

Understanding Multi-Granularity for Open-Vocabulary Part Segmentation

Jiho Choi^{1*}, Seonho Lee^{1*}, Seungho Lee², Minhyun Lee², Hyunjung Shim^{1†}

¹Graduate School of Artificial Intelligence, KAIST, Republic of Korea

²School of Integrated Technology, Yonsei University, Republic of Korea

{jihochoi, glanceeyes, kateshim}@kaist.ac.kr, {seungholee, lmh315}@yonsei.ac.kr

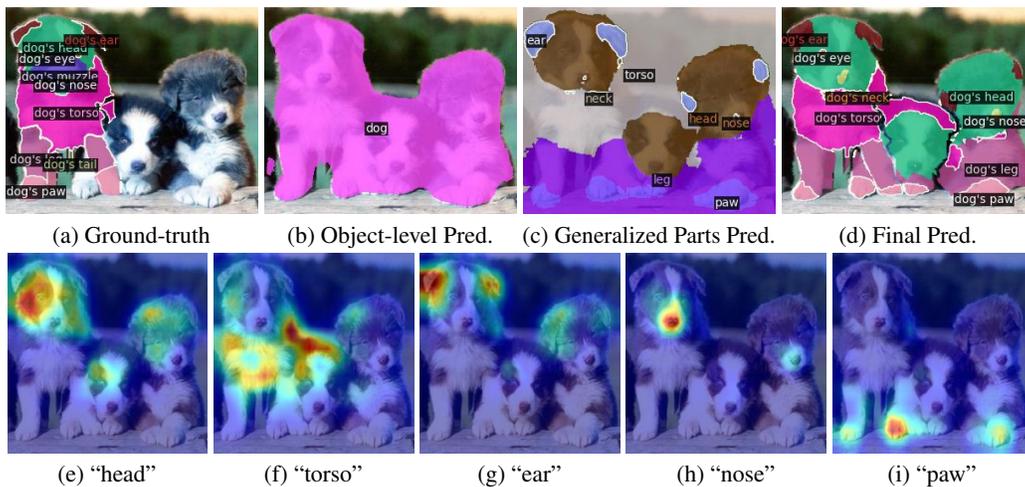


Figure 1: Prediction results of our PartCLIPSeg for unseen categories in the Pascal-Part-116 [7, 46] validation set. A “dog” is **unseen** during training. The final prediction of PartCLIPSeg utilizes (b) object-level context and (c) generalized parts, incorporating disjoint activation among (e)–(i) parts, and enhancing activation for smaller parts (e.g., (h) “nose”).

Abstract

Open-vocabulary part segmentation (OVPS) is an emerging research area focused on segmenting fine-grained entities using diverse and previously unseen vocabularies. Our study highlights the inherent complexities of part segmentation due to intricate boundaries and diverse granularity, reflecting the knowledge-based nature of part identification. To address these challenges, we propose PartCLIPSeg, a novel framework utilizing generalized parts and object-level contexts to mitigate the lack of generalization in fine-grained parts. PartCLIPSeg integrates competitive part relationships and attention control, alleviating ambiguous boundaries and underrepresented parts. Experimental results demonstrate that PartCLIPSeg outperforms existing state-of-the-art OVPS methods, offering refined segmentation and an advanced understanding of part relationships within images. Through extensive experiments, our model demonstrated a significant improvement over the state-of-the-art models on the Pascal-Part-116, ADE20K-Part-234, and PartImageNet datasets. Our code is available at <https://github.com/kaist-cvml/part-clipseg>.

*Equal contribution

†Corresponding author

1 Introduction

The pursuit of understanding parts and multi-granularity in computer vision [7, 13, 21] mirrors the innate complexities of animal instincts. For example, a “cheetah” instinctively targets an “impala’s neck” during a hunt, demonstrating its ability to distinguish specific parts. This ability extends to applications such as robot commands [44], fine-grained controls on image editing [31], and more sophisticated image generation [45]. Part segmentation aims to mimic this ability by recognizing intricate details (e.g., parts) within objects, going beyond simple object-level segmentation to achieve detailed and diverse entity recognition.

Recognizing parts is more challenging than recognizing whole objects due to their complexity and diversity. Parts often have ambiguous boundaries not only defined by visual cues but also require a broader spectrum of contextual information, reflecting their knowledge-based nature. For example, the “head” of a “dog” may include only the “face” or also the “neck” depending on the annotators’ perspective [7, 21].

To address difficulties in part segmentation, Open-Vocabulary Part Segmentation (OVPS) [40, 44, 46] has evolved by leveraging the knowledge of powerful Vision-Language Models (VLMs) like CLIP [38] or ALIGN [24]. Especially, it aims to achieve adaptive recognition and processing of previously unseen categories with the aid of pre-trained VLMs, pushing the boundaries of vocabularies in traditional part segmentation. By utilizing Oracle supervision of base classes during training, recent studies in OVPS exploit part-level knowledge of base classes to generalize to novel classes. Recently, VLPART [40] uses DINO [5] features to map correspondences between base and novel classes and creates pseudo labels for the novel categories. OV-PARTS [46] addresses the ambiguity of part boundaries by introducing object mask prompts and transferring knowledge of base class through a few-shot approach. These methods successfully extract knowledge from VLMs and extend it to novel classes, achieving significant performance improvements in open-vocabulary settings.

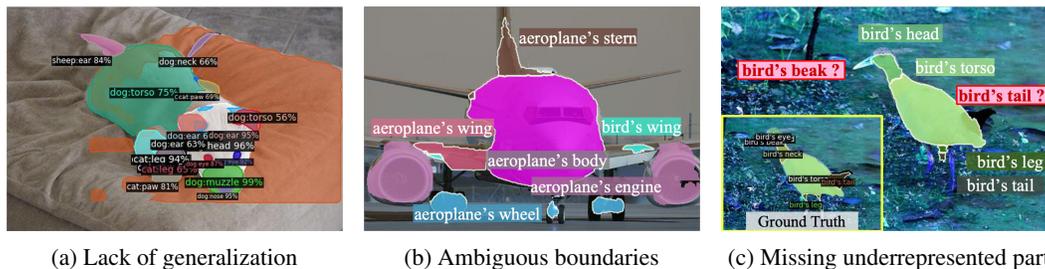


Figure 2: Limitations of existing OVPS methods in predicting unseen categories. (a) Lack of generalization: Classification of a “dog’s parts” involving categories like “cats” and “sheep”, “dog’s tail” misclassified as “sheep’s ear”. (VLPART [40]) (b) Ambiguous boundaries: Vague boundary output of “aeroplane’s body”. (c) Missing underrepresented parts: Neglecting parts such as “beak” and “leg”. (CLIPSeg [32, 46]).

However, through empirical analysis of existing OVPS methods, we observed several common limitations in Figure 2. (Lack of generalization in (a)) Despite understanding part-level information, they often misidentify parts at the object level, e.g., a “dog’s leg” as a “cat’s leg”. Also, part-level misclassification occurs as the knowledge of parts in the base class fails to generalize to a novel class, e.g., “dog’s tail” as a “sheep’s ear”. (Ambiguous boundaries of parts in (b)) They fail to maintain non-overlapping relationships between parts, frequently resulting in overlaps, e.g., an “airplane’s wing” overlapping with its “body” or the presence of empty spaces where no part is predicted. (Missing underrepresented parts in (c)) They ignore small and less frequent parts, causing prediction bias based on part size.

To overcome these limitations, we propose a novel framework called PartCLIPSeg, which consists of three main components. First, we devise generalized parts with object-level contexts to address the lack of generalization issue as the upper side of Figure 1. It explicitly obtains object-level and part-level pseudo-labels from VLMs and trains the OVPS model to satisfy both types of supervision. This guides the model to learn object boundaries while recognizing both part and object-level classes. Then, we suggest an attention control for minimizing the overlap between predicted parts, ensuring that parts are clearly separated as the lower side of Figure 1. In this way, we effectively leverage

internal part information to learn ambiguous part boundaries. Finally, we enhance the activation related to certain parts by normalizing the activation scale of CLIP’s self-attention information. It prevents small and less frequent areas from being ignored in pseudo-labels. This strategy ensures that the smallest granularity levels are retained in the final prediction. Through these three modules, PartCLIPSeg effectively addresses the challenges of existing OVPS methods and achieves robust multi-granularity segmentation. As a result, the proposed method achieves significant improvements in mIoU for both unseen and the harmonic mean when compared to previous state-of-the-art methods on Pascal-Part-116, ADE20K-Part-234, and PartImageNet in both Pred-All and Oracle-Obj settings.

2 Related Work

Open-Vocabulary Semantic Segmentation. Open-vocabulary [19, 55] semantic segmentation (OVSS) goes beyond traditional semantic segmentation, which is restricted to predefined categories, by enabling predictions for unseen classes. Pioneering works focused on aligning predefined text embeddings with pixel-level visual features [4, 48, 56]. By leveraging large-scale Vision-Language Models (VLMs) like CLIP [38] and ALIGN [24], OVSS enables zero-shot segmentation through rich multi-modal features learned from extensive image-text pairs. MaskCLIP [58] modified CLIP’s image encoder to directly handle visual and text features for segmenting novel classes. Some works proposed two-stage strategy [15, 16, 18, 20, 29, 30, 51, 52]: first, models generate class-agnostic mask proposals [9, 10]; then, a pre-trained VLM predicts the category for each region. Some studies have introduced diffusion models to improve mask generation quality [51] or fine-tuned CLIP to enhance classification capabilities [20, 29]. Other studies have adopted a single-stage framework [11, 27, 32, 49, 54, 59]. They use pre-trained CLIP models to align pixel-level visual features with text features. CLIPSeg [32] adds a transformer-based pixel decoder with a FiLM [17] module to fuse multi-modal features. ZegCLIP [59] enhances segmentation by incorporating learnable tokens. SAN [53] adopted a side adapter network for a CLIP-aware end-to-end approach to predict proposal-wise classification. FC-CLIP [54] uses a frozen convolutional CLIP to predict class-agnostic masks and classifies using mask-pooled features [54]. CAT-Seg [11] and SED [49] generate pixel-level cost maps and refine them for segmentation.

