HOPS: Hierarchical Open-vocabulary Part Segmentation with Attention-Aware Filtering and Affinity-Guided Enhancement

Supplementary Material

A. Limitations & Future Work

While HOPS effectively mitigates object over-segmentation and part under-segmentation in OVPS, several limitations remain due to its hierarchical design and task setup. First, the part segmentation relies heavily on the object mask produced in the first stage as a spatial prior. This dependency introduces a risk of error propagation: inaccuracies in object segmentation directly impair part-level predictions, compromising overall performance. Second, HOPS treats parts with the same name but from different object classes (*e.g.* "cat tail" and "dog tail") as entirely distinct categories, overlooking the semantic and morphological consistency shared by cross-object parts (*e.g.* "tail" of different animals exhibit similar structures). This limits the learning of universal part representations and constrains zero-shot generalization. Third, HOPS does not distinguish between multiple instances of the same category within a scene. As a result, parts from different instances are merged into a single mask, limiting the applicability of the method to instance-aware tasks such as robotic manipulation.

To address these limitations, future work can explore several directions: (1) developing a robust joint object–part learning mechanism with structural or topological constraints to reduce error propagation across stages; (2) designing a universal part representation learning framework that captures common visual and semantic patterns of parts across objects; (3) extending the model to support instance-level part segmentation, potentially through instance query mechanisms or discriminative instance feature learning.

B. Method Details

Cost Computation and Embedding. Given an input image I and a category set $\mathcal{C}$, we first use CLIP [36] to extract dense visual features $\mathbf{F}_i \in \mathbb{R}^{N\times d}$, where N is the total number of image patches and d denotes the feature dimension. Meanwhile, category-specific text features $\mathbf{F}_t \in \mathbb{R}^{T\times|\mathcal{C}|\times d}$ are obtained by encoding multiple text prompt templates, where T represents the number of templates. The cosine similarity between the image and text features is then computed to construct the cost map $\mathbf{S} \in \mathbb{R}^{N\times T\times|\mathcal{C}|}$:

$$\mathbf{S}(n,t,c) = \frac{\mathbf{F}_i(n,:)\cdot\mathbf{F}_t(t,c,:)}{\|\mathbf{F}_i(n,:)\|\cdot\|\mathbf{F}_t(t,c,:)\|}. \tag{10}$$

To capture higher-level semantics and facilitate subsequent cost aggregation, we apply a single-layer convolution independently to each cost slice, obtaining the cost embedding $\mathbf{F}_s \in \mathbb{R}^{N\times|\mathcal{C}|\times d_s}$, where d_s denotes the embedding dimension. $\mathbf{F}_s$ is then fed into the cost aggregation module to improve semantic consistency and category discrimination.

Self-Attention Extraction. During image feature extraction with CLIP ViT-B/16 and DINOv2 ViT-S/14 [34], we also obtain the self-attention matrices from all transformer layers of both models. Both CLIP ViT-B/16 and DINOv2 ViT-S/14 adopt a Vision Transformer architecture comprising 12 multi-head self-attention (MHSA) layers. We average the attention heads in each layer to obtain 12 layer-wise self-attention matrices per model. During bidirectional attention fusion in AFM and AEM, we fuse the layer-wise attention matrices from CLIP and DINO, producing a joint semantic–structural attention representation $A \in \mathbb{R}^{L\times N\times N}$, where $L = 12$. To address the mismatch in spatial resolution between the two types of attention matrices, we downsample the DINO attention matrices before fusion to ensure spatial alignment with CLIP.

Cost Aggregation. Cost aggregation [6] enhances the initial cost embedding $\mathbf{F}_s$ by capturing spatial continuity and inter-class relationships, thereby improving semantic discrimination and fine-grained perception. The process consists of two stages: spatial aggregation and class aggregation. *Spatial aggregation* leverages both local and global spatial correlations of the image to improve intra-category consistency while suppressing background noise and scattered activations. For each category c, the feature slice $\mathbf{F}_s[:,c,:] \in \mathbb{R}^{N\times d_s}$ is passed through the spatial aggregation module $\mathcal{T}_{SA}$, which consists of two consecutive Swin Transformer blocks [31]:

$$\mathbf{F}'_s[:,c,:] = \mathcal{T}_{SA}\big(\mathbf{F}_s[:,c,:]\big). \tag{11}$$

The first block captures patch-wise dependencies within local windows, while the second block expands the receptive field via window shifting. The combination of local and shifted-window attention enables the model to effectively aggregate spatial information across scales.

