POPEN: Preference-Based Optimization and Ensemble for LVLM-Based Reasoning Segmentation

Supplementary Material

A. Discussion of Computation

Training Time. As detailed in Sec.3.4 of the main paper, our method consists of three training stages. The first stage follows the method of PixelLM [8], training for 10 epochs, which takes approximately 1.5 days for the 7B model on 8 A100 GPUs. The second and third stages train for 2 epochs each, requiring approximately 5 hours and 8 hours, respectively.

Inference Time. Our proposed preference-based ensemble method needs to generate K different responses and fuses them. In our experiments, K is set to 3. Theoretically, this would require 4 times the computation compared to the the original PixelLM w/o ensemble. However, benefiting from optimizations such as parallel computation and KV cache, in practice, the averge inference time of our method is only 1.57 times that of the method w/o ensemble. This is entirely acceptable considering the significant improvement brought by our preference-based ensemble approach. Also note that even without using the ensemble method that requires additional computational cost, our method can still significantly outperform the baseline PixelLM, as shown by the results for POPEN† in Table.1 of main paper. This further demonstrates the superiority of our approach.

B. More Details of Proposed Method

ChatGPT Prompt for Response Correction. As introduced in Sec.3.2 of main paper, we use ChatGPT to refine LVLM's response y by modifying, adding, or deleting certain words or sentences in y, thus generating a corrected response y_c with fewer errors to construct the text semantic preference data. The ChatGPT prompt format for this operation is as follows:

You are an assistant designed to help me correct an incorrect answer to a question about an image. I will provide you with an image, a question, an answer from an LVLM, and an object list. You need to modify, add, or delete certain words or sentences in the LVLM's answer to correct mistakes, including incorrect objects and faulty reasoning. The corrected answer should include only the objects in the object list. You should return: (1) The original LVLM's answer I provided, in which you should mark the deleted or modified parts in the answer in quotes. (2) Your corrected answer, in which you should mark the modified or added parts compared to the original answer in quotes. Please ensure that only the modified, deleted, or added parts are marked. Do not mark synonyms as modifications. Please retain the sentence structure and content of the LVLM's original answer as much as possible, without adding extra information beyond what is necessary for correction.

In this prompt, the object list refers to a list containing the names of all objects within the ground truth response.

ChatGPT Prompt to Intentionally Introduce Errors. As introduced in Sec.3.2 of the main paper, to enrich dataset, for some of the LVLM's responses that contain only few errors, we instruct ChatGPT to intentionally introduce errors into the ground truth response y_g to formulate y. Specifically, if ChatGPT finds that an LVLM response has no errors, we use the randomness in decoding to generate three different responses and select one containing errors as y for the text semantics preference. If these responses still contain no errors, we use the following prompt to intentionally introduce errors into the ground truth response y_g:

You are an assistant designed to help me intentionally introduce errors into a correct answer to a question about an image. I will provide you with an image, a question, and a correct answer. You need to modify, add, or delete certain words or sentences in the correct answer to introduce some mistakes, such as incorrect objects and faulty reasoning. You should return the modified answer. Please introduce errors into only a small portion of the content (e.g., one or two objects). Please do not perform synonym replacement.

Loss for Preference-Based Ensemble. As indicated in Sec.3.4 of the main paper, in the third training stage of our method, which aims to optimize the preference-based ensemble capability, we employ a specially designed loss function to ensure that the refined text responses and segmentation outperform the originals. The loss function for this stage is the sum of two components: the text improvement loss $\mathcal{L}_{ti}$ and the segmentation improvement loss $\mathcal{L}_{si}$. To be specific, denote the K generated responses as $\{y_k\}_{k=1}^{K}$, the refined response as $\tilde{y}$ and the ground truth response as y^g, $\mathcal{L}_{ti}$ is formulated as follows:

$$
\begin{aligned}
&h_k = \mathrm{BERT}\left(y_k\right),\ \tilde{h} = \mathrm{BERT}\left(\tilde{y}\right),\ h^g = \mathrm{BERT}\left(h^g\right),\\
&\mathcal{L}_{ti} = -\mathbb{E}\frac{1}{K}\sum_{k=1}^{K}\log\sigma\left(10p(\tilde{y})\left(\mathrm{Cos}(\tilde{h},h^g)-\mathrm{Cos}(h_k,h^g)\right)\right),
\end{aligned}
\tag{1}
$$

where $h = \mathrm{BERT}\left(y\right)$ refers to a feature extracted from BERT with y as input, Cos denotes cosine similarity. $p(\tilde{y})$ refers to the probability of the LVLM generating $\tilde{y}$. This

config	value
optimizer	AdamW
base learning rate	3.0e-4
weight decay	0
optimizer monmentum	β_1, β_2=0.9,0.95
batch size	16
learning rate schedule	WarmipDecayLR
warmup iterations	100
augmentations	None

