ReME: A Data-Centric Framework for Training-Free Open-Vocabulary Segmentation

Supplementary Material

Our supplementary material is organized as follows:

- Sec. A provides a formal problem definition of open-vocabulary segmentation (OVS).
- Sec. B provides additional details of our approach and implementation.
- Sec. C includes more experimental results and discussions to supplement Sec. 4 of the main paper.
  - Sec. C.1-Additional Ablation Study Results
  - Sec. C.2-Additional Qualitative Results
  - Sec. C.3-Backbone Usage for Training-Free Methods
  - Sec. C.4-Free-form Queries and In-the-wild Results
  - Sec. C.5-Data Usage for Training-Required Methods
  - Sec. C.6-Comparison with Training-required Methods
- Sec. D discusses the limitation of our work.

A. Problem Definition

To complement the definition of open-vocabulary segmentation (OVS), we formulate it mathematically as follows.

Given an input image $\boldsymbol{I}$ and a candidate set of class labels $\boldsymbol{\mathcal{L}} = \{\mathcal{L}_n\}_{n=1}^{N}$, the objective of OVS is to assign a class label $\mathcal{L}_n \in \boldsymbol{\mathcal{L}}$ to each pixel in $\boldsymbol{I}$. Each $\mathcal{L}_n$ represents the n-th class described by free-form text, where N denotes the total number of candidate classes. Unlike traditional semantic segmentation, where the category set is fixed and predefined during training ($\boldsymbol{\mathcal{L}} = \boldsymbol{\mathcal{L}}_{\text{train}}$), OVS allows for segmentation of arbitrary and unseen categories, operating under a zero-shot setting. This flexibility facilitates adaptive and robust dense scene understanding in dynamic real-world scenarios.

B. Approach and Implementation

B.1. Prompt for Generating Image Descriptions

We design a specific prompt to obtain semantically enriched image descriptions using LLaVA. Our prompt is: *"Describe this image in detail. Mention all visible objects, their parts, contexts, and characteristics like size, color, and texture. Also, describe the background/foreground context, including any natural scene or man-made structures, such as wall, ceiling, sky, and cloud. FOCUS ONLY on visible objects or contexts. Avoid speculation or guesses."*

B.2. Filtering Ambiguous Labels

Object Hallucination in MLLM Outputs. Multi-modal large language models (MLLMs) such as LLaVA often suffer from object hallucination. This includes generating descriptions of tangible objects not present in the input image, which we address through our group-based filtering phase.

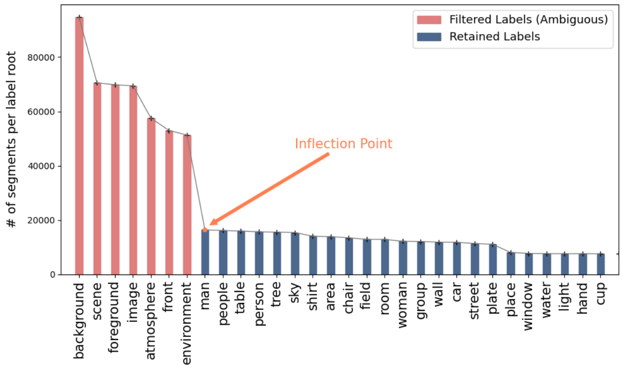


Figure A1. **The number of corresponding segments for each unique label root.** The knee of the distribution curve, *Inflection Point*, indicates the threshold for filtering out ambiguous labels.

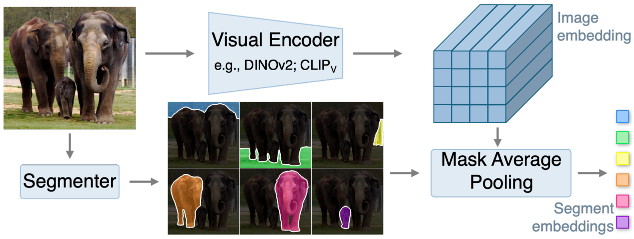


Figure A2. Visual feature encoding for image segments.

Additionally, MLLMs frequently produce ambiguous outputs reflecting abstract or subjective concepts evoked by the image. For instance, descriptions like "*The room has a cozy atmosphere*" lead to ambiguous labels such as "*atmosphere*," which are ungrounded in observable entities and irrelevant to segmentation tasks.

Fast Filtering of Ambiguous Labels. To address this issue, we propose a fast and effective approach to eliminate ambiguous labels arising from evoked descriptions. Due to their abstract nature, these labels appear frequently across MLLM-generated descriptions, often corresponding to an unusually large number of segments in the dataset. This observation forms the basis of our aggregation-based analysis. As described in Sec. 3.2, we group segment-text pairs by consistent label roots. For each group represented by a unique label root, we compute the total number of corresponding segments (i.e., the group size) and plot the distribution of group sizes. By identifying the knee of the curve—referred to as the inflection point (see Fig. A1)—we filter out labels exceeding this point, such as "background,"

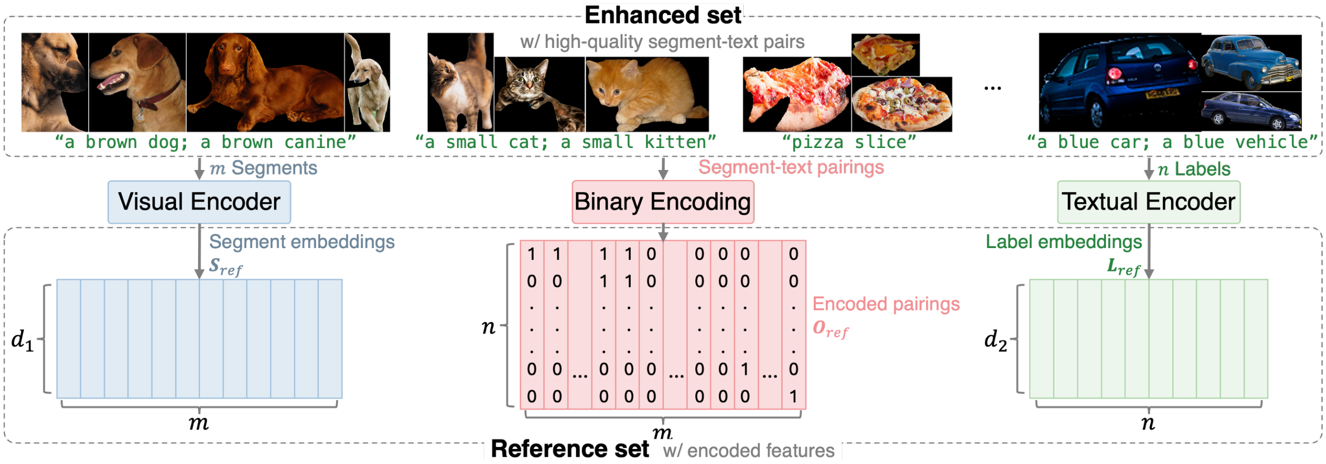


Figure A3. **Illustration of reference set encoding for similarity-based retrieval.** Image segments, textual labels, and their relationships are encoded as $\boldsymbol{S}_{\text{ref}}$, $\boldsymbol{L}_{\text{ref}}$, and $\boldsymbol{O}_{\text{ref}}$, respectively. These embeddings collectively form the reference set, enabling efficient retrieval.

"scene," "image," and "atmosphere." These labels dominate the dataset and detract from meaningful segmentation labels. Removing them ensures the dataset remains focused on concrete and observable objects, improving its relevance and usability for segmentation tasks.