Part Segmentation. Part segmentation aims to identify the individual parts of objects, a task that is more complex and costly due to the smaller and more diverse nature of parts compared to whole objects. To tackle this, various datasets like Pascal-Part [7], PartImageNet [21], ADE20k-Part [57], Cityscapes-Panoptic-Part [13], and PACO [2] provide diverse and detailed part annotations. Earlier studies [7, 12, 22, 23, 43] used self-supervised constraints and contrastive settings for effective part-level entity segmentation. Recent studies extended this to open-vocabulary scenarios [35, 40, 46], opening new avenues for handling diverse parts. By leveraging class-agnostic detectors [35] and Vision-Language Models like CLIP [40, 46], part segmentation has extended its generalization ability to unseen parts. Our work builds upon and extends methodologies from these studies.

3 Methodology

As illustrated in Figure 2, we identified three primary challenges of open-vocabulary part segmentation (OVPS): lack of generalization, overlapping parts, and missing underrepresented parts. Recognizing object-specific parts (such as “dog’s torso”) cannot be determined solely by looking at each part in isolation; it is imperative to consider both generalized part information and the overall context of the object. Furthermore, some parts may have overlapping meanings across different granularity labels (e.g., “eye”, “face”, and “head”). This implies that predictions should consider direct guidance for each part as well as the relationships between different parts. These intricate spatial and functional dependencies between parts are crucial for achieving a holistic understanding and precise predictions in fine-grained entity segmentation tasks.

Based on this motivation, we propose a novel OVPS method, PartCLIPSeg. This method leverages *generalized part* information combined with *object-level context* to tackle the lack of generalization problem (see Section 3.2). Also, we directly minimize the overlap among part predictions to improve the part boundaries (see Section 3.3.1). Finally, we normalize the scale of attention activation from various parts for handling missing underrepresented parts (see Section 3.3.2). The overall architecture of our method is shown in Figure 3.

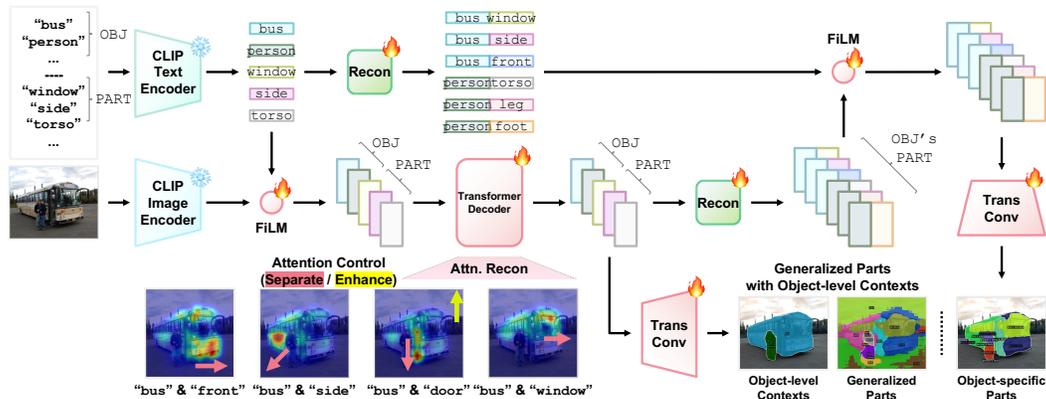


Figure 3: **The overall architecture of PartCLIPSeg.** The embeddings derived from the object category name and the part category name are conditioned using the FiLM operation. Each embedding, modified through attention control, is subsequently reconstructed to predict the final object-specific part results.

3.1 Preliminary

OVPS aims to segment an image into a set of object-specific part categories $C_{\text{obj-part}}^{\text{test}}$ (e.g., “dog’s head,” “car’s front”) in the *test* set, where the image is $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$, and H and W are the height and width. During training, image-mask pairs $\{(\mathcal{I}_k, \mathcal{M}_k)\}$ are used, consisting of images \mathcal{I}_k and corresponding ground-truth mask \mathcal{M}_k which only contains the object-specific part categories $C_{\text{obj-part}}^{\text{train}}$ (e.g., “cat’s head,” “bus’s front”) in the *train* set.

Zero-Shot Part Segmentation. Open-vocabulary is a generalized zero-shot task, allowing the zero-shot segmentation protocol to evaluate zero-shot part segmentation performance. In this setting, *train* and *test* category names are divided into *seen* (base) and *unseen* (novel) sets, respectively, with disjoint object-specific category names; $\{C_{\text{obj-part}}^{\text{unseen}} \cap C_{\text{obj-part}}^{\text{seen}} = \emptyset\}$.

Cross-Dataset Part Segmentation. In this setting, the model is trained on one dataset and evaluated on another without fine-tuning. This means that the category names of the *train* and *test* sets come from different datasets, denoted as $C_{\text{obj-part}}^{\text{train}} \neq C_{\text{obj-part}}^{\text{test}}$. Considering the domain gap between the datasets, such as differences in granularity, this setting is more challenging.

3.2 Generalized Parts with Object-level Contexts

To address the problem of a lack of generalization, we propose leveraging generalized parts with object-level contexts. The concept of generalized parts involves identifying and utilizing common structural components that are shared across different object-level categories. For instance, many animals have parts like “head” or “torso” which, although functionally and visually distinct, may share certain underlying characteristics. By introducing generalized parts from object-specific parts, our PartCLIPSeg can efficiently recognize and segment these object-specific parts across diverse object classes, significantly enhancing the model’s ability to generalize from seen to unseen categories.

Although generalized parts help distinguish the part-level categories, the visual information of a part may not suffice for accurately classifying their object-level categories. For instance, predicting the “leg” part of an animal can be challenging to identify when solely examining the part as it may not clearly indicate to which animal it belongs. For this reason, there have been attempts to incorporate object-level guidance [33, 40, 46] in part segmentation. However, object-level guidance without a generalized part may lose contextual information and miss hierarchical relationships.

By integrating object contexts with generalized parts, PartCLIPSeg employs object-level guidance that captures the holistic essence of the object to which parts belong. This integration allows for a more precise understanding and classification of parts, improving the overall performance of OVPS.

Object and Part Embedding Generation. We modified the architecture of CLIPSeg [32, 46], which adopted CLIP [38] encoder-decoder architecture for semantic segmentation. However, it is worth

noting that our approach of utilizing generalized parts with object-level context is orthogonal to other previously proposed object-level segmentation methods [5, 26, 28, 39].

The proposed approach begins by parsing an object-specific part category name, $c_{\text{obj-part}} \in \mathbf{C}_{\text{obj-part}}$, into separate components: an object category name (c_{obj}) and a generalized part category name (c_{part}), e.g., “cat” and “torso”. Then, the CLIP text encoder, $\text{CLIP}_{\mathcal{T}}^*(\cdot)$, is used to transform these category names into their respective CLIP embeddings ($e_{\text{obj}}^{\mathcal{T}}$ and $e_{\text{part}}^{\mathcal{T}}$). It will condition the image features, $e^{\mathcal{I}}$, derived from the CLIP image encoder, $\text{CLIP}_{\mathcal{I}}^*(\cdot)$ as:

$$e_{[\text{obj} | \text{part}]}^{\mathcal{T}} = \text{CLIP}_{\mathcal{T}}^*(c_{[\text{obj} | \text{part}]}) , e^{\mathcal{I}} = \text{CLIP}_{\mathcal{I}}^*(\mathcal{I}), \quad (1)$$

where $*$ denotes frozen pre-trained models. By using Feature-wise Linear Modulation (FiLM) [36, 42], each category name embeddings respectively modulate the image features as:

$$e_{[\text{obj} | \text{part}]}^{\mathcal{I}} = e^{\mathcal{I}} \oplus \text{FiLM}(e_{[\text{obj} | \text{part}]}^{\mathcal{T}}), \quad (2)$$

where \oplus is an element-wise sum. FiLM is an adaptive affine transformation widely used for multi-modal or conditional tasks. It helps retrieve adequate conditioning for the image features. The modulated image features, $e_{[\text{obj} | \text{part}]}^{\mathcal{I}}$, corresponding to each object and part category name, pass through a decoder module. The decoder module will be discussed in detail in Section 3.3. They then proceed through a transposed convolution model. Finally, the output mask of the object \hat{s}^o and part \hat{s}^p are evaluated with ground-truth mask of objects, s^o , and parts, s^p . Oracle supervision for the object and parts mask is simply computed from a combination of object-specific parts annotations: $s \in \mathcal{M}$.

Object-specific Part Construction. We utilize previously computed generalized part embeddings ($e_{\text{part}}^{\mathcal{I}}$, $e_{\text{part}}^{\mathcal{T}}$) and object embeddings ($e_{\text{obj}}^{\mathcal{I}}$, $e_{\text{obj}}^{\mathcal{T}}$) to reconstruct object-specific part embeddings. This process involves separate operations on modulated image features and category name embeddings.

Initially, we project the concatenated results of the object category name with the generalized part category name. This is to synthesize the embeddings for the target object-specific part category name. The approach ensures that the resultant embeddings are highly representative of parts and contextually relevant. The equivalent operation is applied to both object-level image features and part-level image features to generate object-specific image features as:

$$e_{\text{obj-part}}^{[\mathcal{T}|\mathcal{I}]} = \text{Proj}\left(\left[e_{\text{obj}}^{[\mathcal{T}|\mathcal{I}]} \mid e_{\text{part}}^{[\mathcal{T}|\mathcal{I}]} \right]\right). \quad (3)$$

The resulting object-specific part embeddings are further refined by a FiLM process. Combined with the respective object-specific image features, final modulated object-specific part embeddings, $e_{\text{obj-part}}$ is computed as:

$$e_{\text{obj-part}} = e_{\text{obj-part}}^{\mathcal{I}} \oplus \text{FiLM}(e_{\text{obj-part}}^{\mathcal{T}}). \quad (4)$$

These embeddings are then processed through a deconvolution layer to produce the final segmentation masks $s \in \mathcal{M}$. This step ensures that the embeddings are precisely aligned to enhance the definition and accuracy of the object-specific part masks. It effectively bridges the gap between object and part-level categorical information with object-specific parts information.

