Refer to Any Segmentation Mask Group With Vision-Language Prompts

Appendix

In this appendix, we provide additional details on the implementation of our model RAS (Section A) and our datasets MASKGROUPS-2M (Section B) and MASKGROUPS-HQ (Section C). Furthermore, we include additional experiments (Section D), ablation study (Section E), and qualitative results (Section F).

A. Implementation Details

Visual encoder ensemble. Following Cambrian-1 [56], we use four visual encoders: OpenAI CLIP ViT-L/14@336 [48], OpenCLIP ConvNeXt-XXL@1024 [6, 38], SigLIP ViT-SO400M/14@384 [72], and DINOv2 ViT-L/14@518 [42]. In addition, we provide 2D sinusoidal position embeddings [12] of shape 32×32 and treat them as visual features produced by a fifth visual encoder. All input images are padded to an aspect ratio of $1:1$, resized to the input size required by each encoder (up to $1,024 \times 1,024$), and fed into each encoder. All visual encoders are frozen during the entire training process.

Mask projector and its pretraining. We initialize RAS with weights from LLaVA-1.5-13B [36]. The mask projector is a two-layer multilayer perceptron (MLP) that projects the concatenated mask-level visual features to the language model space. As a new module, the mask projector is randomly initialized. Before training the whole RAS model, we first pretrain the mask projector on the LLaVA-Pretrain dataset [35, 36] with a modified pretext task. We use SAM [24] to generate a set of masks per image and replace the original image tokens with our mask tokens for the image captioning objective. To correctly understand and describe a given image, the model needs to align the mask tokens with the LLM feature space. During the pretraining stage, we set the batch size to 128 and set the base learning rate to 1×10^{-3}. We train on LLaVA-Pretrain for 1 epoch.

Visual instruction tuning. After pretraining the mask projector, the entire RAS model (except the visual encoders) is trained in the visual instruction tuning stage. A binary selection classifier (two-layer MLP) is randomly initialized. Then, we minimize a binary cross-entropy loss. Due to the imbalanced distribution of positive/negative samples (usually only a few masks should be selected from a large pool of candidate masks), we assign a loss weight of 5.0 to positive candidates. During the visual instruction tuning stage, we set the batch size to 128 and set the base learning rate to 2×10^{-5}. We train on MASKGROUPS-2M for 1 epoch.

Further finetuning. For improved performance on specialized tasks (ORES, RES, and GRES), we further finetune RAS on these tasks separately. We set the batch size to 64 and use the same base learning rate as instruction tuning. Due to different data scales, we finetune RAS on ORES or GRES for 4 epochs, and finetune RAS on RES for 2 epochs.

Optimization and computation. Following Vicuna [7] and LLaVA [35], we use a cosine learning rate schedule with warm-up in each training stage. The optimizer is Adam [22] with zero weight decay. All of our training is performed on 8 NVIDIA A100-80GB GPUs. The pretraining stage requires about 4 hours. The visual instruction tuning stage on MASKGROUPS-2M requires about 1.5 days. Further finetuning for ORES, RES, or GRES requires another 1.5 days.

B. Construction of MASKGROUPS-2M

MASKGROUPS-2M is converted from object-level annotations of existing image datasets. The sources of MASKGROUPS-2M are detailed as follows.

MS-COCO [32] and LVIS [16]. Since LVIS uses the same images as MS-COCO, we merge their annotations by combining instances with overlapping masks. For each image, we find object categories with at least 2 object annotations and create a category-based mask group with or without reference masks.

Visual Genome [26]. Because mask annotations are not provided by Visual Genome, we first use SAM [24] to produce segmentation masks based on bounding box annotations and filter low-quality masks. We create category-based mask groups and attribute-based mask groups, similar to MS-COCO and LVIS. Furthermore, we compare the coordinates of bounding boxes to decide if an object is on the left side of, on the right side of, on the top of, or at the bottom of the entire image or another object, and then produce position-based mask groups with or without reference masks. In addition, Visual Genome provides annotations of relationships, which are triplets of (subject, relationship, object). In each image, we find triplets with a) the same subject and the same relationship but different objects, or b) the same object and the same relationship but different subjects, and formulate mask groups accordingly.

RES [69] and GRES [34]. The RES datasets, including RefCOCO, RefCOCO+, and RefCOCOg, provide correspondences between a referring expression and an object, which can be directly converted into a single-mask group. The GRES dataset, gRefCOCO, contains referring expressions and their target object sets, and they can be converted into mask groups including a varying number (zero, one, or more than one) of masks. To avoid data contamination, we exclude images that are used for RES/GRES validation or test sets from the entire MASKGROUPS-2M dataset.