B.3. Feature Encoding

Visual feature encoding. We compute segment embeddings following common practices [4, 14, 32, 33]. According to the requirements, a visual encoder such as DINOv2 or CLIP$_V$ is used, denoted as φ. As shown in Fig. A2, given an input image $\boldsymbol{I}$ and its K corresponding segment masks $\mathcal{M} = \{\boldsymbol{M}_k\}_{k=1}^{K}$, the visual encoder processes the image to obtain its embedding. To align the segment masks with the encoder's output resolution, the masks are resized using a downscaling function, ζ. Lastly, we apply mask average pooling (MAP) to produce embedding for each segment $\boldsymbol{S}_k$. This process is represented as:

$$\boldsymbol{S}_k = \text{MAP}(\varphi(\boldsymbol{I}), \zeta(\boldsymbol{M}_k)). \tag{A1}$$

Textual feature encoding. We generate text embeddings using a textual encoder CLIP$_T$, denoted by ϕ. For a given label $\mathcal{L}$, we deploy four templates to prompt the encoder: *"A photo of* $\{\}$*," "This is a photo of* $\{\}$*," "There is* $\{\}$ *in the scene,"* and *"A photo of* $\{\}$ *in the scene."* The text encoder processes each prompted input, and the resulting embeddings are averaged to form the final label embedding $\boldsymbol{L}$. This process is expressed as:

$$\boldsymbol{L} = \frac{1}{P}\sum_{p=1}^{P} \phi(\psi_p(\mathcal{L})), \tag{A2}$$

where P is the number of templates, and $\psi_p(\mathcal{L})$ represents applying p-th template to label $\mathcal{L}$.

All encoded features, regardless of modality, are L2-normalized to facilitate our cosine similarity computation.

B.4. Reference Set Construction

Following our intra-modality data enhancement phase (refer to Sec. 3.2), we have obtained a high-quality set of segment-text pairs. Fig. A3 depicts how we obtain specific embeddings to construct the reference set for streamlined retrieval. The visual encoder processes image segments to extract d_1-dimensional segment embeddings ($\boldsymbol{S}_{\text{ref}}$), while a textual encoder generates d_2-dimensional label embeddings ($\boldsymbol{L}_{\text{ref}}$). To represent the relationships between segments and their associated labels, we utilize binary encoding to formulate $\boldsymbol{O}_{\text{ref}} \in \mathbb{R}^{m \times n}$, where m and n are the numbers of unique segments and labels, respectively. Each row of $\boldsymbol{O}_{\text{ref}}$ corresponds to a segment, and a column entry of '1' indicates an association with a specific label and '0' otherwise. The resulting reference set is defined by $\{\boldsymbol{S}_{\text{ref}}, \boldsymbol{O}_{\text{ref}}, \boldsymbol{L}_{\text{ref}}\}$, combining visual, textual, and relational encodings. This structured representation enables efficient similarity-based retrieval in the subsequent phase.

B.5. Pseudocode for Similarity-Based Retrieval

To complement Sec. 3.3, we provide a Python-style pseudocode in Alg. A1 to detail the similarity-based retrieval process. The variable names are consistent with those in Sec. 3.3 for ease of reference, and comments within the pseudocode indicate the steps corresponding to the equations discussed in the main paper.

C. More Experimental Results and Discussion

C.1. Additional Ablation Study Results

In this section, we provide comprehensive results and additional examples to supplement the findings presented in the

Algorithm A1 Pseudocode for similarity-based retrieval

```
# Inputs:
#   S_ref [m,d_1] - Segment embeddings in the reference set
#   O_ref [m,n] - Binary encoding of segment-label relationships
#   L_ref [n,d_2] - Label embeddings in the reference set
#   I_test [h,w] - Test image
#   ℒ_test - c test classes in text
# Outputs:
#   l_pred [h,w] - Predicted label mask for the test image

# Segment the test image into k class-agnostic masks
M_seg = segmenter(I_test) # [k,h,w]

# Same encoder as used for S_ref and L_ref
S_test = visual_encoder(I_test, M_seg) # [k,d_1]
L_test = textual_encoder(ℒ_test) # [c,d_2]

# Intra-modality similarities
sim_seg = np.dot(S_test, S_ref.T) # [k,m]
sim_text = np.dot(L_ref, L_test.T) # [n,c]

# Compute and ensemble affinities # Eqns.(1-3)
A1 = np.dot(softmax(sim_seg, axis=1), O_ref) # [k,n]
A2 = softmax(sim_text, axis=1) # [n,c]
P_seg = np.dot(A1, A2) # [k,c]

# Aggregate segment-class probabilities # Eqn.(4)
P_test = np.einsum('ij,ihw->hwj', P_seg, M_seg) # [h,w,c]

# Compute the predicted label mask # Eqn.(5)
l_pred = np.argmax(P_test, axis=2) # [h,w]
```

Method	VOC-20	PC-59	A-150	PC-459
LLaVA [20] as Classifier	72.65	35.50	20.03	7.22
Qwen [3] as Classifier	70.67	36.03	21.21	6.36
LLaVA [20] as Filter	73.06	37.55	22.05	7.60
Qwen [3] as Filter	72.18	38.10	22.81	8.15
ReME	92.34	44.89	26.13	14.12
ReME (OpenFlamingo [2])	92.54	44.77	25.95	13.39

Table A1. **Top**: Analysis of large MLLM capabilities. We use LLaVA-1.5 [20] and Qwen-2.5 VL [3] to (1) classify segmentation masks without any references, and (2) perform label filtering for data enhancing, respectively. They all perform significantly worse than *ReME*. This demonstrates that challenging tasks such as OVS require strategic adaptation rather than direct use. **Bottom**: Analysis of our performance gain from inherent segment-text pretraining. We replace LLaVA-1.5 [20] with OpenFlamingo [1] trained purely on image-text data. The performance remains comparable, indicating *ReME*'s effectiveness without dense annotations.

Method († w/ seg-text training)	VOC-20	PC-59	A-150	PC-459
CAT-Seg (GT COCO) [9]†	94.57	57.45	31.81	19.04
CAT-Seg (*ReME*)†	94.60	59.76	32.24	22.03
FreeDA [4]	87.91	43.49	22.43	10.24
FreeDA (*ReME*)	92.35	44.80	24.91	13.89
ReME	92.34	44.89	26.13	14.12

Table A2. **Data transferability.** We apply *ReME* data to two representative methods by replacing their training/reference data: (1) training-based CAT-Seg [9], and (2) retrieval-based FreeDA [4]. The results demonstrate the strong utility of our data across both training-based and training-free OVS.

main paper. The supplementary tables and figures expand on the quantitative and qualitative analyses in Sec. 4.3, offering a more complete view of our ablation studies. We cross-reference the corresponding tables/figures in the main paper for clarity and context.

- **Data enhancement component analysis.** Full quantitative results for analyzing contributions of individual components in our data enhancement pipeline are presented in Table A4 (supplementing Table 2 in the main paper).
- **Analysis of different data filtering approaches.** A comprehensive comparison of different data filtering approaches is provided in Table A5, extending the analysis from Table 3 in the main paper. We include a variant of our group-based filtering, noted as (d). Compared to our default approach that use the same drop ratio for all groups, (d) adapts each group's drop ratio to its segment consistency, ranging from 0 to 50%, with weights $w = \frac{1}{n}\sum_{i=1}^{n}(1 - \langle S_i, S_{center}\rangle)$, allowing for more drops in sparser groups. We can observe that this variant brings further performance gain.

  Additionally, we provide more examples to showcase the superiority of intra-modality over cross-modality in Fig. A4, to complement Fig. 3 in the main paper.
- **Feature encoder backbones.** Full results of using different feature encoder backbones are detailed in Table A6 (bottom), supplementing Table 4 in the main paper.
- **Analysis of the description generator.** Full results on the impact of the description generator are shown in Table A7, supplementing Table 5. In addition, to further evidence the semantic richness of LLaVA-generated descriptions as discussed in Fig. 7 in the main paper, we provide more examples in Fig. A5.
- **Analysis of the segmenter.** Additional results for the impact of various segmenters are presented in Table A8, which complements Table 6 in the main paper.
- **Analysis of the large MLLM capabilities.** To analyze the capabilities of large MLLM compared to our data enhancement framework, we perform two experiments. (1) We directly leverage advanced MLLMs, including LLaVA-1.5 [20] and Qwen-2.5 VL [3], to assign class labels to class-agnostic segmentation masks, without using any data as references. (2) We perform data filtering with each MLLM, rather than using our group-based data filtering. The results are shown in Table A1, marked as "* as Classifier", and "* as Filter", respectively. They perform significantly worse than ReME. This observation aligns with widely discussed challenges in directly using VLMs for fine-grained data matching—they tend to hallucinate object labels and produce noisy predictions. These results highlight: while pre-trained models present potential, challenging tasks like reasoning segmentation [16] or OVS require strategic adaptation rather than direct use. For instance, LISA [16] fine-tunes vLLM+SAM backbones, while *ReME* studies data-centricity—they contribute in complementary ways.
- **Data transferability.** We apply *ReME* data to two representative methods by replacing their training/reference

data: (1) training-based CAT-Seg [9], and (2) retrieval-based FreeDA [4]. As shown in Table A2, CAT-Seg (*ReME*) even surpasses the version trained on COCO ground-truth, and FreeDA (*ReME*) also outperforms the original version with its default reference set. The results demonstrate the strong utility of *ReME* data across both training-based and training-free OVS settings.