Object, Part, and Object-specific Part Mask Supervision. The mask supervision is provided for three distinct categories: object-specific parts, objects, and generalized parts. This multi-faceted supervision enables our model to effectively disentangle generalized parts from objects, thereby facilitating a more nuanced learning process for OVPS. This disentanglement is crucial for the model to accurately recognize and differentiate between various object categories and their corresponding parts. It enhances the model’s ability to handle complex segmentation tasks with unseen object-specific parts. The overall mask guidance loss can be defined as follows:

$$\mathcal{L}_{\text{mask}} = \sum_{i=1}^{|\mathbf{C}_{\text{obj-part}}|+1} \underbrace{\text{BCE}(s_i, \hat{s}_i)}_{\text{object-specific part}} + \lambda_{\text{obj}} \sum_{i=1}^{|\mathbf{C}_{\text{obj}}|+1} \underbrace{\text{BCE}(s_i^o, \hat{s}_i^o)}_{\text{object guidance}} + \lambda_{\text{part}} \sum_{i=1}^{|\mathbf{C}_{\text{part}}|} \underbrace{\text{BCE}(s_i^p, \hat{s}_i^p)}_{\text{generalized part guidance}}, \quad (5)$$

where $|\mathbf{C}_{\text{obj-part}}| + 1$ and $|\mathbf{C}_{\text{obj}}| + 1$ are for uncategory (or background) prediction. The disentangled object and part generalization with object-specific parts guidance provides a clue to the lack of generalization problem.

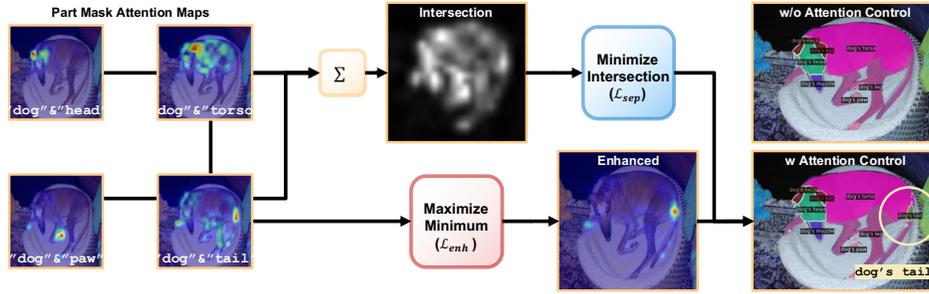


Figure 4: Example of attention control using separation and enhance losses. The proposed method manipulates attention maps to accurately identify and segment small parts.

3.3 Attention Control for Ambiguity and Omission

In this subsection, we address the previously mentioned challenges: (1) ambiguity in part boundaries and (2) omission of small or infrequently appearing parts. The main reason for these challenges is the incomplete guidance from knowledge-based, multi-granularity characteristics of parts. To overcome these, we adopt unsupervised methods traditionally used in fine-grained recognition and part discovery studies [7, 12, 43]. Specifically, we utilize approaches for adjusting self-attention activation inspired by the recent diffusion methods [6, 25, 41].

We assume that the distribution of self-attention activation maps for visual tokens belonging to the same object-specific part mask should exhibit inter-similarity characteristics [41], implying similar distributions. To this end, we first compute the average self-attention map $\mathcal{A}_{\mathcal{M}_c}$ for each object-specific part mask \mathcal{M}_c , where $c \in \mathbf{C}_{\text{obj-part}}$ represents an object-specific part category. This is done by summing the self-attention activation maps from channels specifically corresponding to object c_{obj} and part c_{part} , across all spatial tokens (h, w) within the mask, as follows:

$$\mathcal{A}_{\mathcal{M}_c} = \frac{1}{|\mathcal{M}_c|} \sum_{(h,w) \in \mathcal{M}_c} (\mathcal{A}_{c_{\text{obj}}}[h, w, :, :] + \mathcal{A}_{c_{\text{part}}}[h, w, :, :]). \quad (6)$$

Subsequently, the self-attention map $\mathcal{A}_{\mathcal{M}_c}$ for the object-specific part mask is refined through min-max normalization, followed by the application of a Gaussian filter to smooth the initial activation as in [6, 50]. Therefore, the dimensions of both the original and normalized self-attention maps for the object-specific part masks are as follows: $\mathcal{A}_{\mathcal{M}_c}, \mathcal{A}_{\mathcal{M}_c}^{\text{norm}} \in \mathbb{R}^{H \times W}$.

3.3.1 Minimizing Part Overlaps for Ambiguity

In the self-attention of the decoder layers, competition between object-specific parts helps define boundaries that cannot be sufficiently established by supervision alone. Using the previously obtained normalized attention map, our method generates parts with minimized intersections, inspired by [1, 3, 25, 37, 47]. This approach effectively mitigates the ambiguity issue in part boundaries. Specifically, the normalized attention activation map $\mathcal{A}_{\mathcal{M}_c}^{\text{norm}}$ is first binarized based on an arbitrary threshold γ as:

$$\mathcal{B}_{\mathcal{M}_c}(h, w) = \mathbf{1}_{\{\mathcal{A}_{\mathcal{M}_c}^{\text{norm}}(h,w) \geq \gamma\}}, \quad (7)$$

where $\mathcal{B}_{\mathcal{M}_c}$ denotes binarized attention map for part mask \mathcal{M}_c . From now on, $\mathbf{C}_{\text{obj-part}}$ is simply denoted as \mathbf{C} . The separation loss \mathcal{L}_{sep} , which indicates the degree of intersection between object-specific parts, is as follows:

$$\mathcal{L}_{\text{sep}} = \frac{1}{|\mathbf{C}|} \left| \frac{\{(h, w) \mid \sum_{c \in \mathbf{C}} \mathcal{B}_{\mathcal{M}_c}(h, w) > 1\}}{\{(h, w) \mid \sum_{c \in \mathbf{C}} \mathcal{B}_{\mathcal{M}_c}(h, w) \geq 1\}} \right|, \quad (8)$$

where separating activation mitigates the challenge of ambiguous boundaries between parts.

3.3.2 Enhancing Part Activation for Omission

To address the omission problem, we employ a method inspired by attention controls in modern diffusion-based approaches [3, 6]. This approach enhances the activation within the self-attention

activation map to enhance underrepresented parts before normalization. Specifically, for each object-specific part mask, the maximum value within the attention map is identified. Subsequently, among all object-specific parts, the minimum activation of the part with the maximum value is enhanced as:

$$\mathcal{L}_{\text{enh}} = 1 - \min_{c \in \mathcal{C}} \left(\max_{(h,w) \in \mathcal{M}_c} \mathcal{A}_{\mathcal{M}_c}[h, w] \right), \quad (9)$$

thereby boosting its representational efficacy. In this way, the enhancement loss \mathcal{L}_{enh} provides sufficient guidance for small or infrequently occurring parts, effectively mitigating the omission problem.

The training objective for PartCLIPSeg integrates three key loss components as:

$$\mathcal{L}_{\text{all}} = \mathcal{L}_{\text{mask}} + \lambda_{\text{sep}} \mathcal{L}_{\text{sep}} + \lambda_{\text{enh}} \mathcal{L}_{\text{enh}}, \quad (10)$$

where (1) $\mathcal{L}_{\text{mask}}$ for generalized parts with object-level context, (2) \mathcal{L}_{sep} for addressing ambiguous boundaries, (3) \mathcal{L}_{enh} for handling missing underrepresented parts, and λ_{sep} and λ_{enh} are hyperparameters.

4 Experiments

4.1 Experimental Setups

Datasets. We evaluate our method on three part segmentation datasets: Pascal-Part-116 [7, 46], ADE20K-Part-234 [46, 57], and PartImageNet [21]. Pascal-Part-116 [7, 46] consists of 8,431 training images and 850 test images. It is a modified version of PascalPart [7] by removing direction indicators for certain part classes and merging them to avoid overly complex part definitions. This dataset contains a total of 116 object part classes across 17 object categories. ADE20K-Part-234 [46, 57] consists of 7,347 training images and 1,016 validation images. It provides instance-level object mask annotations along with their corresponding part mask annotations, including 44 objects and 234 parts. PartImageNet [21] contains 16k training images and 2.9k validation images, segmented into 158 object classes from ImageNet [14] and organizes them into 11 super-categories. For this study, we select 40 object classes that represent common categories to assess cross-dataset performance effectively. More details about the datasets can be found in the supplementary materials.

Evaluation Protocols. We use two evaluation protocols for the performance of OVPS: (1) **Pred-All** setting, where the ground truth object-level mask and object class are not provided, and (2) **Oracle-Obj** setting, where the ground truth object-level mask and object class are known. In particular, the **Pred-Obj** setting in OV-PARTS [46] uses predicted masks from the off-the-shelf segmentation model. In contrast, our **Pred-All** setting is a more challenging and practical setting because it does not rely on additional predicted masks or foundation models but solely uses the predicted object masks from the proposed model. For both evaluation protocols, we used mean Intersection over Union (mIoU) as an evaluation metric, which is widely used to measure segmentation performance. Additionally, we utilized the harmonic mean of the results from the seen and unseen categories as the final evaluation metric.

Implementation Details. We build upon CLIPSeg [32, 46], a CLIP-based encoder-decoder model. The implementation details can be found in the supplementary material.

4.2 Performance Evaluation

Zero-Shot Part Segmentation. We compare our PartCLIPSeg to previous methods [11, 32, 46, 52] on three OVPS benchmarks [7, 57]. As shown in Table 1, PartCLIPSeg consistently outperforms previous approaches by significant margins on Pascal-Part-116, demonstrating its zero-shot ability, with performance improvements of 3.94% in the Pred-All setting and 3.55% in the Oracle-Obj setting. The more challenging ADE20K-Part-234 dataset, which is a fine-grained segmentation dataset, further highlights the effectiveness of PartCLIPSeg. As shown in Table 2, PartCLIPSeg achieves a harmonic mean mIoU of 11.38% in the Pred-All setting, outperforming the best-performing baseline by 7.85%. In the Oracle-Obj setting, it achieves 38.60%, which is 4.45% higher than the best baseline. Notably, PartCLIPSeg shows significant performance improvement in unseen categories, demonstrating its strong generalizability. Considering that performance in unseen categories is crucial in a zero-shot

Table 1: Comparison of zero-shot performance with state-of-the-art methods on Pascal-Part-116.

