OPEN-VOCABULARY SEMANTIC SEGMENTATION WITH MASK-ADAPTED CLIP

Anonymous authors
Paper under double-blind review

ABSTRACT

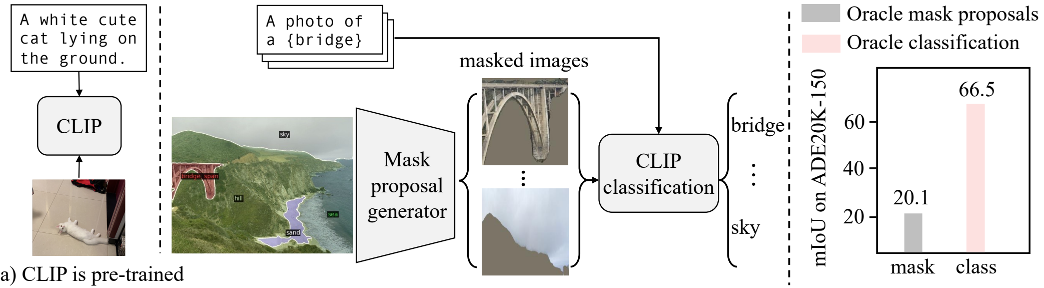
Open-vocabulary semantic segmentation aims to segment an image into semantic regions according to text descriptions, which may not have been seen during training. Recent two-stage methods first generate class-agnostic mask proposals and then leverage pre-trained vision-language models, *e.g.*, CLIP, to classify masked regions. We identify the performance bottleneck of this paradigm to be the pre-trained CLIP model, since it does not perform well on masked images. To address this, we propose to finetune CLIP on a collection of masked image regions and their corresponding text descriptions. We collect training data by mining an existing image-caption dataset (*e.g.*, COCO Captions), using CLIP to match masked image regions to nouns in the image captions. Compared with the more precise and manually annotated segmentation labels with fixed classes (*e.g.*, COCO-Stuff), we find our noisy but diverse dataset can better retain CLIP's generalization ability. Along with finetuning the entire model, we utilize the "blank" areas in masked images using a method we dub *mask prompt tuning*. Experiments demonstrate mask prompt tuning brings significant improvement without modifying any weights of CLIP, and it can further improve a fully finetuned model. In particular, when trained on COCO and evaluated on ADE20K-150, our best model achieves 29.6% mIoU, which is +8.5% higher than the previous state-of-the-art. For the first time, open-vocabulary *generalist* models match the performance of supervised *specialist* models in 2017 without dataset specific adaptations.

1 INTRODUCTION

Semantic segmentation aims to group pixels into meaningful regions with corresponding semantic categories. Although remarkable progress has been made (Long et al., 2015; Chen et al., 2017; 2018; Zhao et al., 2017; Cheng et al., 2021), modern semantic segmentation models are mainly trained with pre-defined categories, failing to generalize to unseen classes. On the contrary, humans understand scenes in an open-vocabulary manner, typically with thousands of categories (Biederman, 1987). To approach human-level perception, this paper studies open-vocabulary semantic segmentation where the model segments an image by arbitrary categories described by texts.

Vision-language models, *e.g.*, CLIP (Radford et al., 2021), learn rich multi-modal features from billion-scale image-text pairs. Witnessing its superior open-vocabulary classification ability, prior works propose to use pre-trained vision-language models for open-vocabulary segmentation (Li et al., 2022; Xu et al., 2021; Ding et al., 2022; Ghiasi et al., 2021). Among them, two-stage approaches have shown great potential: they first generate class-agnostic mask proposals and then leverage pre-trained CLIP to perform open-vocabulary classification (see Figure 1(b)). Their success relies on two assumptions: (1) the model can generate class-agnostic mask proposals (2) pre-trained CLIP can transfer its classification performance to masked image proposals.

To examine these two assumptions, we conduct the following analysis. First, we assume an "oracle" mask generator and an ordinary CLIP classifier. We use ground-truth masks as region proposals, and feed masked images to a pre-trained CLIP for classification. This model only reaches an mIoU of 20.1% on the ADE20K-150 dataset. Next, we assume an "oracle" classifier but an ordinary mask proposal generator – a MaskFormer (Cheng et al. (2021)) pre-trained on the COCO dataset. We first extract masked region proposals, then compare each region with ground-truth object masks, find the



(a) CLIP is pre-trained with natural images

(b) Skeleton of two-stage approaches

(c) Bottleneck analysis

Figure 1: (a) CLIP is pre-trained with natural images with little data augmentation. (b) Two-stage open-vocabulary semantic segmentation approaches first generate class-agnostic mask proposals and then leverage pre-trained CLIP to do open-vocabulary classification. The input of the CLIP model is cropped masked images, which have huge domain gap from the natural images. (c) Our analysis reveals that pre-trained CLIP does not work well on masked images.

object with the highest overlap, and assign the object label to this extracted region. This model, despite imperfect region proposals, reaches a significantly higher mIoU of 66.5%.

This analysis clearly shows that pre-trained CLIP can *not* perform satisfactory classification over masked images, and it is the performance bottleneck of two-stage open-vocabulary segmentation models. We hypothesize that this is caused by the significant domain gap between masked images and CLIP's training images. CLIP is pre-trained on natural images with minimal data augmentation (Radford et al., 2021). On the other hand, mask proposals are cropped and re-sized from original images, and are further corrupted by noisy segmentation masks, see examples in Figure 1 (b).

To address this, we propose to adapt CLIP by finetuning it on masked images and corresponding text labels. One direct solution is to use segmentation labels, *e.g.*, from the COCO-stuff dataset. However, this leads to bad generalization to unseen classes (Section 4.3.1). Such manually annotated masks are accurate but classes are limited to a closed set (*e.g.*, 171 classes for COCO-stuff). We hypothesize that the lack of text diversity causes the finetuned CLIP to lose the generalization ability to open vocabulary concepts. Instead, we collect training data by mining an existing image-caption dataset (*e.g.*, COCO Captions). Given an image-caption pair, we first extract nouns in the caption, and generate class-agnostic masked region proposals using a pre-trained segmentation model. Then, with a pre-trained CLIP model, we assign the best-matching proposal to each extracted noun. By learning from this weakly-supervised alignments between masked images and novel categories, the adapted CLIP better retains its generalization ability for open vocabulary classification.