C.2. Additional Qualitative Results

We perform additional qualitative comparisons with other training-free baselines. The results are shown in Fig. A6. In addition, we present qualitative results of *ReME*-SAM on datasets with a large number of categories. Specifically, we include ADE20K [42] with 847 categories (Fig. A8), Pascal Context [24] with 459 categories (Fig. A9), and COCO Stuff [6] with 171 categories (Fig. A10).

C.3. Backbone Usage for Training-Free Methods

Table A10 presents the backbone usage across various training-free methods. As shown, earlier approaches predominantly relied on a single CLIP backbone, but their overall performance falls short compared to more recent methods that leverage multiple backbones. Compared to these multi-backbone methods, our approach (1) remains entirely off-the-shelf, avoiding structural modifications to the backbone as implemented in ProxyCLIP, and (2) achieves the best performance while maintaining controlled backbone usage.

Additionally, existing methods employ different backbone variants, such as ViT-B/16 and ViT-L/14, with some supporting even larger models like ViT-H/14. In our comparisons, we use ViT-L/14 by default. However, if a method performs better with ViT-B/16, we report the superior result.

C.4. Free-form Queries and In-the-wild Results

Generalizability evaluation. Quantitative. We evaluate generalizability using free-form text. To ensure a fair comparison, we use the same superpixel segmenter as FreeDA. We prompt GPT4o three times independently to generate diverse free-form class variations (e.g.,“cat”→“small domestic feline”) and then perform retrieval. Results across three runs are summarized in Table A9. Shifting to free-from text, FreeDA and ProxyCLIP experience significant performance drops, whereas *ReME* consistently outperforms them. **Qualitative.** Following FreeDA, we collect in-the-wild text and qualitatively evaluate out method. The results are shown in Fig.A7.

C.5. Data Usage for Training-Required Methods

For training-required OVS methods using image-text pairs, they often demand extensive training. Table A3 provides the training data size for such methods, where we can observe that millions of image-text pairs from diverse datasets are leveraged, indicating their higher computational cost.

Methods	Training or Fine-tuning dataset	Size
GroupViT[36]	CC12M+YFCC	26 million
SimSeg[40]	CC15M	15 million
TCL[7]	CC15M	15 million
CoCu[35]	CC15M+YFCC	29 million
ZeroSeg[8]	CC3M+COCO	3.4 million
OVSegmentor[37]	CC4M	4 million
SegCLIP[21]	CC3M+COCO	3.4 million
CoDe[34]	CC15M	15 million
SAM-CLIP[30]	CC15M+YFCC+IN21k	41 million

Table A3. Data usage for training-required OVS methods.

C.6. Comparison with Training-required Methods

Although it falls beyond our primary scope of comparison, we also evaluate our approach against training-required methods, as shown in Table A11. Our method **outperforms all approaches fine-tuned with image-text data**. When compared to methods fine-tuned with segment-text, our approach surpasses LSeg+ [12], ZegFormer [10], and ZSseg [38], but falls short compared to OVSeg [19], SAN [39], and CATSeg [9]. This performance gap is commonly observed across all training-free methods when compared to models that demand fine-tuning on segment-text.

However, it is important to note that training-free methods have significantly fewer resources: (1) no training is performed, and (2) no labor-intensive pixel-level annotations. i.e., segment-text data, are required. As a training-free method, we achieve the smallest performance gap compared to these segment-text fine-tuned models.

To sum up, our contributions remain distinct: **A.** ***ReME* achieves state-of-the-art performance among all training-free methods while also surpassing models trained on millions of image-text pairs**, demonstrating reduced dependence on large-scale training. **B.** Our framework provides a novel perspective on multi-modal data quality, offering contributions that extend beyond OVS.

D. Limitation

One limitation of our framework is the decision to drop misaligned pairs in the base set rather than correcting them by reassigning appropriate labels. For instance, in Fig. 3 of the main paper, misaligned pairs where “dog” is associated with segments not depicting dogs are simply filtered out. A more sophisticated approach could involve identifying the correct segments for those labels and reassigning appropriate labels to the affected segments. This refinement would increase the diversity of the final reference set and further enhance the quality of the resulting segment-text embeddings. However, given the diversity and scale of our image resource, COCO-2017 [6], we opt for a simpler and more efficient data enhancement phase.

In domains with limited data availability and constrained diversity [23], this limitation could be addressed easily through a plug-in component. After group-based filtering, this component could leverage intra-modality similarity to identify the closest neighbors for each element in misaligned pairs, enabling the estimation of correct matches with minimal computational overhead.

Components	mIoU										
	VOC-20	VOC-21	City	PC-59	PC-60	Object	Stuff	A-150	PC-459	A-847	AVG^{10}
Base set (no enhancement)	70.03	62.30	30.94	35.42	30.46	39.38	27.01	22.03	9.14	6.19	33.29
w/ (i) Synonym-guided enriching	79.50	66.91	33.47	36.69	34.81	39.92	28.02	23.41	9.56	6.22	35.85
w/ (ii) Group-based filtering	91.10	76.41	47.36	40.66	38.52	42.48	31.80	24.09	12.96	7.13	41.25
w/ Both (i) and (ii)	**92.34**	**79.63**	**50.42**	**44.89**	**41.64**	**45.50**	**33.12**	**26.13**	**14.12**	**8.43**	**43.62**

Table A4. Impact of data enhancement components.

Data filtering alternatives	mIoU										
	VOC-20	VOC-21	City	PC-59	PC-60	Object	Stuff	A-150	PC-459	A-847	AVG^{10}
Global filtering*$^{(a)}$	79.34	71.19	41.37	39.79	37.25	40.79	31.16	21.18	11.71	7.88	38.47
*Group-based filtering (with cross-modality CLIP score)$^{(b)}$	80.05	72.92	43.06	41.84	39.44	41.81	31.88	22.79	12.19	8.33	39.63
*Group-based filtering (with intra-modality similarity score)$^{(c)}$	**92.34**	**79.63**	**50.42**	44.89	41.64	45.50	33.12	26.13	14.12	8.43	43.62
*Group-based filtering (with intra-modality similarity score; weighted ratio)$^{(d)}$	92.26	79.61	50.38	**44.97**	**41.88**	**45.60**	**33.17**	**26.51**	**14.74**	**8.58**	**43.77**

Table A5. Analysis of different data filtering approaches.

Feature encoder	mIoU										
	VOC-20	VOC-21	City	PC-59	PC-60	Object	Stuff	A-150	PC-459	A-847	AVG^{10}
CLIP	91.61	68.77	38.53	36.51	35.08	39.82	26.85	24.72	13.76	7.51	37.81
DINOv2$_B$	91.72	79.13	50.20	43.65	41.37	44.71	32.58	25.29	13.79	7.68	43.01
DINOv2$_L$	**92.34**	**79.63**	**50.42**	**44.89**	**41.64**	**45.50**	**33.12**	**26.13**	**14.12**	**8.43**	**43.62**

Table A6. Analysis of feature encoder variations.