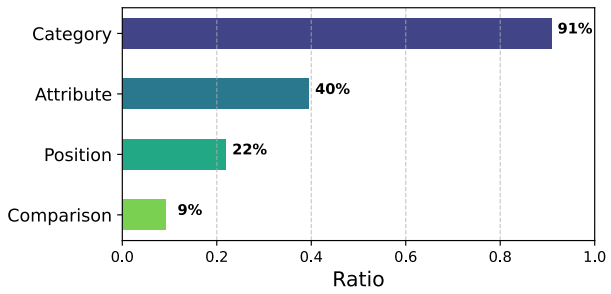

Figure A. **Prompt type distribution in MASKGROUPS-HQ.** A grouping criterion may involve the categories, the attributes, the absolute or relative positions, the cross-entity comparisons, and even their combination.

C. Statistics of MASKGROUPS-HQ

MASKGROUPS-HQ extends the existing mask annotations in EntitySeg [46] with vision-language prompts and mask groups. Human annotators are encouraged to propose creative and meaningful entity groups, so the prompts are very diverse and difficult to categorize. Nevertheless, we provide some statistics for better understanding of MASKGROUPS-HQ: 28% of the samples include reference masks in the prompts, and the other 72% do not contain reference masks. In Figure A, we visualize the distribution of the prompts based on their grouping criterion. Note that each prompt may be labeled with multiple types. For example, the prompt "All paper products smaller than <mask-ref>" simultaneously involves a category ("paper product"), an attribute ("small"), and a comparison ("smaller than <mask-ref>").

D. Additional Experiment Results

SEEM on ORES. As introduced in the main paper, though some interactive segmentation models such as SEEM [77] are able to take text and visual prompts simultaneously, their visual prompts can only be directly used for locating the target object. In contrast, visual prompts in ORES are often for a reference object that has a certain relationship with the target. In Figure B, we visualize examples of prompting SEEM with both text and visual prompts and compare the results with our model RAS. SEEM outputs masks directly corresponding to the visual prompt, instead of correctly understanding the mixed prompt as required by the ORES task. In contrast, our model RAS successfully selects the correct group of masks.

Finetuning GSVA on our data. To understand the impact of training data, we finetune GSVA [62], the previously best GRES model, on our data and evaluate its ORES performance on MASKGROUPS-HQ. As shown in Table A, finetuning GSVA on samples from MASKGROUPS-2M does not yield better performance than its original data recipe, *i.e.*, finetuning on GRES data, and is significantly worse than RAS trained on the same data. Finetuning RAS on the training split of MASKGROUPS-HQ also leads to better ORES performance than GSVA. Note that training on MaskGroups-2M does not necessarily provide an advantage for performance on MaskGroups-HQ due to the domain gap: The samples in MASKGROUPS-2M are constructed from fixed templates, while the samples from MASKGROUPS-HQ are written by human annotators in any free form. Therefore, the stronger performance of our model RAS should be attributed more to its model design.

Figure B. **SEEM, a representative interactive segmentation model, fails in our ORES task.** Instead of understanding the relationship (*e.g.*, "on the reference entity") specified by the vision-language prompt, SEEM [77] simply produces a mask that overlaps with the visual prompt. In contrast, our proposed RAS model can correctly understand the vision-language prompt.

Model	Data	w/o <mask-ref> gIoU	w/o <mask-ref> cIoU
$GSVA_{13B}$ [62]	GRES (original)	41.98	49.55
$GSVA_{13B}$ [62]	0.5M of MASKGROUPS-2M	41.21	36.40
$GSVA_{13B}$ [62]	MASKGROUPS-HQ	56.79	70.11
$RAS_{13B,\ SAM}$ (Ours)	0.5M of MASKGROUPS-2M	54.76	57.73
$RAS_{13B,\ SAM,\ ORES\text{-}FT}$ (Ours)	MASKGROUPS-HQ	**66.71**	**74.59**

Table A. **Results of finetuning GSVA on our data.** Finetuning GSVA [62], the previously best GRES model, on samples of MASKGROUPS-2M, does not achieve better ORES performance than the GSVA model trained with its original data recipe. When finetuned on the training samples of MASKGROUPS-HQ, RAS significantly outperforms GSVA in the ORES task.