Method	Backbone	Pred-All			Oracle-Obj		
		Seen	Unseen	Harmonic	Seen	Unseen	Harmonic
ZSSeg+ [52]	ResNet-50	38.05	3.38	6.20	54.43	19.04	28.21
VLPart [40]	ResNet-50	35.21	9.04	14.39	42.61	18.70	25.99
CLIPSeg [32, 46]	ViT-B/16	27.79	13.27	17.96	48.91	27.54	<u>35.24</u>
CAT-Seg [11, 46]	ViT-B/16	28.17	25.42	<u>26.72</u>	36.20	<u>28.72</u>	32.03
PartCLIPSeg (Ours)	ViT-B/16	43.91 ± 0.45	<u>23.56</u> ± 0.21	30.67 ± 0.09 (+3.94)	<u>50.02</u> ± 0.51	31.67 ± 0.29	38.79 ± 0.13 (+3.55)

¹ The best score is **bold** and the second-best score is underlined. The standard error of an average of 5 results is reported. These are the same for all experiments.

Table 2: Comparison of zero-shot performance with state-of-the-art methods on ADE20K-Part-234.

Method	Backbone	Pred-All			Oracle-Obj		
		Seen	Unseen	Harmonic	Seen	Unseen	Harmonic
ZSSeg+ [52]	ResNet-50	32.20	0.89	1.74	43.19	27.84	33.85
CLIPSeg [32, 46]	ViT-B/16	3.14	0.55	0.93	38.15	30.92	34.15
CAT-Seg [11, 46]	ViT-B/16	7.02	2.36	3.53	33.80	25.93	29.34
PartCLIPSeg (Ours)	ViT-B/16	<u>14.15</u> ± 0.51	9.52 ± 0.13	11.38 ± 0.10 (+7.85)	<u>38.37</u> ± 0.14	38.82 ± 0.31	38.60 ± 0.08 (+4.45)

Table 3: Comparison of zero-shot performance with state-of-the-art method on PartImageNet.

Method	Backbone	Pred-All			Oracle-Obj		
		Seen	Unseen	Harmonic	Seen	Unseen	Harmonic
CLIPSeg [32, 46]	ViT-B/16	32.39	12.27	17.80	53.91	37.17	44.00
PartCLIPSeg (Ours)	ViT-B/16	38.82 ± 0.74	19.47 ± 0.45	25.94 ± 0.32 (+8.14)	56.26 ± 0.29	51.65 ± 0.62	53.85 ± 0.37 (+9.85)

scenario, these results are significant despite some performance degradation in seen categories. We also evaluated PartCLIPSeg on PartImageNet. According to Table 3, PartCLIPSeg shows a notable improvement over CLIPSeg.

We present the segmentation results of PartCLIPSeg in comparison to state-of-the-art open-vocabulary part segmentation methods [11, 32, 40] on Pascal-Part-116. Specifically, we focus on **qualitative performance** on unseen categories such as “dog”, “sheep”, “car”, and “bird”. As shown in Figure 5 for the Pred-All and Figure 6 for the Oracle-Obj setting, the proposed method effectively segments target parts regardless of the need for predefined masks during inference. Notably, PartCLIPSeg excels at identifying smaller, often overlooked part classes such as “eye”, “tail”, and “headlight”. Additionally, our method effectively segments multiple objects and their respective parts, a challenge for other methods, demonstrating the effectiveness of PartCLIPSeg in zero-shot part segmentation. Its improved performance on unseen categories and higher accuracy in challenging environments highlight the robustness and generalization capabilities of PartCLIPSeg. Consistent improvements on Pascal-Part-116, ADE20K-Part-234, and PartImageNet demonstrate that PartCLIPSeg sets a new standard in open-vocabulary part segmentation.

Cross-Dataset Part Segmentation.

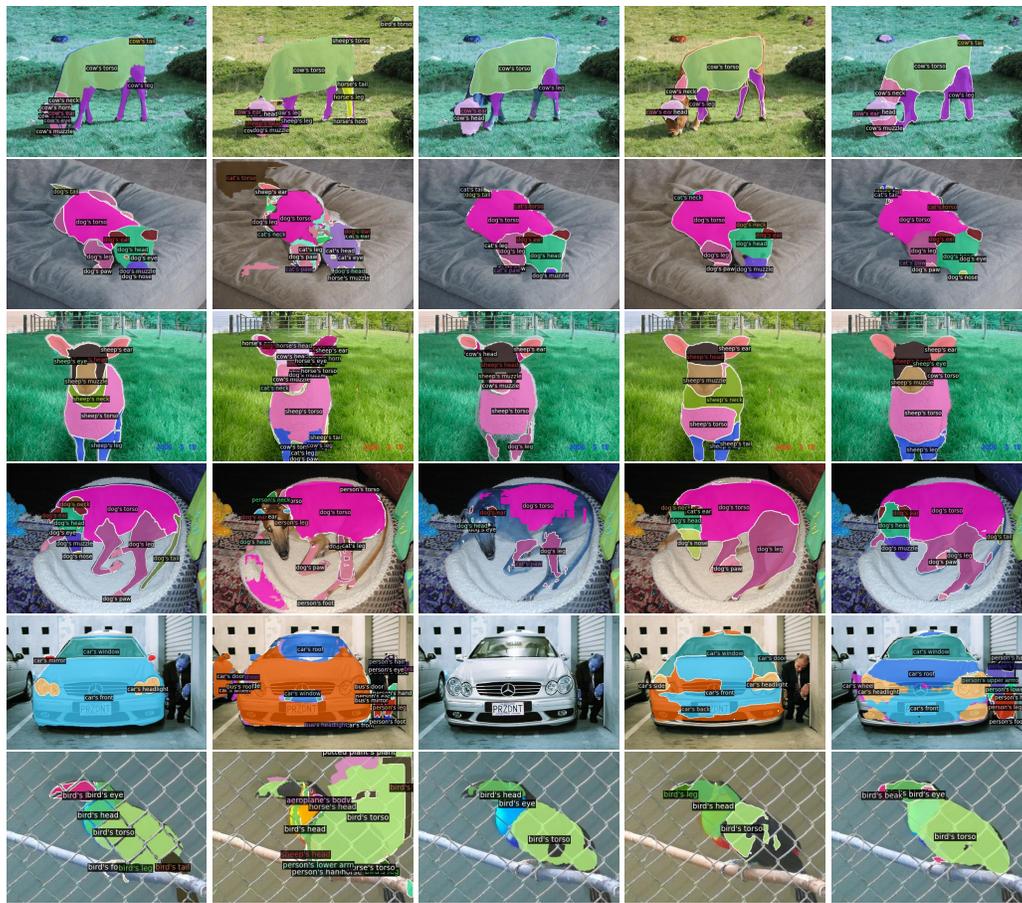
Table 4 validated the efficacy of our approach in a cross-dataset setting, where category names, annotation style, and granularity of mask may vary. Additionally, unlike zero-shot situations within the same dataset, there are differences in the types and diversity of parts. Initially, we trained our model on PartImageNet and ADE20K-Part-234 respectively. Subsequent tests on Pascal-Part-116 [7, 46] showed that PartCLIPSeg outperforms CLIPSeg in both the Pred-All and Oracle-Obj settings, confirming our method’s superiority on generalization in different datasets.

Table 4: Cross-dataset performance.

Method	Pred-All	Oracle-Obj
PartImageNet → Pascal-Part-116		
CLIPSeg [32, 46]	11.72	14.87
PartCLIPSeg (Ours)	14.74 (+3.02)	19.86 (+4.99)
ADE20K-Part-234 → Pascal-Part-116		
CLIPSeg [32, 46]	5.41	17.82
PartCLIPSeg (Ours)	10.37 (+4.96)	17.94 (+0.12)

Table 5: Impact of attention control losses.

Loss		Pred-All			Oracle-Obj		
\mathcal{L}_{sep}	\mathcal{L}_{enh}	Seen	Unseen	Harmonic	Seen	Unseen	Harmonic
Pascal-Part-116							
✗	✗	43.86	21.89	29.20	49.09	31.26	38.20
✓	✗	44.01	23.18	30.37	50.37	31.45	38.72
✓	✓	<u>43.91</u>	23.56	30.67	<u>50.02</u>	31.67	38.79
ADE20K-Part-234							
✗	✗	10.86	8.33	9.43	37.39	<u>36.49</u>	36.93
✓	✗	12.78	9.38	<u>10.82</u>	39.46	36.04	37.67
✓	✓	14.15	9.52	11.38	<u>38.37</u>	38.82	38.60



(a) Ground-truth (b) VLPART [40] (c) CLIPSeg [38, 46] (d) CAT-Seg [11, 46] (e) PartCLIPSeg (Ours)

Figure 5: Qualitative results of zero-shot part segmentation on Pascal-Part-116 in **Pred-All** setting. Annotations for unseen categories (bird, car, dog, sheep, etc.) are not included in the train set.

Table 6: Performance on mean Boundary IoU (\uparrow) on Pascal-Part-116 in Oracle-Obj setting.

Method	Seen	Unseen	Harmonic
ZSSeg+ [52]	33.01	26.76	29.56
CLIPSeg [32, 46]	34.67	32.20	33.39
CAT-Seg [11, 46]	34.17	30.14	32.03
PartCLIPSeg (Ours)	36.15	39.07	37.55

Table 7: Impact of PartCLIPSeg for small parts on Pascal-Part-116 in Oracle-Obj setting. (mIoU)

Part: "eye"	bird	cat	cow	dog	sheep	person
CLIPSeg [32, 46]	3.33	18.77	3.65	16.05	0.00	15.30
PartCLIPSeg (Ours)	1.95	31.01	28.16	32.79	0.67	29.16
Part: "neck"	bird	cat	cow	dog	sheep	person
CLIPSeg [32, 46]	19.09	6.57	0.78	8.12	8.47	30.93
PartCLIPSeg (Ours)	32.51	12.00	2.75	16.37	18.80	50.71
Part: "leg"	bird	cat	cow	dog	sheep	person
CLIPSeg [32, 46]	19.61	38.62	27.85	39.34	52.63	52.67
PartCLIPSeg (Ours)	31.12	44.82	63.78	41.55	54.73	55.35

4.3 Ablation Study

In this section, we analyze the impact of each training loss on PartCLIPSeg. We focus on the roles of the separation and enhancement losses, examining how they contribute to improved segmentation accuracy.