Table 1. Training settings

loss constrains the similarity between the refined response $\tilde{y}$ and the ground truth response y^g to be higher than that between the original responses y_k and y^g, thus optimizing the model to produce more refined response outperforming the original ones. Similarity, the segmentation improvement loss $\mathcal{L}_{si}$ is computed as:

$$\mathcal{L}_{ti} = -\mathbb{E}\frac{1}{K}\frac{1}{N}\sum_{k=1}^{K}\sum_{n=1}^{N}\log\sigma(10(\mathrm{IoU}(\tilde{M}^n, M_g^n) - \mathrm{IoU}(M_k^n, M_g^n))), \tag{2}$$

where N is the number of segmentation targets in the response, $\tilde{M}^n$ is the n-th refined segmentation mask, M_k^n is the n-th segmentation mask from the k-th original response, M_g^n is the corresponding ground truth mask.

More Implementation Details. Some implementation details of our method have been presented in Sec.4.1 of the main paper. Most of the other training settings follow PixelLM and are presented in Table 1. Note that we use the exact same settings shown in Table 1 for all three training stages (detailed in main paper Sec.3.4) in our method. The number of learnable prompt embeddings $\hat{p}$ used in the preference-based ensemble is 10.

ChatGPT Prompt for Response Evaluation. As indicated in Sec.4.1 of main paper, for a more comprehensive evaluation of the LVLM's text responses, we prompt ChatGPT to evaluate the correctness of the LVLM's response given the input image-instruction pair. In this way, a score is generated from ChatGPT to assess the quality of the response. The prompt for ChatGPT in this operation is as follows:

I will give you an image, a question and a text response. You are required to score the performance of the text response given the image and question. You should pay extra attention to the hallucination, which refers to the part of responses that are inconsistent with the image content, such as claiming the existence of something not present in the image or describing incorrectly in terms of the counts, positions, or colors of objects in the image. Please rate the responses of the assistants on a scale of 1 to 10, where a higher score indicates better performance, according to the following criteria: (1) Accuracy: whether the response is accurate with respect to the image content and reasoning logic. Responses with fewer hallucinations should be given higher scores. (2) Detailedness: whether the response is rich and complete in necessary details. Please output a score for such a evaluation. Following the score, please provide an explanation of your evaluation.

C. Further Analysis

Correlation Between Preference Score and Response Quality. In the proposed preference-based ensemble method, we calculate a preference score based on prediction likelihood to modify each attention matrix in LVLMs, enabling the model to focus more on high-reliability content when integrating multiple text responses. This is based on the property that, after finetuning using the preference optimization method, the prediction likelihood of tokens in the response can reflect the extent to which they align with human preferences. We evaluate this property using the Pearson correlation coefficient. Specifically, for each token y^i in the LVLM's text response y, we calculate a score τ^i by summing the likelihood of y^i with the average likelihood of all tokens in the sentence to which y^i belongs. We then employ ChatGPT to score the preference for each token in the text response based on accuracy, obtaining c^i for y^i. The Pearson correlation coefficient r is then calculated as:

$$r = \frac{\sum_{i=1}^{N_i}(\tau_i - \bar{\tau})(c_i - \bar{c})}{\sqrt{\sum_{i=1}^{N_i}(\tau_i - \bar{\tau})^2}\sqrt{\sum_{i=1}^{N_i}(c_i - \bar{c})^2}}, \tag{3}$$

where N_i is the number of tokens in the text response y. A higher r indicates a stronger positive correlation between τ^i and the accuracy of the token y^i. We calculate r across the responses from all image-instruction pairs in the MUSE validation set, and the high average value of 0.76 for r demonstrates the strong correlation. This result highlights the validity of our attention design in Eq.5 of main paper.

