Supplementary Materials: Q-Ground: Image Quality Grounding with Large Multi-modality Models

Anonymous Authors

1 DATA COLLECTION DETAILS

In this section, we provide more details about data collection process including both human and GPT4V.

1.1 Human Annotation

1.1.1 Information of Participants. We recruited 15 participants for this study, with a gender distribution of 10 females and 5 males. All participants are within the age range of 20 to 30 years and possess at least a college degree. The participants were selected to provide a diverse representation in terms of academic backgrounds, including disciplines such as computer science, psychology, and engineering.

1.1.2 Preparation and Quality Control. Participants underwent a comprehensive training session to familiarize them with the annotation guidelines and tools used in this study. The training included detailed explanations of the tasks and practice sessions to ensure clarity and consistency in the annotation process. The participants was trained on 1, 000 samples first, and the supervising teams checked the quality and improved the process. To maintain high standards of annotation quality, we conducted periodic checks of the annotations during the annotation process.

1.1.3 Annotation Pipeline. Figure 1 illustrates the annotation pipeline. To ensure simplicity and consistency in annotation, we have divided the process into two steps. Firstly, participants identify the distorted region by simply clicking on it. For instance, as shown in Fig. 1, the subject first reviews the reference text and determines whether the building is blurry. Upon identifying a blurry region, subjects merely click on that area; this action prompts the system to produce segmentation results using Semantic-SAM [2]. In the second step, subjects refine the annotations and assign distortion classification labels.

1.1.4 Ethical Considerations. All participants were informed about the goals of the research and the use of the annotated data. Consent was obtained from each participant, ensuring they understood their rights, including the right to withdraw from the study at any time without any consequences. Privacy and confidentiality of the participants were strictly maintained throughout the research process.

1.2 GPT4V Annotation

The GPT4V annotation process is illustrated in Fig. 2. Within the system message, GPT4V is characterized as an effective IQA (Image Quality Assessment) assistant that recognizes five types of distortions, along with a "`no distortion`" category. Responses must adhere to the specified JSON format, where a short reasoning message is required to help verify the result. Users will provide quality prompts generated by the most recent Co-Instruct model[1] using the designated prompt:

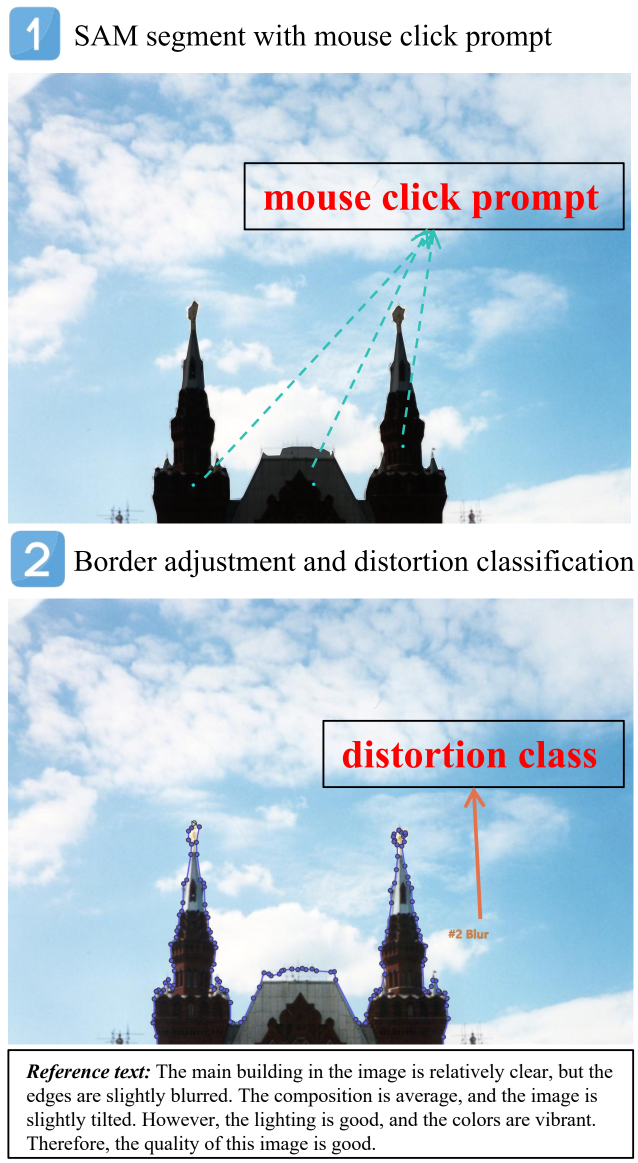

Figure 1: Example of human annotation steps with the help of SAM and reference quality text.

> "The input image: <|image|>. Describe and evaluate the quality of the image."

where "<|image|>" is the placeholder of image input. Figure 2 presents an example of the GPT4V response, in which the correct answer is provided. We checked the response format and tried until it met our requirements. Finally, the corresponding regions were labelled with predicted distortions. We use shortest edit distance

[1] https://huggingface.co/spaces/q-future/Co-Instruct

Example of GPT4V Annotation with SoM (gpt-4-vision-preview)

#System: You are a helpful assistant to help me evaluate the quality of the image. The image is divided into several regions with number marks. You will be given an overall evaluation of the quality as reference. Please help to identify the distortions of each region within the following types [blur, jitter, overexposure, low light, noise, no distortion]. Please give the result in the following json format:

```
[{
    "[mark number]": "distortion type",
    "gpt4v iqa": "message",
}]
```

Please note that the distortion type should be one of the five types mentioned above, and the message should be a brief evaluation of the quality of the region. Please strictly follow the format, otherwise the result will be invalid.

#User: The overall quality reference is: "`The overall clarity of this image is okay. The main subject, which is the boat, is relatively clear. The lighting is weak, making the overall image appear dark. The scenery in the distance is quite blurry, and the texture details are lost. The composition is good, but the quality of this image is poor`". Please help to identify the distortions of each region within the following types [blur, jitter, overexposure, low light, noise, no distortion].

#GPT4V Response:

```
[
    {
        "2": "blur",
        "gpt4v iqa": "The scenery here is quite blurry, detail is lost
            ."
    },
    {
        "3": "low light",
        "gpt4v iqa": "This area is dark and lacks adequate lighting."
    },
    {
        "4": "low light",
        "gpt4v iqa": "The image appears dark due to weak lighting."
    },
    {
        "5": "no distortion",
        "gpt4v iqa": "The main subject, the boat, appears relatively
            clear with no significant distortion."
    }
]
```

Manually generated mask *according to the text label from GPT4V and the corresponding SAM segmentation masks.*

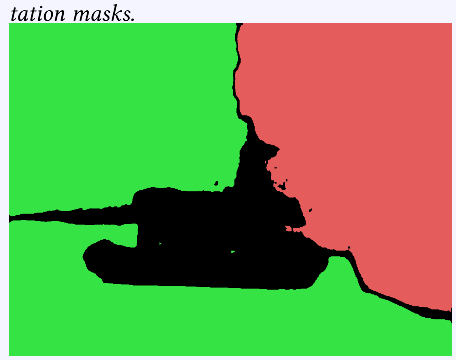

Figure 2: Example of GPT4V annotation

to find the best matching distortion types because the generated distortion words may not always exactly match the candidates.

2 TRAINING DETAILS

2.1 Loss Functions

As described in the main text, we use the following loss function to train the network:

$$\mathcal{L} = \lambda_{txt}\mathcal{L}_{ce}(y_{txt}, \hat{y}_{txt}) + \lambda_{seg}\mathcal{L}_{seg}(y_{seg}, \hat{y}_{seg}). \quad (1)$$

where $\mathcal{L}_{ce}$ is the auto-regressive cross-entropy loss, $\mathcal{L}_{seg}$ is the segmentation loss. We follow the same practice as [1, 3] and use a combination of per-pixel binary cross-entropy loss and DICE loss for $\mathcal{L}_{seg}$ as following:

$$\mathcal{L}_{seg} = \lambda_{bce}\text{BCE}(y_{seg}, \hat{y}_{seg}) + \lambda_{dice}\text{DICE}(y_{seg}, \hat{y}_{seg}), \quad (2)$$

where the loss weights are set to $\lambda_{txt} = 1.0$, $\lambda_{seg} = 1.0$, $\lambda_{bce} = 2.0$, $\lambda_{dice} = 0.5$.

