.4	–
LLaVA-1.5-13B-HD	**81.8***	64.7*	**62.5**	86.3	1500.1	**62.6**	–	**72.0**	**39.4**	–
Ferret-v2-13B (Ours)	**81.8***	**64.8***	62.2	**88.1**	**1521.4**	61.7	**90.7**	69.9	35.7	34.7/16.8

Other Fine-grained Pixel-Level MLLMs

VistaLLM:

- Represent masks as a sequence of texts.
- Adaptive Sampling.



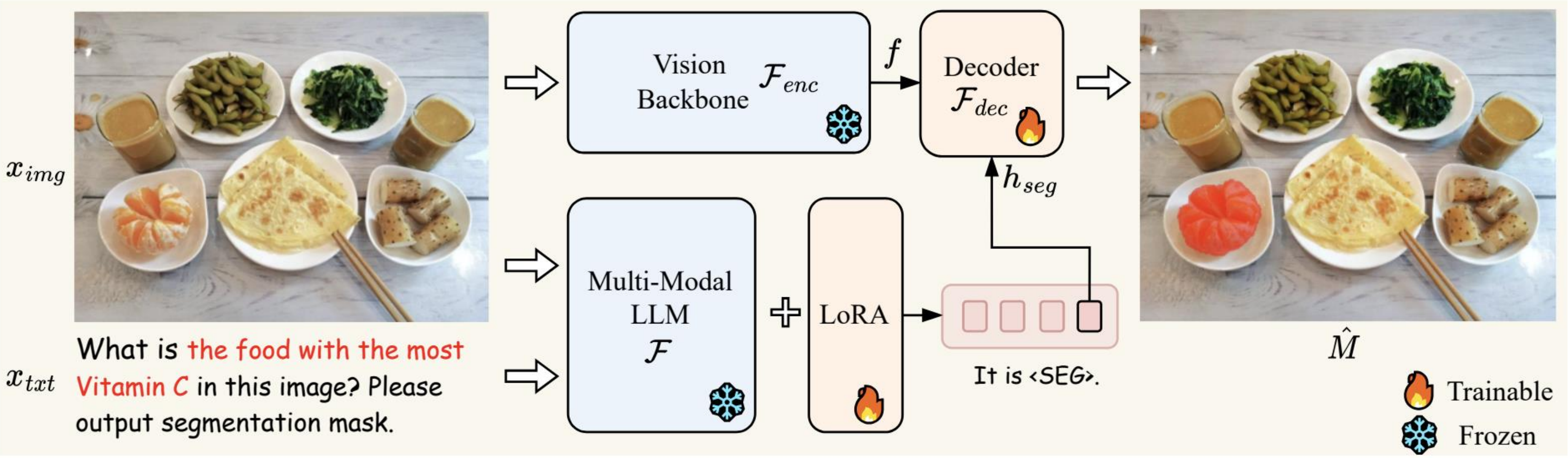


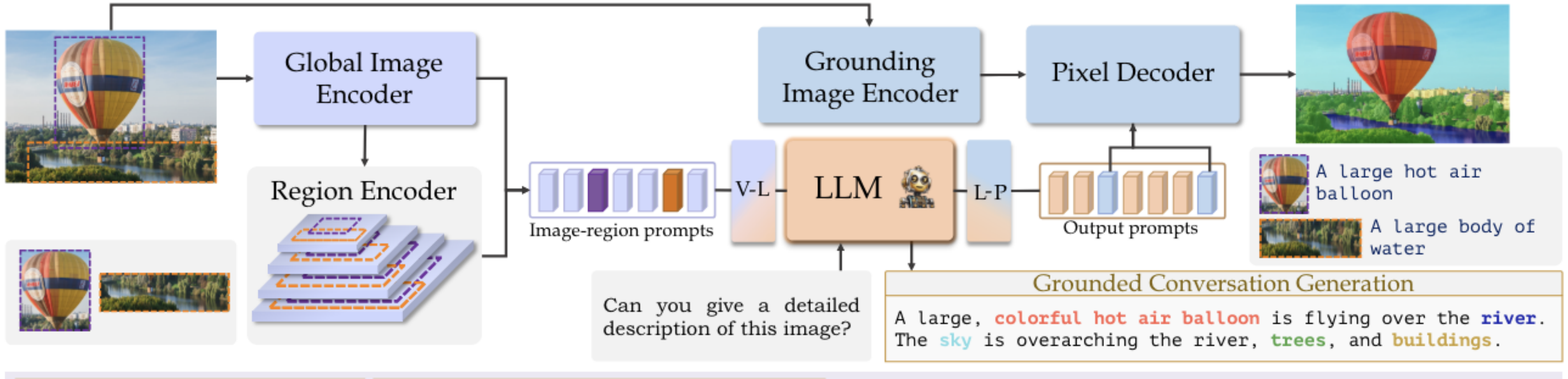
Other Fine-grained Pixel-Level MLLMs

