Appendix

A. Mask2Former+ Implementation Details

Our enhancement of the original Mask2Former model focuses on broadening its segmentation capabilities beyond the 134 common object categories it currently handles, which include 80 things and 55 stuffs as defined in the COCO dataset. The primary goal is to enable the model to recognize an expanded range of object categories, as well as segmentation masks of various levels of granularities, such as semantic parts and visual text regions.

Training data. We have compiled a comprehensive dataset by combining multiple existing segmentation datasets. This ensemble encompasses a wide spectrum of entities (things and stuff), their semantic parts, and visual text, drawn from sources such as COCO [5], LVIS [25], Entity-v2 [60], Pascal [16], PACO [63], MHP-v2 [40], and TextOCR [67]. The resulting dataset comprises over 200K images and 4.5M masks, as summarized in Table 11. Notably, the annotations from COCO, LVIS, and PACO are based on a shared set of COCO images. We merged these annotations to ensure comprehensive mask proposal coverage, thereby providing holistic instance coverage within each image, as can be illustrated in Figure 7.

Dataset		Granularity			Dataset Size	
Name	Split	Entity	Part	Text	#Image	#Masks
LVIS [25] & PACO [63]	part	✓	✓		15,089	596,687
	no_part	✓			103,178	2,062,536
Entity-v2 [60]	cls	✓			31,913	579,076
Pascal [18]	train	✓	✓		4,998	93,322
	val	✓	✓		5,105	95,462
MHP [40]	train		✓		15,403	410,113
TextOCR [67]	train			✓	21,749	714,770
Total		✓	✓	✓	197,435	4,551,966

Table 11. Summary of the training datasets for Mask2Former+. Entity includes both thing and stuff categories.

Model. Building on the foundation of the original Mask2Former [10], we developed Mask2Former+, a panoptic segmentation model designed for multi-grained segmentation. We initialize our model from the Mask2Former checkpoint with the Swin-L backbone pre-trained on the COCO panoptic segmentation dataset [34]. Besides the 200 entity queries that are trained for thing and stuff proposals, we added 50 additional expert queries for the segmenting parts and the visual text regions, respectively. Given that not all images have annotations for every type of segmentation (for instance, the TextOCR dataset provides annotations only for visual text regions), our model computes the group-wise matching loss exclusively for the annotations available in each dataset. This approach ensures that the model benefits from partial annotations without compromising its ability to recognize other levels of granularity when certain annotations are unavailable. Although most samples in our dataset also have semantic annotations such as object categories, we do not use them but only train the model for class-agnostic mask proposals. We train the model for 20k iterations on our combined segmentation dataset with a batch size of 16 using the Detectron2 library [77].[1]

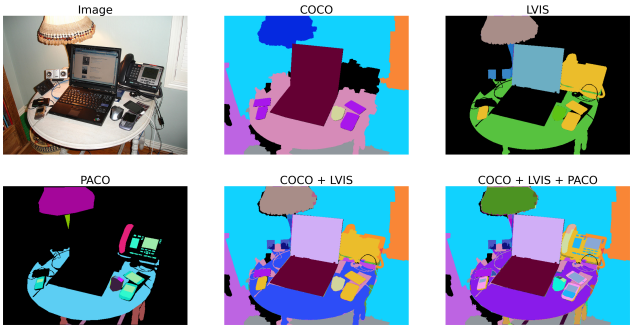


Figure 7. Illustrations of the merged segmentation annotations from COCO Panoptic, LVIS, and PACO datasets.

B. The M3G2 Dataset

In this section, we introduce the M3G2 dataset with Multi-Modal Multi-Grained Grounding. M3G2 is a comprehensive dataset consisting of 36 sub-problems, derived and augmented from 27 existing datasets with grounded vision-language annotations. The dataset is categorized into four main types: (1) Grounded Image Captioning (GIC), (2) Grounded Visual Question Answering (GVQA), (3) Referential Expression Segmentation (RES), and (4) Referential Dialog (RD). Details on the dataset sources, image origins, types of grounding annotations, semantic granularity, and data statistics are summarized in Table 12. All datasets are formatted into the conversation format between a human user and a model assistant, where the user provides task objectives as instructions, and model responses are generated automatically based on the annotations.

Grounded Image Captioning (GIC). GIC focuses on generating image captions that ground to visual entities presented in the image. We incorporate the Panoptic Narrative Grounding (PNG) [34] and Flickr30K-Entity [58] datasets. PNG, derived from Localize Narrative [59] and COCO Segmentation [5], provides long and detailed narratives with an average of 36.5 words per description, exemplified in Figure 15a. These narratives are rich in detail, offering a high coverage of the visual content including the background. Flickr30K-Entity, offering concise captions with box annotations, complements PNG with its larger vocabulary and finer granularity, as shown in 15b. The example instruction templates used to construct the conversation are listed in Ta-

[1] https://github.com/facebookresearch/detectron2

Metadata			Grounding Annotations			Semantic Granularity					Data Size	
Task Type	Dataset Name	Image Source	Mask	Box	Pointer	Thing	Stuff	Part	Multi.	Text	Train	Val / Test
Grd. Captioning (GCAP)	PNG	COCO	✓	✓		✓	✓		✓		132,045	8,435
	Flickr30K-Entity	Flickr30K		✓		✓	✓	✓	✓		148,915	1,000 / 1,000
Referential Expression Segmentation (RES)	RefCOCO	COCO	✓	✓		✓					113,311	-
	RefCOCO+	COCO	✓	✓		✓					112,441	-
	RefCOCOg	COCO	✓	✓		✓					80,322	-
	RefCLEF	ImageCLEF	✓	✓		✓					104,531	-
	gRefCOCO	COCO	✓	✓		✓					194,233	-
	PhraseCut	VG	✓	✓		✓	✓	✓	✓		84,688	-
	D-Cube	GRD	✓	✓		✓			✓		9,499	-
	ReasonSeg	OpenImages & ScanNetV2	✓	✓		✓	✓	✓	✓		1,315	344
	RIO	COCO	✓	✓		✓			✓		27,696	34,170
	SK-VG	VCR		✓		✓					23,404	-
Grounded Visual Question Answering (GVQA)	VizWiz-Grounding	VizWiz	✓	✓		✓	✓			✓	6,494	1,131 / 2,373
	TextVQA-X	OpenImages	✓	✓						✓	14,476	3,620
	GQA	VG		✓		✓	✓	✓	✓		301,623	-
	VQS	COCO		✓		✓					20,380	8,203
	Shikra-BinaryQA	Flickr30K		✓		✓	✓	✓	✓		4,044	1,159
	EntityCount	Entity-v2	✓	✓		✓	✓	✓	✓		11,088	453
	FoodSeg-QA	Recipe1M	✓	✓		✓			✓		7,114	-
	LVIS-QA	COCO	✓	✓		✓	✓		✓		94,860	3,611
Referential Dialog (RD)	RefCOCO-REG	COCO	✓	✓	✓	✓					17,395	-
	RefCOCO+-REG	COCO	✓	✓	✓	✓					17,383	-
	RefCOCOg-REG	COCO	✓	✓	✓	✓					22,057	-
	gRefCOCO-REG	COCO	✓	✓	✓	✓					20,282	-
	VG-SpotCap	VG		✓	✓	✓	✓	✓	✓		247,381	232,935
	V7W	COCO		✓	✓	✓					22,805	10,193 / 57,265
	PointQA-Local	VG			✓	✓					27,426	4,855 / 4,880
	PointQA-Twice	VG			✓	✓					36,762	14,668 / 5,710
	VCR-Open	VCR		✓	✓	✓					58,340	-
	VCR-Multichoice	VCR		✓	✓	✓					97,648	26,534 / 25,263
	ShikraRD	Flickr30K		✓	✓	✓	✓	✓	✓		1,878	-
	SVIT-RD	VG		✓	✓	✓	✓	✓	✓		32,571	-
	Guesswhat-Guesser	COCO	✓	✓	✓	✓					92,136	19,665
	Guesswhat-Oracle	COCO	✓	✓	✓	✓					101,256	21,643
	VG-RefMatch	VG		✓	✓	✓	✓	✓	✓		247,381	-
	HierText	OpenImages	✓	✓	✓					✓	6,058	3,885

Table 12. The full list of datasets used in M3G2.