The next question is how to effectively finetune CLIP? The most notable difference between a masked image and a natural image is that background pixels in a masked image are masked out, leading to many blank areas, which will be converted to "zero tokens" when feeding to CLIP transformers. Such tokens not only contain no useful information, but also bring domain distribution shift to the model (since such tokens don't exist in natural images) and cause performance degradation. To mitigate this, we propose mask prompt tuning, á la visual prompt tuning (Jia et al., 2022). When tokenizing a masked image, we replace the "zero tokens" with learnable prompt tokens. During finetuning, we either train prompts only and freeze CLIP's weights, or train both of them. We find that mask prompt tuning alone significantly improves CLIP's performance on masked images. This is a crucial property for multi-task scenarios where we cannot change CLIP's weight since it is shared with other tasks. When combined with full model finetuning, mask prompt tuning can further improve the performance by a non-trivial margin (see Section 4.3.2).

In our experiments, we measure the open-vocabulary segmentation performances on holdout segmentation datasets in a "zero-shot" manner – we do not adapt the model for each evaluation dataset. We train our model using COCO-stuff (Caesar et al., 2018) dataset with captions from Chen et al. (2015), and test on challenging ADE20K (A-150, A-847 for 150/846 categories) (Zhou et al., 2019), Pascal Context (PC-59, PC-459 for 59/459 categories) (Mottaghi et al., 2014) and PASCAL VOC (PAS-20) (Everingham et al., 2010). Our best model achieves 29.6% mIoU on A-150, which is +8.5% than the state-of-the-art OpenSeg (Ghiasi et al., 2021) under the same setting. On more challenging A-847 and PC-459, our model sets up a new state-of-the-art of 9.0%, 12.4% mIoU, surpassing the previous best solution by +2.7% and 3.4%. Moreover, for the first time, we show open-vocabulary *generalist* models can match the performance of supervised *specialist* models (Long et al., 2015; Chen et al., 2017; Zoph et al., 2020) without dataset specific adaptations.

In summary our contributions include: (1) Our analysis reveals the pre-trained CLIP does *not* perform well on mask proposals, making it the performance bottleneck of two-stage approaches. (2) We collect diverse mask-category pairs from captions to adapt CLIP for masked images and retain its generalization ability. (3) We propose mask prompt tuning specifically for masked image adaptation. This method does not change CLIP's weight, enabling multi-task weight sharing. (4) For the first time, we show open-vocabulary *generalist* models can match the performance of supervised *specialist* models without dataset specific adaptations.

2 Related Work

Pre-trained vision-language models (Radford et al., 2021; Jia et al., 2021; Li et al., 2021) connect the visual concepts with textual description. Pre-trained CLIP (Radford et al., 2021) has strong open-vocabulary classification ability, *i.e.*, classifying an image with arbitrary categories described by language. Pre-trained CLIP has empowered many computer vision tasks with the language ability, such as image manipulation (Patashnik et al., 2021), image generation (Crowson et al., 2022), object detection (Gu et al., 2021; Zhong et al., 2022) and image segmentation (Li et al., 2022; Xu et al., 2021; Ding et al., 2022; Ghiasi et al., 2021). Our work is similar with RegionCLIP (Zhong et al., 2022), which adapts CLIP for object detection by finetuning on region proposals. Our method differs from RegionCLIP since (1) we adapt CLIP to process masked images wile RegionCLIP process complete region crops; (2) We leverage blank areas in masked images and propose mask prompt tuning, which adapts CLIP without changing its weights. This enables sharing CLIP's weight with other tasks in multi-task scenarios. This is not supported by RegionCLIP.

Open-vocabulary segmentation aims to understand an image with arbitrary categories described by texts. Pioneering work ZS3Net (Bucher et al., 2019) uses generative models to synthesize pixel-level features by word embeddings of unseen class. SPNet (Xian et al., 2019) utilizes the word embeddings, *e.g.*, word2vec (Mikolov et al., 2013), to align the semantic meaning with visual features. More recently, researchers propose to leverage the pre-trained CLIP (Radford et al., 2021) for open-vocabulary semantic segmentation. LSeg (Li et al., 2022) aligns pixel embeddings to the text embedding of the corresponding semantic class, which is generated by CLIP's text encoder. Unlike pixel-level LSeg, OpenSeg (Ghiasi et al., 2021) proposes to align the segment-level visual features with text embedding via region-word grounding. Our approach falls into the category of two-stage approaches, such as ZSSeg (Xu et al., 2021) and ZegFormer (Ding et al., 2022): they first generate class-agnostic mask proposals and then utilize pre-trained CLIP to perform open-vocabulary classification. Unlike ZSSeg and ZegFormer which directly use the original CLIP for masked image classification, we propose to adapt CLIP to achieve better results.

Prompt tuning is a strategy to adapt large-scale pre-trained models to new tasks. The idea originated from natural language processing (Liu et al., 2021a; Li & Liang, 2021; Lester et al., 2021), and recent work extends prompt tuning to computer vision. CoOp (Zhou et al., 2022) pre-appends the category words with learnable vectors to adapt CLIP for many recognition tasks. The textual prompt tuning is also widely used in open-vocabulary object detection (Du et al., 2022) and semantic segmentation Xu et al. (2021). Our mask prompt tuning is more relevant to prompt tuning in the visual domain (Bahng et al., 2022; Jia et al., 2022) where learnable vectors are applied to the image domain. Unlike visual prompt tuning (Jia et al., 2022) that inserts additional tokens before the actual image tokens, we *replace* masked tokens with learnable prompts. Furthermore, mask prompt tuning brings additional improvement over a fully finetuned model (Section 4.3.2). Such additional improvements have not been reported by prior work.

3 Method

In this section, we first revisit the two-stage open-vocabulary segmentation methods (Xu et al., 2021; Ding et al., 2022). Then we discuss how to obtain a dataset of mask-category pairs to finetune CLIP. Last, we discuss the proposed mask prompt tuning technique to adapt CLIP for masked images.