Table 1: Hyper-parameter configurations for Stage 1.

Hyper-parameter config	Value
Image encoder (frozen)	CLIP-L/14-336
LLM (frozen)	LLaVA-v1.5-7B[3]
Input image size	448×448
Layers used for ϕ_v	7, 14, 23
Optimizer	AdamW
Learning rate	5e-4
Weight decay	0
(β_1, β_2)	(0.9, 0.95)
Scheduler	WarmupCosineLR
Warm up steps	100
ZeRO stage (deepspeed)	2
Precision	bfloat16
Batch size (with accumulation)	$2 \times 4 \times 10$
Training dataset	LAION-CC-SBU[4]
Total epochs	1

2.2 Training Configurations

We employed the DeepSpeed framework[2] to accelerate training and reduce memory requirements. The training was conducted using 4 NVIDIA 4090 GPUs. Given our modifications to the multi-modal projection block ϕ_v, it was necessary to adhere to training protocols from LLaVA to align the visual and language representations. Consequently, the training process was structured into three phases:

(1) **Stage 1: Feature alignment between the vision encoder and the LLM.** During this phase, both the vision encoder and LLM were fixed. Training focused solely on the projector ϕ_v to align the vision and text representations.
(2) **Stage 2: Visual instruction tuning.** This phase involved fine-tuning the model to enhance its capability to follow instructions, utilizing multi-modal instruction-following data.
(3) **Stage 3: Mixture dataset tuning.** Once a robust base model was established, it was further finetuned to integrate visual quality grounding with other tasks.

Hyperparameters for each stage are detailed in Tab. 1 and Tab. 2. The entire training duration was approximately two days.

3 MORE QUALITATIVE RESULTS

Figures 3 and 4 demonstrate the versatile capabilities of our model.

REFERENCES

[1] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. 2023. LISA: Reasoning Segmentation via Large Language Model. *arXiv preprint arXiv:2308.00692* (2023).

[2] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. 2023. Semantic-SAM: Segment and Recognize Anything at Any Granularity. *arXiv preprint arXiv:2307.04767* (2023).

[2] https://www.deepspeed.ai/
[3] https://huggingface.co/liuhaotian/llava-v1.5-7b
[4] https://huggingface.co/datasets/liuhaotian/LLaVA-Pretrain
[5] https://huggingface.co/datasets/liuhaotian/LLaVA-Instruct-150K

Table 2: Hyper-parameter configurations for stage 2 and 3.

Hyper-parameter config	Value
Image encoder (frozen)	CLIP-L/14-336
LLM initialization	Previous training stage
Input image size	448×448
Segmentation token number	6
Layers used for ϕ_v	7, 14, 23
LoRA rank	64
LoRA α	16
Optimizer	AdamW
Learning rate	3e-4
Weight decay	0
(β_1, β_2)	(0.9, 0.95)
Scheduler	WarmupCosineLR
Warm up steps	100
ZeRO stage (deepspeed)	2
Precision	bfloat16
Batch size (with accumulation)	$4 \times 4 \times 10$
Training dataset	**Stage 2**: Instruction dataset[5] **Stage 3**: Mixture dataset
Total samples	$80 \times 500 \times 20$

[3] Yunchao Wei Yao Zhao Dongmei Fu Jiashi Feng Xiaojie Jin Zhongwei Ren, Zhicheng Huang. 2024. PixelLM: Pixel Reasoning with Large Multimodal Model. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2024).

