Revisiting Open-Set Panoptic Segmentation

Yufei Yin[1], **Hao Chen**[2], **Wengang Zhou**[1,3,*], **Jiajun Deng**[4], **Haiming Xu**[4], **Houqiang Li**[1,3,*]
[1] CAS Key Laboratory of Technology in GIPAS, EEIS Department, University of Science and Technology of China
[2] Zhejiang University
[3] Institute of Artificial Intelligence, Hefei Comprehensive National Science Center
[4] Australian Institute for Machine Learning, University of Adelaide
yinyufei@mail.ustc.edu.cn, haochen.cad@zju.edu.cn, zhwg@ustc.edu.cn,
jiajun.deng@adelaide.edu.au, hai-ming.xu@adelaide.edu.au, lihq@ustc.edu.cn

Abstract

In this paper, we focus on the open-set panoptic segmentation (OPS) task to circumvent the data explosion problem. Different from the close-set setting, OPS targets to detect both *known* and *unknown* categories, where the latter is not annotated during training. Different from existing work that only selects a few common categories ($\leq$ 16) as *unknown* ones, we move forward to the real-world scenario by considering the various tail categories ($\sim 1k$). To this end, we first build a new dataset with long-tail distribution for the OPS task. Based on this dataset, we additionally add a new class type for *unknown* classes and re-define the training annotations to make the OPS definition more complete and reasonable. Moreover, we analyze the influence of several significant factors in the OPS task and explore the upper bound of performance on *unknown* classes with different settings. Furthermore, based on the analyses, we design an effective two-phase framework for the OPS task, including thing-agnostic map generation and unknown segment mining. We further adopt semi-supervised learning to improve the OPS performance. Experimental results on different datasets validate the effectiveness of our method.

1 Introduction

Recent decades have witnessed a surge of high-quality datasets (Deng et al. 2009; Everingham et al. 2010; Lin et al. 2014; Cordts et al. 2016; Gupta, Dollar, and Girshick 2019), which lead to tremendous advances in visual perception algorithms (He et al. 2016; Ren et al. 2015; Redmon et al. 2016; He et al. 2017). However, the thirst for data is far from being satisfied since the models cannot perform robustly in complex real-world scenarios. Datasets with a large variety are crucial to the generalization performance for neural networks, but simply adding more labeled samples is not a viable solution. As the size and complexity of the datasets increase, the problem of long-tail distribution and label ambiguity becomes more significant. We refer to this problem as *data explosion*. Since it is unrealistic to extensively generate categorical labels for thousands of classes, we look for a more feasible approach by revisiting the relation between the categorical labels and the perception tasks.

*Corresponding authors: Wengang Zhou and Houqiang Li

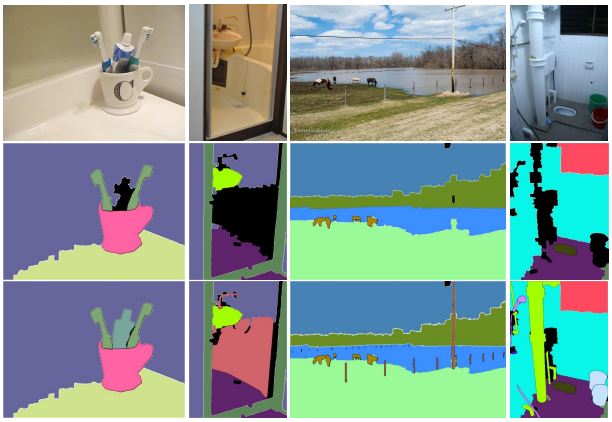

Figure 1: Comparison between COCO and LVIS-PS. The first row presents images, and the subsequent two rows present the annotations of COCO and LVIS-PS, respectively.

In this paper, we propose to circumvent the data explosion problem by studying a more realistic setting, termed open-set panoptic segmentation task (OPS). As an extension of panoptic segmentation (Kirillov et al. 2019), OPS requires to detect instances that are not annotated in the training set, *a.k.a.* the *unknown* category. In this setting, the annotation complexity does not increase as the dataset grows. On one hand, the tail categories[1] can be regarded as *unknown* categories, and no annotations for them are needed for training. On the other hand, the annotations of panoptic segmentation do not have overlapping ambiguity compared to the box-level ones, and each pixel is one-to-one mapped to a target. Therefore, OPS is a proper setting for robust perception network training where the dataset complexity exceeds manual label capability.

Only a few works have been explored on the OPS task. The pioneering work EOPSN (Hwang et al. 2021) first extends panoptic segmentation to the open-set setting and proposes an exemplar-based approach to discover unlabeled objects in the training set. Nevertheless, the existing OPS benchmark is in small scale and suffers some limitations in setting: (1) The COCO dataset (Lin et al. 2014) utilized in the benchmark only includes 80 common categories, omit-

[1]The tail parts of categories in long-tail distribution

ting a significant portion of rare classes. The incomplete annotations for rare classes in COCO may result in some correct open-set predictions being overlooked or incorrectly identified as "false positive" during inference. (2) Only a few common categories (≤ 16) in COCO are selected as *unknown* ones, which is a significant deviation from the real-world scenario where un-annotated categories can be rare and diverse. Moreover, the instances of the *unknown* classes all appear in the training images, which may leak some information to implicitly help the model to identify them. (3) Pixels with *unknown* classes are re-annotated as "void" ("ignore") type during training, which provides too much extra prior information that *unknown* classes only exist in the small parts of "void" areas in the image. In addition, certain important factors that have substantial impacts on the OPS task remain undiscussed in previous works, such as class information, which may affect the generalization capability from *known* categories to the *unknown* ones; annotation propotion, which affects the information of novel categories.