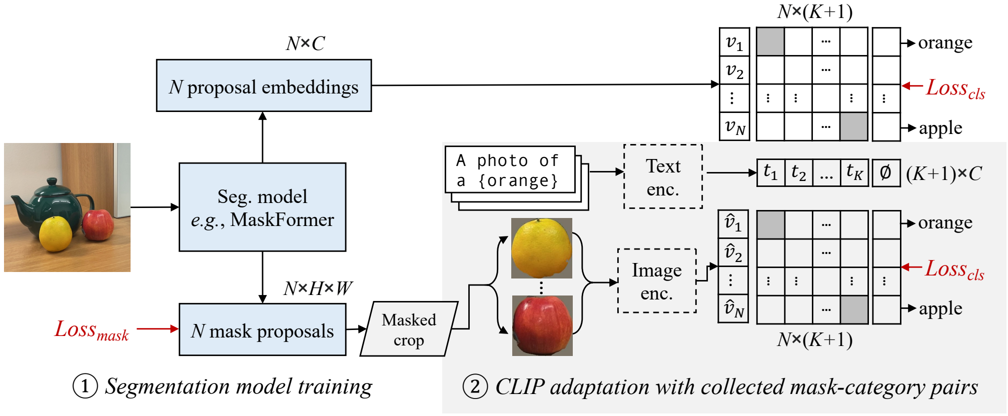


Figure 2: Two-stage approaches consist of one segmentation model, *e.g.*, MaskFormer, and one CLIP model. Firstly, the modified MaskFormer is trained with CLIP's text embeddings so as to perform open-vocabulary segmentation. (Section 3.1). We then use the pre-trained segmentation model to generate class-agnostic proposals and align proposals with extracted nouns from corresponding captions (Section 3.2). After collecting diverse mask-category pairs, we finetune CLIP with the proposed mask prompt tuning (Section 3.3).

3.1 TWO-STAGE MODELS FOR OPEN-VOCABULARY SEMANTIC SEGMENTATION

Our two-stage open-vocabulary semantic segmentation model is shown in Figure 2. It consists of a segmentation model that generates mask proposals, and an open vocabulary classification model.

Following Xu et al. (2021); Ding et al. (2022), we choose MaskFormer (Cheng et al., 2021) as the segmentation model. Unlike per-pixel segmentation models (Long et al., 2015; Chen et al., 2017), MaskFormer predicts a set of N mask proposals and corresponding class predictions. Each proposal is represented by an $H \times W$ binary mask, indicating the location of the target object. The class prediction is a C-dimensional distribution, where C is the number of classes in the training set. Following (Xu et al., 2021), we modify MaskFormer such that for each mask, it generates a C-dimensional proposal embedding, where C is the embedding dimension of a CLIP model (512 for ViT-B/16 and 768 for ViT-L/14). This change allows MaskFormer to perform open-vocabulary segmentation. Specifically, suppose we would like to classify the mask to K categories, we can first use a CLIP model's text encoder to generate K text embeddings for each class as $\{t_k | t_k \in \mathbf{R}^C\}_{k=1,\cdots,K}$. Next, we compare each mask embedding v_i with the text embedding, and predict the class-k probability as $p_{i,k} = \exp(\sigma(v_i, t_k)/\tau)/\sum_k(\exp(\sigma(v_i, t_k)/\tau))$. Here $\sigma(\cdot,\cdot)$ denotes the cosine similarity between two embedding vectors, and τ is the temperature coefficient (Radford et al., 2021). We train the modified MaskFormer on the COCO-Stuff dataset (Caesar et al., 2018) with 171 classes. We use CLIP's text encoder to process class names to generate text embeddings. We also append a learnable embedding $\emptyset$ to represent the category of "no object". For other training settings, we follow the original MaskFormer (Cheng et al., 2021).

Note that the mask proposal generator trained this way is not strictly "class-agnostic", as the definition of an object is determined by the class definitions in the training set. For example, if the training set only contains "person" as a class, it is not likely the model will automatically segment a person into "face", "hand", "body", or finer body parts. How to train a general and class agnostic model to generate mask proposals is an important topic but is beyond the scope of this paper.

In addition to MaskFormer's prediction, following (Ding et al., 2022; Xu et al., 2021), we add a parallel prediction branch using CLIP. MaskFormer generates mask proposals $\{M_i | M_i \in \{0,1\}^{H\times W}\}_{i=1,\cdots,N}$ where 1 and 0 denotes foreground and background. For each mask, we select a tight bounding box that includes all foreground pixels, crop the image, mask out backgrounds, and re-size to CLIP's resolution. We feed mask proposal-i to CLIP and compute class-k probability as $\hat{p}_{i,k}$. We ensemble both predictions to compute final prediction as $p_{i,k}^{(1-\lambda)} * \hat{p}_{i,k}^{\lambda}$ where $\lambda \in [0,1]$. We fuse mask-wise predictions to semantic segmentation using MaskFormer's fusion module.

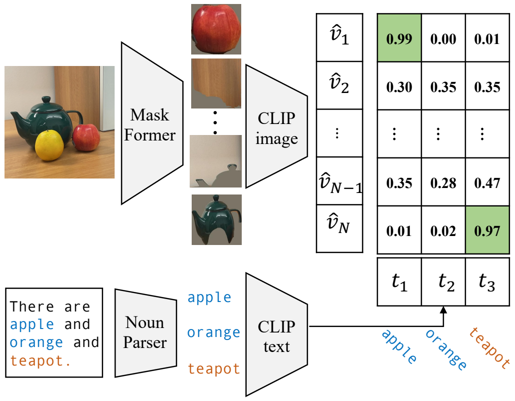
As discussed in Section 1 and Figure 1 (c), our analysis show that CLIP does *not* work well on such masked images. Specifically, CLIP is trained on natural images with little data augmentation (Radford et al., 2021). However, masked images as shown in Figure 1 (b) contain a lot of "blank regions". Such a significant domain gap makes it difficult for CLIP to transfer its classification performance. We also tried cropping the proposals without masking out background pixels. However, we observe worse performance (see Appendix A.1). We conjecture that keeping background pixels makes it more confusing for CLIP to correctly classify the foreground.