LISA & GLaMM:

- Use the MLLM embedding to prompt SAM - Segment Anything decoder.

- Can perform reasoning segmentation.

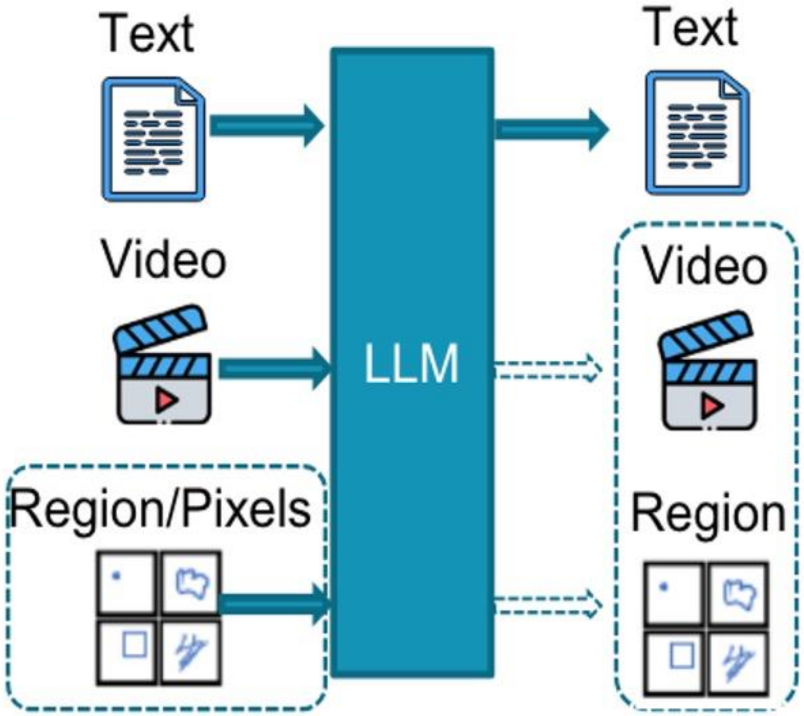




[1] LISA: Reasoning Segmentation via Large Language Model. 2023.
[2] GLaMM: Pixel Grounding Large Multimodal Model. 2023.
[3] Segment Anything. 2023

Video-based Region-Level MLLMs

- PG-Video-LLaVA
- Merlin
- Motion Epic
- ...