Captioners	mIoU										
	VOC-20	VOC-21	City	PC-59	PC-60	Object	Stuff	A-150	PC-459	A-847	AVG^{10}
LLaVA [20]	**92.34**	**79.63**	**50.42**	**44.89**	**41.64**	**45.50**	**33.12**	**26.13**	**14.12**	**8.43**	**43.62**
BLIP-2 [18]	89.41	56.32	40.06	40.85	38.42	37.64	30.76	24.31	12.42	6.47	37.67
GT Caption	89.02	55.57	40.19	40.15	38.37	37.77	29.68	24.09	11.64	5.37	37.18

Table A7. Ablation study of the image description generator.

Segmenters	mIoU										
	VOC-20	VOC-21	City	PC-59	PC-60	Object	Stuff	A-150	A-847	PC-459	AVG^{10}
Superpixel [11]	92.34	79.63	50.42	44.89	41.64	45.50	33.12	26.13	14.12	8.43	43.62
SAM [15]	93.15	82.20	59.04	**53.10**	**44.58**	48.21	33.32	28.21	15.82	8.80	46.64
SAM2 [25]	**93.18**	**82.26**	**61.19**	52.03	43.42	**48.40**	**33.36**	**28.21**	**8.83**	**15.97**	**46.69**

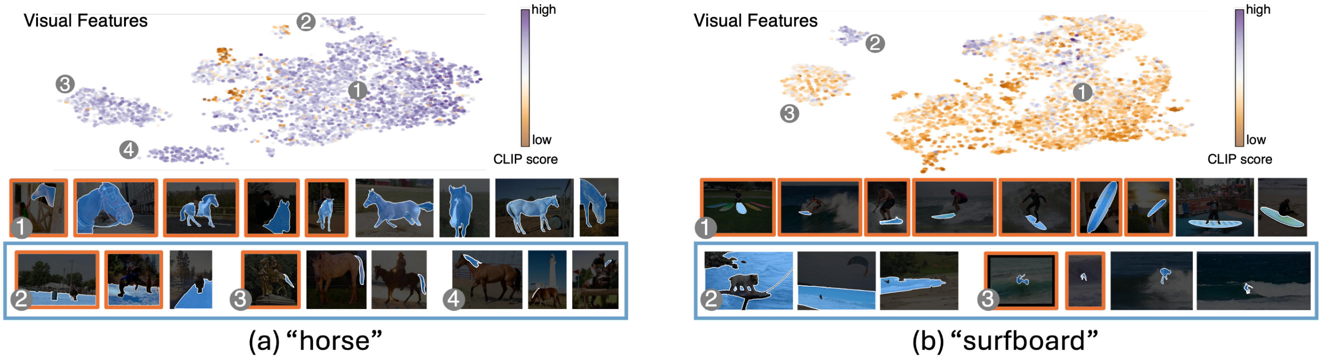
Table A8. Ablation study of the segmenter.

Methods	mIoU											
	PC-59	PC-59*	Δ(%)	A-150	A-150*	Δ(%)	PC-459	PC-459*	Δ(%)	A-847	A-847*	Δ(%)
Ours	**44.89**	**42.89**±**0.9**	↓ **4.46**	**26.13**	**26.12**±**0.3**	↓ **0.04**	**14.12**	**13.14**±**0.1**	↓ **6.94**	**8.43**	**7.35**±**0.1**	↓ 12.81
FreeDA [4]	43.50	36.18±0.8	↓ 16.83	22.4	16.27±1.0	↓ 27.37	10.20	7.16±0.2	↓ 29.80	5.30	2.09±0.1	↓ 54.52
ProxyCLIP [17]	37.7	33.15±1.2	↓ 12.05	22.6	17.12±0.3	↓ 24.26	11.20	8.41±0.3	↓ 24.84	6.70	6.39±0.2	↓ **4.63**

Table A9. Generalizability evaluation with free-form queries.

Methods	Backbone	Post-proc	mIoU									
			VOC-20	VOC-21	City	PC-59	PC-60	Object	Stuff	A-150	PC-459	A-847
GEM [5]	CLIP	✗	46.2	24.7	-	32.6	21.2	-	15.1	10.1	4.6	3.7
MaskCLIP [43]	CLIP, DeepLabV2	✓	74.9	38.8	12.6	25.5	23.6	20.6	14.6	9.8	-	-
ReCo [27]	CLIP, DenseCLIP	✓	62.4	27.2	23.2	24.7	21.9	17.3	16.3	12.4	-	-
SCLIP [29]	CLIP	✓	83.5	61.7	34.1	36.1	31.5	32.1	23.9	17.8	9.3	6.1
CaR [28]	CLIP	✓	<u>91.4</u>	67.6	15.1	39.5	30.5	36.6	11.2	17.7	11.5	5.0
NACLIP [13]	CLIP	✓	83.0	64.1	38.3	38.4	35.0	36.2	25.7	19.1	9.0	6.5
CLIPtrase [26]	CLIP	✓	81.2	53.0	21.1	34.9	30.8	<u>39.6</u>	24.1	17.0	9.9	5.9
PnP [22]	CLIP, GPT4om, BLIP	✓	79.1	51.3	19.3	31.0	28.0	36.2	17.9	14.2	5.5	4.2
FreeDA [4]	CLIP, Stable Diffusion, DINO	✓	87.9	55.4	36.7	<u>43.5</u>	<u>38.3</u>	37.4	<u>28.8</u>	22.4	10.2	5.3
ProxyCLIP [17]	CLIP, DINO	✗	83.2	60.6	<u>40.1</u>	37.7	34.5	39.2	25.6	<u>22.6</u>	11.2	<u>6.7</u>
DiffSegmenter [31]	Stable Diffusion, BLIP, U-Net, DeepLabV2	✓	71.4	60.1	-	27.5	25.1	37.9	-	-	-	-
OVDiff [14]	CLIP, Stable Diffusion, GPT, CutLER	✓	80.9	<u>68.4</u>	23.4	32.9	31.2	36.2	20.3	14.1	<u>12.0</u>	6.6
***ReME* (Ours)**	CLIP, LLaVA, DINO	✗	**92.3**	**79.6**	**50.4**	**44.9**	**41.6**	**45.5**	**33.1**	**26.1**	**14.1**	**8.4**

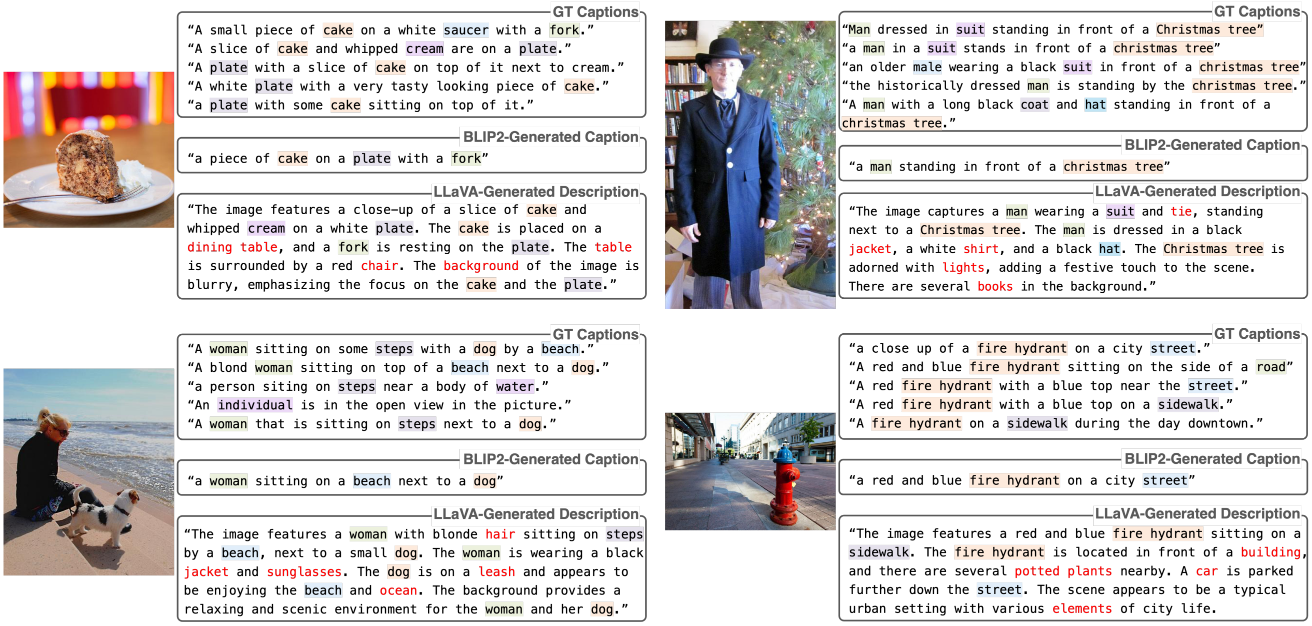
Table A10. **Comparison to training-free methods without SAM.** The best overall results are **bolded**, with the second-best results <u>underlined</u>.