Separation & Enhancement Losses. We conducted an ablation study to investigate the effect of the separation loss \mathcal{L}_{sep} and the enhancement loss \mathcal{L}_{enh} on the performance of PartCLIPSeg in Table 5. On Pascal-Part-116, eliminating both losses resulted in a lower harmonic mean of 29.20 in Pred-All and a harmonic mean of 38.20 in Oracle-Obj. Introducing \mathcal{L}_{sep} without \mathcal{L}_{enh} improved the harmonic mean in both Pred-All and Oracle-Obj setups. Using both losses led to the highest harmonic means of 30.67 and 38.79, respectively. Similarly, for ADE20K-Part-234, employing both losses resulted in the



Figure 6: Qualitative results of zero-shot part segmentation on Pascal-Part-116 in **Oracle-Obj** setting.

best performance, with harmonic means of 19.63 in Pred-All and 38.60 in Oracle-Obj. These results highlight the importance of both separation and enhancement losses in improving performance.

To verify the effectiveness of boundary creation of PartCLIPSeg, we examined an additional qualitative metric, Boundary IoU [8]. The results demonstrated high Boundary IoU performance, confirming that PartCLIPSeg effectively resolves ambiguous boundary issues as shown in Table 6.

Impact of PartCLIPSeg for Underrepresented Parts. We investigate the effect of the enhancement loss \mathcal{L}_{enh} on OVPS model performance, especially with respect to underrepresented parts. In Table 7, we compare our PartCLIPSeg with CLIPSeg [32, 46] on small parts such as “eye”, “neck”, and “leg” of animals in Pascal-Part-116. As shown in the table, PartCLIPSeg consistently outperforms CLIPSeg with significant improvements in most cases. Notably, there is an impressive performance increase of 35.93%p for “cow’s leg”. These improvements highlight the effectiveness of the enhancement loss in accurately segmenting small and intricate parts, demonstrating its crucial role in improving overall performance.

5 Conclusion

In this study, we introduced PartCLIPSeg, a state-of-the-art OVPS method that addresses three primary challenges in OVPS. PartCLIPSeg utilizes generalized parts and object-level guidance to effectively solve identification issues. Then, it separates parts by minimizing their overlaps in attention maps, thus learning ambiguous part boundaries. Additionally, we implemented an enhancement loss function to improve the detection of underrepresented parts. Through extensive experimentation, we have confirmed the superior performance of PartCLIPSeg.

Acknowledgements

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the MSIP (NRF-2022R1A2C3011154, RS-2023-00219019), KEIT grant funded by the Korean government (MOTIE) (No. 2022-0-00680, No. 2022-0-01045), the IITP grant funded by the Korean government (MSIT) (No. 2021-0-02068 Artificial Intelligence Innovation Hub, RS-2019-II190075 Artificial Intelligence Graduate School Program (KAIST)), and Samsung Electronics Co., Ltd (IO230508-06190-01).

Bibliography

- [1] Aishwarya Agarwal, Srikrishna Karanam, KJ Joseph, Apoorv Saxena, Koustava Goswami, and Balaji Vasan Srinivasan. A-star: Test-time attention segregation and retention for text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2283–2293, 2023.
- [2] Juan Antonio Balbuena, Raúl Míguez-Lozano, and Isabel Blasco-Costa. Paco: a novel procrustes application to cophylogenetic analysis. *PloS one*, 8(4):e61048, 2013.
- [3] Zhipeng Bao, Yijun Li, Krishna Kumar Singh, Yu-Xiong Wang, and Martial Hebert. Separate-and-enhance: Compositional finetuning for text-to-image diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–10, 2024.
- [4] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 32, 2019.
- [5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.
- [6] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–10, 2023.
- [7] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1971–1978, 2014.
- [8] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. Boundary iou: Improving object-centric image segmentation evaluation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15334–15342, 2021.
- [9] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in neural information processing systems*, 34: 17864–17875, 2021.
- [10] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022.
- [11] Seokju Cho, Heeseong Shin, Sunghwan Hong, Seungjun An, Seungjun Lee, Anurag Arnab, Paul Hongsuck Seo, and Seungryong Kim. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. *arXiv preprint arXiv:2303.11797*, 2023.
- [12] Subhabrata Choudhury, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Unsupervised part discovery from contrastive reconstruction. *Advances in Neural Information Processing Systems*, 34:28104–28118, 2021.
- [13] Daan de Geus, Panagiotis Meletis, Chenyang Lu, Xiaoxiao Wen, and Gijs Dubbelman. Part-aware panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5485–5494, 2021.

- [14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [15] 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.
- [16] Zheng Ding, Jieke Wang, and Zhuowen Tu. Open-vocabulary universal image segmentation with maskclip. *arXiv preprint arXiv:2208.08984*, 2022.
- [17] Vincent Dumoulin, Ethan Perez, Nathan Schucher, Florian Strub, Harm de Vries, Aaron Courville, and Yoshua Bengio. Feature-wise transformations. *Distill*, 3(7):e11, 2018.
- [18] 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.
- [19] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021.
- [20] Kunyang Han, Yong Liu, Jun Hao Liew, Henghui Ding, Jiajun Liu, Yitong Wang, Yansong Tang, Yujiu Yang, Jiashi Feng, Yao Zhao, et al. Global knowledge calibration for fast open-vocabulary segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 797–807, 2023.
- [21] Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaoding Yuan, Jie-Neng Chen, Shuai Liu, Cheng Yang, Qihang Yu, and Alan Yuille. Partimagenet: A large, high-quality dataset of parts. In *European Conference on Computer Vision*, pages 128–145. Springer, 2022.
- [22] Ju He, Jieneng Chen, Ming-Xian Lin, Qihang Yu, and Alan L Yuille. Compositor: Bottom-up clustering and compositing for robust part and object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11259–11268, 2023.
- [23] Wei-Chih Hung, Varun Jampani, Sifei Liu, Pavlo Molchanov, Ming-Hsuan Yang, and Jan Kautz. Scops: Self-supervised co-part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 869–878, 2019.
- [24] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.
- [25] Yunji Kim, Jiyoung Lee, Jin-Hwa Kim, Jung-Woo Ha, and Jun-Yan Zhu. Dense text-to-image generation with attention modulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7701–7711, 2023.
- [26] 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.
- [27] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. *arXiv preprint arXiv:2201.03546*, 2022.
- [28] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. *arXiv preprint arXiv:2307.04767*, 2023.
- [29] 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.

- [30] Yong Liu, Sule Bai, Guanbin Li, Yitong Wang, and Yansong Tang. Open-vocabulary segmentation with semantic-assisted calibration. *arXiv preprint arXiv:2312.04089*, 2023.
- [31] Zongdai Liu, Feixiang Lu, Peng Wang, Hui Miao, Liangjun Zhang, Ruigang Yang, and Bin Zhou. 3d part guided image editing for fine-grained object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11336–11345, 2020.
- [32] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7086–7096, 2022.
- [33] Umberto Michieli and Pietro Zanuttigh. Edge-aware graph matching network for part-based semantic segmentation. *International Journal of Computer Vision*, 130(11):2797–2821, 2022.
- [34] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11): 39–41, 1995.
- [35] Tai-Yu Pan, Qing Liu, Wei-Lun Chao, and Brian Price. Towards open-world segmentation of parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15392–15401, 2023.
- [36] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.
- [37] Quynh Phung, Songwei Ge, and Jia-Bin Huang. Grounded text-to-image synthesis with attention refocusing. *arXiv preprint arXiv:2306.05427*, 2023.
- [38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [39] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.
- [40] Peize Sun, Shoufa Chen, Chenchen Zhu, Fanyi Xiao, Ping Luo, Saining Xie, and Zhicheng Yan. Going denser with open-vocabulary part segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15453–15465, 2023.
- [41] Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar Gonzalez-Franco. Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion. *arXiv preprint arXiv:2308.12469*, 2023.
- [42] Mehmet Ozgur Turkoglu, Alexander Becker, Hüseyin Anil Gündüz, Mina Rezaei, Bernd Bischl, Rodrigo Caye Daudt, Stefano D’Aronco, Jan Wegner, and Konrad Schindler. Film-ensemble: probabilistic deep learning via feature-wise linear modulation. *Advances in neural information processing systems*, 35:22229–22242, 2022.
- [43] Robert van der Klis, Stephan Alaniz, Massimiliano Mancini, Cassio F Dantas, Dino Ienco, Zeynep Akata, and Diego Marcos. Pdisconet: Semantically consistent part discovery for fine-grained recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1866–1876, 2023.
- [44] Zifu Wan, Yaqi Xie, Ce Zhang, Zhiqiu Lin, Zihan Wang, Simon Stepputtis, Deva Ramanan, and Katia P Sycara. Instructpart: Affordance-based part segmentation from language instruction. In *AAAI-2024 Workshop on Public Sector LLMs: Algorithmic and Sociotechnical Design*, 2024.
- [45] Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. Instancediffusion: Instance-level control for image generation. *arXiv preprint arXiv:2402.03290*, 2024.

- [46] Meng Wei, Xiaoyu Yue, Wenwei Zhang, Shu Kong, Xihui Liu, and Jiangmiao Pang. Ov-parts: Towards open-vocabulary part segmentation. *Advances in Neural Information Processing Systems*, 36, 2024.
- [47] Weijia Wu, Yuzhong Zhao, Mike Zheng Shou, Hong Zhou, and Chunhua Shen. Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1206–1217, 2023.
- [48] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8256–8265, 2019.
- [49] Bin Xie, Jiale Cao, Jin Xie, Fahad Shahbaz Khan, and Yanwei Pang. Sed: A simple encoder-decoder for open-vocabulary semantic segmentation. *arXiv preprint arXiv:2311.15537*, 2023.
- [50] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7452–7461, 2023.
- [51] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2955–2966, 2023.
- [52] 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.
- [53] 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.
- [54] Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. *Advances in Neural Information Processing Systems*, 36, 2024.
- [55] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14393–14402, 2021.
- [56] Hang Zhao, Xavier Puig, Bolei Zhou, Sanja Fidler, and Antonio Torralba. Open vocabulary scene parsing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2002–2010, 2017.
- [57] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.
- [58] 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.
- [59] Ziqin Zhou, Yinjie Lei, Bowen Zhang, Lingqiao Liu, and Yifan Liu. Zegclip: Towards adapting clip for zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11175–11185, 2023.

