OpenDAS: Domain Adaptation for Open-Vocabulary Segmentation

Anonymous ECCV 2024 Submission

Paper ID #2410

Abstract. The advent of visual language models (VLMs) has transformed image understanding from fixed classifications to dynamic image-language interactions, enabling open-vocabulary segmentation. Despite their flexibility, VLMs often trail behind closed-set classifiers in accuracy, due to their heavy dataset demands and absence of domain-specific knowledge. Hence, we introduce a new task *domain adaptation for open-vocabulary segmentation*, where we enhance VLMs with domain priors while preserving their open-vocabulary nature. We propose a novel approach that leverages prompt tuning in combination with triplet loss training, to bypass extensive fine-tuning. Our work marks a significant advancement in making VLMs more precise and adaptable, demonstrating improved performance and practicality for real-world applications, achieving +3.5% accuracy in ScanNet++ office subset with unseen classes during training compared to other parameter-efficient learning strategies, as well as on 2D semantic segmentation benchmarks +7.7% on KITTI-360 and +4.1% on ADE20K-150 validation split.

Keywords: Open vocabulary segmentation · Domain adaptation · Prompt Tuning

1 Introduction

The advent of visual-language models (VLMs), such as CLIP [30] and ALIGN [15], have catalyzed a paradigm shift in image understanding. Moving beyond conventional categorization, these models have enabled significant advancements in open-vocabulary segmentation. This approach facilitates precise object localization within images using language cues, leveraging the generalization capabilities of VLMs. However, despite these advancements, current models still suffer in more exotic or specialized domains.

Our research is driven by the transformative impact of VLMs on a range of practical applications, from autonomous home robots such as TidyBot [38] and OK-Robot [28], which are capable of understanding textual commands for object interaction, to systems like Text2Loc [40] and Language Frontier Guide [33] that facilitate text-based localization and navigation in unfamiliar environments. These applications underscore the potential of VLMs to revolutionize how machines understand and interact with the world around them.

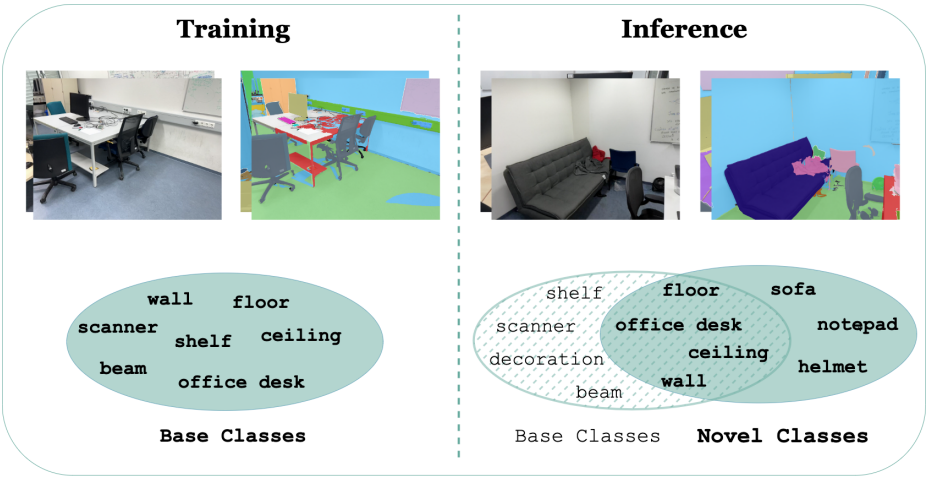


Fig. 1: Domain Adaptation for Open-Vocabulary Segmentation. We introduce a new task, *domain adaptation for open-vocabulary segmentation*. During training, we learn from ground truth masks and only the base classes. During inference, we test our model with visually closer domains where we have some base classes, and novel classes, demonstrating its generalization capabilities while still adapting to the domain.

However, the efficacy of VLMs in tasks like scene segmentation and object identification in response to text prompts remains suboptimal compared to traditional methods that operate within a limited set of categories. A primary obstacle to enhancing the performance of VLMs, like CLIP [30], is their reliance on extensive datasets for learning a comprehensive representation space. Although this contributes to their strong generalization capabilities, it also limits their effectiveness in specialized domains where the task like semantic segmentation becomes more fine-grained, where expert knowledge could offer significant advantages over the general knowledge encoded in existing VLMs.

To overcome these challenges, we introduce a new paradigm, namely "domain adaptation for open-vocabulary segmentation". The scenarios we investigate align closely with the practical applications where open-vocabulary segmentation has shown promising impacts. For instance, it's both reasonable and expected for a home robot to adapt to its environment while still being able to understand and act upon arbitrary language prompts.