(a) "horse"

(b) "surfboard"

Figure A4. **The superiority of intra-modality over cross-modality for data issue detection.** Each figure provides a UMAP projection of segment embeddings labeled as "horse" or "surfboard", respectively, colored by cross-modal similarity scores (CLIP scores) between the segment and its corresponding label. Individual segments are shown below. **Blue boxes** highlight misalignments detected by our filtering; **orange boxes** are those detected by low CLIP scores, which remove correct pairings while leaving many misalignments unaddressed.



GT Captions

"A small piece of cake on a white saucer with a fork."
"A slice of cake and whipped cream are on a plate."
"A plate with a slice of cake on top of it next to cream."
"A white plate with a very tasty looking piece of cake."
"a plate with some cake sitting on top of it."

BLIP2-Generated Caption

"a piece of cake on a plate with a fork"

LLaVA-Generated Description

"The image features a close-up of a slice of cake and whipped cream on a white plate. The cake is placed on a dining table, and a fork is resting on the plate. The table is surrounded by a red chair. The background of the image is blurry, emphasizing the focus on the cake and the plate."

GT Captions

"Man dressed in suit standing in front of a Christmas tree"
"a man in a suit stands in front of a christmas tree"
"an older male wearing a black suit in front of a christmas tree"
"the historically dressed man is standing by the christmas tree."
"A man with a long black coat and hat standing in front of a christmas tree."

BLIP2-Generated Caption

"a man standing in front of a christmas tree"

LLaVA-Generated Description

"The image captures a man wearing a suit and tie, standing next to a Christmas tree. The man is dressed in a black jacket, a white shirt, and a black hat. The Christmas tree is adorned with lights, adding a festive touch to the scene. There are several books in the background."

GT Captions

"A woman sitting on some steps with a dog by a beach."
"A blond woman sitting on top of a beach next to a dog."
"a person siting on steps near a body of water."
"An individual is in the open view in the picture."
"A woman that is sitting on steps next to a dog."

BLIP2-Generated Caption

"a woman sitting on a beach next to a dog"

LLaVA-Generated Description

"The image features a woman with blonde hair sitting on steps by a beach, next to a small dog. The woman is wearing a black jacket and sunglasses. The dog is on a leash and appears to be enjoying the beach and ocean. The background provides a relaxing and scenic environment for the woman and her dog."

GT Captions

"a close up of a fire hydrant on a city street."
"A red and blue fire hydrant sitting on the side of a road"
"A red fire hydrant with a blue top near the street."
"A red fire hydrant with a blue top on a sidewalk."
"A fire hydrant on a sidewalk during the day downtown."

BLIP2-Generated Caption

"a red and blue fire hydrant on a city street"

LLaVA-Generated Description

"The image features a red and blue fire hydrant sitting on a sidewalk. The fire hydrant is located in front of a building, and there are several potted plants nearby. A car is parked further down the street. The scene appears to be a typical urban setting with various elements of city life."

Figure A5. Image descriptions from different resources. Red text highlights concepts uniquely present in the LLaVA description.

Methods	Post-processing	mIoU									
		VOC-20	VOC-21	City	PC-59	PC-60	Object	Stuff	A-150	PC-459	A-847
Methods that require finetuning on segment-text data											
LSeg+[12]	✗	-	59.0	-	36.0	-	-	-	13.0	5.2	2.5
ZegFormer [10]	✗	86.2	62.7	-	42.8	-	-	-	16.9	9.1	4.9
ZSseg [38]	✗	88.4	-	-	44.7	-	-	-	20.5	-	7.0
OVSeg [19]	✗	94.5	-	-	55.7	-	-	-	29.6	12.4	9.0
SAN [39]	✗	94.6	-	-	57.7	-	-	-	32.1	15.7	12.4
CAT-Seg [9]	✗	97.0	82.5	-	63.3	-	-	-	37.9	23.8	16.0
Methods that require finetuning on image-text data											
GroupViT [36]	✗	74.1	52.3	11.1	23.4	22.4	24.3	15.3	10.6	4.9	4.3
SimSeg [40]	✓	57.4	35.2	10.8	26.2	23.4	29.7	18.5	11.4	5.0	4.7
TCL [7]	✓	83.2	55.0	23.1	33.9	30.4	31.6	19.6	17.1	8.7	6.3
CoCu [35]	✗	73.0	51.4	22.1	26.5	23.6	22.7	15.2	12.3	5.1	4.5
ZeroSeg [8]	✗	-	40.8	-	20.4	-	20.2	-	-	-	-
OVSegmentor [37]	✗	-	53.8	-	-	20.4	25.1	-	5.6	-	-
SegCLIP [21]	✗	-	52.6	-	-	24.7	26.5	-	8.7	-	-
CoDe [34]	✓	57.7	-	28.9	30.5	-	32.3	23.9	17.7	-	-
SAM-CLIP [30]	✗	-	60.6	-	-	29.2	-	31.5	17.1	-	-
Training-free Methods without SAM											
GEM [5]	✗	46.2	24.7	-	32.6	21.2	-	15.1	10.1	4.6	3.7
MaskCLIP [43]	✓	74.9	38.8	12.6	25.5	23.6	20.6	14.6	9.8	-	-
ReCo [27]	✓	62.4	27.2	23.2	24.7	21.9	17.3	16.3	12.4	-	-
SCLIP [29]	✓	83.5	61.7	34.1	36.1	31.5	32.1	23.9	17.8	9.3	6.1
CaR [28]	✓	91.4	67.6	15.1	39.5	30.5	36.6	11.2	17.7	11.5	5.0
NACLIP [13]	✓	83.0	64.1	38.3	38.4	35.0	36.2	25.7	19.1	9.0	6.5
CLIPtrase [26]	✓	81.2	53.0	21.1	34.9	30.8	39.6	24.1	17.0	9.9	5.9
PnP [22]	✓	79.1	51.3	19.3	31.0	28.0	36.2	17.9	14.2	5.5	4.2
FreeDA [4]	✓	87.9	55.4	36.7	43.5	38.3	37.4	28.8	22.4	10.2	5.3
ProxyCLIP [17]	✗	83.2	60.6	40.1	37.7	34.5	39.2	25.6	22.6	11.2	6.7
DiffSegmenter [31]	✓	71.4	60.1	-	27.5	25.1	37.9	-	-	-	-
OVDiff [14]	✓	80.9	68.4	23.4	32.9	31.2	36.2	20.3	14.1	12.0	6.6
***ReME* (Ours)**	✗	**92.3**	**79.6**	**50.4**	**44.9**	**41.6**	**45.5**	**33.1**	**26.1**	**14.1**	**8.4**
Training-free Methods with SAM											
RIM [33]	✗	77.8	-	-	34.3	-	44.5	-	-	-	-
CaR w/ SAM [28]	✗	-	70.2	16.9	40.5	31.1	37.6	12.4	17.9	11.8	5.7
CLIPtrase w/ SAM [26]	✗	82.3	57.1	-	36.4	32.0	44.2	24.8	17.2	10.6	6.0
ProxyCLIP w/ SAM [17]	✗	80.4	59.3	37.0	37.0	33.6	35.4	25.0	19.1	6.9	4.8
CorrCLIP [41]	✗	91.6	74.1	47.7	45.5	40.3	43.6	30.6	-	-	-
***ReME* (Ours)** w/ SAM	✗	**93.2**	**82.2**	**59.0**	**53.1**	**44.6**	**48.2**	**33.3**	**28.2**	**15.8**	**8.8**

Table A11. **Comparison to state-of-the-art OVS approaches.** The best overall results are **bolded**, with the second-best results underlined. We also analyze data robustness by varying the image resources of our reference set from the default COCO-2017 to VOC and ADE, respectively, where leading performances over baselines are **bolded**.