Instruction Templates for Brief Captioning	Instruction Templates for Detailed Captioning
Describe the image briefly.	Describe the image in detail.
Describe the image in a few words.	Describe the picture's every detail.
Describe the image in a short sentence.	Describe the given picture in very detail.
Describe the image in a clear and concise manner.	Make a fine description of the image.
Generate a short caption for the picture.	Generate a long caption for the given image.
Caption the image in a few words.	Give me a detailed caption of this image.

Table 13. Instruction templates for the GIC task.

Instruction Templates For RES	
Highlight "{}" in the image.	Segment "{}" in the image.
Segment: {}.	Help me segment out {}.
Localize "{}" in the image.	Help me localize {}.
Help me highlight the region of {}.	Demonstrate where "{}" is located in this image.
Show me where to find in this photo.	Identify and mark the region of {} for me.
Can you highlight "{}"?	Can you extract the segment: {} for me?
Can you localize "{}" in this image?	Could you please segment out {} in the image?

Table 14. Templates used for the RES task.

ble 13, where we use key words such as "short/briefly" and "in detail" to distinguish between short and long captioning.

Referential Expression Segmentation (RES). RES is a task combining language understanding with precise visual segmentation. Our dataset includes 10 diverse sources. To improve the learning efficiency and enhance contextual understanding, we format queries from the same image into a simulated multi-turn dialog, as illustrated in Figures 16 and 17. We employ the widely used RefCOCO/+/g datasets [32, 53] and RefCLEF [65] for single-object RES. gRefCOCO [43] is employed for multi-object and negative queries. To enhance the visual diversity, we also incorporate PhraseCut [76] and D-Cube [79] that use an image source different than COCO. Additionally, ReasonSeg [37], RIO [61], and SK-VG [78] are included, where a textual context is given and the models need to not only understand that context, but also equips with a certain degree of commonsense knowledge to successfully solve the query, such as shown in Figure 17b, 17c and 17d. The dialogue templates are listed in Table 14.

Grounded Visual Question Answering (GVQA). The GVQA task extends the visual question answering by additionally requiring visual grounding of the answer. We include 8 datasets for the grounded VQA task in M3G2. First, we collect and organize some existing datasets that can directly fit into our grounded vision-language task framework, including VizWiz-Grounding [6], TextVQA-X [64], GQA [29], VQA [19] and Shikra-BinaryQA [8] (Figure 18). To further improve the data scale and visual concept coverage, we enlarge the GVQA collection by re-purposing existing panoptic segmentation datasets with templated instruc-

Instruction Templates For Short Response VQA.
{}Answer with a single word or a short phrase.
Given the image, answer the question "{}" with a single word or a short phrase.
Give a short answer to the question "{}" based on the image.
Instruction Templates For Chain-of-Thought Response VQA.
{}Let's think step by step.
{}Please include the reasoning process.
{}Before giving the answer, please explain your reasoning.
{}Explain your logic before giving the answer.
Please answer the following question "{}", and describe your thought process.
Instruction Templates For Grounding Answer to Masks.
Show where in the image you found your answer.
Mark the part of the image that supports your answer.
Please highlight your evidence in the image.
Point out the evidence from the image.
Indicate the area in the image that justifies your response.
Highlight the section of the image that backs up your answer.
Shade the section of the image that confirms your reply.
Emphasize the part of the image that relates to your answer.
Instruction Templates For Object Presence QA.
Is {} present in the image?
Is there any {} in this image?
Instruction Templates For Object Counting QA.
How many {} can you see in this image?
Count the number of {}.
Instruction Templates For Object Segmentation Request.
Segment {}.
Highlight all the {} in this image.
Show me all the {} presented in the picture.

Table 15. Templates used for the GVQA task.

tions and model responses. Specifically, based on the annotations from LVIS [25] and EntityV2 [60], we design questions about object presence, object counting, and segment query with a possibly negative request (i.e. the target object does not exist in the image), for the model to learn to recognize a diverse set of concepts more faithfully. See Figure 19 for examples of such multi-turn QA, and example question templates used in Table 15.

Referential Dialog (RD) . RD features multi-modal conversations where the user can refer to objects or regions in the image by a spatial prompt (e.g. a bounding box). We include various types of RD in our dataset and the templates used are listed in Table 16. First, we add several existing RD datasets such as V7W [94], PointQA [51], VCR [88], ShikraRD [8] and SVIT [90] without much modifications. We then revisit the RefCOCO series [43, 52, 86] for referential expression generation, where the referred object is given and the goal is to generate a unique description that leads to that object. We use the region caption annotations from the VG dataset [36] for region captioning and a region-matching game. We select a set of region pointers and several descriptions to provide to the model, and the goal is to match the pointed regions with the descriptions (Figure 21b). We repurpose the GuessWhat dataset [17] to make it fit into our RD formulation, as shown in Figure 21a. We also construct a referred text reading task based on the HierText [47] dataset and enhance the model's capability of text recognition, as shown in Figure 21c.

Instruction Templates For REG.
Provide a distinct description for that <PTR>
Describe the selected area in a unique way. <PTR>
Share a unique description of the region <PTR>
Offer a one-of-a-kind descriptor <PTR>
Describe the selected area <PTR> uniquely.
Point out <PTR> in the picture with a unique description.
Tell me how <PTR> stands out in the photo.
Use your words to highlight just <PTR> in the image.
Please describe <PTR> in the image in a way that it can be uniquely identified.
If you had to describe just <PTR> to someone, how would you do it?
What makes <PTR> different from everything else in the picture?
How can you describe <PTR> in the image in a way that it can be uniquely identified?
Can you provide a referring expression for <PTR> such that it sets it apart from others?
Let's play a game! Describe <PTR> in the photo so I can find it.
Instruction Templates For Region Captioning.
Describe it <PTR>.
Describe the region <PTR> in a few words.
Describe the region <PTR> in a short phrase.
Describe the selected area <PTR>.
Provide a brief description of this part <PTR>.
Give a short caption for this <PTR>.
Provide a brief description of the area marked <PTR>.
Tell me about the contents in the selected zone <PTR>.
Provide a concise description for this spot <PTR>.
Narrate what you see in the indicated area <PTR>.
What is in the region <PTR>? Describe in a phrase.
What can you see in this area <PTR>?
How would you describe the content at <PTR>?
How would you caption this particular region <PTR>?
What's depicted in the marked area <PTR>?

Table 16. Templates used for the RD task.

C. GROUNDHOG Implementation Details

Data Balancing. In constructing the M3G2 dataset, we recognized the need to address the varying scales of the multiple constituent datasets to ensure a balanced data distribution during training. To achieve this, we have implemented dataset-specific sampling strategies, adjusting the volume of data from each source dataset through either up-sampling or down-sampling. The ratios we applied are as follows:

- PNG: up-sampled by a factor of 2.
- Flickr30k-Entities: up-sampled by 1.5 times.
- RefCOCO$^+$: up-sampled by 1.5 times.
- RefCOCOg: up-sampled by 1.5 times.
- SK-VG: up-sampled by a factor of 2.
- Dcube (multiturn): up-sampled by a factor of 10.
- ReasonSeg: up-sampled by a factor of 10.
- Shikra-Binary: up-sampled by a factor of 10.
- VCR-Open (multiturn): down-sampled by half.
- VCR-Multiturn: down-sampled to 10%.
- VizWiz: up-sampled by a factor of 3.
- LVIS-QA: down-sampled by half.
- TextVQAX: up-sampled by a factor of 2.
- EntityCount: up-sampled by a factor of 2.
- VG-SpotCap: down-sampled by half.
- Shikra-RD: up-sampled by a factor of 10.
- HierText: up-sampled by a factor of 5.
- GuessWhat-Oracle: down-sampled to 20%.
- GuessWhat-Guesser: down-sampled to 20%.
- SVIT: up-sampled by a factor of 3.