Class aggregation models semantic correlations among categories to reduce confusion in open-vocabulary settings and maintain robustness to category ordering. For each spatial position n, the feature slice $\mathbf{F}'_s[n,:,:] \in \mathbb{R}^{|\mathcal{C}|\times d_s}$ is processed by a linear Transformer block [22] $\mathcal{T}_{CA}$ without positional encoding, to suppress interference from semantically similar categories:

$$\mathbf{F}''_s[n,:,:] = \mathcal{T}_{CA}\big(\mathbf{F}'_s[n,:,:]\big). \tag{12}$$

Structural Guidance. Following PartCATSeg [8], we incorporate DINO features as structural guidance during spa-

tial aggregation. Specifically, when constructing the Query and Key matrices, we concatenate the cost embedding with the DINO features along the feature dimension:

$$\mathbf{F}'_s[:,c,:] = \mathcal{T}_{SA}([\mathbf{F}_s[:,c,:]; \mathbf{F}_{\text{dino}}[:,c,:]]), \quad (13)$$

where $\mathbf{F}_{\text{dino}} \in \mathbb{R}^{N\times|\mathcal{C}|\times d_{\text{dino}}}$ denotes the DINO features after linear projection to align with the cost embeddings, and $[\cdot;\cdot]$ indicates concatenation. In this way, spatial aggregation jointly leverages semantic information from the cost embeddings and structural priors (*e.g.* geometric structures, contour boundaries) from DINO features. Consequently, the boundary precision and structural consistency of part segmentation are significantly improved.

Upsampling Decoder. The decoder adopts a hierarchical upsampling scheme similar to U-Net [38], which progressively upsamples the aggregated cost embedding $\mathbf{F}''_s$ and integrates multi-scale features from the CLIP image encoder to recover spatial detail and enhance segmentation precision. Specifically, intermediate features are extracted from the 4th and 8th layers of the CLIP ViT-B/16 encoder, which retain rich structural cues such as part boundaries and local textures. These features are upsampled via learnable transposed convolutions to spatial resolutions of 96×96 and 48×48, denoted as $\mathbf{F}_{i,4}$ and $\mathbf{F}_{i,8}$, respectively. Subsequently, $\mathbf{F}''_s$ undergoes multi-round upsampling and feature fusion. First, $\mathbf{F}''_s$ is upsampled to 48×48, concatenated with $\mathbf{F}_{i,8}$, and fused using a 3×3 convolution. The result is further upsampled to 96×96, concatenated with $\mathbf{F}_{i,4}$, and fused again via a convolution layer. Each fusion is followed by Group Normalization [50] and ReLU activation. Finally, the fused representation is passed through a prediction head consisting of a 1×1 convolution and a sigmoid activation, producing the logit scores for each category.

C. Experimental Details

C.1. Code & Reproduction

For additional implementation details, please refer to the public codebase https://github.com/TJU-IDVLab/HOPS.

C.2. Device Information

All experiments were executed on a server equipped with 8 NVIDIA A6000 GPUs. Both training and inference were performed using the same hardware configuration.

C.3. Hyperparameters

All experiments adopt the following default hyperparameters. For the Attention-Aware Filtering Module (AFM), the semantic-structural attention fusion weight $\alpha = 0.7$, and the cross-layer consistency threshold $y = 4$. For the Affinity-Guided Enhancement Module (AEM), the initial segmentation threshold $\gamma = 0.45$ and the small-part compensation threshold $\sigma = 0.1$. For the loss function, the object segmentation weight $\lambda_{\text{obj}} = 0.4$ and the part segmentation weight $\lambda_{\text{part}} = 0.6$.

C.4. Training Details

Our implementation of HOPS builds upon the CAT-Seg framework [6]. During training, the input images are resized to 384×384, resulting in cost maps with a spatial resolution of 24×24. The CLIP and DINO branches are initialized with the pre-trained weights of CLIP ViT-B/16 and DINOv2 ViT-S/14, respectively. Fine-tuning is performed on the benchmark datasets using the AdamW optimizer [32] with an initial learning rate of 1×10^{-4} and a batch size of 4. Training is conducted for 20,000 iterations, and model checkpoints are saved every 1,000 iterations. The checkpoint achieving the best validation performance is selected for final evaluation.

D. Additional Ablation Study

D.1. Overall Effect of AFM and AEM

Table 9. Overall ablation of AFM and AEM on PartImageNet.