Supplementary Material

Table of Contents

• Discussion	Appendix A
– Limitations & Future Work	Appendix A.1
– Social Impact	Appendix A.2
• Experimental Details	Appendix B
– Datasets Details	Appendix B.1
– Implementation Details	Appendix B.2
– Computational Resource	Appendix B.3
• Additional Quantitative Evaluation	Appendix C
• Additional Ablation	Appendix D
– Impact of Object-Level and Part-Level Guidance	Appendix D.1
– Qualitative Ablation on Attention Control Losses	Appendix D.2
– Ablation on the Hyperparameter in Attention Control	Appendix D.3
• Additional Qualitative Results and Qualitative Analysis	Appendix E
– CLIP Embedding	Appendix E.1
– Additional Qualitative Results	Appendix E.2

A Discussion

A.1 Limitations & Future Work

We share some limitations of our model and outline directions for future research. Our model is based on semantic segmentation, which does not allow for the discrimination of individual parts as instances. Consequently, parts such as “Paw 1” from *Dog 1* and “Paw 2” from *Dog 2* are assigned the same label. We plan to address this limitation in our future work to enhance the model’s capability to distinguish between similar parts from different instances.

Furthermore, we believe that adding more inductive biases related to the relationships between parts, similar to key point detection which incorporates structural understanding, could yield higher-quality results.

Currently, our focus has been on object-specific parts, essentially mapping different granularity of vocabulary visually. Advanced methods could allow us to more effectively handle a broader variety of input categories, further enhancing our model’s applicability and performance.

A.2 Social Impact

This study explores open-vocabulary part segmentation, a technique that expands segmentation models to include fine-grained categories not encountered during training. The approach’s robust nature allows for segmentation across various categories, proving invaluable for applications requiring flexibility and adaptability.

Open-vocabulary part segmentation could greatly influence several advanced fields. In robotics, for example, robots can precisely identify and handle a wide array of objects and components, essential for tasks from manufacturing assembly lines to complex medical surgeries. This adaptability allows robots to function in new settings without extensive retraining.

In healthcare, this technology enhances diagnostic processes by allowing for the segmentation of novel anatomical structures in medical imaging. This could facilitate earlier disease detection by identifying subtle, non-cataloged abnormalities essential for diagnosis.

In image editing, open-vocabulary part segmentation enables sophisticated manipulation by letting editors modify image fine-grained components not predefined in their software. This is especially

beneficial in the creative industries, where precise adjustments can improve output quality and foster innovation.

Adopting open-vocabulary part segmentation promises to enhance the efficiency, accessibility, and effectiveness of these technologies, particularly in handling real-world variability and unpredictability.

B Experimental Details

B.1 Datasets Details

B.1.1 Pascal-Part-116

In the Pascal-Part-116 dataset [7, 46], we target the following object-specific category names in Table A1. Among these, “bird”, “car”, “dog”, “sheep”, and “motorbike” are designated as unseen categories, encountered for the first time during inference in the zero-shot part segmentation setting.

Table A1: List of object-specific classes in Pascal-Part-116.

Object-specific Part Categories				
aeroplane’s body	aeroplane’s stern	aeroplane’s wing	aeroplane’s tail	aeroplane’s engine
aeroplane’s wheel	bicycle’s wheel	bicycle’s saddle	bicycle’s handlebar	bicycle’s chainwheel
bicycle’s headlight	bird’s wing	bird’s tail	bird’s head	bird’s eye
bird’s beak	bird’s torso	bird’s neck	bird’s leg	bird’s foot
bottle’s body	bottle’s cap	bus’s wheel	bus’s headlight	bus’s front
bus’s side	bus’s back	bus’s roof	bus’s mirror	bus’s license plate
bus’s door	bus’s window	car’s wheel	car’s headlight	car’s front
car’s side	car’s back	car’s roof	car’s mirror	car’s license plate
car’s door	car’s window	cat’s tail	cat’s head	cat’s eye
cat’s torso	cat’s neck	cat’s leg	cat’s nose	cat’s paw
cat’s ear	cow’s tail	cow’s head	cow’s eye	cow’s torso
cow’s neck	cow’s leg	cow’s ear	cow’s muzzle	cow’s horn
dog’s tail	dog’s head	dog’s eye	dog’s torso	dog’s neck
dog’s leg	dog’s nose	dog’s paw	dog’s ear	dog’s muzzle
horse’s tail	horse’s head	horse’s eye	horse’s torso	horse’s neck
horse’s leg	horse’s ear	horse’s muzzle	horse’s hoof	motorbike’s wheel
motorbike’s saddle	motorbike’s handlebar	motorbike’s headlight	person’s head	person’s eye
person’s torso	person’s neck	person’s leg	person’s foot	person’s nose
person’s ear	person’s eyebrow	person’s mouth	person’s hair	person’s lower arm
person’s upper arm	person’s hand	pottedplant’s pot	pottedplant’s plant	sheep’s tail
sheep’s head	sheep’s eye	sheep’s torso	sheep’s neck	sheep’s leg
sheep’s ear	sheep’s muzzle	sheep’s horn	train’s headlight	train’s head
train’s front	train’s side	train’s back	train’s roof	train’s coach
tvmonitor’s screen				

B.1.2 ADE20K-Part-234

In the ADE20K-Part-234 dataset [57], we target specific object categories listed in Table A2. The dataset includes 44 object classes and detailed subdivisions into over 200 part categories. Notably, “bench”, “bus”, “fan”, “desk”, “stool”, “truck”, “van”, “swivel chair”, “oven”, “ottoman”, and “kitchen island” are identified as novel classes and are encountered for the first time during inference in our zero-shot part segmentation setting.

B.1.3 PartImageNet

PartImageNet [21] is a dataset derived from ImageNet [14], consisting of approximately 24,000 images across 158 classes. Each class has annotations for parts. All classes belong to one of 11 superclasses, organized using the hierarchical information from WordNet [34].

Previous open-vocabulary part segmentation research [40] primarily used PartImageNet to evaluate cross-dataset settings. In our study, we use PartImageNet not only for cross-dataset evaluation but also to assess model performance in zero-shot settings specific to PartImageNet.

To measure more generalized performance, we select 40 classes out of the 158. We maintain the proportion of existing superclasses as much as possible. For each superclass, at least 50% of the categories are designated as seen categories, with the remaining being unseen categories. Therefore, there are 25 seen classes and 15 unseen classes in our PartImageNet evaluation dataset.

Table A2: List of object-specific classes in ADE20K-Part-234.

Object-specific Part Categories				
person's arm	person's back	person's foot	person's gaze	person's hand
person's head	person's leg	person's neck	person's torso	door's door frame
door's handle	door's knob	door's panel	clock's face	clock's frame
toilet's bowl	toilet's cistern	toilet's lid	cabinet's door	cabinet's drawer
cabinet's front	cabinet's shelf	cabinet's side	cabinet's skirt	cabinet's top
sink's bowl	sink's faucet	sink's pedestal	sink's tap	sink's top
lamp's arm	lamp's base	lamp's canopy	lamp's column	lamp's cord
lamp's highlight	lamp's light source	lamp's shade	lamp's tube	sconce's arm
sconce's backplate	sconce's highlight	sconce's light source	sconce's shade	chair's apron
chair's arm	chair's back	chair's base	chair's leg	chair's seat
chair's seat cushion	chair's skirt	chair's stretcher	chest of drawers's apron	chest of drawers's door
chest of drawers's drawer	chest of drawers's front	chest of drawers's leg	chandelier's arm	chandelier's bulb
chandelier's canopy	chandelier's chain	chandelier's cord	chandelier's highlight	chandelier's light source
chandelier's shade	bed's footboard	bed's headboard	bed's leg	bed's side rail
table's apron	table's drawer	table's leg	table's shelf	table's top
table's wheel	armchair's apron	armchair's arm	armchair's back	armchair's back pillow
armchair's leg	armchair's seat	armchair's seat base	armchair's seat cushion	ottoman's back
ottoman's leg	ottoman's seat	shelf's door	shelf's drawer	shelf's front
shelf's shelf	swivel chair's back	swivel chair's base	swivel chair's seat	swivel chair's wheel
fan's blade	fan's canopy	fan's tube	coffee table's leg	coffee table's top
stool's leg	stool's seat	sofa's arm	sofa's back	sofa's back pillow
sofa's leg	sofa's seat base	sofa's seat cushion	sofa's skirt	computer's computer case
computer's keyboard	computer's monitor	computer's mouse	desk's apron	desk's door
desk's drawer	desk's leg	desk's shelf	desk's top	wardrobe's door
wardrobe's drawer	wardrobe's front	wardrobe's leg	wardrobe's mirror	wardrobe's top
car's bumper	car's door	car's headlight	car's hood	car's license plate
car's logo	car's mirror	car's wheel	car's window	car's wiper
bus's bumper	bus's door	bus's headlight	bus's license plate	bus's logo
bus's mirror	bus's wheel	bus's window	bus's wiper	oven's button panel
oven's door	oven's drawer	oven's top	cooking stove's burner	cooking stove's button panel
cooking stove's door	cooking stove's drawer	cooking stove's oven	cooking stove's stove	microwave's button panel
microwave's door	microwave's front	microwave's side	microwave's top	microwave's window
refrigerator's button panel	refrigerator's door	refrigerator's drawer	refrigerator's side	kitchen island's door
kitchen island's drawer	kitchen island's front	kitchen island's side	kitchen island's top	dishwasher's button panel
dishwasher's handle	dishwasher's skirt	bookcase's door	bookcase's drawer	bookcase's front
bookcase's side	television receiver's base	television receiver's buttons	television receiver's frame	television receiver's keys
television receiver's screen	television receiver's speaker	glass's base	glass's bowl	glass's opening
glass's stem	pool table's bed	pool table's leg	pool table's pocket	van's bumper
van's door	van's headlight	van's license plate	van's logo	van's mirror
van's taillight	van's wheel	van's window	van's wiper	airplane's door
airplane's fuselage	airplane's landing gear	airplane's propeller	airplane's stabilizer	airplane's turbine engine
airplane's wing	truck's bumper	truck's door	truck's headlight	truck's license plate
truck's logo	truck's mirror	truck's wheel	truck's window	minibike's license plate
minibike's mirror	minibike's seat	minibike's wheel	washer's button panel	washer's door
washer's front	washer's side	bench's arm	bench's back	bench's leg
bench's seat	traffic light's housing	traffic light's pole	light's aperture	light's canopy
light's diffusor	light's highlight	light's light source	light's shade	

We conduct the dataset evaluation as follows: Models are trained on a training dataset composed of seen classes. Segmentation performance are then assessed on a validation dataset containing both seen and unseen classes. Evaluations were conducted in both Pred-All and Oracle-Obj settings.