The balanced sampled dataset contains 1.8 million samples in total.

(a) Grounded short caption generation on Flickr30K-Entity. While only box supervisions are available for this dataset, GROUNDHOG generalize to pixel-level grounding after joint training on M3G2.

(b) Grounded detailed narrative generation on PNG. GROUNDHOG successfully generalize to grounding a novel category *watch* in the generated caption, which is not included in the 80 categories of PNG annotation.

Figure 8. Examples of GROUNDHOG's performance in grounded image captioning.

Learning from Both Box and Mask Supervision. In the M3G2 dataset, not all sub-datasets include mask supervision, necessitating a hybrid loss approach to effectively benefit from grounded supervision from both mask and box annotations. We address this by employing different loss functions based on the type of annotation available. When mask annotations are available, we apply the dice loss $\mathcal{L}_{\text{dice}}$ and binary cross-entropy loss $\mathcal{L}_{\text{bce}}$ between the predicted grounding masks and the ground truth masks of each phrase, following Cheng et al. [10]. In cases where only box annotations are present, we apply the projection loss $\mathcal{L}_{\text{proj}}$ as introduced by Tian et al. [69], which selects the mask whose projection on the axis matches the best with the annotated box. Essentially, this can be seen as a 1D dice loss calculated between the projected masks and the edges of the ground truth boxes along both the x and y axes. Given that the primary objective of grounding is to accurately select the correct mask, we assign different weights to these loss components. The mask dice loss and box projection loss are both weighted at 1, while the mask bce loss is given a lower weight of 0.1. The final loss calculation is a summation of the language modeling loss $\mathcal{L}_{\text{lm}}$ and these mask-related losses.

LLM Configuration We adopt the Vicuna-7B model [11] as our base LLM, and use the OpenAI CLIP@336 [62] model and DINOv2-L/14-reg[15] pretrained checkpoints. We use the original conversation template from Vicuna, where all the interactions are formatted as `<system_message> <s> USER: <utterance> ASSISTANT: <utterance> </s>`.

Parameter-Efficient Training. We adopt the LLaMA2-7B model [70] as our base LLM, and initialized the weight from LLaVA-1.5 [44]. For the vision encoders, we use the OpenAI CLIP@336 [62] model and DINOv2-L/14-reg [15] pretrained checkpoints. We freeze all the parameters of the Mask2Former+, the CLIP, and the DINOv2 model during training. We use Low-Rank Adaptation (LoRA) [28] with $r = 16$ and $\alpha = 16$ to tune the LLM, including all the linear layers, input embeddings, and the LM head. We train all the new components introduced for connecting these models, including the MLP projection layer of CLIP and DINOv2, and the mask retrieval head. As a result, less than 2% of the total parameters are trainable in the whole model. We use the AdamW optimizer [48] with an initial learning rate of 2e-4 and a cosine annealing rate. We train our model on the balanced sampled M3G2 dataset for 2 epochs, which takes around 2 days using 8 40G A100 GPUs.

D. Additional Qualitative Results

We provide additional qualitative examples generated by GROUNDHOG. See Figure 9 & 10 for GIC, Figure 11 for RES, Figure 12 for GVQA, and Figure 13 & 14 for RD, respectively.

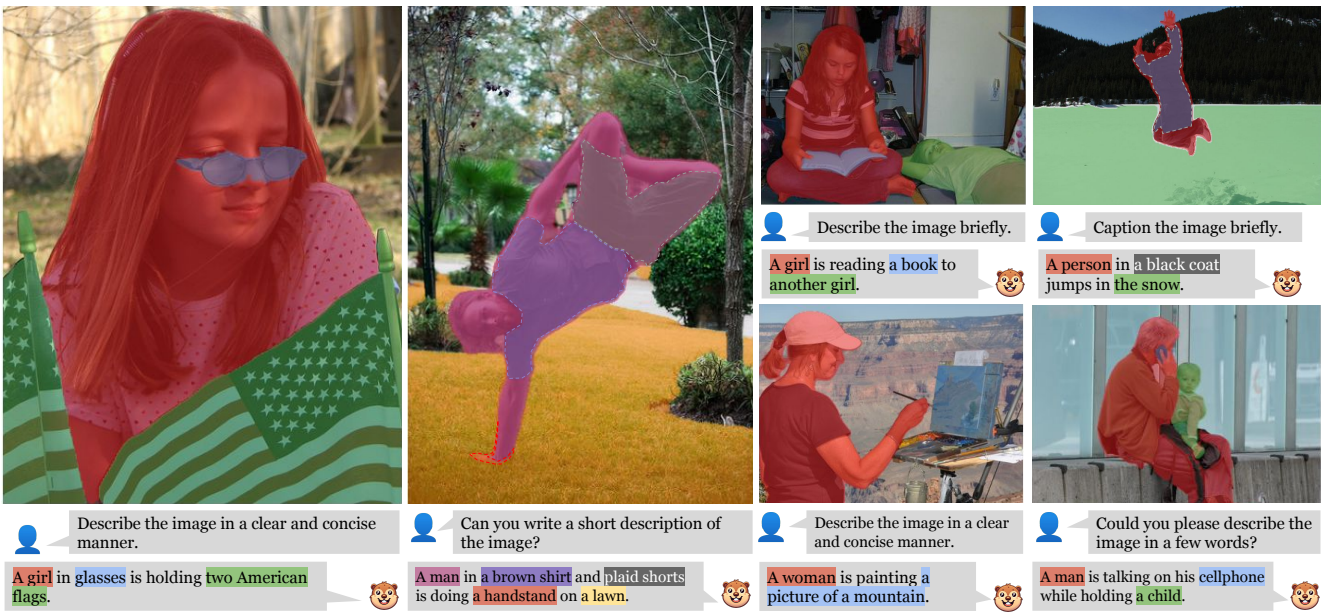


Figure 9. Additional qualitative examples for short grounded captioning.