To address the above issues, in this paper, we first revisit the OPS task and re-formulate its benchmark settings. To involve more diverse categories and complete annotations, we construct a new LVIS-PS dataset for the OPS task based on the LVIS dataset (Gupta, Dollar, and Girshick 2019) and COCO. As shown in Fig. 1, LVIS-PS adds more segments with various tail categories to the *void* or *stuff* areas of COCO. We treat all these tail categories ($\sim 1k$) in LVIS-PS as *unknown* ones. We also introduce a new class type for *unknown* classes (*i.e.*, *unseen*), which is absent from the training images, and propose a new metric to evaluate it. Furthermore, we re-define the available training annotations to make the OPS settings more reasonable yet challenging.

Subsequently, we conduct a thorough analysis of several crucial factors that impact the performance of OPS, including different usage of class information, different annotation and category numbers. Finally, based on these analyses, we propose an effective two-phase framework for the OPS task, which consists of thing-agnostic map generation and unknown segment mining. We also build a Semi-PanoFCN-2s model with semi-supervised training to further improve the OPS performance. The proposed framework can be regarded as a simple yet effective baseline for the new challenging OPS benchmark. Our framework outperforms (Hwang et al. 2021) by a considerable margin on the *unknown* classes on LVIS-PS. Moreover, compared with the pure class-agnostic model (Qi et al. 2021), our framework not only has class-specific segmentation capability, but also shows better generalization capability to the other dataset (*i.e.*, ADE20K (Zhou et al. 2017)).

2 Related Work

2.1 Open-set Detection and Segmentation

Recently, the open-set problem has been explored in various computer vision tasks (Bendale and Boult 2015; Dhamija et al. 2020; Joseph et al. 2021; Gupta et al. 2022; Zhao et al. 2022; Vaze et al. 2021; Qi et al. 2021; Saito et al. 2021; Wang et al. 2022b; Hwang et al. 2021; Wang et al. 2022a, 2021). Dhamija *et al.* (Dhamija et al. 2020) first formalize the open-set object detection problem and propose the open-set object detection protocol to better estimate the performance under real-world conditions. Joseph *et al.* (Joseph et al. 2021) propose the ORE model to achieve the open-world detection task based on the energy-based identifier and contrastive clustering. For the segmentation task, Lu *et al.* (Qi et al. 2021) propose a class-agnostic entity segmentation task and construct a Global Kernel Bank with both dynamic and static kernels to generate entity masks. LDET (Saito et al. 2021) introduces a new data augmentation and uses decoupled training for open-world instance segmentation. Hwang *et al.* (Hwang et al. 2021) extend panoptic segmentation to the open-set setting and propose an EOPSN model which uses RPN to obtain proposals for *unknown* classes and applies clustering to mine reliable exemplars.

Our work focuses on the open-set panoptic segmentation task following (Hwang et al. 2021). However, different from (Hwang et al. 2021), we re-formulate the open-set panoptic segmentation task from several aspects and introduce various tail categories to make it closer to the real-world condition but more challenging.

3 Rethinking Open-Set Panoptic Segmentation

In this section, we first formalize the open-set panoptic segmentation (OPS) task (Sec. 3.1). To address the drawbacks of the original OPS settings, we construct a new OPS benchmark to make it closer to the real-world scenario yet more challenging (Sec. 3.2). After that, we introduce the applied evaluation metrics for the new OPS task (Sec. 3.3).

3.1 Problem Formulation

Panoptic segmentation is a combination of instance segmentation and semantic segmentation. It aims to classify each pixel to its corresponding *thing* or *stuff* class and segment each individual instance for *thing* classes. The main difference between open-set panoptic segmentation (OPS) and the common panoptic segmentation setting (close-set) is that the former involves a special *unknown* class, which is not available for training. Concretely, suppose a set of *known* classes $C = \{0, \cdots, C-1\}$ and a set of *unknown* categories is predefined. All these *unknown* categories are selected from the *thing* categories and regarded as a special "*unknown* class" U. In the training stage, only the data of the *known* classes are available, and their annotations are the same as the close-set settings. In the inference stage, all segments with the *known* classes or the special *unknown* class are supposed to be found in a given image.