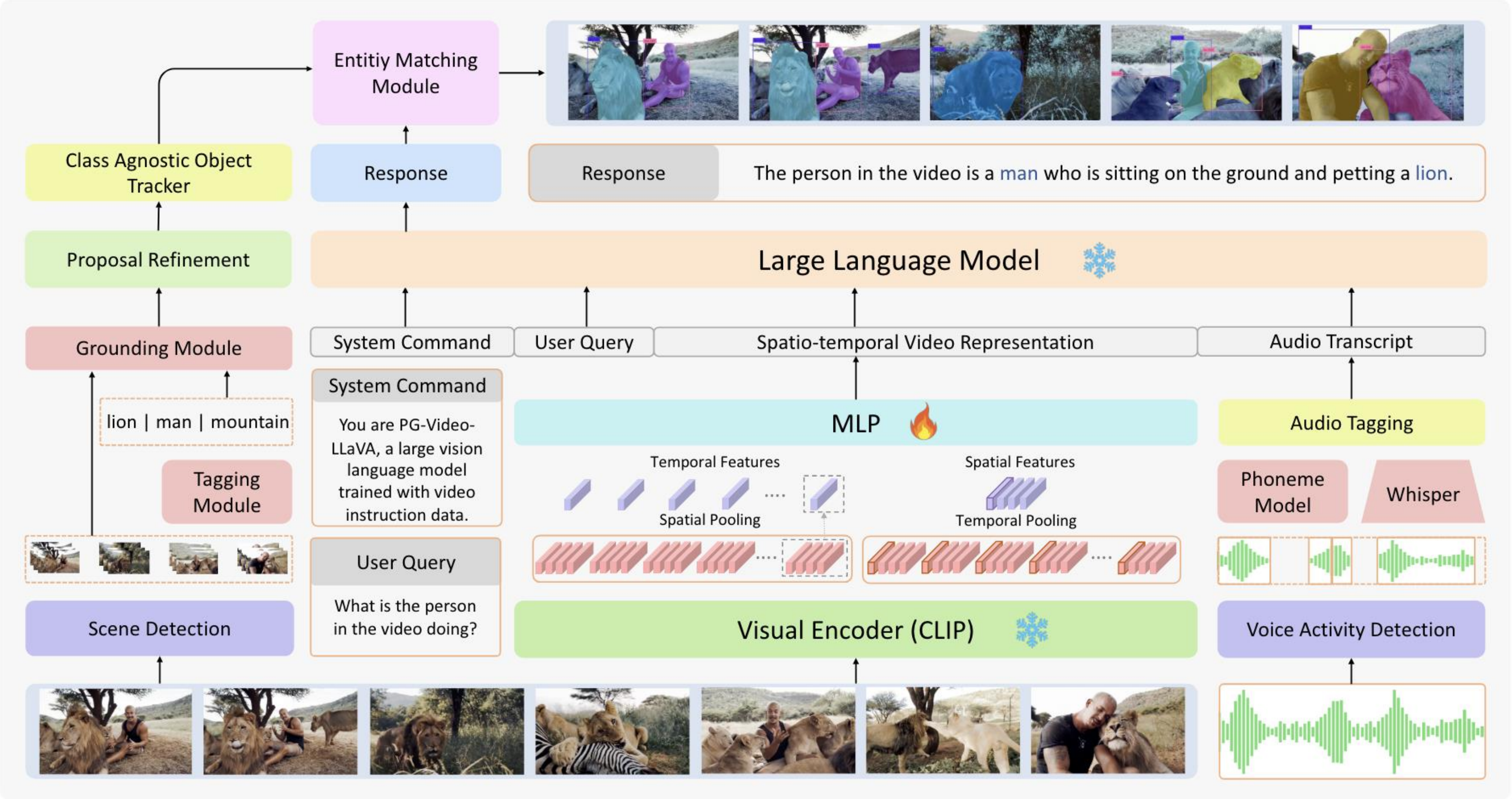


[1] *PG-Video-LLaVA: Pixel Grounding in Large Multimodal Video Models. 2023*
[2] *Merlin: Empowering Multimodal LLMs with Foresight Minds. 2023*
[3] *Video-of-Thought: Step-by-Step Video Reasoning from Perception to Cognition. 2024*

Video-based Region-Level MLLMs

PG-Video-LLaVA:

- Apply Spatial & Temporal Pooling to reduce the multiple frame tokens.
- Entity Matching Module is similar as GLIP / Grounding DINO's.
- Aligned Audio inputs.