3.2 Collecting diverse mask-category pairs from captions

To adapt CLIP to better process masked images, we propose to finetune CLIP on a dataset consists of masked image and text pairs. One direct solution is to leverage manually annotated segmentation labels, *e.g.*, from COCO-Stuff. Such labels are accurate, but have a closed set of categories. We try this solution and collect 965K mask-category pairs spanning 171 classes (*e.g.*, banana, orange) from COCO-Stuff. Then we finetune the CLIP's image encoder, while freezing the text encoder, following Zhong et al. (2022). However, we observe that this naive approach limits the generalization ability of CLIP, as the performance drops if there are more unseen classes (see Section 4.3.1). We hypothesize that due to the limited text vocabulary, the finetuned CLIP over-fits to the 171 classes, losing the ability to generalize to unseen categories.

Compared with segmentation labels, image captions contain much richer information about images and involve a much larger vocabulary. For example, in Figure 3, the image caption is "`There are apple and orange and teapot.`". Though "`apple`" and "`orange`" are valid classes in COCO-Stuff, other concepts are not valid classes and are ignored.

Based on this observation, we designed a self-labeling strategy (Ghiasi et al., 2021; Zhong et al., 2022) to extract mask-category pairs. As in Figure 3, given an image, we first use a pre-trained MaskFormer to extract masked proposals. Meanwhile, from the corresponding image caption, we extract all nouns using an off-the-shelf language parser (Bird et al., 2009), and treat them as potential classes. Then, we use CLIP to pair the most matching mask proposal to each class. From COCO-Captions Chen et al. (2015), we collect 1.3M mask-category pairs with 27K unique nouns using 5 captions per image, or 440K pairs with 12K nouns using 1 caption per image. Experiments show this noisy but diverse mask-category dataset leads to significantly better performance than manual segmentation labels (see Section 4.3.1).



$\hat{v}_1$	0.99	0.00	0.01
$\hat{v}_2$	0.30	0.35	0.35
⋮	⋮	⋮	⋮
$\hat{v}_{N-1}$	0.35	0.28	0.47
$\hat{v}_N$	0.01	0.02	0.97
	t_1	t_2	t_3

Figure 3: For the given image-cation pair, only "`apple`" and "`orange`" are categories in COCO. By extracting nouns from captions, we can also get a novel "`teapot`" category.

3.3 Mask prompt tuning

After collecting the dataset, a natural question is how to finetune CLIP effectively? The most notable difference between a masked image and a natural image is that background pixels in a masked images are set to zeros, leading to many "blank areas". When feeding masked images to CLIP, images will be divided into non-overlapping patches and subsequently tokenized. Those blank areas will then become zero tokens. Such tokens not only contain no useful information, but also bring domain distribution shift to the model (since such tokens don't exist in natural images) and cause performance degradation. To mitigate this, we propose a technique called *mask prompt tuning*, à la visual prompt tuning (Jia et al., 2022). Specifically, when feeding into CLIP, a masked image will be tokenized to a tensor $T \in \mathbf{R}^{N_p \times E}$, where N_p is the number of patches, and E is the token dimension. The masked image also comes with a condensed binary mask $M_p \in \{0,1\}^{N_p}$, where each element indicating whether a given patch is kept of masked out. We allocate a learnable tensor representing prompt tokens as $P \in \mathbf{R}^{N_p \times E}$. Finally, the final input to the transformer is computed as $T \otimes M_p + P \otimes (1 - M_p)$, where $\otimes$ denotes element-wise multiplication. Following the "deep prompts" setting in Jia et al. (2022), we can add such prompt tokens to deeper layers of the transformer. This is also illustrated in Figure 4.

Compared with fully finetuning the entire model (Zhong et al., 2022), mask prompt tuning has several advantages. First, it is specifically designed for segmentation tasks, where parts of input

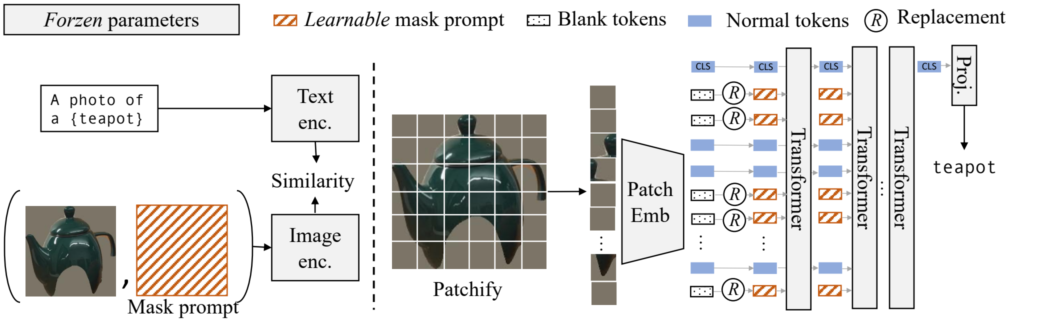


Figure 4: The proposed mask prompt tuning can adapt CLIP to masked images without changing its weights. We replace the zero tokens from masked patches to learnable mask prompt.

images are masked. Next, compared with full model finetuning, the amount of trainable parameters in mask prompt tuning is orders of magnitude smaller, leading to much better training efficiency. Moreover, as a foundational model, CLIP may be simultaneously used for many tasks, and we may not be allowed to tune CLIP's weights. Mask prompt tuning does not require changing weights of CLIP, thus is suitable for such multi-task scenarios. Lastly, our experiments show that mask prompt tuning alone leads to significant improvement. And if applied together with full model finetuning, it can further improve the open-vocabulary segmentation performance (Section 4.3.2).

4 Experiments

4.1 Experimental setup

Training Dataset We train our model on the COCO dataset (Lin et al., 2014). We first train the modified MaskFormer using the segmentation labels from COCO-Stuff (Caesar et al., 2018). Next, we finetune CLIP on the mask-category dataset that we obtained from COCO Captions (Chen et al., 2015). There are 118k training images labeled with 171 valid categories in the dataset, ranging from things (*e.g.*, orange, car) to stuffs (*e.g.*, sky, road). If not specified otherwise, we use all the 171 categories data during training.