3.2 Towards a New OPS Benchmark

As discussed in Sec. 1, the current OPS setting (Hwang et al. 2021) remains drawbacks and suffers large gaps from the real-world scenario. Therefore, we construct a new benchmark for the OPS task according to the following steps:

Annotation aggregation. We aim to adopt the more complicated LVIS dataset (Gupta, Dollar, and Girshick 2019) for the OPS task, which shares the same images with the COCO dataset (Lin et al. 2014) while re-annotating them with more

Method	Train		Test	Unknown Classes		
	Anno. Source	Unk. Annotated	Anno. Source	Number	Type	Source
EOPSN	COCO	Void	COCO	4/8/16	Seen	COCO
Ours	COCO	Void & Stuff	LVIS-PS	1020	Seen & Unseen	Long Tail in LVIS-PS

Table 1: Comparison of the benchmark settings of EOPSN (Hwang et al. 2021) and ours. "Anno. Source" denotes the source of annotations during training or testing. "Unk. Annotated" denotes the classes that *unknown* segments may be annotated in training annotations.

diverse categories and complete annotations. However, LVIS cannot be used for OPS directly since it is constructed primarily for the instance segmentation task with overlapped instance annotations and no *stuff* categories. To address the problem, we build a new panoptic segmentation dataset, named "LVIS-PS", based on (Gupta, Dollar, and Girshick 2019) and (Lin et al. 2014). Concretely, we follow a "Thing First, COCO First" principle to generate the panoptic segmentation level annotations of the LVIS-PS dataset. For each image, we place the annotations from different sources on the *remaining blank areas* of a panoptic map in the following order: COCO-THING, LVIS-THING, COCO-STUFF [2]. Specially, if a newly added instance has high overlaps with existing ones on the map, it will be discarded to avoid ambiguity. Detailed information of LVIS-PS and the procedure to construct it are presented in the Supplementary Material. Consequently, all categories in COCO are retained in LVIS-PS while more tail categories are added with no overlap. As shown in Fig.1, LVIS-PS additionally labels more instances that are originally regarded as *stuff* or ignored in COCO.

Category split. Considering the un-annotated categories can be rare and diverse in the real-world scenario, we select various tail categories ($\sim 1k$) as *unknown* classes, *i.e.*, the newly added ones in LVIS-PS compared to COCO. Correspondingly, categories in COCO are regarded as *known* classes. Moreover, though the annotations of the *unknown* classes are not available at the training stage, their corresponding objects still exist in the training images in the previous OPS setting, which will help some class-agnostic classifiers (*i.e.*, region proposal network in (Hwang et al. 2021)) to identify them implicitly. In contrast, the OPS method also needs to possess the capability to find segments with classes that never appear in training, which are denoted as *unseen* classes. These classes are genuinely "open-set" to some degree. To this end, we select a portion of *unknown* classes with few samples as *unseen* classes, and remove all images that contain these classes during training (about 10% training images). Accordingly, the remained *unknown* classes are denoted as *seen* classes.

Unknown region removal. According to the definition of the OPS task, annotations of *unknown* classes need to be removed before training. A problem naturally arises as to what their corresponding pixels should be re-annotated as. In the previous OPS setting, these pixels are re-annotated as "void" (or "ignore") type, which is ignored at the training stage. However, this operation will introduce unreasonable prior information that the *unknown* classes are solely present in the limited regions of "void" areas in the image. Instead, there is another reasonable situation where segments of those *unknown* classes are annotated as *stuff* classes under a loose criterion. Based on this assumption, original COCO annotations become an optimal training source for LVIS-PS since the annotations of LVIS-PS are extended based on COCO annotations, and these newly added segments are naturally annotated as "void" type or *stuff* classes in original COCO annotations.

In summary, we adopt the LVIS-PS dataset for the OPS task. We use the corresponding original COCO annotations during training, while using generated LVIS-PS annotations for inference. Four class types (*i.e.*, *known-thing*, *known-stuff*, *seen*, *unseen*) are considered during evaluation. Comparison of the benchmark settings proposed in (Hwang et al. 2021) and ours is shown in Tab. 1.

3.3 Evaluation Metrics

Following (Hwang et al. 2021), we use three standard panoptic segmentation metrics (Kirillov et al. 2019), including panoptic quality (PQ), segmentation quality (SQ), and recognition quality (RQ). However, for *unseen* class, it's not appropriate to use original RQ and PQ, which contains "false positive" (FP), to measure its performance for the following reasons. First, FPs may face an important condition that the prediction is actually an object but with no ground-truth assigned to it. However, these kinds of predictions are not real "false positives" for *unseen* class in the open-set setting. Second, for *unseen* class, we tend to find as many potential objects as possible, hence the "false positives" are not a major concern. Third, compared with other tasks (*e.g.*, object detection), recall in panoptic segmentation can better reflect real-world performance, since the latter requires each pixel to be one-to-one mapped to a class. More FPs of *unseen* can be reflected from the performance of other class types (*e.g.*, *stuff*). Moreover, *unseen* class shares the same FPs with *seen* class since they are both supposed to be predicted as the special *unknown* class. It's reasonable to treat all these FPs as FPs of *seen* class since there are significantly more ground-truths in *seen* class than in *unseen* class.