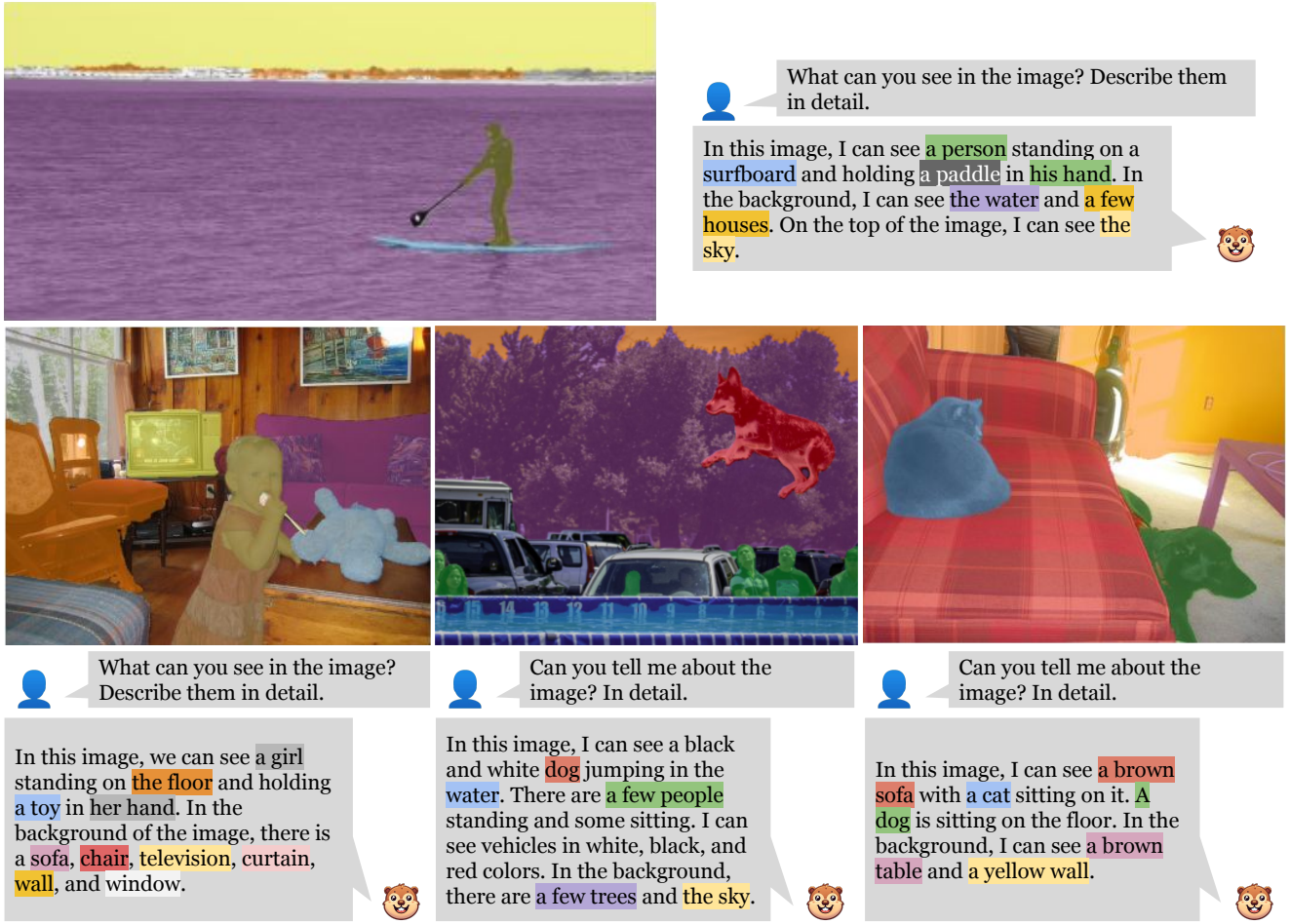


Figure 10. Additional qualitative examples for detailed grounded captioning.

Figure 11. Additional qualitative examples for referring expression segmentation.

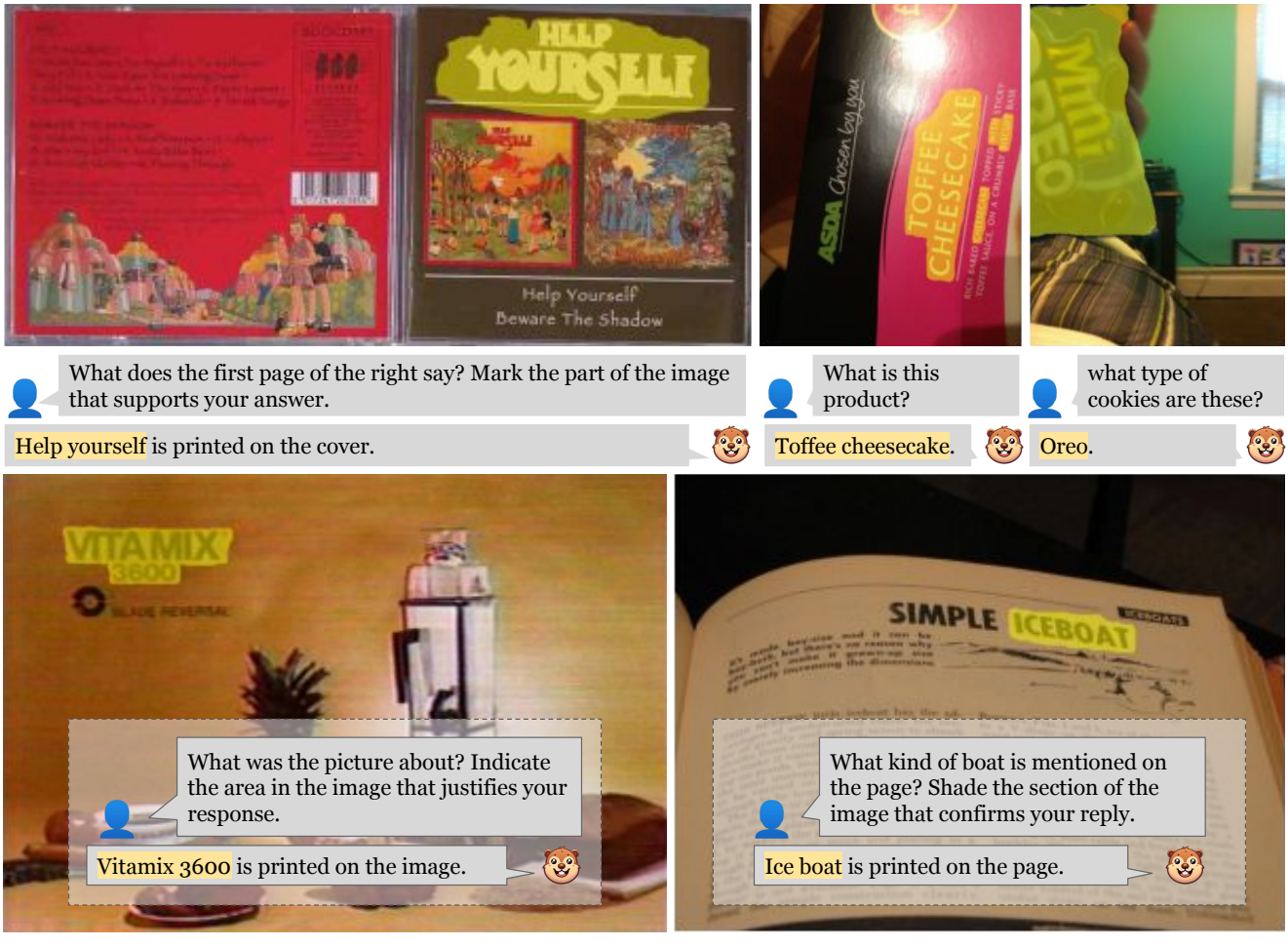


Figure 12. Additional qualitative examples for grounded VQA of visual text.

Figure 13. Additional qualitative examples for referential dialogue with pointer input.