Correlation Between Preference Score and Segmentation Performance. We use the same method as in the previous section, employing the Pearson correlation coefficient r to measure the correlation between the average prediction likelihood of all tokens in a sentence and the accuracy of the segmentation target contained in the sentence. Across the entire MUSE validation set, the model finetuned with preference optimization achieves a high average r value of 0.69, indicating a strong positive correlation and demonstrating the rationale behind our designs in Eq. 6 of the main paper for multi-segmentation integration. One possible explanation for this property is that the sequential prediction process of the LVLM would propagate errors and uncertainties from earlier tokens in the text response to subsequent segmentation tokens, while also transmitting errors in the

Model	Validation Set					Test Set				
	M	C	AP50	mIoU	Recall	M	C	AP50	mIoU	Recall
BuboGPT [12]	17.2	3.6	19.1	54.0	29.4	17.1	3.5	17.3	54.1	27.0
Kosmos-2 [6]	16.1	27.6	17.1	55.6	28.3	15.8	27.2	17.2	56.8	29.0
LISA [3]	13.0	33.9	25.2	62.0	36.3	12.9	32.2	24.8	61.7	35.5
GLaMM [7]	16.2	47.2	30.8	66.3	41.8	15.8	43.5	29.2	65.6	40.8
POPEN	**20.3**	**52.8**	**34.9**	**70.1**	**45.2**	**20.1**	**49.4**	**33.8**	**69.7**	**44.0**

Table 2. **Performance on the grounded conversation generation (GCG) task of GranD$_f$ Dataset.** Metrics include METEOR (M), CIDEr (C), AP50, mIoU, and Mask Recall. Our POPEN achieves the best performance.

Method	gIoU	cIoU
LISA [3]	52.9	54.0
GSVA [10]	50.5	56.4
POPEN	**60.2**	**64.5**

Table 3. Performance on ReasonSeg benchmark.

Model	**ObjHal**		**MMHal**	
	Resp.↓	Mention↓	Info.↑	Resp.↓
LLaVA	63.0	29.5	31.9	70.8
RLHF-V	12.2	7.5	40.0	52.1
POPEN	**9.2**	**4.9**	**42.1**	**47.7**

Table 4. Results on hallucination benchmarks.

Method	gIoU ↑	cIoU ↑	C_S ↓	C_I ↓
Baseline (PixelLM)	41.9	48.9	22.0	9.8
OPERA [2]	42.3	49.5	15.0	6.8
VCD [4]	42.3	49.3	16.9	7.3
HALC [1]	42.2	49.6	14.2	6.7
POPEN [1]	**45.4**	**55.2**	**9.3**	**4.3**

Table 5. Comparison with other hallucination mitigation methods on MUSE.

segmentation embedding to later tokens. Consequently, the segmentation accuracy becomes strongly positively correlated with the likelihood of the sentence it belongs to, which reflects the sentence's accuracy and quality.

D. More Experiments

D.1. Comparison on More Benchmarks

Results on Grounded Conversation Generation of GranD$_f$ Benchmark. Grounded conversation generation is a task aimed at generating text captions for images as well as segmentation masks for each object within them. To evaluate our method on this task, we follow GLaMM [7] by first pretraining the model using the approach described in the main paper, and then finetune it on the GranD$_f$ dataset. The results of the finetuned model on the validation set and test set of GranD$_f$ are presented in Table 2. Our POPEN significantly outperforms previous state-of-the-art approaches such as LISA and GLaMM, demonstrating the high effectiveness and superiority of our method.

Results on ReasonSeg Benchmark. We further evaluate our method on the ReasonSeg [3] validation set and compare its performance with LISA and GSVA. The results are presented in Table 3. Our POPEN achieves the best performance, with significant advantages over the second-best method, showing a +7.3% improvement on the gIoU metric and +8.1% on the cIoU metric. These results demonstrate the outstanding performance of our method.

Results on Hallucination Benchmarks. We further evaluate the effectiveness of our method in mitigating hallucination on two hallucination benchmarks, ObjHal and MMHal. As shown in Table 4, our POPEN outperforms both the baseline LLaVA [5] and RLHF-V [11], which is specifically designed to address hallucination. This demonstrates the superior effectiveness of our method in mitigating hallucination through the use of additional segmentation training data and the novel techniques we propose in this work.

D.2. Hallucination Mitigation on MUSE

Some previous works have explored ways to mitigate hallucinations in LVLMs. We compare our method with these approaches on the MUSE validation set, and the results are presented in Table 5. Although these previous methods can alleviate hallucination to some extent compared to the baseline, our approach significantly outperforms them, with substantially reduced C_S and C_I metrics. Moreover, due to the lack of segmentation-specific designs in previous methods, they fail to achieve significant improvements in segmentation accuracy. In contrast, benefiting from preference-based optimization and ensemble techniques specifically designed for segmentation, our method, POPEN, greatly enhances segmentation metrics including gIoU and cIoU, demonstrating the superiority of our approach.