Hence, we propose a modified PQ (denoted as PQ*), which replaces the RQ with Recall during its computation, to measure the performance of *unseen* class. The detailed modification is presented in Equ. 1:

$$\begin{aligned} \text{PQ} &= \underbrace{\frac{\sum_{(p,g)\in TP} IoU(p,g)}{|TP|}}_{\text{segmentation quality (SQ)}} \cdot \underbrace{\frac{|TP|}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}}_{\text{recognition quality (RQ)}}, \\ \text{PQ*} &= \underbrace{\frac{\sum_{(p,g)\in TP} IoU(p,g)}{|TP|}}_{\text{segmentation quality (SQ)}} \cdot \underbrace{\frac{|TP|}{|TP| + |FN|}}_{\text{Recall}}. \end{aligned} \tag{1}$$

[2] We use COCO-Thing to represent "thing classes in COCO dataset" for simplicity. LVIS-Thing and COCO-Stuff share the same representation.

Settings	*Seen*		*Unseen*	
	PQ	PQ-thing	PQ*	PQ*-thing
Class-Specific	18.53	23.21	8.59	20.32
Comb-Seen	17.91	21.21	17.67	26.88
Comb-All	-	25.18	-	29.77

Table 2: Performance on *unknown* classes with different class information.

4 Analysis of Influencing Factors in OPS

In this section, we study the influence of several significant factors on the OPS task, and explore the upper bound of performance on *unknown* classes with different settings. In these experiments, we assume that all annotations of *seen* categories are provided. We use a two-stage Panoptic FCN model (denoted as PanoFCN-2s) for all the experiments, which is modified from the original Panoptic FCN (Li et al. 2021) and the details will be discussed in the next section.

4.1 Influence of Class Information

Recent studies (Li et al. 2020; Kim et al. 2022) indicate that a class-agnostic detector will help detect more open-world instances. This inspires us to investigate the impact of class information on the OPS task. We consider three types of class information for training: (1) Class-Specific. All segments are annotated with their specific classes. (2) Comb-Seen. All *seen* classes are combined as a single class (referred as "unknown-comb"). In other words, we re-annotate all segments of *seen* classes with "unknown-comb" class. (3) Comb-All. We combine all *thing* (*i.e.*, *known-thing*, *seen*) classes as a single "thing-comb" class, while leaving the *stuff* classes unchanged. *Unseen* classes are not considered here, as they only occur in the test set.

Considering that they are trained with different category numbers, we need to unify their evaluation methods on *unknown* classes. Following the OPS settings, if segments of *unknown* classes are classified as any one of the *unknown* classes (for (1)) or "unknown-comb" class (for (2)), they will be regarded as "true positives (TPs)". To compare (1), (2) with (3), we follow another principle that segments of *unknown* classes are TPs if they are classified as any one of the *thing* classes when calculating PQ, which is denoted as "PQ-thing" ("PQ*-thing"). It's worth noting that when calculating the "PQ-thing" of *seen* class for (3), we use the expectation of FPs since its true value cannot be obtained.

The results are shown in Tab. 2. We find that the performance of Comb-All performs the best among the three settings on both two *unknown* classes. These results verify that if we follow a class-agnostic setting to reduce or eliminate the class-variation information, models will have better segmentation and generalization capabilities on *unknown* classes. We attribute this to the fact that this setting will drive model to ignore the differences between each *thing* class, thus forcing it to learn stronger objectness cues.

4.2 Influence of Annotation Propotion

It's widely acknowledged that a dataset with more annotations is likely to enhance model performance, as the model can be exposed to a greater variety of samples in the training stage. It motivates us to quantitatively study the influence of annotation numbers in this task. Specifically, we construct four different splits with different selection ratios $(20\%, 40\%, 60\%, 80\%)$. We randomly select the *seen* annotations with corresponding ratios, while the *known* annotations remain unchanged. We use PanoFCN-2s model with Comb-Seen setting in these experiments.

Ratios		20%	40%	60%	80%	100%
seen anno.	*Seen*	12.31	16.64	17.66	**18.03**	17.91
	Unseen	5.78	11.17	14.69	16.82	**17.67**
seen category	*Seen*	6.48	14.03	15.32	17.39	**17.91**
	Unseen	4.00	7.80	12.18	15.20	**17.67**

Table 3: Performance with different ratios of *seen* annotation numbers or *seen* category numbers. The experiments are based on the PanoFCN-2s model and Comb-Seen setting.

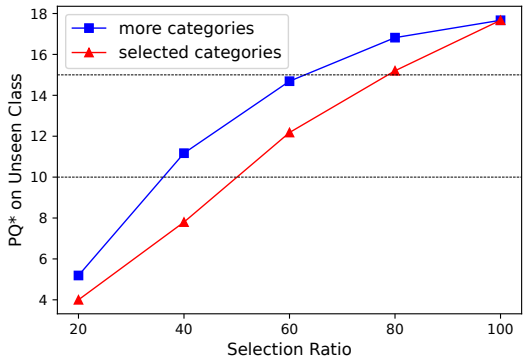

Figure 2: The red points denote the performance with different ratios of *seen* classes. For each selection ratio, the blue point denotes the performance with similar annotation amounts but more category numbers.

The results are shown in the top parts of Tab. 3. On one hand, the performance of *seen* classes is notably improved when the ratio increases initially, but this improvement gradually diminishes and may even become negative. On the other hand, the increment of ratio brings continuous performance improvement of *unseen* classes. We attribute it to the fact that the increment of annotations will guide the network to mine more potential instances, thereby aiding the discovery of *unseen* classes. However, this may also lead to more false positives, thus hindering the performance improvement of *seen* classes.