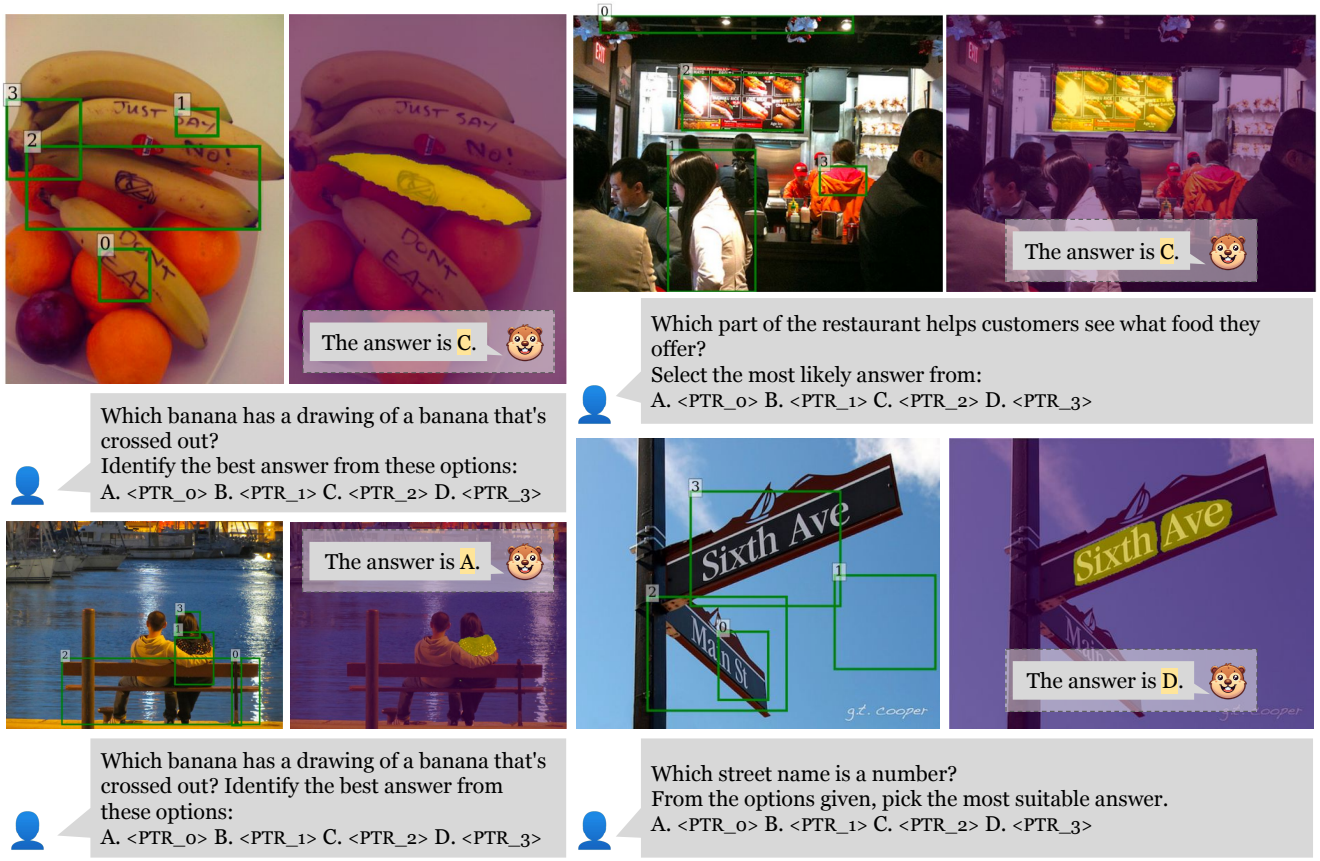
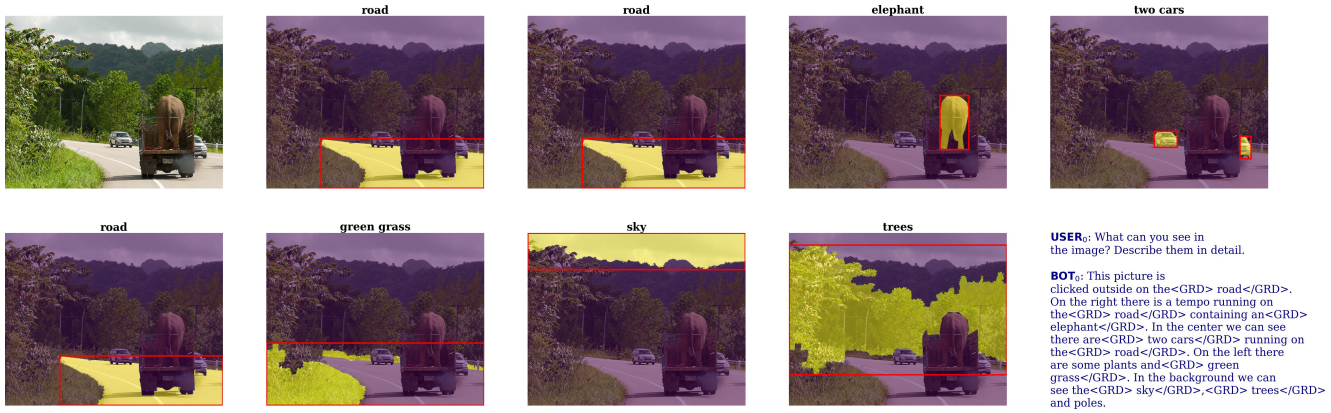


Figure 14. Additional qualitative examples for referential dialogue with pointers as multiple choices input.

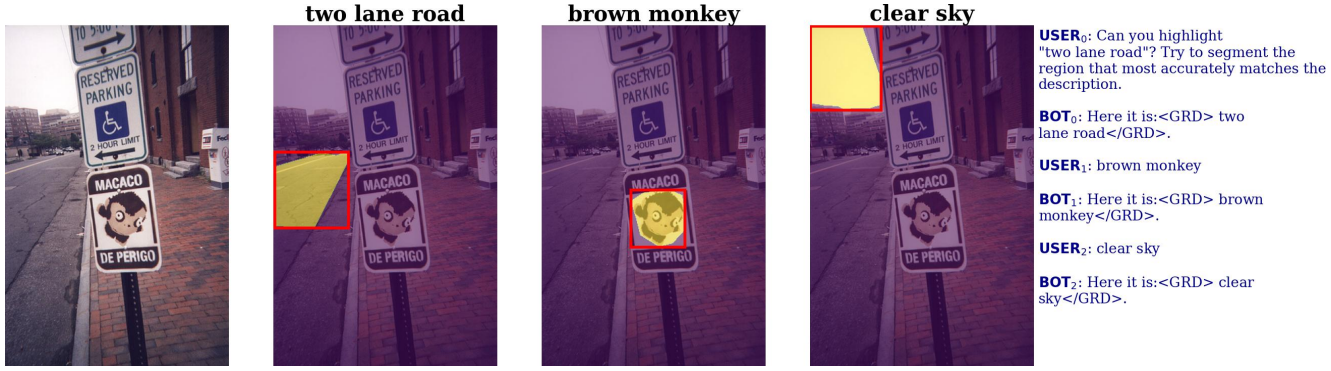


(a) Example of the detailed captioning task originated from PNG.

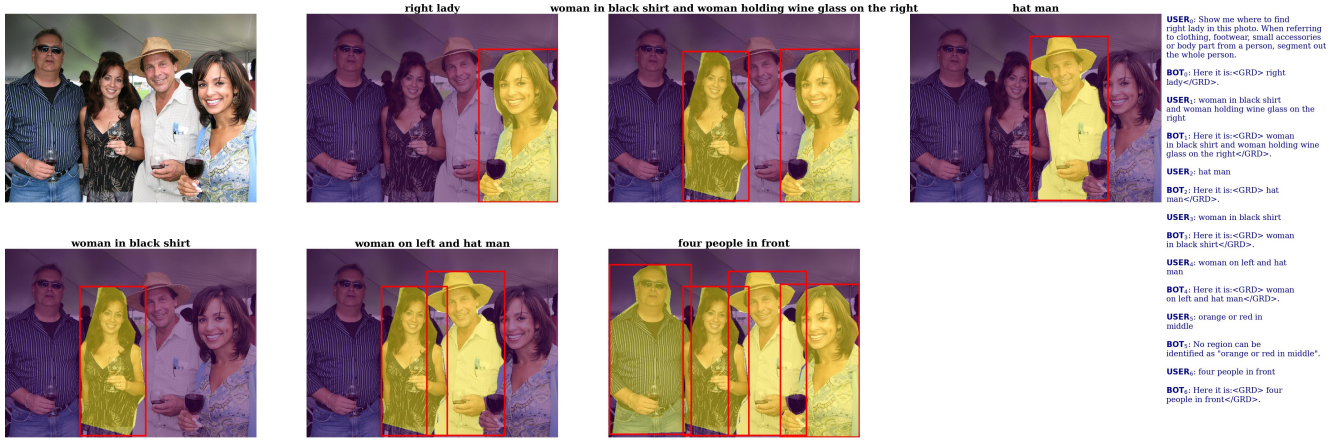
(b) Example of the short captioning task originated from Flickr30K-Entity.