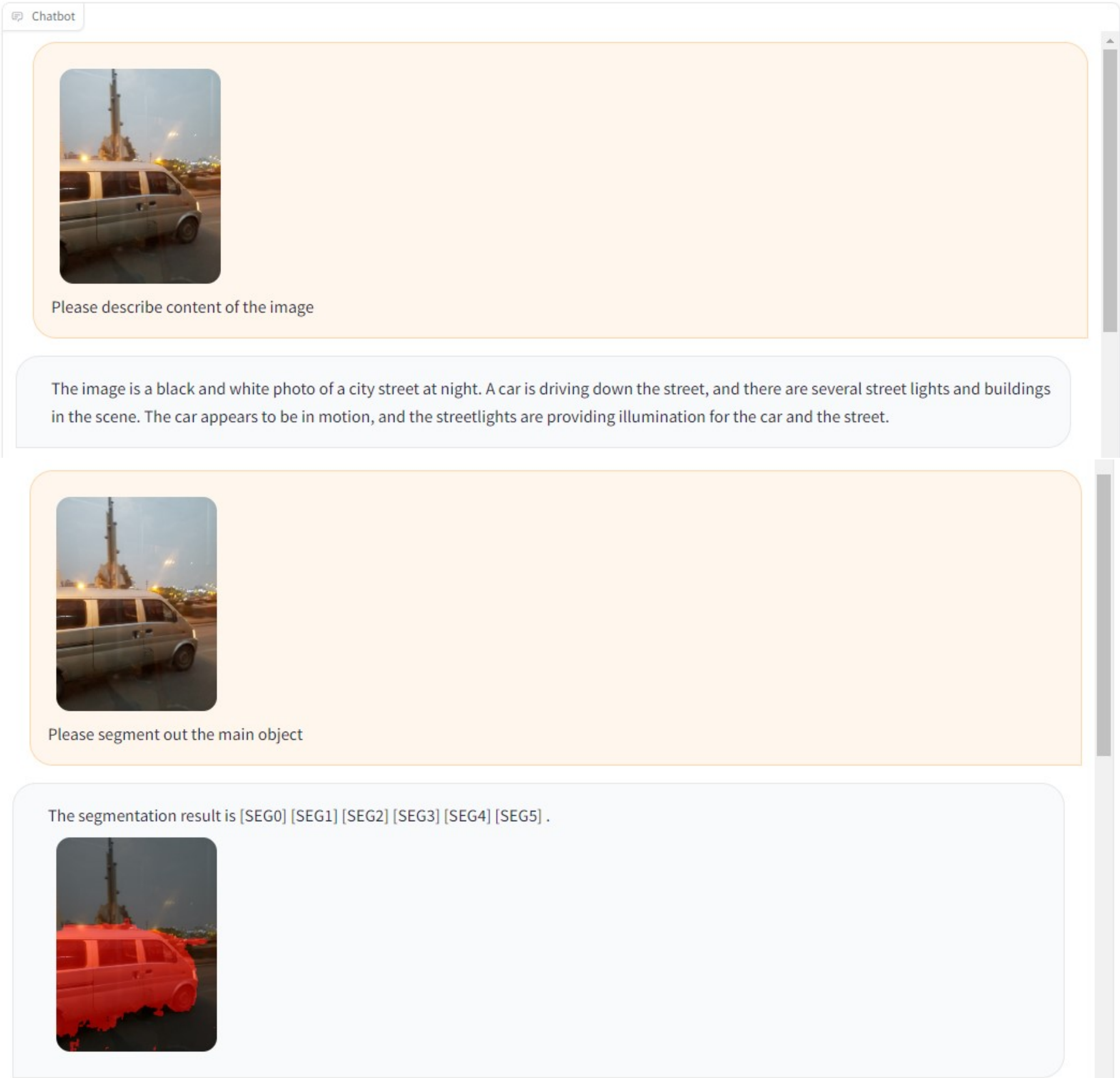

Figure 3: Example of visual question answering and semantic segmentation.