Furthermore, we adopt another strategy to select annotations, that is, we randomly choose different ratios of *seen* categories with their corresponding annotations. As shown in the bottom parts of Tab. 3, the performance trends are similar to the above results. Moreover, these results drive us to think of a question: does the increment of category numbers play an important role in the performance improvement? To this end, for each setting of category numbers, we conduct another experiment with similar annotation amounts but containing more categories. The results are shown in

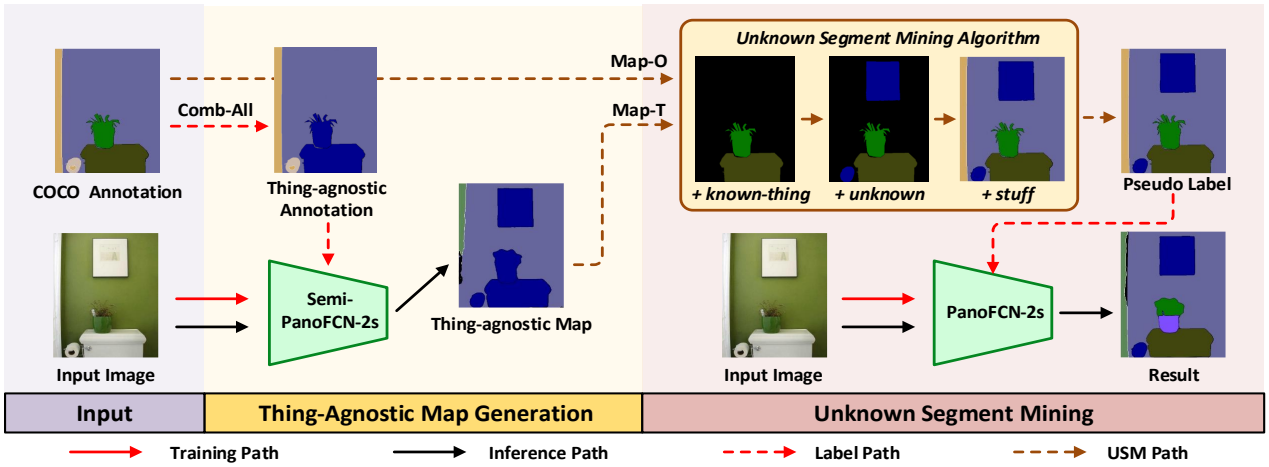

Figure 3: Overview of the proposed two-phase framework, including thing-agnostic map generation (first phase) and unknown segment mining (second phase). In the first phase, we apply Semi-PanoFCN-2s to mine more potential unknown instances.

Fig. 2, from which we can draw two conclusions: (1) With similar annotation numbers, more category numbers perform better. (2) The increment of category numbers has a more significant impact than only increasing corresponding amounts of annotations when the annotation number reaches a certain value (40%). The great influence of category variety reminds us that introducing more categories is more important than generating more annotations for the OPS task. However, the diversity of categories is far from sufficient under the COCO annotations. Labels with more various categories are needed to improve the model's generalization capability. Therefore, we propose to design a framework that generates pseudo labels of instances with novel categories automatically.

5 Method

Based on our analyses in Sec. 4, we can conclude that the class-agnostic setting leads to better performance on *unknown* classes, and annotations containing more categories will significantly help the OPS task. However, the number of *known* classes is very limited in the OPS setting. To this end, we first modify (Li et al. 2021) into a two-stage structure (Sec. 5.1) and then design a two-phase semi-supervised framework (Sec. 5.2 - 5.4) to enrich the category variety in the annotations, thus better completing the OPS task. The whole framework is shown in Fig. 3.

5.1 PanoFCN-2s

Due to its one-stage structure, Panoptic FCN (Li et al. 2021) will suffer from the foreground-background class imbalance problem, hence not excel at detecting more potential unknown segments. Therefore, we modify it into a two-stage structure (denoted as *PanoFCN-2s*) to better fit the OPS task. We construct an RoI Kernel Head to generate kernels for *thing* classes following the structure in (He et al. 2017), and use it to replace the Kernel Generator module in (Li et al. 2021). Details please refer to the Supplementary Material.

5.2 First Phase: Thing-Agnostic Map Generation

To find potential *unknown* segments sufficiently and with high quality, we need to choose a reasonable training strategy. As discussed in Sec. 4, we find that the Comb-All setting performs the best on *unknown* classes. Therefore, we first combine all *thing* (*i.e.*, *known-thing*, *seen*) classes into a single "thing-comb" class and re-annotate *thing* segments with it. *Stuff* classes remain unchanged. Next, we use these re-annotated training samples to train a PanoFCN-2s model. Particularly, the output dimension of the classification branch in PanoFCN-2s is set to $S+1$, where S is the number of *stuff* classes. After training, we pass training images through the model to obtain the prediction maps. Benefited from the Comb-All setting, these maps contain many potential *thing* segments but are *thing-agnostic* that all these segments belong to one "thing-comb" class.