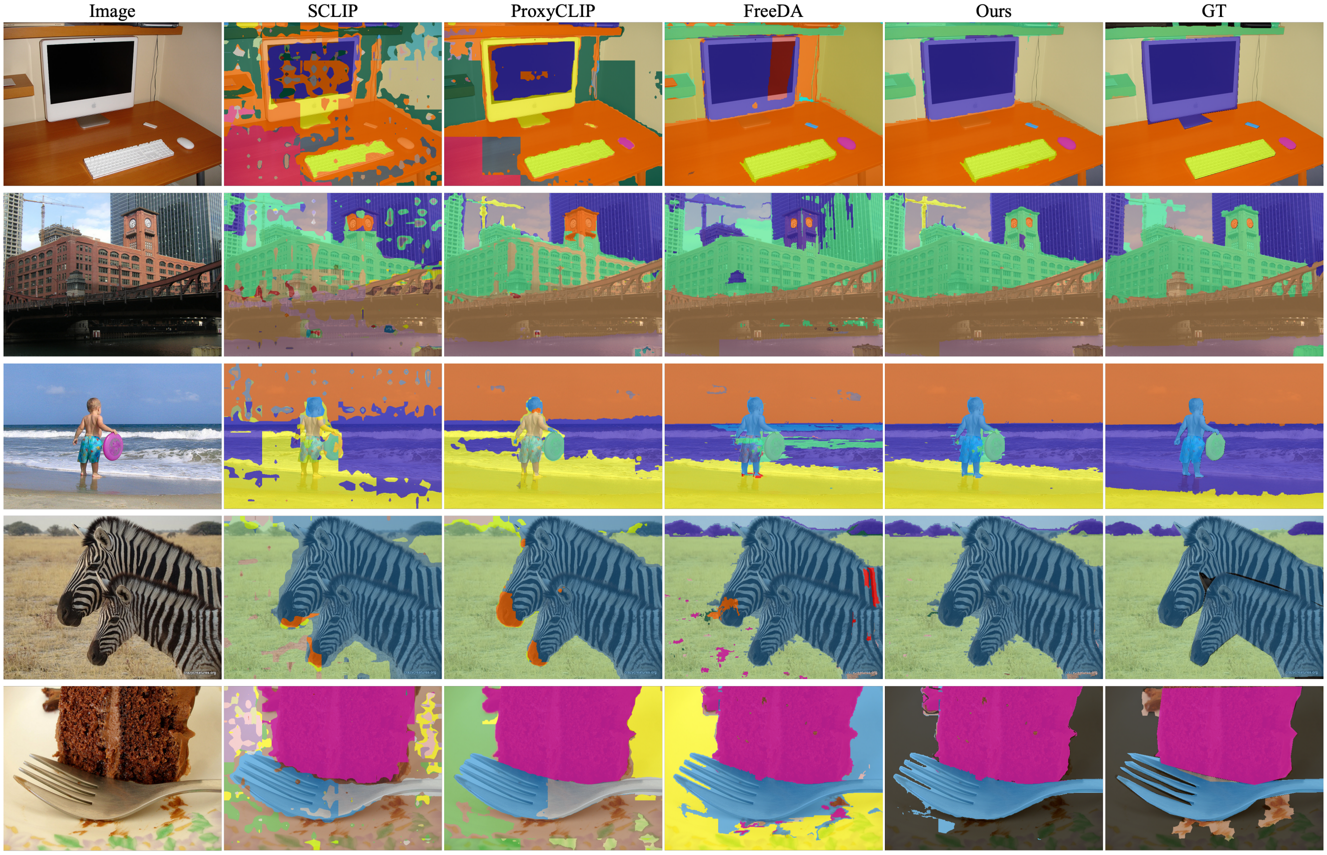


Figure A6. **Qualitative results of *ReME* in comparison with other training-free OVS methods.** SCLIP is based on CLIP attention; ProxyCLIP enhances CLIP attention with DINO features; FreeDA and *ReME* are retrieval-based methods, adopting the same superpixel-algorithm [11] for class-agnostic segmentation. We observe increasing quality of OVS results from left to right, with less noise in both masks and assigned labels.

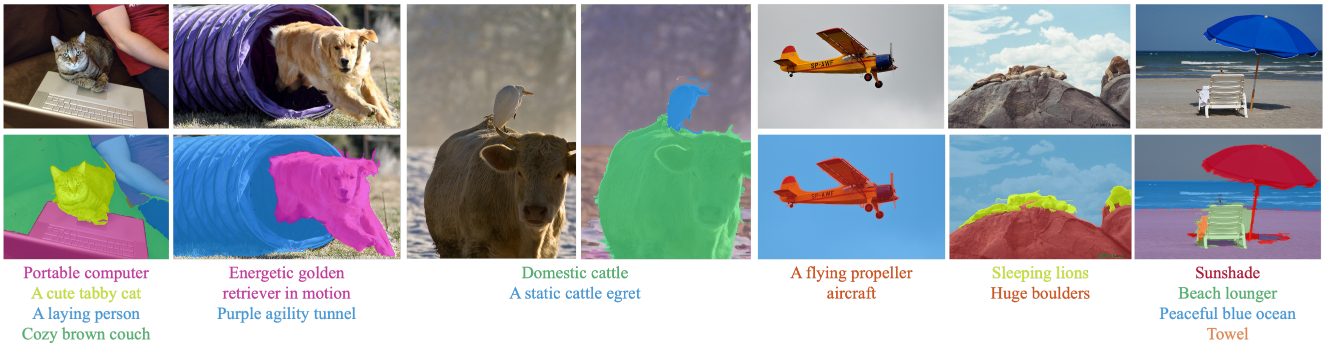


Figure A7. **In-the-wild segmentation results obtained by prompting *ReME* with diverse free-form textual inputs.**