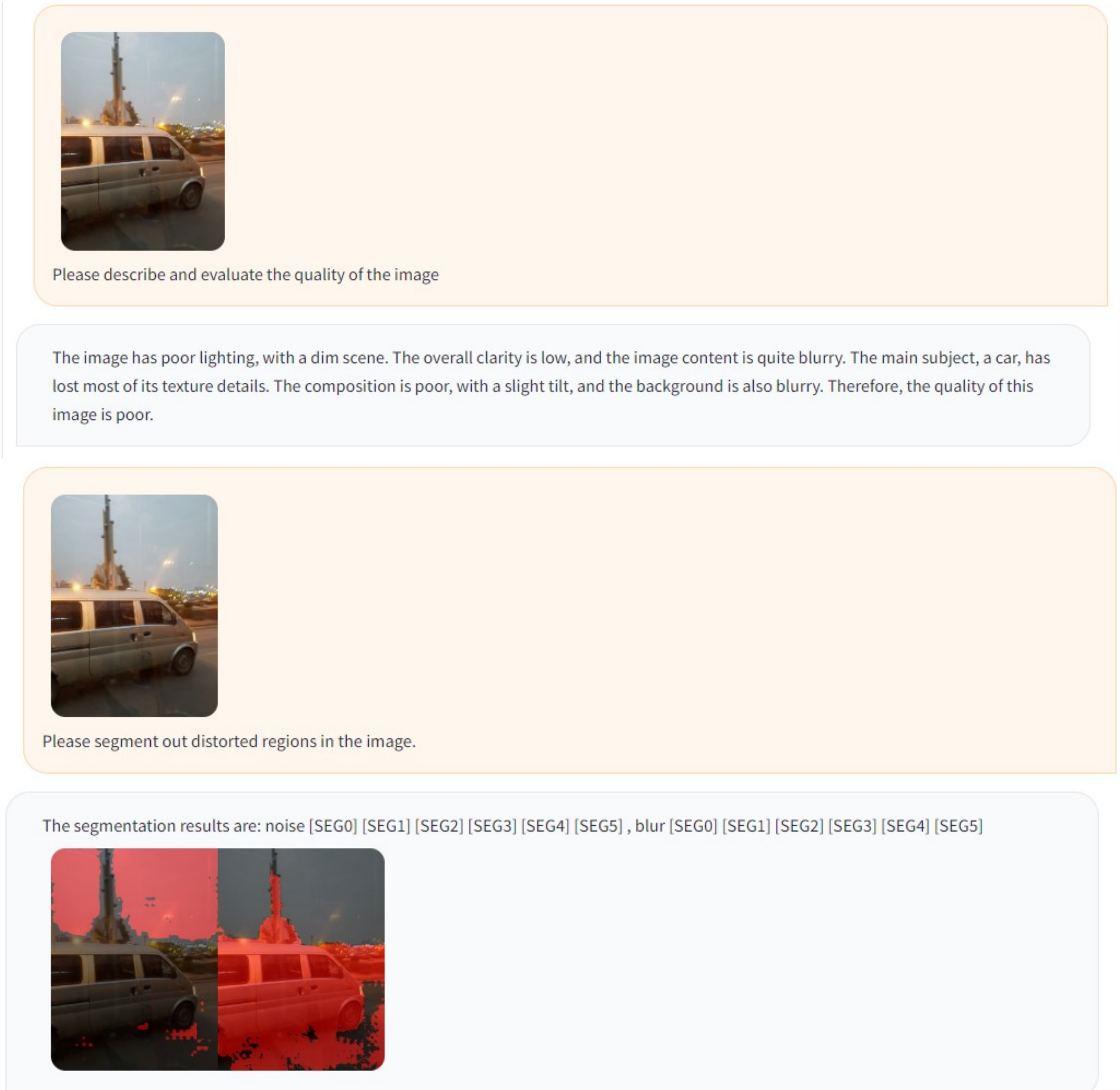

Figure 4: Example of visual visual quality reasoning and visual quality grounding.