Table A3: List of selected object classes per superclass. We choose 40 object classes from 158 categories to evaluate performance on PartImageNet and in a cross-dataset setting. Object categories that are both underlined and in **bold** represent the unseen classes, which are emphasized for their unique characteristics within each superclass.

Superclass	Object Categories
Quadruped	tiger, giant panda, leopard, gazelle, <u>ice bear</u> , <u>impala</u> , <u>golden retriever</u>
Snake	green mamba, Indian cobra
Reptile	green lizard, Komodo dragon, tree frog, <u>box turtle</u> , <u>American alligator</u>
Boat	yawl, pirate, <u>schooner</u>
Fish	barracouta, goldfish, killer whale, <u>tench</u>
Bird	albatross, goose, bald eagle
Car	garbage truck, minibus, ambulance, <u>jeep</u> , <u>school bus</u>
Bicycle	mountain bike, moped, <u>motor scooter</u>
Biped	gorilla, orangutan, <u>chimpanzee</u>
Bottle	beer bottle, water bottle, <u>wine bottle</u>
Aeroplane	warplane, <u>airliner</u>

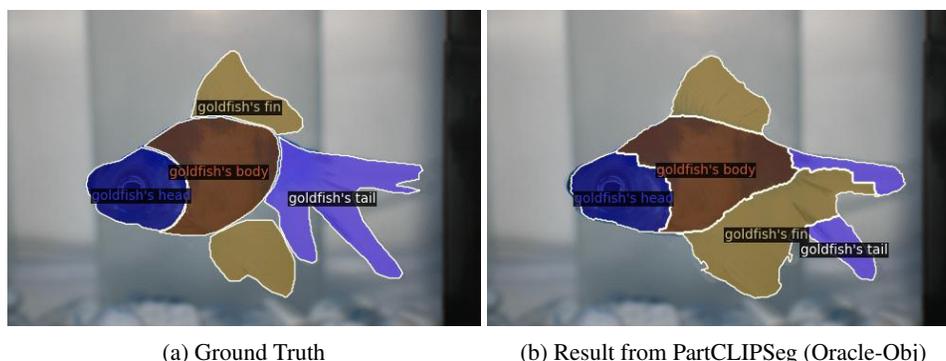


Figure A1: Example of part annotations in PartImageNet on our experiment.

B.2 Implementation Details

Our model implementation is based on the CLIPSeg [32] architecture, as described in the OV-PARTS [46]. We utilized the pre-trained CLIP ViT-B/16 [38, 59] image encoder and text encoder for our experiments.

The model is trained using the ADAMW optimizer with a base learning rate of 0.0001 over 20,000 iterations, with a batch size of 8 images. We employ a WarmupPolyLR learning rate scheduler to manage the learning rate throughout the training process. To ensure model stability, we apply gradient clipping with a maximum gradient norm of 0.01.

We save model parameters every 1,000 iterations during training. The best-performing parameters are selected based on the highest validation evaluation scores. For example, the evaluation result on the Pascal-Part-116 dataset in the Oracle-Obj setting is derived from the checkpoint saved at the 5,000-step mark, which yields the best validation performance.

We evaluated several baseline methods—ZSSeg+, CLIPSeg [32], and CAT-Seg [11]—which are fine-tuned on our datasets. ZSSeg+ is a modified version of ZSseg [52], utilizing different fine-tuning methods according to [46]. It employs a ResNet-101 backbone and Compositional Prompt Tuning based on CoOp.

CLIPSeg and CAT-Seg models are pre-trained on object datasets; however, we fine-tuned these models on each part-level dataset. CAT-Seg, based on ResNet-101 and using ViT-B/16 as CLIP’s visual encoder, achieved comparable performance by computing cost volumes and subsequently applying cost aggregation—a process that enhances segmentation by aggregating matching costs between image features. Specifically, CAT-Seg uses the frozen upsampling decoder but fine-tuned CLIP’s image and text encoders. Conversely, we fine-tune the CLIPSeg decoder to better identify small segments and define clear boundaries. CLIPSeg, based on the ViT-B/16 architecture, is fine-tuned on the visual adapter, text embeddings, and transformer decoder to enhance its segmentation capabilities.

B.3 Computational Resource

All our experiments are conducted on $8 \times$ NVIDIA A6000 GPUs.

As shown in Table A4, PartCLIPSeg offers advantages in both the number of parameters and memory consumption compared to other baselines on the Pascal-Part-116 dataset. With 152.4 million parameters, it is more efficient than ZSSeg+ and CAT-Seg, and comparable to CLIPSeg. In terms of GPU memory usage, PartCLIPSeg requires 24.4 GB, which is lower than both CAT-Seg and CLIPSeg.

For PartCLIPSeg, although the number of parameters is larger than CLIPSeg because of computations related to attention control, there is an advantage in not having to maintain weights for each object-

Table A4: Computational resources on Pascal-Part-116 with batch size 8.

Method	Params	Memory
ZSSeg+ [52]	191.6 M	11.1 G
CLIPSeg [32, 46]	151.7 M	25.5 G
CAT-Seg [11]	180.6 M	29.0 G
PartCLIPSeg	152.4 M	24.4 G

specific part due to the use of generalized parts. These efficiencies become more pronounced as the number of generalized parts shared among object classes increases. By leveraging shared representations for generalized parts, PartCLIPSeg reduces redundancy and memory requirements. This makes our model particularly advantageous in datasets where object classes have many common parts, leading to more efficient training and inference without compromising performance.

C Additional Quantitative Evaluation

Table A5: Recall performance on Pascal-Part-116 under the Oracle-Obj setting.

Method	Seen	Unseen	Harmonic
ZSSeg+ [52]	65.47	32.13	43.10
CLIPSeg [32, 46]	55.71	43.35	48.76
CAT-Seg [11]	56.00	43.20	48.77
PartCLIPSeg (w/o $\mathcal{L}_{sep} + \mathcal{L}_{enh}$)	58.97	46.47	51.98
PartCLIPSeg (w/ $\mathcal{L}_{sep} + \mathcal{L}_{enh}$)	58.46	47.93	52.67

Table A6: Recall performance on ADE20K-Part-234 under the Oracle-Obj setting.

Method	Seen	Unseen	Harmonic
ZSSeg+ [52]	55.78	40.71	47.07
CLIPSeg [32, 46]	49.59	48.11	48.84
CAT-Seg [11]	43.48	39.87	41.60
PartCLIPSeg (w/o $\mathcal{L}_{sep} + \mathcal{L}_{enh}$)	51.64	50.99	51.31
PartCLIPSeg (w/ $\mathcal{L}_{sep} + \mathcal{L}_{enh}$)	53.31	51.52	52.40

In this section, we present an additional evaluation metric that focuses on specific challenges within the Open-Vocabulary Part Segmentation (OVPS) task as shown in Figure 2. The Recall metric is used to assess how well the model captures underrepresented parts, addressing the challenge of underrepresented parts. Higher values in recall indicate that the model effectively captures these seldom-occurring parts, thereby addressing the challenge of underrepresented parts in OVPS.

PartCLIPSeg consistently achieves higher recall on both seen and unseen classes across both datasets as shown in Tables A5 and A6. The improved harmonic mean indicates that our model is more effective at identifying underrepresented parts, thereby addressing one of the core challenges in OVPS.

We further analyze the impact of the attention control losses \mathcal{L}_{sep} and \mathcal{L}_{enh} on the recall. By comparing the recall metric with and without these losses, we assess their effectiveness in enhancing the representation of seldom-occurring parts. From Tables A5 and A6, we observe that incorporating the attention control losses enhances the model’s performance on unseen classes, which often include underrepresented parts. The increases in harmonic mean suggest that the attention control losses help the model to better capture these seldom-occurring or small parts.

D Additional Ablation

D.1 Impact of Object-Level and Part-Level Guidance

Table A7: Ablation on λ_{obj} , λ_{part} , and attention control on Pascal-Part-116 in Oracle-Obj setting.

λ_{obj}	λ_{part}	$\mathcal{L}_{sep} + \mathcal{L}_{enh}$	Seen	Unseen	Harmonic mIoU
0.0	0.0	✓	48.36	29.42	36.58
1.0	0.0	✓	48.61	31.28	38.07
0.0	1.0	✓	48.94	31.68	38.46
1.0	1.0	✗	49.09	31.26	38.20
1.0	1.0	✓	50.02	31.67	38.79

We conduct additional experiments to verify the impact of object-level and part-level label guidance on model performance as shown in Table A7. Specifically, we vary the weights λ_{obj} and λ_{part} in Equation (5), setting each to 0 or 1, to assess the influence of object-level and part-level supervision on the overall performance. Additionally, we evaluate the effect of the attention control losses, \mathcal{L}_{sep} and \mathcal{L}_{enh} , by including or excluding them.

As shown in Table A7, both object-level and part-level guidance positively impact model performance on the Pascal-Part-116 dataset under the Oracle-Obj setting. When neither object-level nor part-level supervision is applied, the harmonic mean is 36.58. Introducing object-level guidance alone increases the harmonic mean IoU to 38.07, while part-level guidance alone raises it to 38.46. Combining both guidances yields the best performance with a harmonic mean IoU of 38.79.



Figure A2: Comparison of results using only \mathcal{L}_{sep} (top) with both \mathcal{L}_{sep} and \mathcal{L}_{enh} (bottom). The heatmap illustrates attention activation for the “sheep’s neck” class.

Table A8: Effect of varying threshold γ on Pascal-Part-116 in Oracle-Obj setting.

Threshold (γ)	Seen	Unseen	Harmonic mIoU
0.1	47.34	32.24	38.35
0.2	47.45	32.20	38.37
0.3	50.02	31.67	38.79
0.4	51.10	31.18	38.73
0.5	48.71	31.16	38.01

Additionally, removing the attention control losses \mathcal{L}_{sep} and \mathcal{L}_{enh} while keeping both λ_{obj} and λ_{part} at 1.0 results in a lower Harmonic mean of 38.20. This indicates that the attention control losses contribute to better distinguishing between seen and unseen classes.