5.3 Second Phase: Unknown Segment Mining

We now have two kinds of panoptic segmentation maps of training images, one is the accurate original annotations but without *unknown* classes (denoted as Map-O), the other is the generated *thing-agnostic* maps (denoted as Map-T). To mine potential *unknown* segments and generate complete segmentation maps, we design an Unknown Segment Mining (USM) algorithm to take advantage of both two maps. First of all, we need to clarify the areas where the *unknown* segments may be found. As shown in Fig. 1, the original COCO annotations tend to place the *unknown* segments into *void* (*ignore*) or *stuff* areas. Hence, we choose to mine *unknown* segments from these two areas. Concretely, we first fetch the segments with *thing* class from the generated Map-T, denoted as $TH = [th_1, \cdots, th_n]$. Next, we calculate the intersection areas of each segment in TH with *void* and *stuff* areas in Map-O separately. Segments with high intersections will be chosen as potential *unknown* segments.

After obtaining *unknown* segments, we then need to combine them with the Map-O. We follow a "Thing First, Known First" principle to construct the complete annotations. Specifically, for a training image, we first take *known-thing* segments and *known-stuff* segments from Map-O, and

Model	*Known Classes*						*Unknown Classes*					
	Known-Thing			*Known-Stuff*			*Seen*			*Unseen*		
	PQ	SQ	RQ	PQ	SQ	RQ	PQ	SQ	RQ	PQ*	SQ	Recall
Supervised model												
`PanoFCN-2s`	42.41	78.15	52.17	27.44	70.31	33.87	17.91	78.35	22.87	17.67	79.61	22.20
Open-set Panoptic segmentation methods												
`Void-Supp` (Hwang et al. 2021)	**44.81**	79.90	**53.97**	26.67	72.36	33.94	4.57	73.80	6.19	9.70	71.96	13.48
`EOPSN` (Hwang et al. 2021)	44.74	**80.74**	54.05	26.45	72.62	33.64	0.51	74.64	0.68	0.26	74.41	0.35
`Two-Phase (PanoFCN)`	44.00	80.88	53.74	**29.59**	75.41	**36.70**	3.49	**81.42**	4.29	4.20	79.97	5.26
`Two-Phase (PanoFCN-2s)`	43.80	78.37	53.73	28.07	**75.43**	34.62	7.89	78.55	10.04	16.85	79.02	21.33
`Semi-Two-Phase`	43.23	78.36	53.03	27.53	74.22	33.94	**9.98**	80.17	**12.45**	**19.80**	**80.19**	**22.20**

Table 4: Open-set Panoptic segmentation results on LVIS-PS *val* set under the proposed OPS setting, which needs to predict *known* classes and *unknown* classes with only annotations of *known* classes are used. "Supervised model" represents that a PanoFCN-2s model is trained with annotations where *seen* classes are available. It is worth mentioning that our performance on *unseen* classes even outperforms that of the supervised model with some margins.

initialize a blank panoptic segmentation map. Then, we place these segments on the blank areas of the map following the order: (1) *known-thing* segments, (2) *unknown* segments, (3) *known-stuff* segments. Finally, these complete panoptic segmentation maps are used as pseudo labels to train another PanoFCN-2s model. In this way, many potential *unknown* segments are added in the annotations, enriching their category varieties, hence benefiting the OPS training. Particularly, the output dimension of its classification branch is set to $T + S + 1$, where T, S are the number of *known-thing*, *known-stuff* classes, respectively. Only this PanoFCN-2s model is applied during inference.

5.4 Semi-PanoFCN-2s

Though we have built a simple yet effective baseline to achieve the OPS task, we further improve the first PanoFCN-2s model to make it more suitable for this task, thereby boosting the performance on the *unknown* classes. In the first phase, the PanoFCN-2s model is able to find potential *thing* segments from the images, benefiting from the proper model structure and the Comb-All setting. However, it relies much on the model's generalization capability while lacking task-specific guidance. Hence, we adopt the semi-supervised learning strategy into the training procedure and modify the PanoFCN-2s model to achieve it. Specifically, we add a new classification branch CLS_2 in the RoI Kernel Head of PanoFCN-2s model, paralleling with the original one (CLS_1). Different from CLS_1, CLS_2 aims to mine more potential *thing* segments following the online semi-supervised training strategy. Hence, we denote the modified PanoFCN-2s model as Semi-PanoFCN-2s. During training, we first select top-k proposals according to their classification scores on the *thing* class, generate their corresponding masks, and filter out low-scoring ones. The kept masks are considered as proposals for *unknown* segments. As mentioned in Sec. 5.3, *unknown* segments are likely to hide in the *void* or *stuff* areas. Hence, we calculate the intersection areas of each of those *unknown* proposals with the two areas separately and remove the proposals with low intersections. Besides, we additionally set a scoring threshold on the proposals which have high intersections with the *stuff* areas to guarantee the quality of *stuff* classes. Finally, we relabel the remained proposals as the *thing* class, and use these pseudo labels to train the CLS_2. During inference, we only use CLS_2 to obtain classification scores.

The method to mine potential *unknown* segments is similar to the USM algorithm, but the most significant difference is that it participates in the training procedure following a semi-supervised training strategy, hence is able to enhance the ability of the network to find more *unknown* segments. It is worth noting that we only replace the PanoFCN-2s model with Semi-PanoFCN-2s in the first phase of the framework.