Converting visual prompts into language. In the main paper, we have discussed the limitations of existing GRES models [34, 62, 74]: They cannot take visual prompts that represent reference entities as inputs, and therefore cannot process all samples in the ORES task (Table 3). One may argue that visual prompts in ORES can be replaced by text prompts (*e.g.*, "Locate all pillows on <mask-ref>" → "Locate all pillows on the bed", Figure 1). However,

Prompt	Model	w/ <mask-ref> gIoU	w/ <mask-ref> cIoU
Text + Converted <mask-ref>	ReLA [34]	21.15	24.14
	PSALM$_{1.3B}$ [74]	24.68	24.19
	GSVA$_{13B}$ [62]	22.66	25.10
	RAS $_{13B, SAM}$ (Ours)	27.13	27.74
	RAS $_{13B, SAM, ORES-FT}$ (Ours)	43.76	47.80
Text + Visual <mask-ref>	RAS $_{13B, SAM}$ (Ours)	35.91	37.77
	RAS $_{13B, SAM, ORES-FT}$ (Ours)	**58.72**	**68.77**

Table B. **Results of converting visual prompts into language.** We manually translate visual prompts for reference entities into language (*e.g.*, "Locate all pillows on <mask-ref>" → "Locate all pillows on the bed," see Figure 1), and test multiple GRES models and our RAS model on the converted prompts. The original visual prompts lead to better performance than the converted prompts, demonstrating that visual prompting is necessary in referring expression segmentation. When provided with pure-text prompts, our model RAS still outperforms all prior GRES models. The subscript $_{ORES-FT}$ means evaluation of RAS that is further finetuned on the original training set (not including the converted prompts) of MASKGROUPS-HQ.

when the scene is complex and involves multiple semantically similar objects, visual prompts can hardly be clearly and concisely "translated" into language. To investigate this discrepancy between visual prompts and text prompts, we manually convert <mask-ref> into language for 200 samples in MASKGROUPS-HQ, and test GRES models and our RAS on these samples. As shown in Table B, visual prompts are better perceived by RAS, indicating that such visual prompts are necessary to guide the model in accurately locating the target entities that are related to the reference entity. When provided with the same pure-text prompts, despite the increased complexity of the converted prompts, RAS still outperforms the existing GRES models.

E. Additional Ablation Study

Special tokens in mask tokenization. In RAS, we prepend a learnable special token <mask-pool-pre> to each candidate mask token and prepend a <mask-ref-pre> token to each reference mask token. These special tokens indicate the different roles of the following tokens. In Table C, we compare RAS with two variants: The first variant does not prepend any special tokens to the mask tokens, and the second variant prepends the same token to both candidate mask tokens and reference mask tokens. Using two different special tokens in mask tokenization achieves the best performance.

LMM scales. In the main paper, we report the results of training our model RAS based on LLaVA-1.5-13B [36], which originates from Vicuna-13B [7]. In principle, RAS can be built on other LLMs of different parameter scales. As an example, we train another RAS based on LLaVA-1.5-7B. The model performance is summarized in Table D.

Special tokens	w/o <mask-ref>	w/ <mask-ref>	Overall cIoU
No <pre> tokens	55.61	34.98	50.13
Same <pre> tokens	54.68	32.37	48.49
Different <pre> tokens	**57.73**	**44.47**	**53.75**

Table C. **Comparison of RAS with different special tokens prepended to mask tokens.** Prepending <mask-pool-pre> to candidate mask tokens and <mask-ref-pre> to reference mask tokens leads to the best result. All models are trained on the same 0.5M samples from MASKGROUPS-2M and evaluated on MASKGROUPS-HQ.

Model	ORES	RES	GRES
RAS $_{7B, SAM / Co-DETR}$	52.19	73.7	67.30
RAS $_{13B, SAM / Co-DETR}$	**53.93**	**75.0**	**67.78**

Table D. **Comparison of RAS with different LLM scales.** The larger 13B LLM leads to a stronger performance on all tasks. The metric is the overall cIoU. We use SAM as the mask proposal model in ORES, and use Co-DETR in RES and GRES, consistent with the main results in Tables 3, 4, and 5.

F. Additional Qualitative Results

Our RAS shows strong generalization beyond MS-COCO benchmarks, where prior works primarily focus. As shown in Figure C, our model outperforms GSVA on out-of-distribution (OOD) images. This is achieved by decoupling mask generation and selection, allowing RAS to leverage strong generalization capabilities of SAM.

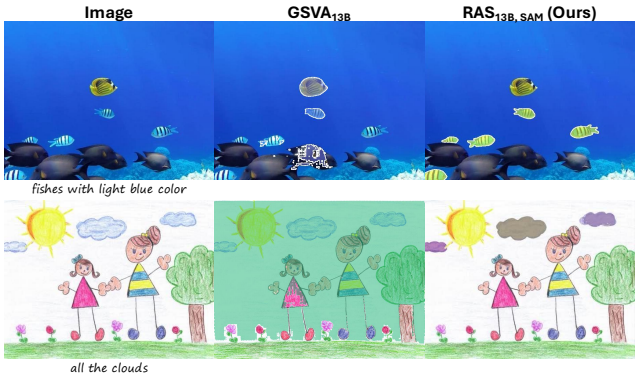

Figure C. **Qualitative comparison on OOD examples.** Our RAS framework generalizes better to novel image domains, such as under-water images and cartoon-style images.