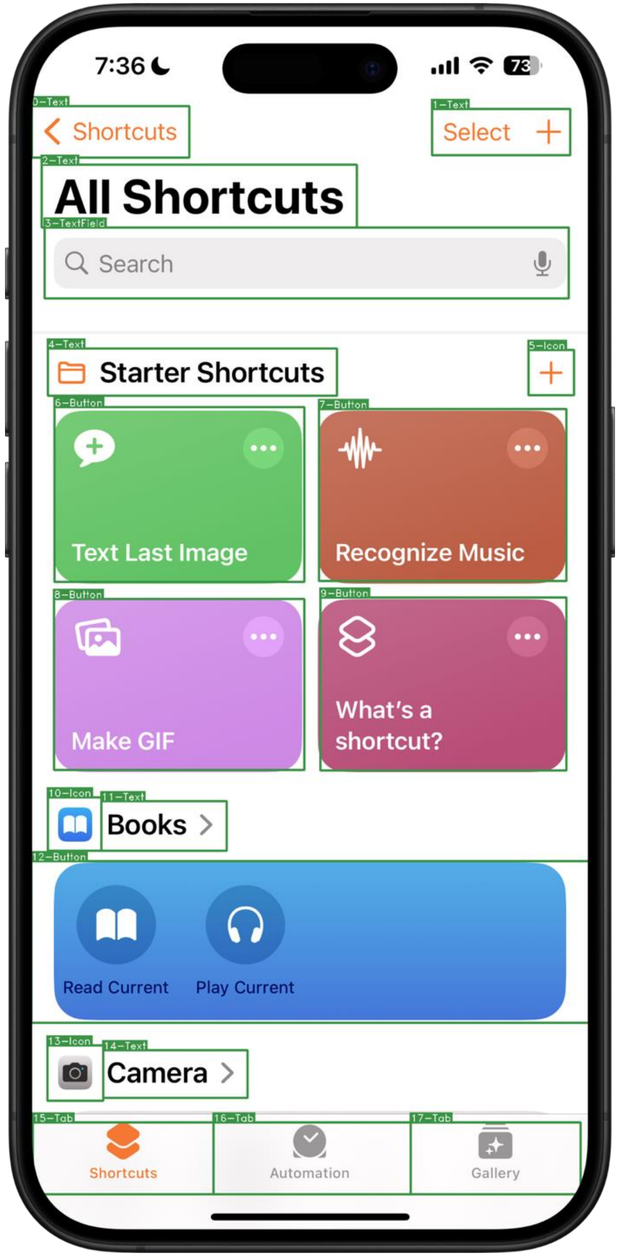


[1] *PG-Video-LLaVA: Pixel Grounding Large Video-Language Models. 2023.*

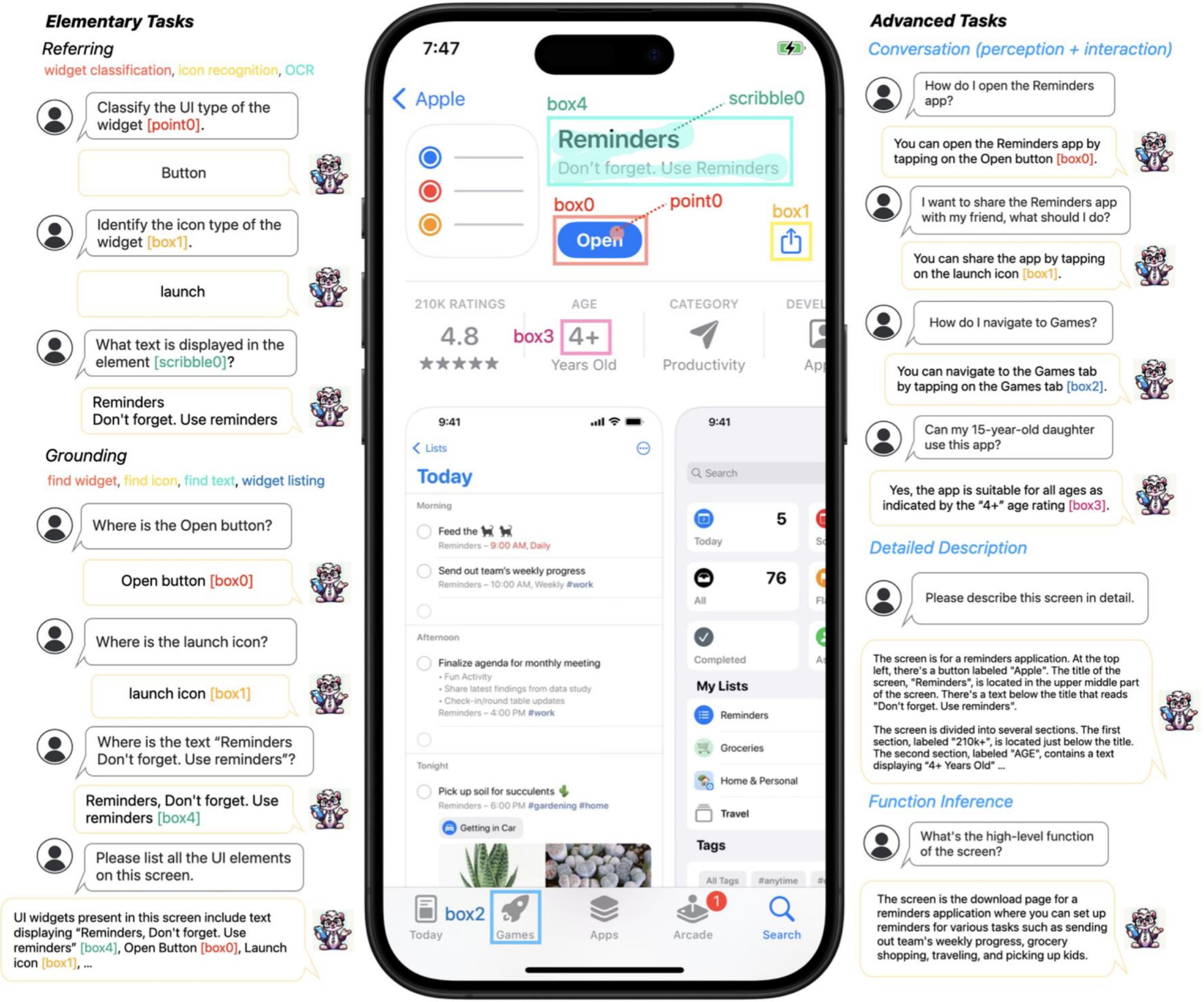
New Application - UI Understanding

- Screenshots
  - iPhone screens from the AppModeling dataset
  - Android screens from RICO dataset