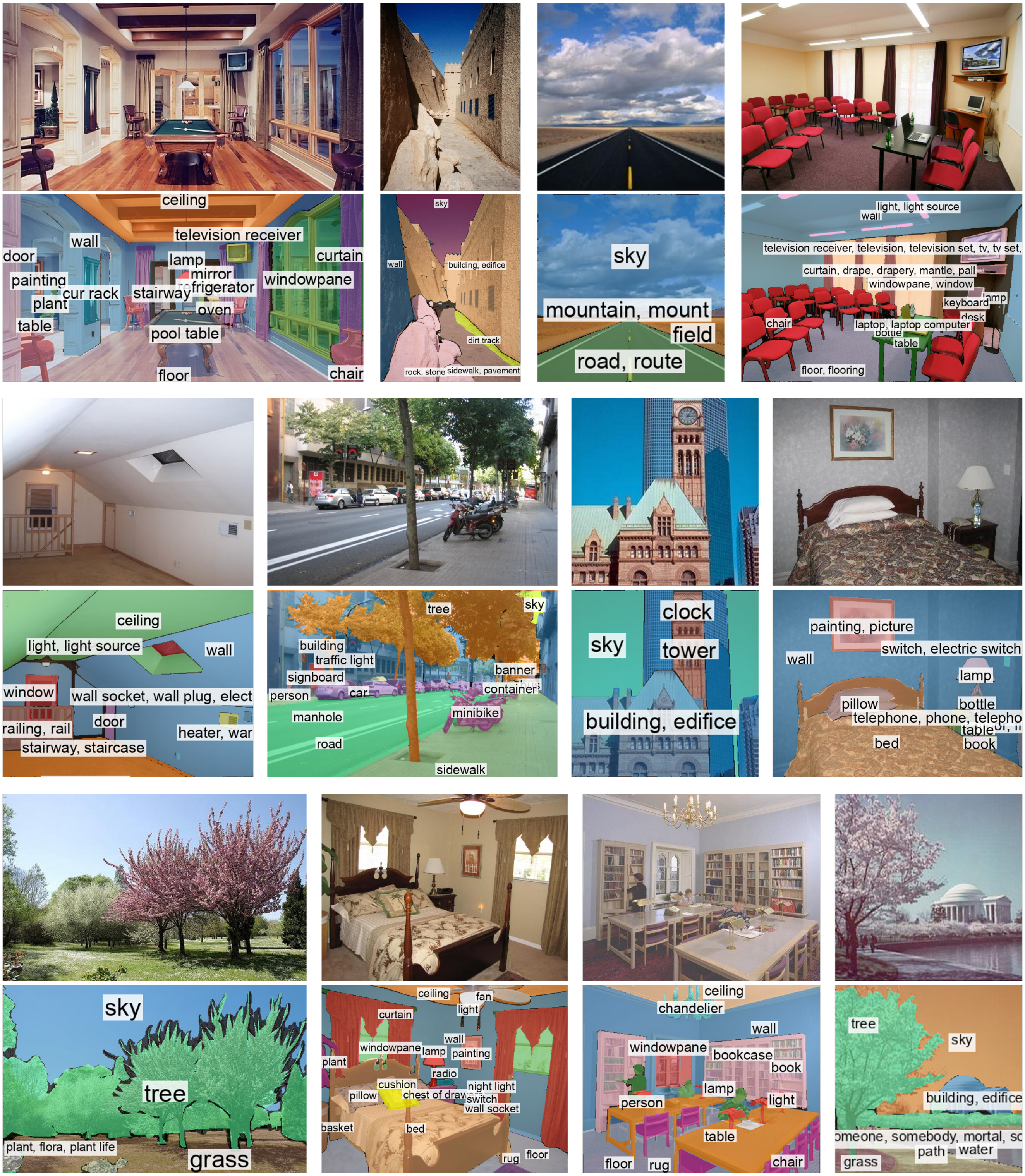


Figure A8. **Qualitative results on ADE20K [42] with 847 categories.**

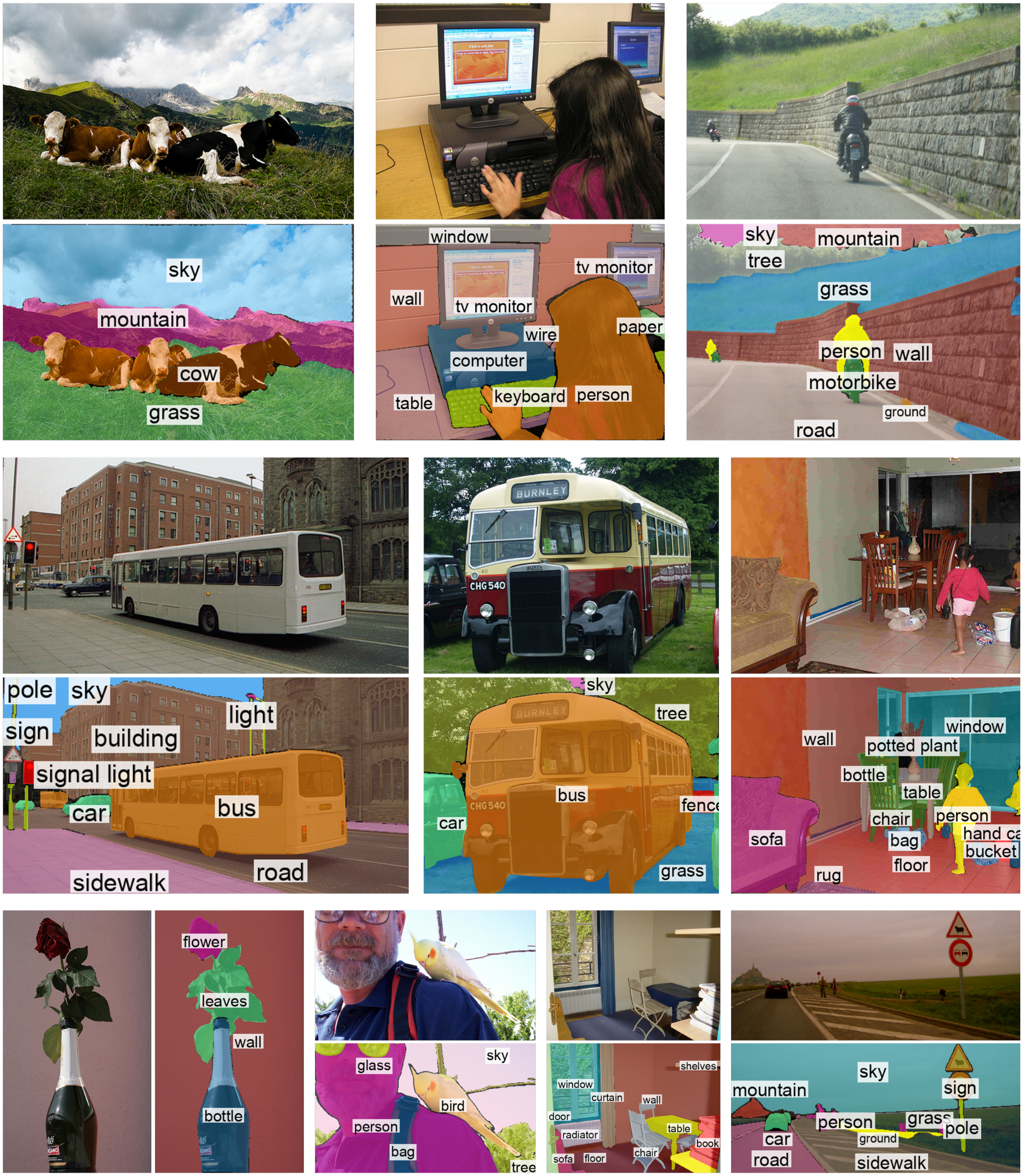


Figure A9. **Qualitative results on Pascal Context [24] with 459 categories.**

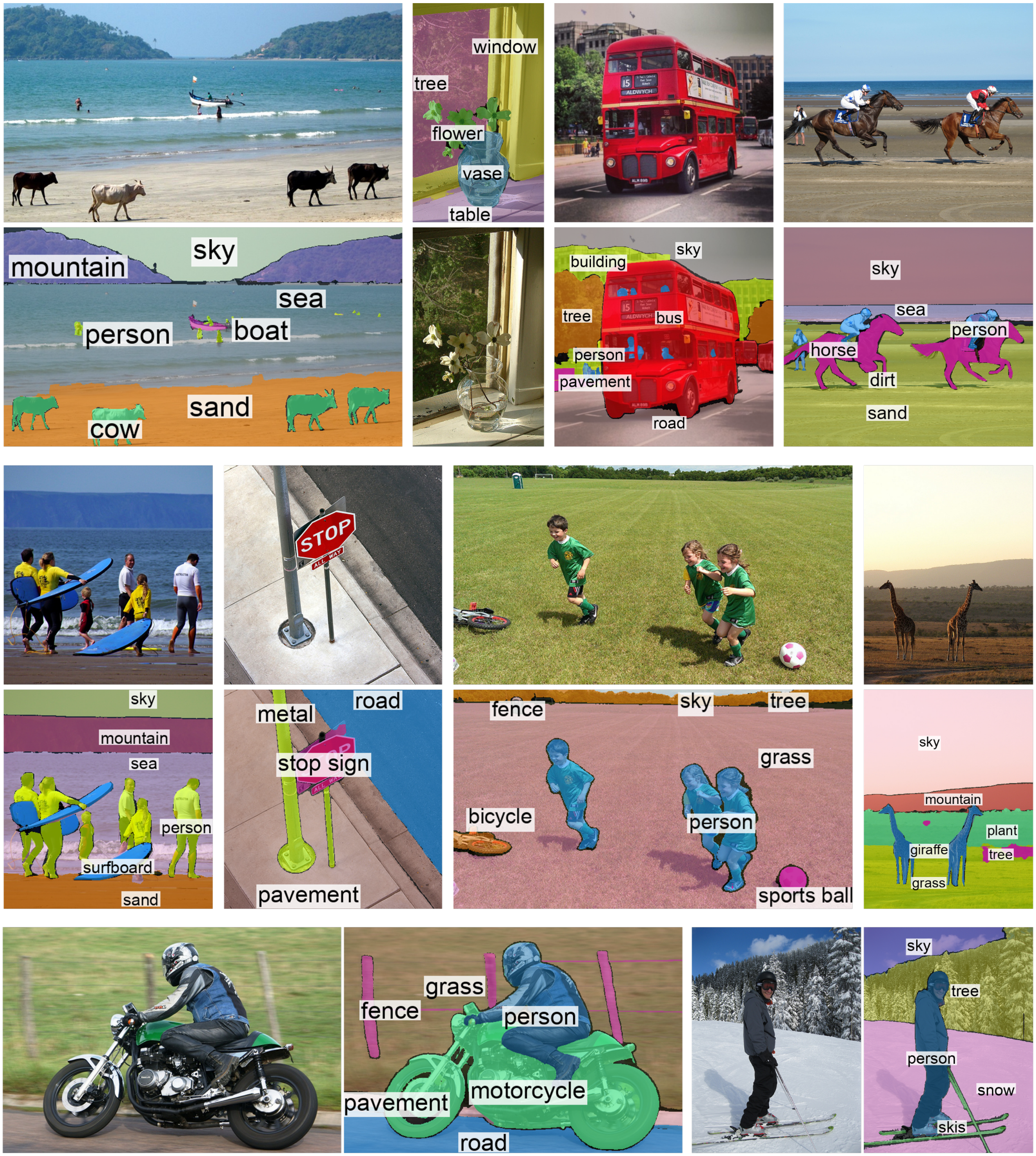


Figure A10. **Qualitative results on COCO Stuff [6] with 171 categories.**

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. 3

[2] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023. 3

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 3

[4] Luca Barsellotti, Roberto Amoroso, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Training-free open-vocabulary segmentation with offline diffusion-augmented prototype generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3689–3698, 2024. 2, 3, 4, 6, 7, 8

[5] Walid Bousselham, Felix Petersen, Vittorio Ferrari, and Hilde Kuehne. Grounding everything: Emerging localization properties in vision-language transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3828–3837, 2024. 7, 8

[6] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 4, 5, 12

[7] Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11165–11174, 2023. 4, 8

[8] Jun Chen, Deyao Zhu, Guocheng Qian, Bernard Ghanem, Zhicheng Yan, Chenchen Zhu, Fanyi Xiao, Sean Chang Culatana, and Mohamed Elhoseiny. Exploring open-vocabulary semantic segmentation from clip vision encoder distillation only. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 699–710, 2023. 4, 8

[9] Seokju Cho, Heeseong Shin, Sunghwan Hong, Anurag Arnab, Paul Hongsuck Seo, and Seungryong Kim. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4113–4123, 2024. 3, 4, 8

[10] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11583–11592, 2022. 4, 8

[11] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59:167–181, 2004. 6, 9

[12] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision*, pages 540–557. Springer, 2022. 4, 8

[13] Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. Pay attention to your neighbours: Training-free open-vocabulary semantic segmentation. *arXiv preprint arXiv:2404.08181*, 2024. 7, 8

[14] Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. Diffusion models for open-vocabulary segmentation. In *European Conference on Computer Vision*, pages 299–317. Springer, 2025. 2, 7, 8

[15] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 6

[16] 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

[17] Mengcheng Lan, Chaofeng Chen, Yiping Ke, Xinjiang Wang, Litong Feng, and Wayne Zhang. Proxyclip: Proxy attention improves clip for open-vocabulary segmentation. In *ECCV*, 2024. 6, 7, 8

[18] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 6

[19] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7061–7070, 2023. 4, 8

[20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 3, 6

[21] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. Segclip: Patch aggregation with learnable centers for open-vocabulary semantic segmentation. In *International Conference on Machine Learning*, pages 23033–23044. PMLR, 2023. 4, 8

[22] Jiayun Luo, Siddhesh Khandelwal, Leonid Sigal, and Boyang Li. Emergent open-vocabulary semantic segmentation from off-the-shelf vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4029–4040, 2024. 7, 8

[23] Divyanshu Malik, Xiwei Xuan, and Kwan-Liu Ma. Towards interactive 3d surgical scene reconstruction: An incremental training and monitoring framework. In *2025 IEEE 22nd International Symposium on Biomedical Imaging (ISBI)*, pages 1–4. IEEE, 2025. 5

[24] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the*