Figure 15. Data samples for the GCAP task.

(a) Example of the referential expression segmentation task originated from RefCOCO+.

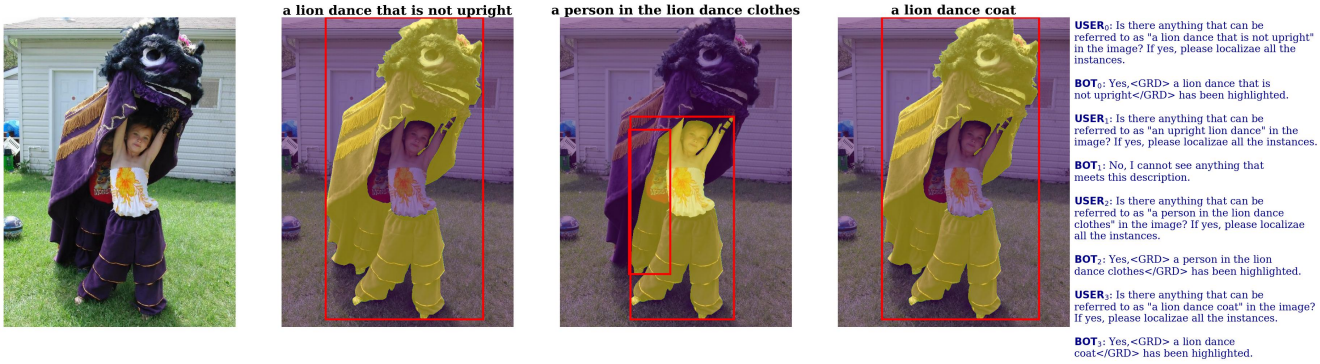


(b) Example of the referential expression segmentation task originated from PhraseCut.



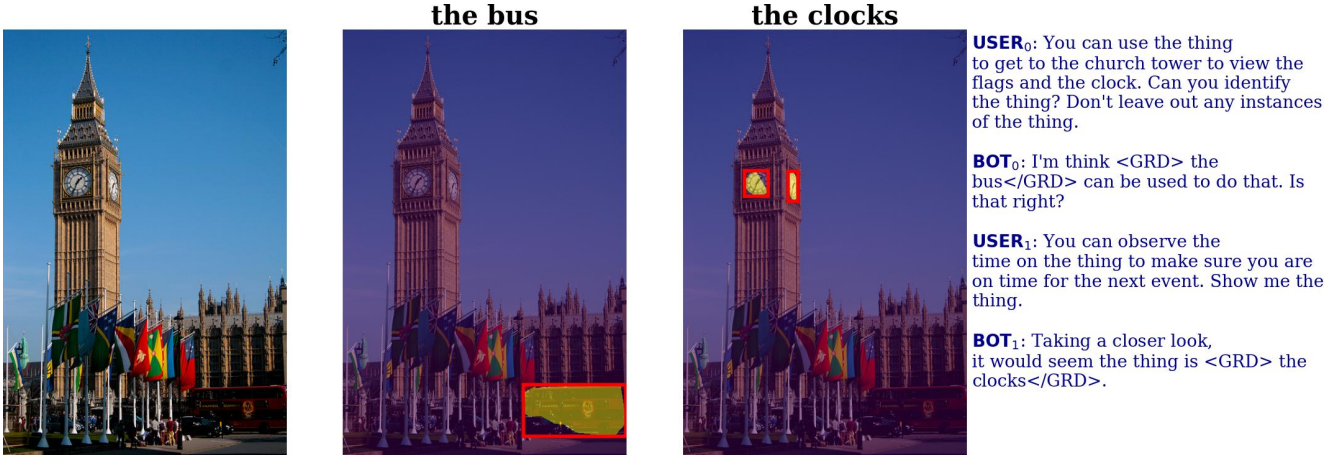
(c) Example of the generalized referential expression segmentation task originated from gRefCOCO.

Figure 16. Data samples for the RES task (part 1).

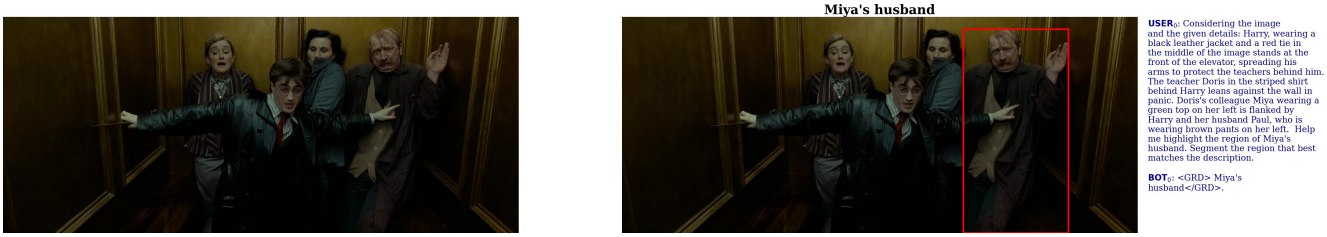


(a) Example of the generalized referential expression segmentation task originated from D-Cube.

(b) Example of the reasoning-based referential expression segmentation task originated from ReasonSeg.



(c) Example of the reasoning-based referential expression segmentation task originated from RIO.



(d) Example of the reasoning-based referential expression segmentation task originated from SK-VG.

Figure 17. Data samples for the RES task (part 2).

USER$_0$: Give a short answer to the question "Can you tell me the content of this notice?" based on the image. Show where in the image you found your answer.

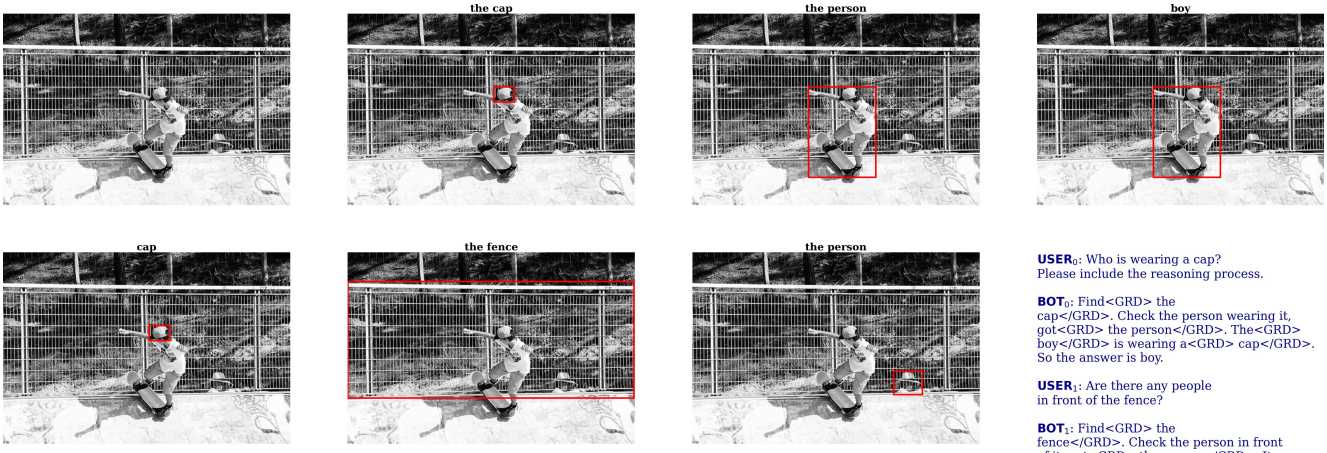
BOT$_0$: <GRD>No smoking</GRD>.