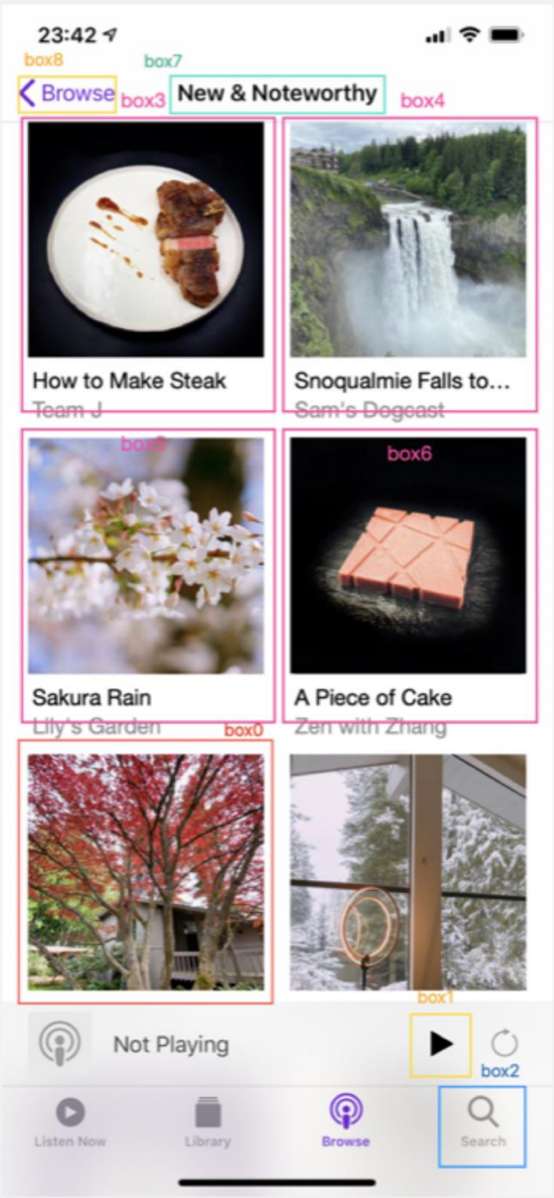
- Annotations
  - Internal UI detection model identifies all UI elements on a screen
  - With their type, bounding boxes, and text
- Elementary tasks
  - Rely on referring and grounding capabilities
- Advanced tasks
  - Requires a deeper level of comprehension