D.3. Improvement of Target Localization

As indicated in Sec.3.2 of the main paper, during the first half of finetuning, we collect perturbed images with varying localization accuracy as preference data $\mathcal{P}_s$ for optimiza-

Method	gIoU ↑	cIoU ↑	C_S ↓	C_I ↓
LLaVA + PixelLM	41.9	48.9	22.0	9.8
Qwen2-VL + PixelLM	42.8	50.5	15.2	7.3
LLaVA + POPEN	45.4	55.2	9.3	4.3
Qwen2-VL + POPEN	**46.1**	**56.4**	**8.1**	**3.8**

Table 6. Effectiveness on the stronger LVLM Qwen2-VL.

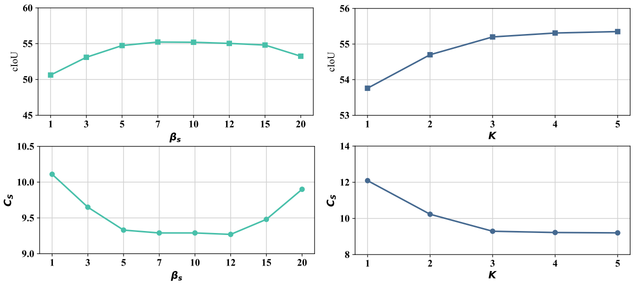
tion. To further validate the effectiveness of this method, we conduct a quantitative comparison of target localization precision between models finetuned using randomly generated $\mathcal{P}_s$ and those finetuned with $\mathcal{P}_s$ generated by our method. Specifically, for each object mask generated by the model, we calculate its IoU IoU_g with the ground truth object mask, as well as its maximum IoU IoU_o with other objects in the image (provided by SAM) beyond the ground truth object. We then define a mask as having target localization error if $IoU_g < 0.75$ and $IoU_o > 0.25$, and we compute the proportion p of such wrongly-located objects among all objects in the MUSE validation set. On this metric p, the model finetuned with randomly generated $\mathcal{P}_s$ achieves a score of 23.3%, while the model finetuned with $\mathcal{P}_s$ generated by our method reduces this to 6.5%. This demonstrates the significant improvement on the model's target localization capability brought by our method.

D.4. Results on Stronger LVLMs

In addition to LLaVA, several more advanced LVLMs have been proposed recently, offering better performance and reduced hallucination for different tasks. We further evaluate the effectiveness of POPEN when integrated with these stronger LVLMs. As shown in Table 6, replacing LLaVA with a stronger Qwen2-VL-7B [9] does lead to some performance improvement, including better segmentation quality and reduced hallucination. However, even when using a weaker LVLM, LLaVA + POPEN still outperforms Qwen2-VL + PixelLM (with a stronger LVLM), demonstrating that simply relying on a stronger LVLM is not sufficient; while using our novel methods in POPEN can yield larger improvements. Additionally, the notable advantage of Qwen2-VL + POPEN over Qwen2-VL + PixelLM further highlights the ability of our method to improve performance even under a stronger LVLM, demonstrating its high effectiveness and generalizability for different base models.

D.5. Ablation Study of Hyperparameters

In our method, the hyperparameters β_t in Eq.2 and λ in Eq.3 of the main paper follow the same settings as RLHF-V [11]. Therefore, we primarily focus on validating the remaining hyperparameters in our approach, including β_s in Eq.4 of the main paper, and the number K of generated responses in the preference-based ensemble method. Both the text metric C_S and segmentation metric cIoU are reported. (higher cIoU and lower C_S are better.)



(a) Ablation Study of β_s **(b) Ablation Study of K**

Figure 1. Ablation study of hyperparameters β_s and K on the cIoU metric (the first row) and C_S metric (the second row). Higher cIoU and lower C_S indicate better results.

Ablation Study of β_s. β_s is a scaling factor used in Eq.4 of the main paper to compute the segmentation preference loss. As shown in Figure 1(a), overly small or large values of β_s can lead to performance degradation. However, the performance can remain consistently stable when $5 < \beta_s < 15$, demonstrating the robustness of our method to the choice of β_s.