In Figure D, we provide additional visualized results of applying RAS and other GRES models in the ORES task. RAS (both before and after finetuned on MASKGROUPS-HQ) achieves better results on MASKGROUPS-HQ than all previous GRES models, which is consistent with our quantitative evaluation in Table 3 of the main paper.

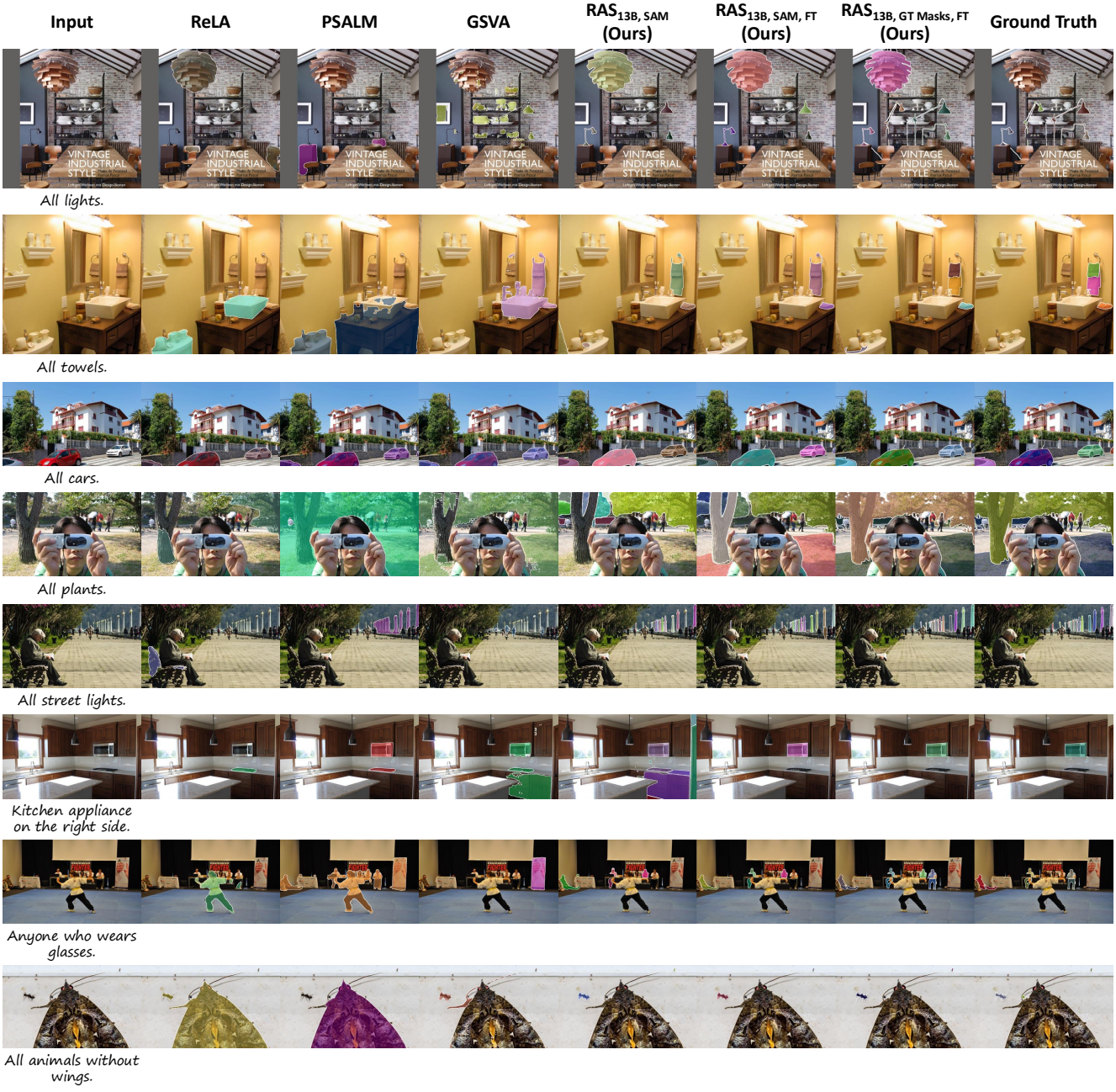

Figure D. **Qualitative comparison on MASKGROUPS-HQ.**