Evaluation Dataset Our open-vocabulary model is able to perform zero-shot segmentation on arbitrary datasets without dataset-specific adaption. Thus, we test our model on challenging ADE20K (Zhou et al., 2019), Pascal VOC (Everingham et al., 2010) and Pascal Context (Mottaghi et al., 2014) datasets. ADE20K is a densely pixel-wise annotated dataset for scene understanding, which spans diverse annotations of indoor and outdoor scenes. There are 2K images in its validation set. We choose two versions of categories, one with 150 frequently used categories (A-150) and one with more diverse 847 categories (A-847). Pascal VOC is a classical dataset for segmentation. We evaluate on the 1.5K validation images with 20 categories (PAS-20). Pascal Context is a set of additional annotations for PASCAL VOC 2010. It goes beyond the original PASCAL semantic segmentation task by providing annotations for the whole scene. There are 5K images in its validation set. We also choose two versions of categories, one with 59 frequently used categories (PC-59) and one with the whole 459 categories (PC-459).

Implementation Details As indicated before, our model consists of two part: one segmentation model based on MaskFormer (Cheng et al., 2021) and one mask-adapted CLIP model (Radford et al., 2021). For the segmentation model, we have two backbone choices, ResNet-101c (Chen et al., 2017) and Swin-Base (Liu et al., 2021b). For the CLIP model, we have two choices: ViT-B/16 and ViT-L/14 (Dosovitskiy et al., 2020). We detail our largest model setting here, while the training recipe of the R101c model can be found in Appendix A.2. For Swin-Base segmentation model, the backbone weights are initialized from an ImageNet-21K pre-trained model. We use AdamW (Loshchilov & Hutter, 2017) optimizer with the poly learning rate schedule (Chen et al., 2017). The initial learning rate and weight decay are set to $6 \cdot 10^{-5}$ and 10^{-2}, respectively. We use a crop size of 640×640, a batch size of 32 and train the model for 120K iterations. For data augmentations and other hyper-parameters, we mainly following the setting of Cheng et al. (2021).

For adapting CLIP ViT-L/14 model, we utilize the OpenCLIP (Ilharco et al., 2021) implementation. After collecting 440K mask-category pairs from captions (see Section 3.2), we propose three ways

Table 1: The mIoU results of our model and previous open-vocabulary generalist models and supervised specialist models. Results for SPNet and ZS3Net on PAS-20 are reported from Li et al. (2022). Results for ZegFormer on PAS-20 are recalculated by us. Under the R101c model scale, our model significantly outperforms other open-vocabulary models. Our large Swin-Base model can match the performance of some supervised specialist models.

method	backbone	training dataset	A-847	PC-459	A-150	PC-59	PAS-20
Open-vocabulary generalist models.							
SPNet (Xian et al., 2019)	R-101	PASCAL-15	-	-	-	24.3	18.3
ZS3Net (Bucher et al., 2019)	R-101	PASCAL-15	-	-	-	19.4	38.3
LSeg (Li et al., 2022)	R-101	PASCAL-15	-	-	-	-	47.4
LSeg+ (Ghiasi et al., 2021)	R-101	COCO Panoptic	2.5	5.2	13.0	36.0	59.0
SimBaseline (Xu et al., 2021)	R-101c	COCO-Stuff-156	-	-	15.3	-	74.5
ZegFormer (Ding et al., 2022)	R-50	COCO-Stuff-156	-	-	16.4	-	80.7
OpenSeg (Ghiasi et al., 2021)	R-101	COCO Panoptic	4.0	6.5	15.3	36.9	60.0
OVSeg (Ours)	R-101c	COCO-Stuff-156	7.0	10.4	24.0	51.7	89.2
OVSeg (Ours)	R-101c	COCO-Stuff-171	**7.1**	**11.0**	**24.8**	**53.3**	**92.6**
LSeg+ (Ghiasi et al., 2021)	Eff-B7	COCO Panoptic	3.8	7.8	18.0	46.5	-
OpenSeg (Ghiasi et al., 2021)	Eff-B7	COCO Panoptic	6.3	9.0	21.1	42.1	-
OVSeg (Ours)	Swin-B	COCO-Stuff-171	**9.0**	**12.4**	**29.6**	**55.7**	**94.5**
Supervised specialist models.							
FCN (Long et al., 2015)	FCN-8s	Same as test	-	-	29.4	37.8	-
Deeplab (Chen et al., 2017)	R-101	Same as test	-	-	-	45.7	77.7
SelfTrain (Zoph et al., 2020)	Eff-L2	Same as test	-	-	-	-	90.0
MaskFormer (Cheng et al., 2021)	R-101c	Same as test	17.4	-	46.0	-	-

to adapt CLIP: mask prompt tuning (MPT) only, full model fine-tuning (FT) only and joint MPT + FT. For MPT only, we initialize the CLIP model with official OpenAI weights (Radford et al., 2021) and the learnable tokens are randomly initialized. We also use the deep prompts as proposed in Jia et al. (2022). The prompt depth is set to 3 if not specified otherwise. The training optimizer is AdamW with initial learning rate $2 \cdot 10^{-2}$ and weight decay 0. The cosine annealing scheduler is adopted to adjust the learning rate. The model is trained with input size of 224×224, a batch size of 256 for 5 epochs. For FT only, we keep similar training procedure but with a much lower learning rate $5 \cdot 10^{-6}$ and larger weight decay 0.2. For MPT + FT, we first initialize the CLIP with fully finetuned model and then apply the mask prompt tuning over it, which we fined more stable and effective (see Appendix A.3). All other hyper-parameters are the same with MPT only. The text encoder of CLIP is frozen in all our experiments.