Ferret-UI:



[1] *Ferret-UI: Grounded Mobile UI Understanding with Multimodal LLMs. 2023.*



Q: Classify the type of UI widget located in the element [box0].
A: Picture

Q: Where is the play icon located?
A: play icon [box1]

Q: Where is the search tab located?
A: search tab [box2]

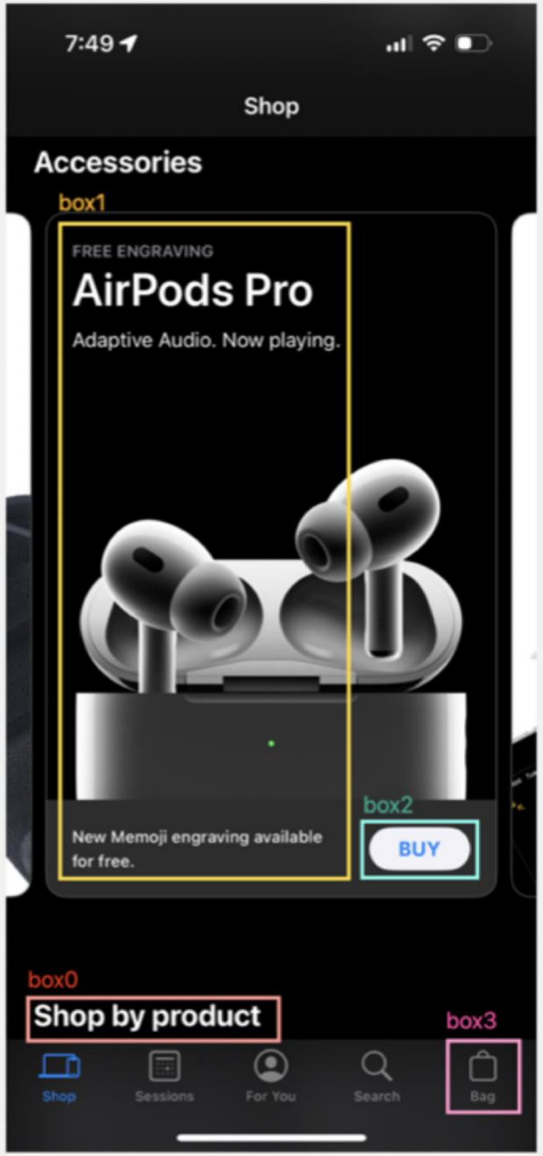
Q: What podcasts are available on this page?.
A: There are several podcasts available on this page. They include "How to Make Steak" by Team J [box3], "Snowquamie Falls to..." by Sam's Dogcast [box4], "Sakura Rain" by Lily's Garden [box5], and "A Place of Cake" by Zen with Zhang [box6].

Q: I want to listen to Sakura Rain, what should I do?
A: You can start listening to "Sakura Rain" by tapping on the corresponding picture [box4]

Q: What's the category of the podcasts listed on this screen?
A: The podcasts listed on this screen are about various topics such as cooking, gardening, and a place of cake. They are categorized under "New & Notable" [box7].

Q: How do I go back?
A: You can go back by tapping on the "Browse" button [box8]

Q: What's the function of this screen when the user interacts with application?
A: The screen is for a podcast application where users can browse and play new and notable podcasts, with options to play, download, and search for specific podcasts.