D.2 Qualitative Ablation on Attention Control Losses

The separation loss reduces the overlap between different parts, while the enhancement loss strengthens the activation of underrepresented parts. As shown in Figure A2, when only the separation loss \mathcal{L}_{sep} is applied (top), smaller parts adjacent to larger parts may be diminished. Specifically, “sheep’s neck” is not properly highlighted because minimizing the intersection can cause larger parts, such as the “sheep’s torso” and “sheep’s head”, to overshadow smaller ones. When both losses \mathcal{L}_{sep} and \mathcal{L}_{enh} are utilized (bottom), the model accurately segments the small part—“sheep’s neck”—as the enhancement loss boosts its representation, preventing it from being overwhelmed by larger neighboring parts.

This demonstrates that the separation and enhancement losses complement each other. Their combined use is essential to effectively distinguish and represent both large and small parts within an object, leading to improved segmentation performance.

D.3 Ablation on the Hyperparameter in Attention Control

To evaluate the sensitivity of our method to the hyperparameter threshold γ in Equation (8), we conducted experiments on the Pascal-Part-116 dataset under the Oracle-Obj setting. We varied γ from 0.1 to 0.5 and measured the performance in terms of mIoU for seen and unseen classes, as well as the harmonic mean.

As shown in Table A8, our method is robust to the choice of γ within the range of 0.1 to 0.5. The harmonic mean remains relatively stable, with the best performance achieved at $\gamma = 0.3$. While there is a slight variation in performance across different values of γ , the changes are not significant, indicating that our method does not heavily depend on the exact value of this hyperparameter.

E.2 Additional Qualitative Results

E.2.1 Oracle-Obj Setting

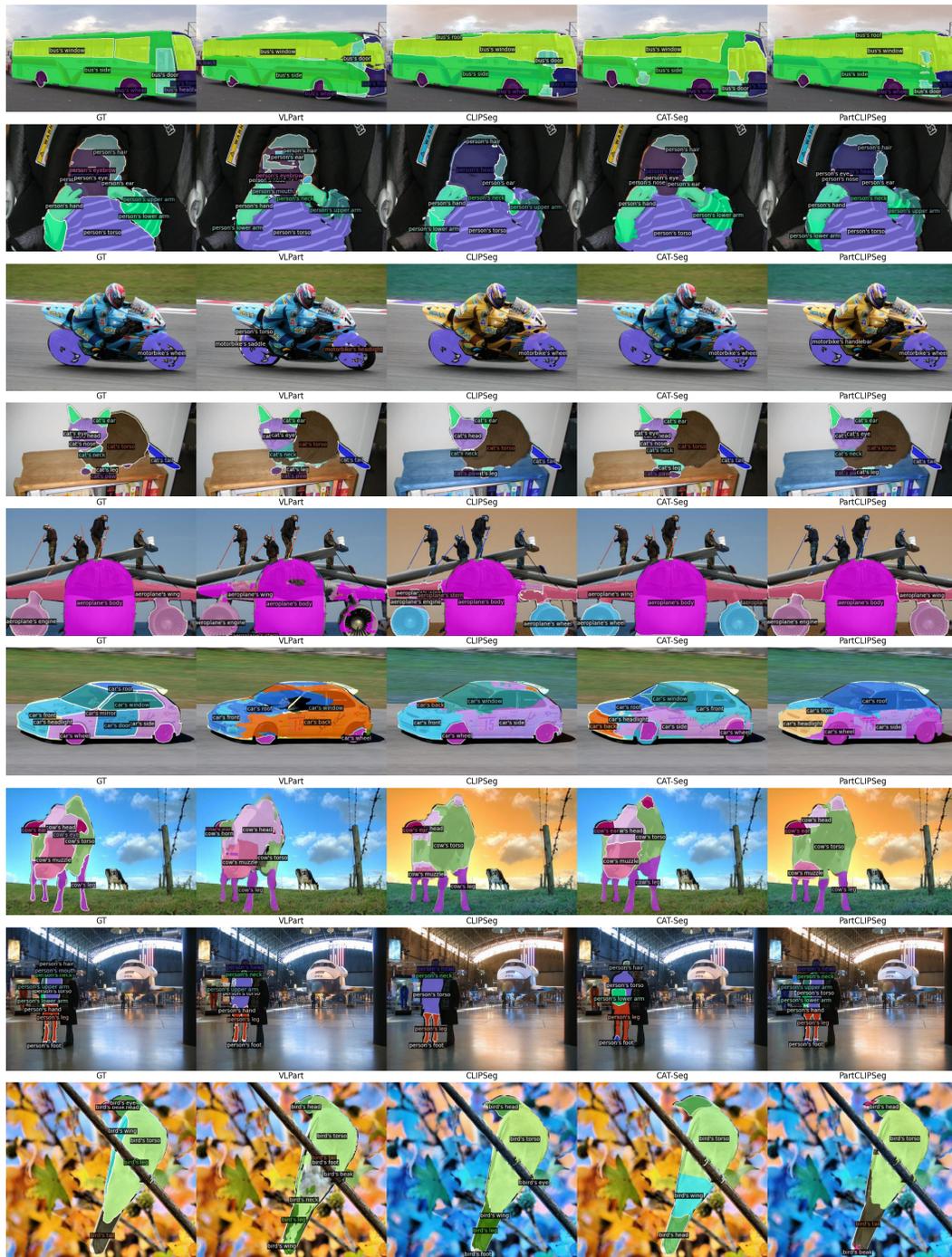


Figure A4: Comparison of VLPpart, CLIPSeg, CAT-Seg, and our model on the Pascal-Part-116 dataset in Oracle-Obj setting.

E.2.2 Pred-All Setting

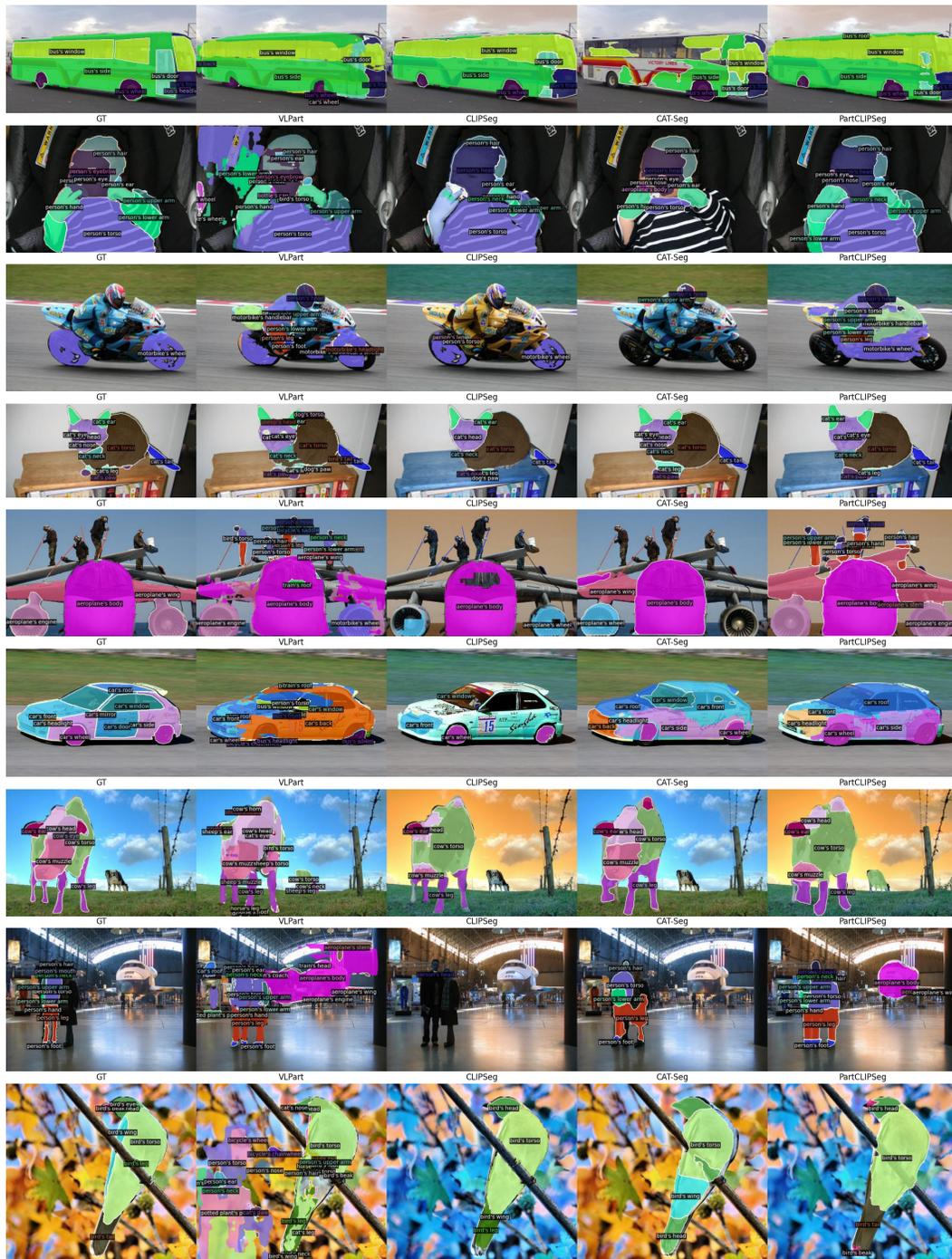


Figure A5: Comparison of VLPpart, CLIPSeg, CAT-Seg, and our model on the Pascal-Part-116 dataset in Pred-All setting.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: Yes, the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope in Section 1.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The limitations & future works are included at supplementary materials.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The anonymized repository of our implementation and instructions of reproduction are provided. (in the abstract section)

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The anonymized repository of our implementation and instructions of reproduction are provided. (in the abstract section)

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The experimental setting and details are provided in section 4 and supplementary materials. Also, the anonymized repository of our implementation and reproduction instructions are provided. (URL is in the abstract section)

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The standard error of an average of 5 results is reported in Section 4 of the proposed model.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The experimental resources, setting and details are provided in section 4 and supplementary materials. Also, the anonymized repository of our implementation and reproduction instructions are provided. (URL is in the abstract section)

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: -

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The broader of impact is discussed in supplementary materials.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We properly credited previous works and codes in section 4.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The anonymized repository of our implementation and reproduction instructions are provided. (URL is in the abstract section)

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: [NA]

Guidelines: [NA]

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.