4.2 Main results on open vocabulary semantic segmentation

OVSeg achieves best performance among open-vocabulary models. We conduct the comparison with other open-vocabulary generalist models using the common ResNet-101 (R-101) model scale in Table 1. If not specified otherwise, our best performance is achieved using joint mask prompt tuning and fine-tuning (see Section 4.3.2). First of all, compared with per-pixel approaches (SPNet (Xian et al., 2019), ZS3Net (Bucher et al., 2019), LSeg (Li et al., 2022) and LSeg+ (Ghiasi et al., 2021)), proposal-based approaches (OpenSeg (Ghiasi et al., 2021), SimBaseline (Xu et al., 2021) and ZegFormer (Ding et al., 2022)) show better performance. Our OVSeg also falls into the proposal-based category. Compared with other proposal-based approaches, our model shows significant improvements across all five benchmarks. In particular, our R101c model achieves 7.1% and 11.0% mIoU on challenging A-847 and PC-459, which even performs better than the EfficientNet-B7 based OpenSeg model. All proposal-based approaches are using the same COCO images, *i.e.*, the 2017 splits with 118K images, but with different annotations. Our experiments show different annotations result in relatively small performance differences: we only observe a $\sim$1% performance drop on A-150 when changing COCO-Stuff-171 to COCO-Stuff-156.

Largest OVSeg model sets up new SOTA results on zero-shot benchmarks. When we scale up the model, our method can further achieve better results. With Swin-Base (Swin-B) backbone and CLIP ViT-L/14, our model can achieve 29.6% and 55.5% mIoU on A-150 and Pascal PC-59, which is +8.5% and +13.6% higher than the SOTA zero-shot results. On the challenging A-847 and PC-459, our model sets up a new zero-shot state-of-the-art 9.0% and 12.4% mIoU. We further detail the class-wise IoU of A-150 categories in Appendix A.4.

Table 2: Ablation on mask-category pairs. The baseline is MaskFormer Swin-Base with original CLIP ViT-L/14. The masks come from ground-truth (GT) or generated proposals. The category nouns come from ground-truth (GT) classes or captions. We also calculate the statistics (number of pairs and unique nouns) of collected pairs.

Case	Source		Statistics		A-847	A-150	PC-59
	Mask	Category	Pairs	Unique nouns			
Baseline	-	-	-	-	7.3	21.8	51.4
(1)	GT	GT	965K	171	5.3 (-2.0)	23.0 (+1.2)	**57.3** (+5.9)
(2)	GT	1 caption	440K	12K	7.9 (+0.6)	24.2 (+2.4)	53.2 (+1.8)
(3)	proposals	1 caption	440K	12K	**8.8** (+1.5)	**28.8** (+7.0)	55.7 (+4.3)
(4)	proposals	5 captions	1.3M	27K	**8.8** (+1.5)	28.6 (+6.8)	55.5 (+4.1)

Table 3: Ablation on mask prompt tuning. The baseline is MaskFormer Swin-Base with CLIP ViT-L/14. We report the zero-shot mIoU on representative ADE-847, ADE-150 and PC-59 datasets. All the improvements are measured upon the baseline model.

case	Finetuning method		A-847	A-150	PC-59
	mask prompt	full model			
Baseline			7.3	21.8	51.4
(a)	✓		8.4 (+1.1)	26.5 (+4.7)	55.4 (+4.0)
(b)		✓	8.8 (+1.5)	28.8 (+7.0)	**55.7** (+4.3)
(c)	✓	✓	**9.0** (+1.7)	**29.6** (+7.8)	**55.7** (+4.3)

Open-vocabulary generalist models can match supervised specialist models in 2017. We show our generalist model can achieve competitive performance without the need of any dataset specific training. On the challenging A-150, our model achieves similar performance with fully supervised FCN-8s (Long et al., 2015). On the PAS-20, our model achieves 94.5% mIoU, which is even +4.5% than the SOTA specialist model (Zoph et al., 2020). We note OVSeg is not directly comparable with supervised models because OVSeg is not trained on evaluation datasets. OVSeg also has different backbones and segmentation model architectures. Thus, comparison with supervised models is for reference purposes only. Our generalist model still underperforms the advanced specialist models, such as supervised MaskFormer (Cheng et al., 2021).

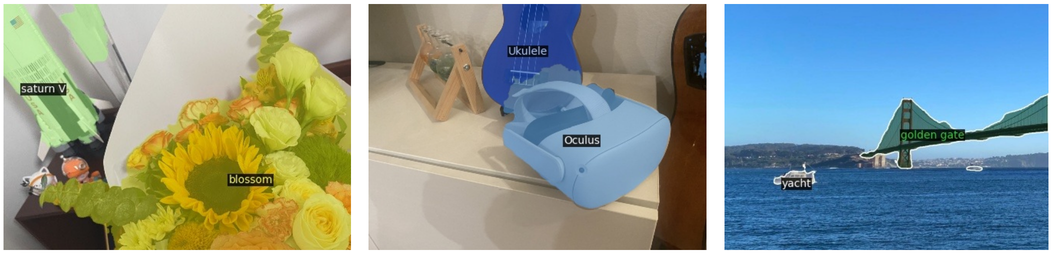
4.3 Ablation Study

4.3.1 Collecting mask-category pairs

We discuss the impact of finetuning data in Table 2. The baseline model is MaskFormer Swin-Base with the original CLIP ViT-L/14. Our initial trial (case (1)) is collecting ground-truth (GT) masks with supervised GT classes. We can collecting 965K mask-category pairs with 171 unique nouns (the number of classes defined in COCO-stuff). Then we finetune the CLIP model with the collected pairs. We observe a -2.0% performance drop on the A-847 dataset. This is because the adapted CLIP is over-fitting to the 171 GT classes. Although the model achieves good results on PC-59 (whose categories are highly overlapped with COCO-Stuff), it perform badly for more diverse concepts in A-847. As detailed in Section 3.2, we propose to utilize captions (Chen et al., 2015) to collect diverse mask-category pairs. After parsing the nouns in the caption, we match the nouns with GT masks (case (2)) or proposals (case (3)) generated by the baseline model. By replacing the GT masks with proposals, the A-150 mIoU is significantly improved (from 24.2% to 28.8%) We conjecture that many regions are not labeled as GT masks (see examples in Figure 3), and are therefore ignored. In contrast, the generated proposals (usually 100) can cover most of regions-of-interest in the image, leading to better performance. If all the 5 captions per image are used (case (4)), we observe a mild -0.2% degradation on A-150 and PC-59 We hypothesis that 12K nouns are adequate for the CLIP to retain its open-vocabulary ability. Thus, we use 1 caption per image as our default setting.