Q: What's the function of this screen when the user interacts with application?
A: The screen is for an online shopping application where users can browse and purchase products, specifically the AirPods Pro, with options to shop by product, accessories, and other features like Store Mode, Search, and Bag.

Q: Can I customize my AirPods?
A: Yes, you can customize your AirPods by tapping on the "Shop by product" text [box0].

Q: Do I need to pay for the engraving?
A: No, the text on the screen indicates that the engraving is free [box1]. However, you will need to pay for the AirPods Pro [box1].

Q: Amazing! I want to buy AirPods!
A: You can do that by tapping on the "Buy" button [box2] on the screen.

Q: Let me check what's in my bag.
A: You can view the items in your bag by tapping on the "Bag" tab [box3].

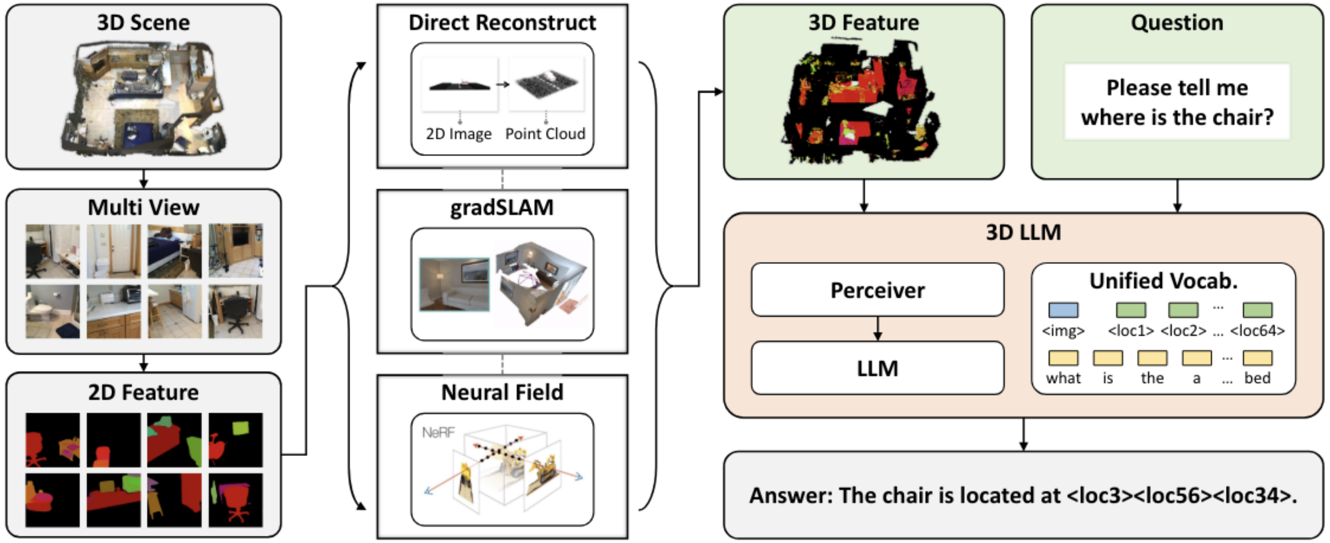
Q: Please provide a detailed description for this screen.
A: The screen appears to be a shopping page for a product, specifically the "AirPods Pro". The product is displayed prominently in the middle of the screen with a description stating "New Memoji engraving available, for free.". There is also a "Buy" button located towards the right side of the product description.