We focus on refining the practical and realistic application of open-vocabulary segmentation by adapting a CLIP-based model to better match text and image crops, *independently* of predicting the masks. This approach is informed by findings from SAM [20], which indicate that segmentation yields improved results when it is learned to be category-invariant. Moreover, our work addresses the limitations of current open-vocabulary segmentation models' common pipeline, which often involves classifying segments proposed by another segmentation model [9, 26, 37]. While adapting mask proposal methods to new domains is

relatively straightforward, how to enhance segment classification without compromising generalization capabilities has remained not trivial.

To adapt CLIP into a specific domain, we leverage the prompt tuning strategy, a new approach to adapt VLMs into target domains in a parameter-efficient way. Prior works demonstrate its advantage in image classification [16–19, 22, 48, 49]. We adopt such methods for the first time for the task of 2D segmentation. Besides, we introduce a surprisingly simple yet effective architecture that is rooted in the prior multi-modal prompt tuning works. To further exploit this strategy, we propose combining the classic cross-entropy loss with a triplet loss to adapt VLMs. Compared to only using greedy cross-entropy loss as the optimization objective, our enhanced objective appears to help the model learn new classes while retaining its generalization capabilities across novel, unseen classes. Moreover, further analysis reveals that, even though direct fine-tuning the entire visual encoder can lead to a performance boost in segment classification in a new domain, our prompt learning strategy can effectively augment the zero-shot capabilities of VLMs, demonstrating improved performance on novel classes, while being much more parameter-efficient and faster to train.

With extensive experiments in 3 challenging datasets, from indoor to outdoor scenes, our approach proves to yield not only better performance over other uni-modal and multi-modal parameter-efficient adaptation strategies, but also robust fine-tuning methods, improving its understanding of both base and novel classes. Moreover, besides boosting the segment classification performance given GT masks, we also demonstrate that our method can also be beneficial for the recent open-vocabulary segmentation method in understanding semantic labels, even with their predicted masks.

In summary, our main contributions are as follows:

- We highlight the benefits of a triplet loss in the task of domain adaptation for open-vocabulary segmentation.
- We propose a surprisingly simple yet effective model architecture rooted in prior multi-modal prompt tuning works.
- Our method significantly outperforms previous uni-modal and multi-modal methods and also surpasses the original CLIP model's generalization to novel classes in the new domain.

2 Related Work

2.1 Open-Vocabulary Semantic Segmentation

Open-vocabulary semantic segmentation is the task of localizing objects from language cues in images. Wu et al. [37] suggest that the most common framework for open vocabulary segmentation is to generate class-agnostic masks and visual embeddings and compare them against a zero-shot Visual Language Model (VLM), e.g. CLIP [30] or ALIGN [15] text embeddings. In this direction, LSeg [24] uses CLIP text embeddings and aligns pixel-level features to the corresponding

semantic class. OpenSeg [9] instead aligns segment-level features with text embedding via region-word grounding. Many other approaches rely specifically on CLIP to generate textual embeddings for open-vocabulary segment class matching and encode image or per segment in the same latent space [5, 7, 8, 26, 41, 46]. More recently, PADing [12] proposes a unified framework to tackle zero-shot semantic segmentation, zero-shot instance segmentation, and zero-shot panoptic segmentation. All these approaches require reliable zero-shot segment classification and, hence, even with ground truth masks supplied, the performance cannot improve further than pre-trained VLM's capabilities.