AFM	AEM	Seen	Unseen	h-IoU	mOSI ↓	mUSI ↓
		56.24	52.13	54.10	24.92	36.27
✓		58.53	54.82	56.61	**19.08**	34.24
	✓	58.37	54.51	56.37	24.92	32.76
✓	✓	**60.81**	**57.46**	**59.08**	**19.08**	**30.90**

Tab. 2 in the main paper validates the overall synergy between the two-stage segmentation paradigm and the AFM/AEM modules, while Tabs. 7 and 8 further analyze the contributions of individual sub-components within AFM and AEM, respectively. To more directly assess the effectiveness of AFM and AEM as well as their mutual synergy, we conduct ablation experiments on PartImageNet [18]. As shown in Tab. 9, AFM alone (row 2) substantially reduces object over-segmentation and improves h-IoU by 2.51%. Since AEM does not affect the object segmentation stage, applying AEM alone (row 3) keeps mOSI unchanged but effectively alleviates part under-segmentation, reducing mUSI by 3.51% and increasing h-IoU by 2.27%. When both modules are combined, HOPS achieves the best performance across all metrics, confirming their complementary synergy.

D.2. Effect of Different Attention

To evaluate the effectiveness of our bidirectional semantic–structural attention fusion mechanism, we conduct an ablation study that compares different attention types, with results shown in Table 10. DINO attention provides strong structural perception but lacks semantic discrimina-

Table 10. Ablation of attention type on PartImageNet.

Attention Type	Seen	Unseen	h-IoU	mOSI ↓	mUSI ↓
DINO	56.77	52.89	54.76	24.08	35.91
CLIP	58.13	54.04	56.01	21.25	34.83
Sem. Str. Attn.	**60.81**	**57.46**	**59.08**	**19.08**	**30.90**

tion, achieving the lowest h-IoU (54.76%) and the highest mOSI and mUSI. CLIP attention demonstrates superior semantic alignment, improving h-IoU to 56.01% and reducing mOSI to 21.25%, yet still suffers from part under-segmentation due to limited structural awareness. Our semantic–structural attention performs best across all metrics. These results demonstrate that the proposed bidirectional attention fusion mechanism effectively integrates CLIP's semantic alignment with DINO's structural perception, addressing the limitations of both models.

E. Datasets Details

E.1. Pascal-Part-116

Pascal-Part-116 [47] is a fine-grained part segmentation dataset constructed from the original Pascal-Part [4] through data cleaning and reorganization. To better suit open-vocabulary scenarios, the original annotations are optimized by merging overly specific parts with directional modifiers (*e.g.* merging "cow left foreleg" and "cow right hind leg" into "cow leg"). Redundant or ambiguous annotations are also removed to ensure concise and semantically consistent part definitions. In total, Pascal-Part-116 comprises 116 part categories (*e.g.* "bird wing", "cat paw", "car wheel"), covering 74 base categories and 42 novel categories. Pascal-Part-116 serves as a standard benchmark for evaluating zero-shot generalization and cross-dataset transfer in part segmentation tasks.

E.2. ADE20K-Part-234

ADE20K-Part-234 [47] is built upon the SceneParse150 subset of ADE20K [54], aiming to address the sparsity and noise present in the original part annotations. The specific processing includes: retaining only objects with high-frequency part annotations (each part appears at least 100 times), removing rare parts (fewer than 10 occurrences), and merging duplicate or synonymous annotations (*e.g.* "chair armrest" and "chair arm"). Ultimately, ADE20K-Part-234 contains 234 part categories, including 176 base categories and 58 novel categories. It covers a wide range of fine-grained entities across both indoor and outdoor scenes (*e.g.* "lamp shade", "sofa cushion", "car bumper"). With its fine-grained annotations and extensive scene diversity, ADE20K-Part-234 serves as a challenging benchmark for evaluating part segmentation accuracy and cross-granularity generalization in complex scenarios.

E.3. PartImageNet

PartImageNet [18] is a large-scale, high-quality part segmentation dataset derived from ImageNet [11], covering 158 object classes grouped into 11 superclasses (*e.g.* quadrupeds, birds, and vehicles). All object classes within the same superclass share the same part structure. PartImageNet provides fine-grained, pixel-level part annotations with an emphasis on non-rigid objects such as animals, achieving high precision and consistency in labeling. This dataset serves as a versatile benchmark for various vision tasks, including part segmentation, object segmentation, and few-shot learning. Notably, the dataset does not predefine base and novel categories. Following the setup of PartCATSeg [8], we select 40 representative object classes for zero-shot evaluation, dividing them into 25 base and 15 novel classes.