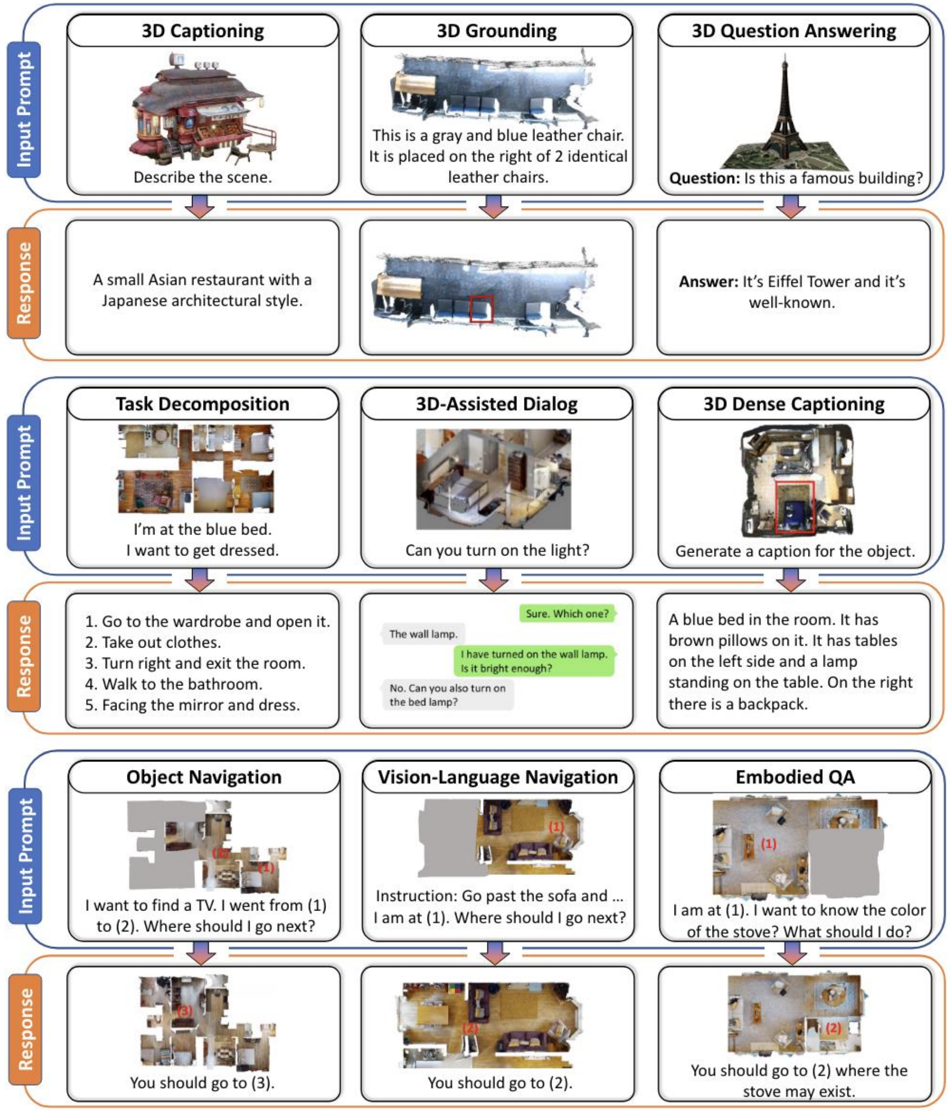
At the top of the screen, there is a "Shop" text centered, likely indicating the current page or section of the app. Below this, there is a section titled "Accessories", which is likely a category for the product.

At the bottom of the screen, there is a navigation bar with five tabs. From left to right, these are labeled "Shop", "Store Mode", "Memoji", "Search", and "Bag". The "Shop" tab is currently selected, as indicated by its position at the far left of the navigation bar.

New Application – 3D & Embodiments

- Holistic Scene Understanding
- 3D Spatial Relationships
- Affordances and 3D Planning





[1] 3D-LLM: Injecting the 3D World into Large Language Models. 2023.

Take-away message

- What has been covered…
  - Referring and Grounding Capabilities are the core tasks of recent MLLMs.
  - Multiple kinds of Fine-grained MLLMs: Spatial (points, boxes, masks), Temporal
  - Enable new applications: UI Understanding, 3D & Embodied Agents, etc.
- Future Challenges
  - Higher resolution needed to be supported given the limitations of LLM context window.
  - Training efficiency may be added up when we scaling the image resolution & detailed captions.
  - What kind of tasks are required to enable the fine-grained capabilities in MLLMs?

TM and © 2024 Apple Inc. All rights reserved.