Ablation Study of K. We further evaluate the impact of K, with the results presented in Figure 1(b). Increasing K enhances performance, as the quality of the refined results can be improved through the fusion of more responses. However, performance plateaus when $K > 3$, with only marginal gains observed upon further increases. Therefore, we select $K = 3$ as our setting.

D.6. Further Ablation of Curriculum Collection

As detailed in Sec.3.2 of the main paper, we employ a curriculum method for obtaining segmentation embedding preference data $\mathcal{P}_s$, collecting different types of $\mathcal{P}_s$ for the first and second halves of finetuning. In Table 5 of the main paper, we conduct an ablation study to compare our method with random collection or using the same strategy throughout both halves of the finetuning process. In this Supp, we further compare with a hybrid approach, where samples collected using the first-half strategy and those collected using the second-half strategy are both used throughout the entire finetuning process. As shown in Table 7, this hybrid approach outperforms random collection but remains significantly inferior to our curriculum-based method. This demonstrates the importance of sequentially learning fundamental and advanced skills in our proposed method.

D.7. More Qualitative Comparison

In Figure 2, we present more examples comparing text responses and segmentation results between our POPEN and PixelLM [8]. In these examples, PixelLM suffers from se-

Collection Method	gIoU ↑	cIoU ↑	C_S ↓	C_I ↓
Curriculum Collection	45.42	55.20	9.29	4.31
Random	43.06	51.35	9.78	4.62
Hybrid Approach	44.19	53.75	9.48	4.55

Table 7. Effectiveness of different methods for segmentation preference data collection. "Hybrid approach" refers to samples collected using the first-half strategy and those collected using the second-half strategy are both used throughout the entire finetuning process.

rious hallucinations, generating objects in its text responses that do not exist within the images, such as the "books" in the second example and "bench in the left side" in the fourth example. Furthermore, the segmentation accuracy is suboptimal, with coarse results in the object boundary regions and even wrong localization of the target objects (such as the segmentation for "cat" in the first example and "Lionel Messi" in the third example). By employing the proposed preference-based optimization and ensemble methods, our POPEN achieves significantly improved results, effectively mitigating hallucination in text responses and enhancing segmentation accuracy. These comparative results demonstrate the high effectiveness and advantage of our method compared to PixelLM.

References

[1] Zhaorun Chen, Zhuokai Zhao, Hongyin Luo, Huaxiu Yao, Bo Li, and Jiawei Zhou. Halc: Object hallucination reduction via adaptive focal-contrast decoding. *arXiv preprint arXiv:2403.00425*, 2024. 3

[2] Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multimodal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427, 2024. 3