E.4. PartImageNet (OOD)

PartImageNet (OOD) [18] is an out-of-distribution (OOD) benchmark derived from PartImageNet, designed to evaluate the generalization ability of part segmentation models under zero-shot and cross-distribution settings. The dataset adopts a strict non-overlapping class split between the training and validation sets, ensuring complete class disjointness to realistically simulate challenging scenarios involving unseen categories. Specifically, the training set contains 109 base object classes, while the validation set consists of 19 novel object classes.

F. More Qualitative Results

As shown in Figs. 8 and 9, we present qualitative comparisons of our HOPS with PartCLIPSeg [7], PBAPS [27], and PartCATSeg [8] on PartImageNet. Additional qualitative results on Pascal-Part-116 are provided in Fig. 10.

Figure 8. Qualitative evaluation of zero-shot part segmentation on PartImageNet.

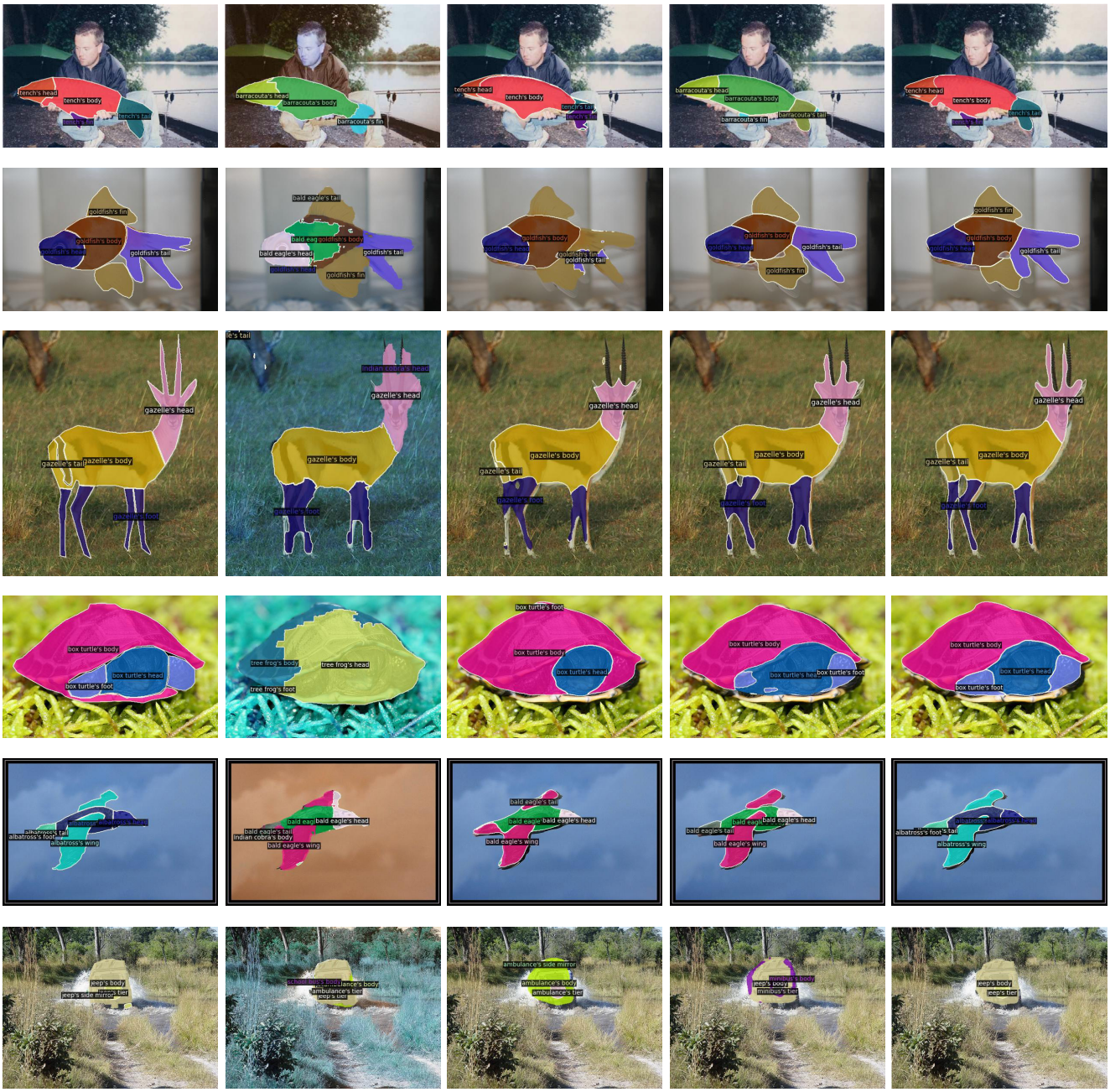

Figure 9. Qualitative evaluation of zero-shot part segmentation on PartImageNet.