6 Experiments

We evaluate our method on the proposed LVIS-PS dataset. During training, as discussed in Sec. 3.2, we use the corresponding original COCO annotations of LVIS-PS *train* set, which contain 80 *thing* classes and 53 *stuff* classes. LVIS-PS *val* set is utilized for evaluation, which has 994 classes in total. Three kinds of standard panoptic segmentation metrics (Kirillov et al. 2019), including panoptic quality (PQ), segmentation quality (SQ) and recognition quality (RQ) are applied for three class types, *i.e.*, *known-thing*, *known-stuff* and *seen*. PQ* and Recall are used for *unseen* classes.

6.1 Experiment Setup

As described in Sec. 5, we follow a two-phase paradigm to achieve the OPS task. In the first phase, we train the proposed Semi-PanoFCN-2s with the Comb-All setting to produce *thing-agnostic* maps. In the second phase, we use USM algorithm to generate pseudo labels and train the proposed PanoFCN-2s with the Comb-Seen setting using these generated labels. During inference, only the PanoFCN-2s model is applied. In both two phases, we follow the original settings of (Li et al. 2021) with $1\times$ and multi-scale strategies. For hyperparameters, the overlap thresholds are set to 0.8 and 0.9 for *void* and *stuff* areas, respectively. The score threshold for *stuff* areas is set to 0.3 in Semi-PanoFCN-2s. k is set to 50, which is the same with (Li et al. 2021).

Models	Epoch	AP_e^m	$AP_e^m{}_{50}$	$AP_e^m{}_{75}$	$AP_e^m{}_s$	$AP_e^m{}_m$	$AP_e^m{}_l$
Trained on COCO							
`Panoptic FCN` (Li et al. 2021)	12	11.88	24.19	10.61	3.49	7.75	22.73
`ES` (Qi et al. 2021)	12	13.78	26.57	12.55	1.73	11.27	26.98
`ES` (Qi et al. 2021)	36	14.66	27.96	13.43	2.06	13.03	28.08
Trained on LVIS-PS (a part of COCO) with COCO annotations							
`Two-Phase`	12	16.02	30.35	14.48	**4.49**	9.55	29.34
`Semi-Two-Phase`	12	**16.28**	**30.79**	**14.63**	4.23	**9.67**	**29.91**

Table 5: Cross-dataset results on ADE20K *val* set. The models of (Li et al. 2021) and (Qi et al. 2021) are trained with COCO, while our model is trained with LVIS-PS with COCO annotations, which has fewer training images.

6.2 Evaluation on LVIS-PS Dataset

Tab. 4 shows the performances on the LVIS-PS *val* set of different models. The supervised one (1st row) is a PanoFCN-2s model, which is directly trained following the Comb-Seen setting and with complete annotations, in which annotations of *seen* classes are available. Compared with the supervised model, the proposed two-phase framework (5th row) can achieve comparable performance on *unseen* classes, and even performs slightly better on both two types of *known* classes. For *seen* classes, the performance of the supervised model can be seen as an upper bound. With lacking annotations of over 700 kinds of *seen* classes, the performance gap of the two-phase framework with the supervised one is reasonable. Overall, these results demonstrate that the proposed framework can achieve the OPS task with a relatively good performance.

When the proposed Semi-PanoFCN-2s is employed in the first phase (6th row), the performance on the *unknown* classes improves in all aspects, while still achieving competitive performance on *known* classes. It is worth mentioning that the performance on *unseen* classes even outperforms that of the supervised model with some margins (+2.13% on PQ*, +0.58% on SQ). In addition, when we replace PanoFCN-2s with Panoptic FCN (Li et al. 2021) (4th row), the performance on *unknown* classes drops to a great degree, which demonstrates the importance of PanoFCN-2s on the OPS task. These results verify the effectiveness of our proposed contributions in the OPS setting for constructing a two-phase framework with a two-stage model and adopting semi-supervised learning to enable the network to find more potential accurate *unknown* segments.

We also compare our framework with the previous OPS method EOPSN (Hwang et al. 2021). Their results are shown in the second and third rows, where "Void-Supp" represents the baseline model of EOPSN. We re-train and infer these models on the LVIS-PS datasets, strictly following the original settings. As shown in Tab. 4, our method is superior to EOPSN on *unknown* classes by a large margin, and has comparable performance with it on *known* classes. We attribute its poor performance on LVIS-PS datasets to the fact that the samples of tail *unknown* classes are rare and diverse, and thus are hard to be clustered as in EOPSN.

The visualization results of different models are shown in Fig. 3 in Supplementary Material. Segments with *unknown* classes are in deep blue. Compared with Panoptic FCN (Li et al. 2021) (3rd row) and Void-Supp (Hwang et al. 2021) (4th row), our method is able to find more *unknown* instances (in deep blue) with comparable or better *stuff* quality (1st-3rd column). Moreover, in many cases, we observe that our method also has better segmentation capability on *known* classes (4th-5th column).