[3] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9579–9589, 2024. 3

[4] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13872–13882, 2024. 3

[5] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 3

[6] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 3

[7] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13009–13018, 2024. 3

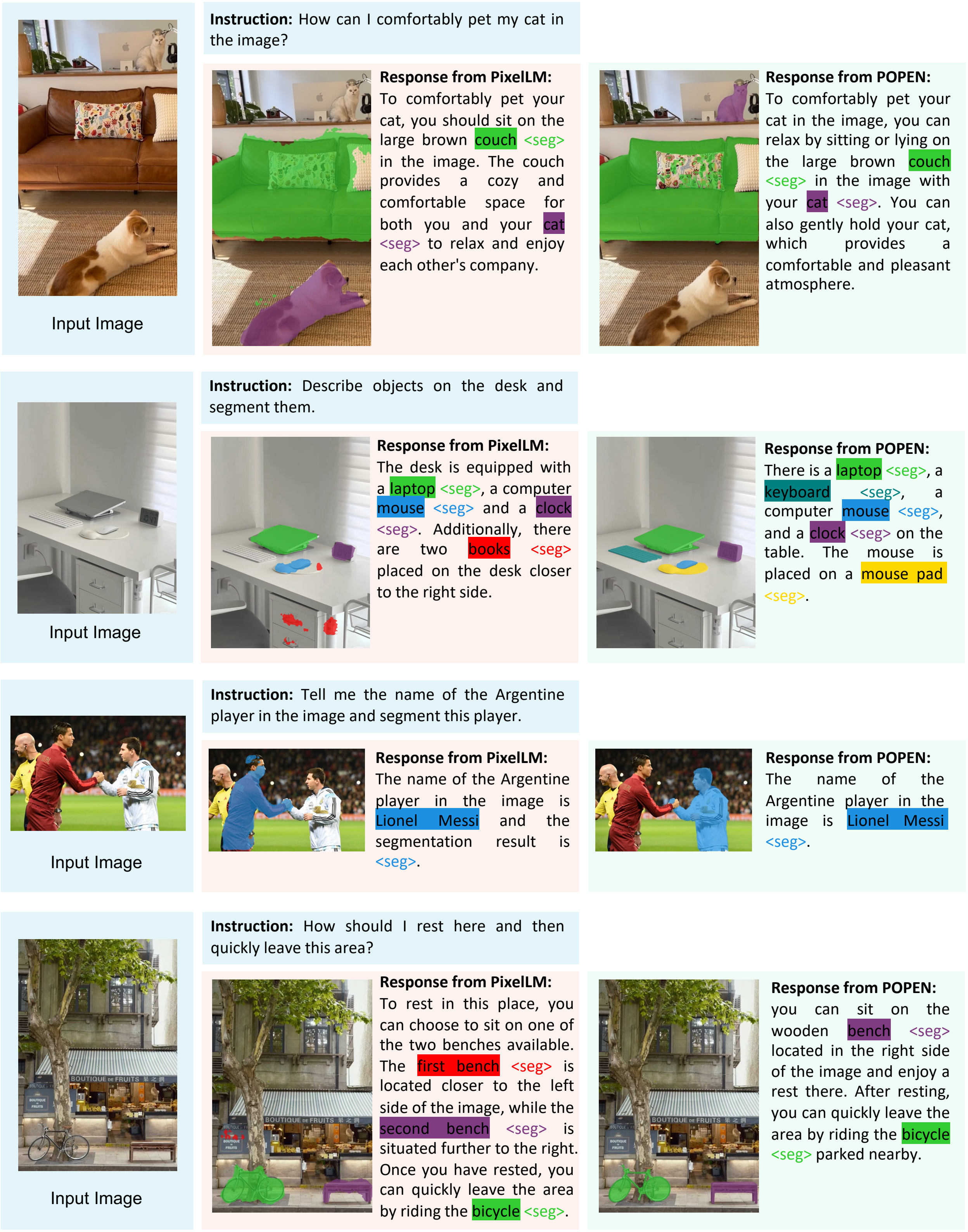
[8] Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. Pixellm: Pixel reasoning with large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26374–26383, 2024. 1, 4

[9] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 4

[10] Zhuofan Xia, Dongchen Han, Yizeng Han, Xuran Pan, Shiji Song, and Gao Huang. Gsva: Generalized segmentation via multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3858–3869, 2024. 3

[11] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, et al. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13807–13816, 2024. 3, 4

[12] Yang Zhao, Zhijie Lin, Daquan Zhou, Zilong Huang, Jiashi Feng, and Bingyi Kang. Bubogpt: Enabling visual grounding in multi-modal llms. *arXiv preprint arXiv:2307.08581*, 2023. 3



Instruction: How can I comfortably pet my cat in the image?

Response from PixelLM: To comfortably pet your cat, you should sit on the large brown couch <seg> in the image. The couch provides a cozy and comfortable space for both you and your cat <seg> to relax and enjoy each other's company.

Response from POPEN: To comfortably pet your cat in the image, you can relax by sitting or lying on the large brown couch <seg> in the image with your cat <seg>. You can also gently hold your cat, which provides a comfortable and pleasant atmosphere.

Instruction: Describe objects on the desk and segment them.

Response from PixelLM: The desk is equipped with a laptop <seg>, a computer mouse <seg> and a clock <seg>. Additionally, there are two books <seg> placed on the desk closer to the right side.

Response from POPEN: There is a laptop <seg>, a keyboard <seg>, a computer mouse <seg>, and a clock <seg> on the table. The mouse is placed on a mouse pad <seg>.

Instruction: Tell me the name of the Argentine player in the image and segment this player.

Response from PixelLM: The name of the Argentine player in the image is Lionel Messi and the segmentation result is <seg>.

Response from POPEN: The name of the Argentine player in the image is Lionel Messi <seg>.

Instruction: How should I rest here and then quickly leave this area?

Response from PixelLM: To rest in this place, you can choose to sit on one of the two benches available. The first bench <seg> is located closer to the left side of the image, while the second bench <seg> is situated further to the right. Once you have rested, you can quickly leave the area by riding the bicycle <seg>.

Response from POPEN: you can sit on the wooden bench <seg> located in the right side of the image and enjoy a rest there. After resting, you can quickly leave the area by riding the bicycle <seg> parked nearby.

Figure 2. More comparative examples of text responses and segmentation results between PixelLM and our POPEN.