IEEE conference on computer vision and pattern recognition, pages 891–898, 2014. 4, 11

[25] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 6

[26] Tong Shao, Zhuotao Tian, Hang Zhao, and Jingyong Su. Explore the potential of clip for training-free open vocabulary semantic segmentation. *arXiv preprint arXiv:2407.08268*, 2024. 7, 8

[27] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. *Advances in Neural Information Processing Systems*, 35:33754–33767, 2022. 7, 8

[28] Shuyang Sun, Runjia Li, Philip Torr, Xiuye Gu, and Siyang Li. Clip as rnn: Segment countless visual concepts without training endeavor. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13171–13182, 2024. 7, 8

[29] Feng Wang, Jieru Mei, and Alan Yuille. Sclip: Rethinking self-attention for dense vision-language inference. In *European Conference on Computer Vision*, pages 315–332. Springer, 2024. 7, 8

[30] Haoxiang Wang, Pavan Kumar Anasosalu Vasu, Fartash Faghri, Raviteja Vemulapalli, Mehrdad Farajtabar, Sachin Mehta, Mohammad Rastegari, Oncel Tuzel, and Hadi Pouransari. Sam-clip: Merging vision foundation models towards semantic and spatial understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3635–3647, 2024. 4, 8

[31] Jinglong Wang, Xiawei Li, Jing Zhang, Qingyuan Xu, Qin Zhou, Qian Yu, Lu Sheng, and Dong Xu. Diffusion model is secretly a training-free open vocabulary semantic segmenter. *arXiv preprint arXiv:2309.02773*, 2023. 7, 8

[32] Xiaoqi Wang, Wenbin He, Xiwei Xuan, Clint Sebastian, Jorge Piazentin Ono, Xin Li, Sima Behpour, Thang Doan, Liang Gou, Han-Wei Shen, et al. Use: Universal segment embeddings for open-vocabulary image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4187–4196, 2024. 2

[33] Yuan Wang, Rui Sun, Naisong Luo, Yuwen Pan, and Tianzhu Zhang. Image-to-image matching via foundation models: A new perspective for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3952–3963, 2024. 2, 8

[34] Ji-Jia Wu, Andy Chia-Hao Chang, Chieh-Yu Chuang, Chun-Pei Chen, Yu-Lun Liu, Min-Hung Chen, Hou-Ning Hu, Yung-Yu Chuang, and Yen-Yu Lin. Image-text co-decomposition for text-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26794–26803, 2024. 4, 8

[35] Yun Xing, Jian Kang, Aoran Xiao, Jiahao Nie, Ling Shao, and Shijian Lu. Rewrite caption semantics: Bridging semantic gaps for language-supervised semantic segmentation. *Advances in Neural Information Processing Systems*, 36, 2024. 4, 8

[36] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18134–18144, 2022. 4, 8

[37] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Yi Wang, Yu Qiao, and Weidi Xie. Learning open-vocabulary semantic segmentation models from natural language supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2935–2944, 2023. 4, 8

[38] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, pages 736–753. Springer, 2022. 4, 8

[39] Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. Side adapter network for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2945–2954, 2023. 4, 8

[40] Muyang Yi, Quan Cui, Hao Wu, Cheng Yang, Osamu Yoshie, and Hongtao Lu. A simple framework for text-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7071–7080, 2023. 4, 8

[41] Dengke Zhang, Fagui Liu, and Quan Tang. Corrclip: Reconstructing correlations in clip with off-the-shelf foundation models for open-vocabulary semantic segmentation. *arXiv preprint arXiv:2411.10086*, 2024. 8

[42] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019. 4, 10

[43] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European Conference on Computer Vision*, pages 696–712. Springer, 2022. 7, 8