4.3.2 Mask prompt tuning

We ablate the effect of mask prompt tuning in Table 3. The baseline model is MaskFormer Swin-Base with CLIP ViT-L/14. If we only use mask prompt tuning (case (a)), our model outperforms



Query: saturn V, blossom Query: Oculus, Ukulele Query: golden gate, yacht

Figure 5: Open-vocabulary segmentation with user-defined queries. Our model accurately segments unseen categories, such as the Saturn V rocket, Oculus headset, and Golden gate bridge.

the baseline by a large +4.7% and +4.0% mIoU improvement on ADE-150 and PC-59, respectively. Case (b) shows the result of full model fine-tuning. Although it achieves the best accuracy, the trainable parameters are orders of magnitude higher. In contrast, the proposed mask prompt tuning only modifies the input without changing CLIP's weight. Furthermore, mask prompt tuning can further improve over a fully finetuned model, as shown in case (c). Case (c) achieves 29.6% mIoU ADE-150, which further improves the fully finetuned model by a considerable margin of +0.8%.

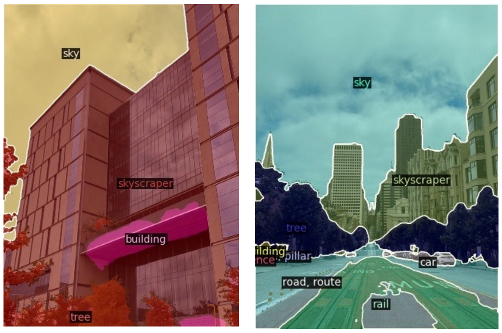
4.4 Discussions

4.4.1 Segmentation with user-defined queries.

Our method allows users to define arbitrary queries and search the query in the image, see Figure 5. Without training our models to learn specific concepts, our model can locate and segment Saturn V as the lego rocket, Oculus as the VR headset, and golden gate as the bridge in corresponding images. This demonstrates the strong potentials of open vocabulary semantic segmentation.

4.4.2 Ambiguity of open vocabulary segmentation evaluation

We show some "failure" predictions from the A-150 dataset in Figure 6. For the left figure, the ground-truth category is "building" while our model predicts "skyscrapers". The "skyscrapers" is a reasonable description, but the standard A-150 evaluation protocol will treat it as a wrong prediction. A similar case happens in the right figure, the ground-truth "rail" is recognized as "road". This is caused by the fact that language defined categories are ambiguous and can overlap with each other. Designing a better evaluation metric for open-vocabulary segmentation models is an important topic for our future research. Note that due to IP constraints, we use our own images, instead of ADE20K images in Figure 6. But this phenomenon widely exists on ADE20K images.



GT: building Pred: skycraper GT: rail Pred: road

Figure 6: Ambiguity of the class definition.

5 Conclusion

This paper studies open-vocabulary semantic segmentation where the model segments an image by arbitrary categories described by texts. We identify the performance bottleneck of current two-stage methods to be the pre-trained CLIP, since it doesn't perform well on masked images. We propose to adapt CLIP for masked images. To retain CLIP's open-vocabulary classification ability, we adapt CLIP with diverse mask-category pairs mined from image-caption dataset. We further propose mask prompt tuning, a method can adapt CLIP without changing its original weights. The proposed model is general and can do zero-shot segmentation on arbitrary datasets without dataset-specific adaption. For the first time, we showopen-vocabulary generalist models can match the performance of supervised specialist models.

Ethics Statement

We only use the public computer vision datasets (COCO, ADE20K, Pascal) and leverage the open-sourced vision-language models (CLIP) for our experiments. To the best of our knowledge, we do not foresee our approach as being inherently subject to concerns of discrimination / bias / fairness, inappropriate potential applications, impact, privacy and security issues, research integrity or research practice issues. However, the public datasets and pre-trained models may be subject to bias that may be inherited by models trained with our approach.

Reproducibility Statement

Our code is reproducible and can be implemented based on the method description in Section 3 as well as training details in Section 4.1.

References

H Bahng, A Jahanian, S Sankaranarayanan, and P Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, pp. 2022, 2022.

Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 32, 2019.

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

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4): 834–848, 2017.

Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

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.

Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. Vqgan-clip: Open domain image generation and editing with natural language guidance. *arXiv preprint arXiv:2204.08583*, 2022.

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*, pp. 11583–11592, 2022.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. Learning to prompt for open-vocabulary object detection with vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14084–14093, 2022.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338, 2010.

Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Open-vocabulary image segmentation. *arXiv preprint arXiv:2112.12143*, 2021.

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.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL `https://doi.org/10.5281/zenodo.5143773`. If you use this software, please cite it as below.

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*, pp. 4904–4916. PMLR, 2021.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*, 2022.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. *arXiv preprint arXiv:2201.03546*, 2022.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021a.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021b.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 891–898, 2014.

Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2085–2094, 2021.

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*, pp. 8748–8763. PMLR, 2021.

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*, pp. 8256–8265, 2019.

Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model. *arXiv preprint arXiv:2112.14757*, 2021.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2881–2890, 2017.

Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16793–16803, 2022.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019.

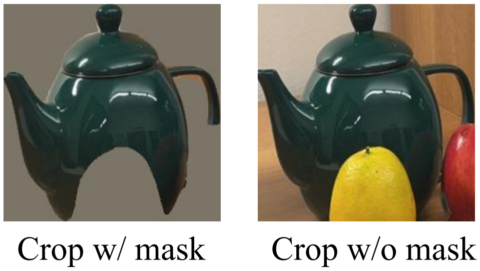
Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking pre-training and self-training. *Advances in neural information processing systems*, 33: 3833–3845, 2020.

A Appendix

A.1 Crop with or without mask

In the paper, we use the default crop with mask (see the left of Figure 7). We also try the direct crop without mask (see the right of Figure 7). Following the bottleneck analysis in Figure 1, we feed the unmasked crops a pre-trained CLIP for classification. This experiment gives a 13.8% mIoU, which is -6.3% worse than using the masked crops. We hypothesize that the crop with mask introduces many background pixels, making the prediction more difficult. For the example in the right of Figure 7, the "orange" will also be an appropriate category for the unmasked crop.