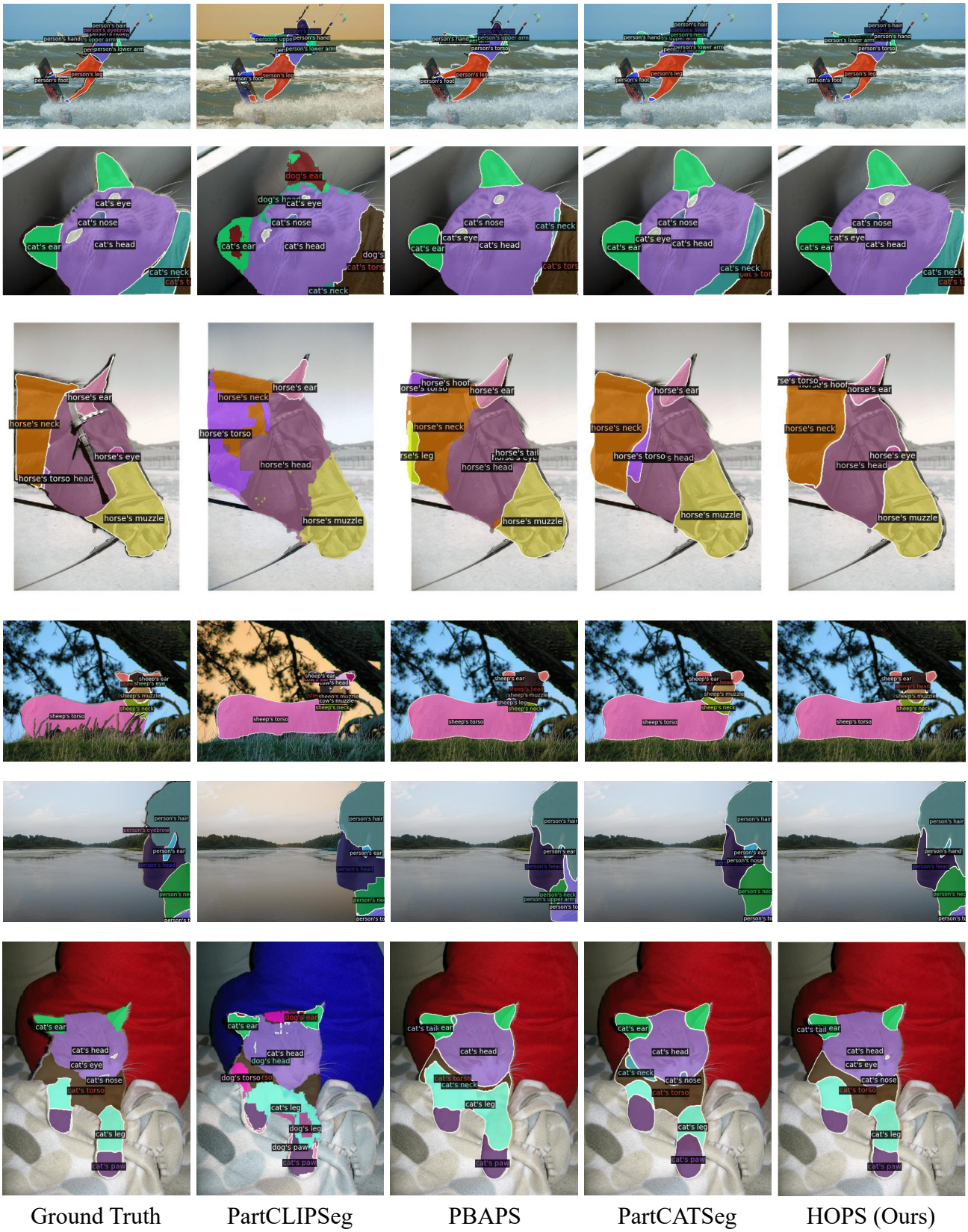

Figure 10. Qualitative evaluation of zero-shot part segmentation on Pascal-Part-116.