(a) Example of grounded VQA originated from VizWiz-Grounding.

USER$_0$: What are the two words on the bottom of the screen? Please highlight your evidence in the image.

BOT$_0$: <GRD>Livecash midland</GRD> are words at the bottom of the screen.

(b) Example of grounded VQA originated from TextVQA-X.



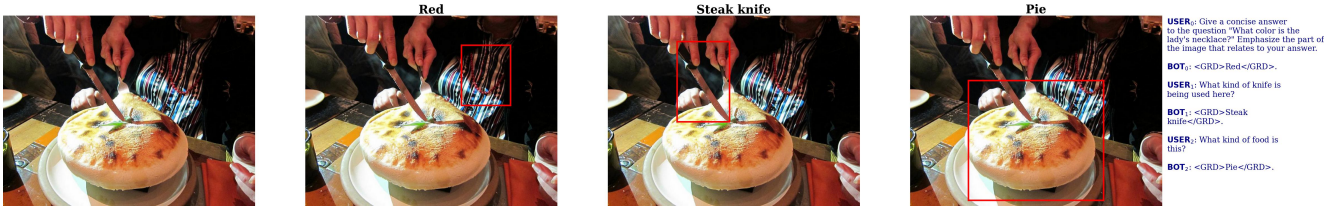
USER$_0$: Who is wearing a cap? Please include the reasoning process.

BOT$_0$: Find<GRD> the cap</GRD>. Check the person wearing it, got<GRD> the person</GRD>. The<GRD> boy</GRD> is wearing a<GRD> cap</GRD>. So the answer is boy.

USER$_1$: Are there any people in front of the fence?

BOT$_1$: Find<GRD> the fence</GRD>. Check the person in front of it, got<GRD> the person</GRD>. It exists. The answer is no.

(c) Example of grounded VQA with chain-of-thought reasoning originated from GQA.



USER$_0$: Give a concise answer to the question "What color is the lady's necklace?" Emphasize the part of the image that relates to your answer.

BOT$_0$: <GRD>Red</GRD>.

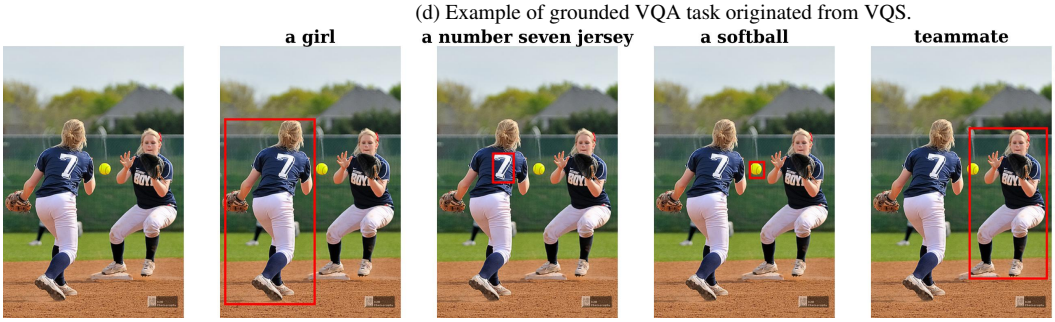
USER$_1$: What kind of knife is being used here?

BOT$_1$: <GRD>Steak knife</GRD>.

USER$_2$: What kind of food is this?

BOT$_2$: <GRD>Pie</GRD>.

(d) Example of grounded VQA task originated from VQS.

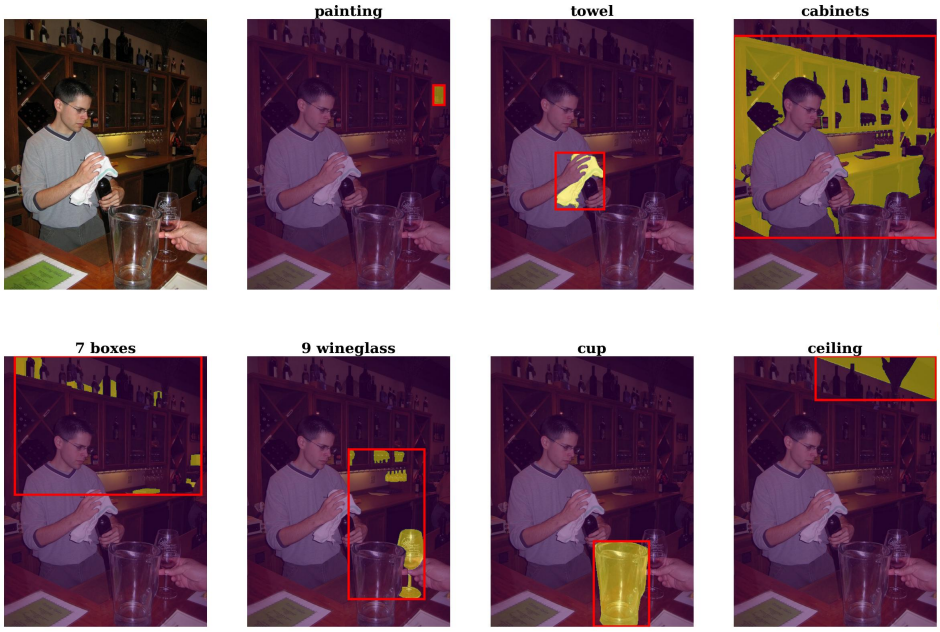


USER$_0$: Is there an event "a girl with a number seven jersey throws a softball to her teammate" happening in the picture? Let's think step by step.

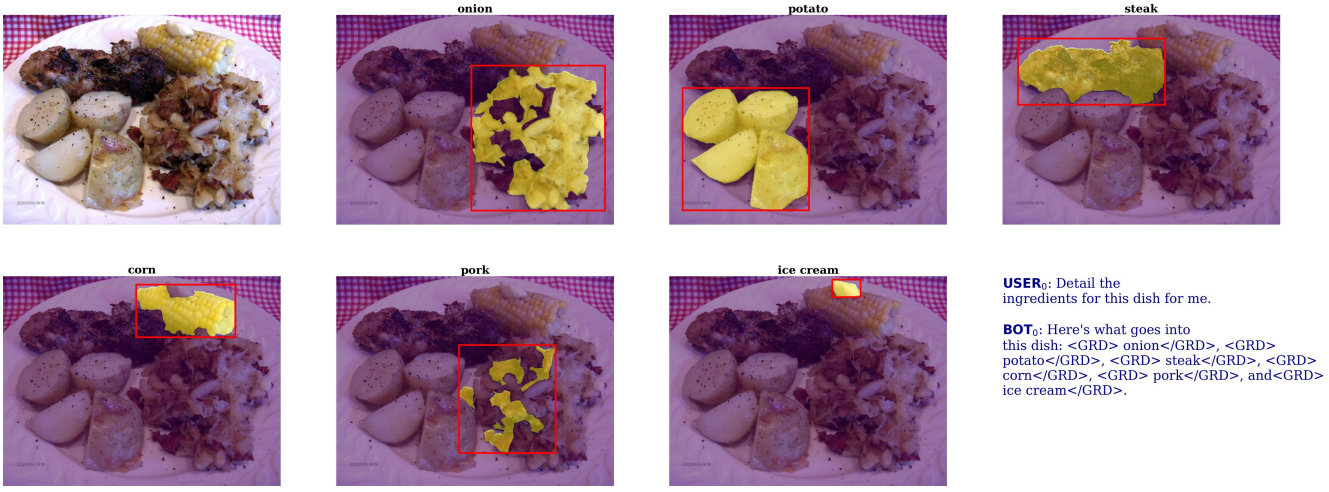
BOT$_0$: In the picture,<GRD> a girl</GRD> is wearing<GRD> a number seven jersey</GRD>. She is indeed throwing<GRD> a softball</GRD> to her<GRD> teammate</GRD>. Based on these details, the event "a girl with a number seven jersey throws a softball to her teammate" is happening in the picture. Therefore, the answer is yes.