Crop w/ mask Crop w/o mask

Figure 7: Crop without mask will introduce background pixels, making the prediction more difficult.

We note in ZegFormer (Ding et al., 2022) has also done a ablation study about different strategies to obtain final crop. We have the similar conclusions.

A.2 Training hyperparams of R101c model

Our small model is MaskFormer R101c with CLIP ViT-B/16. For MaskFormer training, the backbone weights are initialized from an ImageNet-1K pre-trained model. We use AdamW optimizer with the poly learning rate schedule. The initial learning rate and weight decay are set to $2 \cdot 10^{-4}$ and 10^{-4}, respectively. We also use a learning rate multiplier 0.1 on the backbone. We use a crop size of 512×512, a batch size of 32 and train the model for 120K iterations. For data augmentations and other hyper-parameters, we majorly following the setting of Cheng et al. (2021). For adapting CLIP ViT-B/16 model, we basically follow the hyperparameters of finetuning ViT-L/16 except we use a larger batch size 1024.

A.3 More ablation studies on mask prompt tuning

Table 4: Ablation on combining mask prompt tuning (MPT) and fine-tuning (FT). FT ->MPT indicates first FT and then MPT, and vice versa. FT + MPT sim. means optimizing prompts and CLIP simultaneously.

combination	A-847	A-150
FT ->MPT (default)	**9.0**	**29.6**
MPT ->FT	8.5 (-0.5)	28.1 (-1.5)
FT + MPT sim.	8.8 (-0.2)	29.0 (-0.6)

Table 5: Ablation on prompt depth. We test with and without fully fine-tuned (FT) model.

prompt depth	A-150	
	w/o FT	w/ FT
1	25.7	29.3
3 (default)	26.5	**29.6**
6	**26.8**	29.4
12	**26.8**	29.3

We explore two other ways to combine mask prompt tuning (MPT) and fine-tuning (FT) as in Table 4. Our default setting (FT ->MPT) is first doing FT and then applying MPT to the already fine-tuned model. We don't change the weights of fine-tuned CLIP. The other option is first doing MPT and then doing FT with fixed mask prompts (MPT ->FT). This combination produces poor results (-1.5% drop on A-150). We conjecture the mask prompts learned with original CLIP provide a bad prior when we fune-tune the entire CLIP model. We also explore learning mask prompts and fine-tune CLIP weight *simultaneously* (FT + MPT sim.). This doesn't bring favorable results.

We further ablate the effects of prompt depth in Table 5. The depth can be selected from $\{1, 3, 6, 12\}$. We use two different scenarios: without fine-tuning (w/o FT) for mask prompt tuning only, with fine-tuning (w/ FT) for applying mask prompt tuning over a already fine-tuned model. For w/o FT case, one layer prompt can bring significant improvement, *e.g.*, from baseline's 21.8% to 25.7%. Deeper prompts result in better performance, because more parameters are introduced with more prompts.

Interestingly, deeper prompts (going from 3 to 12) don't bring further improvement for w/ FT case. We choose prompt depth as 3 for default setting.

A.4 CLASS-WISE IOU OVER SEEN AND UNSEEN CATEGORIES.

We detail the class IoU on all 150 categories in ADE20K-150 (model trained on COCO) in Figure 8, and we annotated seen vs. unseen classes and their IoUs. Seen categories mean there are *similar* categories in COCO-stuff training set. Unseen categories denotes the novel categories in ADE20K. The average IoU of seen and unseen categories are 37.6% and 21.9%, respectively, showing that our model performs better on seen categories. This is also observed in other open vocabulary segmentation work, such as (Ding et al., 2022).

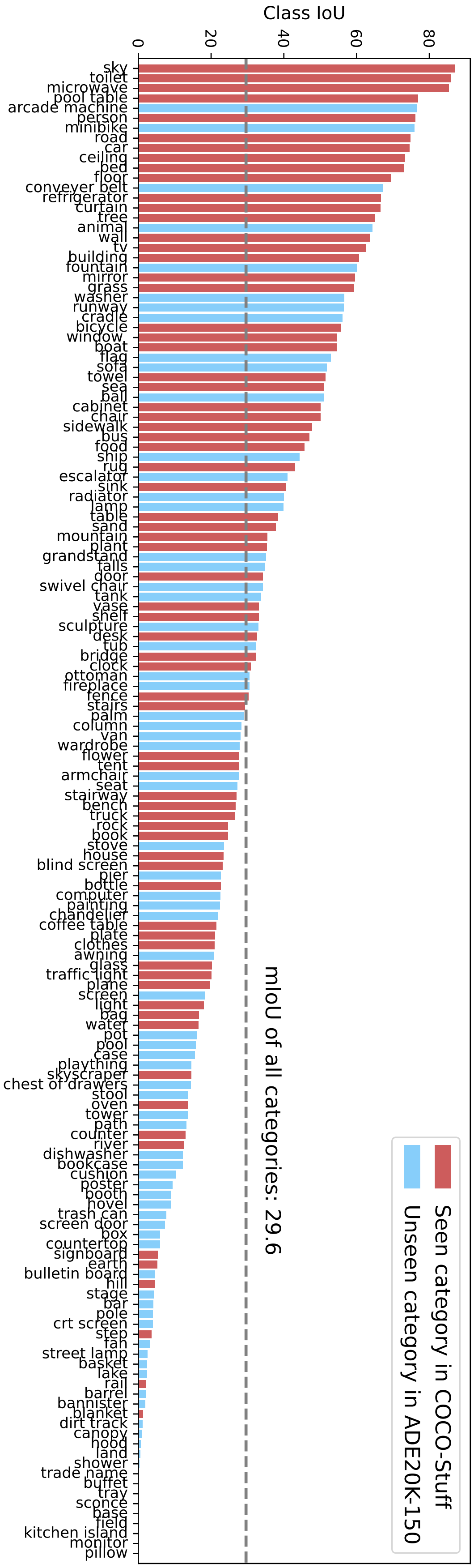


Figure 8: Class IoU on all 150 categories in ADE20K (model trained on COCO). It is expected the model performs better on seen categories in training set.