6.3 Cross-Dataset Evaluation

To validate the generalization capability of our method, we evaluate our trained model on another dataset ADE20K (Zhou et al. 2017). Considering that the classes of ADE20K and COCO (LVIS-PS) are different, we apply the entity segmentation metric AP_e^m (Qi et al. 2021) for evaluation. AP_e^m is similar to AP^m used in instance segmentation, while it regards all segments as one class, including those in *thing* or *stuff* classes, and gives no tolerance to the overlaps of different segments.

Tab. 5 shows the generalization results on the ADE20k dataset with different models. For Panoptic FCN (Li et al. 2021) and ES (Qi et al. 2021), we use their released models trained on COCO and evaluate them on the whole ADE20K *val* set. It's worth mentioning that we actually use fewer training samples than them, since the training set of LVIS-PS is a part of that of COCO. Despite this, our proposed method (Line 4-5) outperforms them (Line 1-2) by at least 2.24% AP_e^m, and even surpasses (Qi et al. 2021) (Line 3) trained with more epochs. Especially, compared with the class-agnostic model (Qi et al. 2021), our method not only shows better generalization performance, but also possesses class-specific segmentation capability.

7 Conclusion

In this paper, we first build a new dataset LVIS-PS for the OPS task and redefine the OPS settings in a more reasonable and practical way. We regard tail categories in LVIS-PS as *unknown* classes and redefine the training annotations to avoid unreasonable prior information. Subsequently, we analyze the influence of several significant factors for the OPS task, such as class information and annotation propotion. Based on these analyses, we design an effective two-phase semi-supervised framework to accomplish the OPS task, which comprises of thing-agnostic map generation and unknown segment mining. Experimental results on different datasets demonstrate the effectiveness of our method.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Contract U20A20183 and 62021001. It was also supported by the GPU cluster built by MCC Lab of Information Science and Technology Institution and the Supercomputing Center of the USTC. This work was also supported in part by the National Key R&D Program of China (NO.2022ZD0160101).

References

Bendale, A.; and Boult, T. 2015. Towards open world recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1893–1902.

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3213–3223.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Dhamija, A.; Gunther, M.; Ventura, J.; and Boult, T. 2020. The overlooked elephant of object detection: Open set. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 1021–1030.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2): 303–338.

Gupta, A.; Dollar, P.; and Girshick, R. 2019. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5356–5364.

Gupta, A.; Narayan, S.; Joseph, K.; Khan, S.; Khan, F. S.; and Shah, M. 2022. OW-DETR: Open-world detection transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9235–9244.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask R-CNN. In *Proceedings of the IEEE Conference on International Conference on Computer Vision*, 2961–2969.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Hwang, J.; Oh, S. W.; Lee, J.-Y.; and Han, B. 2021. Exemplar-based open-set panoptic segmentation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1175–1184.

Joseph, K.; Khan, S.; Khan, F. S.; and Balasubramanian, V. N. 2021. Towards open world object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5830–5840.

Kim, D.; Lin, T.-Y.; Angelova, A.; Kweon, I. S.; and Kuo, W. 2022. Learning Open-World Object Proposals without Learning to Classify. *IEEE Robotics and Automation Letters*.

Kirillov, A.; He, K.; Girshick, R.; Rother, C.; and Dollár, P. 2019. Panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9404–9413.

Li, S.; Zhou, J.; Jia, Z.; Yeung, D.-Y.; and Mason, M. T. 2020. Learning accurate objectness instance segmentation from photorealistic rendering for robotic manipulation. In *Proceedings of the 2018 International Symposium on Experimental Robotics*, 245–255. Springer.

Li, Y.; Zhao, H.; Qi, X.; Wang, L.; Li, Z.; Sun, J.; and Jia, J. 2021. Fully convolutional networks for panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 214–223.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, 740–755.

Qi, L.; Kuen, J.; Wang, Y.; Gu, J.; Zhao, H.; Lin, Z.; Torr, P.; and Jia, J. 2021. Open-world entity segmentation. *arXiv preprint arXiv:2107.14228*.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779–788.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28.

Saito, K.; Hu, P.; Darrell, T.; and Saenko, K. 2021. Learning to detect every thing in an open world. *arXiv preprint arXiv:2112.01698*.

Vaze, S.; Han, K.; Vedaldi, A.; and Zisserman, A. 2021. Open-set recognition: A good closed-set classifier is all you need. *arXiv preprint arXiv:2110.06207*.

Wang, W.; Feiszli, M.; Wang, H.; Malik, J.; and Tran, D. 2022a. Open-world instance segmentation: Exploiting pseudo ground truth from learned pairwise affinity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4422–4432.

Wang, W.; Feiszli, M.; Wang, H.; and Tran, D. 2021. Unidentified video objects: A benchmark for dense, open-world segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10776–10785.

Wang, X.; Zhao, K.; Zhang, R.; Ding, S.; Wang, Y.; and Shen, W. 2022b. ContrastMask: Contrastive Learning to Segment Every Thing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 11604–11613.

Zhao, X.; Liu, X.; Shen, Y.; Ma, Y.; Qiao, Y.; and Wang, D. 2022. Revisiting open world object detection. *arXiv preprint arXiv:2201.00471*.

Zhou, B.; Zhao, H.; Puig, X.; Fidler, S.; Barriuso, A.; and Torralba, A. 2017. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 633–641.