(e) Example of the grounded VQA with chain-of-thought reasoning originated from Shikra-BinaryQA.

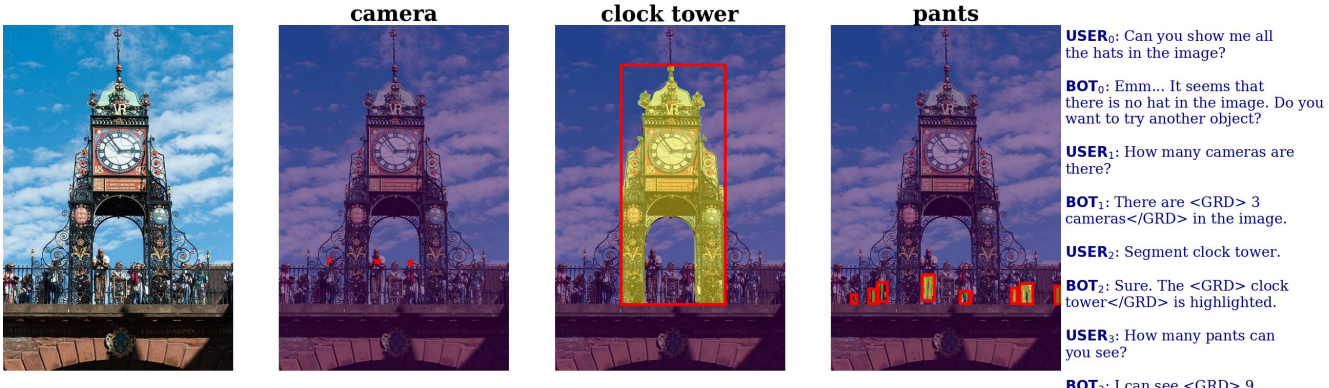
Figure 18. Data samples of the GVQA task (part 1).



(a) Example of the grounded VQA task originated from EntityCount.



(b) Example of the grounded VQA task originated from FoodSeg-QA.



(c) Example of the grounded VQA task originated from LVIS-QA.

Figure 19. Data samples of the GVQA task (part 2).

USER$_0$: Please describe<0> in the image in a way that it can be uniquely identified.

BOT$_0$: <GRD> brown cap pouring into cup</GRD>.

USER$_1$: Another one: <1>.

BOT$_1$: <GRD> a brown sofa</GRD>.

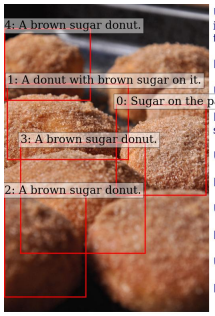
USER$_2$: Another one: <2>.

BOT$_2$: <GRD> cup or vase cut off</GRD>.

USER$_3$: Another one: <3>.

BOT$_3$: <GRD> the cup with a straw in it</GRD>.

(a) Refer expression generation (RefCOCO+).



USER$_0$: Narrate what you see in the indicated area<0>. No grounding tokens are needed.

BOT$_0$: Sugar on the pastry.

USER$_1$: <1>

BOT$_1$: A donut with brown sugar on it.

USER$_2$: Good. Next:<2>.

BOT$_2$: A brown sugar donut.

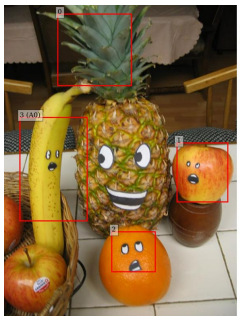
USER$_3$: What about<3>?

BOT$_3$: A brown sugar donut.

USER$_4$: <4>.

BOT$_4$: A brown sugar donut.

(b) Region captioning (VG).



USER$_0$: Which fruit is mostly yellow? Select the most likely answer from: A.<0> B.<1> C.<2> D.<3>

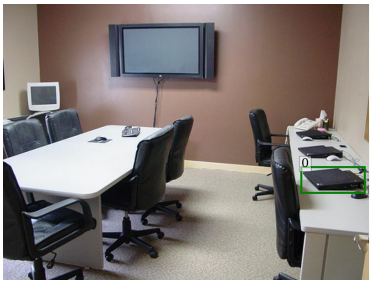
BOT$_0$: The answer is<GRD> D</GRD>.

(c) Example of the referring QA task from V7W.

USER$_0$: What color is this kite? <0> Answer with a single word.

BOT$_0$: Yellow.

(d) Example of the referring QA task originated from PointQA-Local.



USER$_0$: How many of these are there? <0> Answer with a single word.

BOT$_0$: 3.

(e) Example of the referring QA task originated from PointQA-Twice.

USER$_0$: What would happen if person 0<0> tripped?

BOT$_0$: <GRD> person 1</GRD>, and<GRD> person 2</GRD> would laugh.

USER$_1$: Can you explain the basis for your thinking?

BOT$_1$: Children laugh at each other when something silly of clumsy happens.

(f) Example of the referring QA task originated from VCR.

USER$_0$: Provide an answer to the question "What is the girl doing with the cheese<0>?". Kindly include your thought process as well.

BOT$_0$: <GRD> The girl</GRD> is cutting<GRD> the Velveeta cheese</GRD> into cubes with<GRD> a knife</GRD>. She is putting the cheese cubes into<GRD> a bowl</GRD>.

(g) Example of the referential dialog task originated from ShikraRD.

USER$_0$: What color is the street sign attached to the pole in front of the store?

BOT$_0$: The street sign attached to the pole is black and white as seen from the region <GRD>the sign is black and white</GRD>.

USER$_1$: Is there any object placed outside near the store?

BOT$_1$: Yes, there are <GRD>things placed outside</GRD> near the store.

USER$_2$: What is the name mentioned on the sign above the store?

BOT$_2$: The sign above the store says <GRD>coach</GRD>.

USER$_3$: What does the writing on the window mention?

BOT$_3$: The writing on the window says <GRD>The Madison Collection</GRD>.

USER$_4$: What is the color of the horse and carriage?

BOT$_4$: The <GRD>horse and carriage</GRD> are red.

USER$_5$: Is there a depiction of a woman with an accessory? If so, what is it?

BOT$_5$: Yes, there is a depiction of a <GRD>woman</GRD> modeling a handbag.

USER$_6$: Where can one find the address of the restaurant in the image?

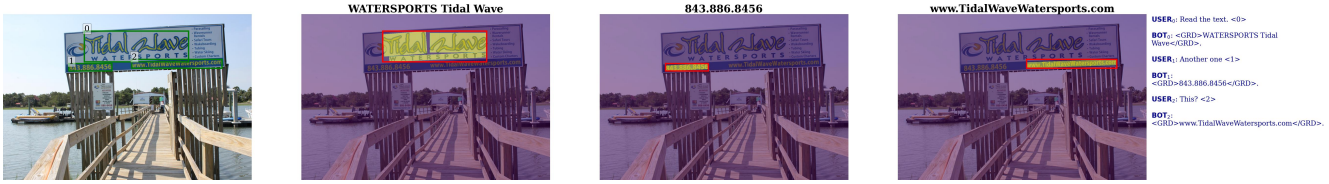
BOT$_6$: The <GRD>address of the restaurant</GRD> is displayed on the image.

(h) Example of the referential dialog task originated from SVIT.

Figure 20. Data samples of the RD task (part 1).

(a) Example of the guesswhat game originated from GuessWhat.

(b) Example of the referential region matching game originated from VG.



(c) Example of the referred text reading task originated from HierText.

Figure 21. Data samples of the RD task (part 2).