2.2 Domain Adaptation on Visual Language Models

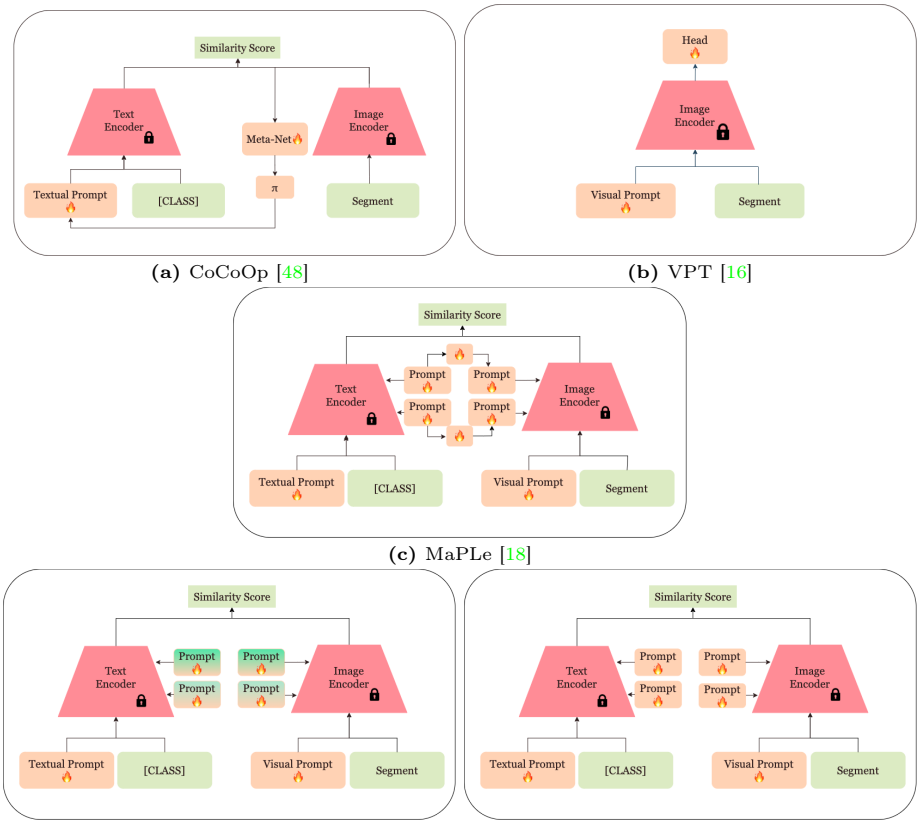
In the evolving landscape of computer vision research, the adaptation of Visual Language Models (VLMs) to new domains through fine-tuning has been a focal point of study. While standard fine-tuning techniques have been shown to significantly enhance accuracy on target distributions, they often come at the cost of reduced model robustness to distribution shifts and can result in sub-optimal performance [21, 35]. Wortsman et al. propose WiSE-FT to address this issue by ensembling the weights of the zero-shot and fine-tuned models to become more robust against distribution shifts [36]. On the other hand, Goyal et al. propose mimicking the pre-training by using contrastive loss for fine-tuning the model to improve performance [10]. Despite their effectiveness, these approaches require comprehensive fine-tuning of the entire model, which is computationally expensive.

Prompt Learning. This technique [11, 23, 25, 29] has emerged as a groundbreaking strategy to overcome the limitations of full model fine-tuning, which enables model adaptation with minimal computational cost. It was initially proposed for LLMs such as BERT [6], GPT [3, 31]. Recently, it was also proposed for VLMs, e.g. CLIP [30], and showed promising results [14, 16–19, 22, 48, 49]. Pioneering works in this direction explore uni-modal prompt learning [16, 48, 49], laying the groundwork for more complex methodologies.

We compare some of the outstanding works in this domain in Fig. 2. In evaluating these developments, we highlight two uni-modal prompt tuning approaches as notable examples, namely CoCoOp [48] and VPT [16]. As shown in Fig. 2a, CoCoOp adds dynamic textual prompts based on image features instead of "A photo of a [CLASS]", while VPT (in Fig. 2b) incorporates learnable visual prompts in addition to a linear probe layer for enhanced adaptability.

More recent works employ multi-modal learning approaches where they train textual and visual prompts jointly and observe an improvement compared to only textual prompt tuning. As presented in Fig. 2c, MaPLe adds coupling between interim textual and visual prompts [18]. On the contrary to this approach, we employ a simpler architecture, where we have separate prompts inserted in the input space as well as interim layers. This allows us to apply a sequential learning approach where we have full control over the adaptation of each encoder.

However, while RPO, shown in Fig. 2d, maintains distinct encoders like our approach, it incorporates attention masks to regulate information flow ex-



(a) CoCoOp [48] **(b)** VPT [16]

(c) MaPLe [18]

(d) RPO [22] **(e)** Our Architecture

Fig. 2: Model Architecture Comparison. We compare our architecture against the existing prompt tuning methods. In all these images, text and image encoders are frozen and learnable prompts are colored with orange and marked with 🔥. CoCoOp (Fig. 2a) adds dynamic prompts based on image features, while VPT (Fig. 2b) only adds learnable prompts to the visual encoder. On the other hand, MaPLe (Fig. 2c) adds prompts to visual and textual encoders combined with a coupling layer. RPO (Fig. 2d) adds prompts separately as well as an attention mask that regulates the information flow from the zero-shot parameters to the learnable prompts.

clusively from model parameters to prompts [22]. Although beneficial for domain generalization, this method inhibits the model's ability to develop domain-specific representations in the interim layers, potentially diminishing domain adaptation effectiveness.

3 Method

Our goal is to adapt pre-trained vision-language models, e.g. CLIP, into the target domain for open-vocabulary segmentation without losing its generaliz-

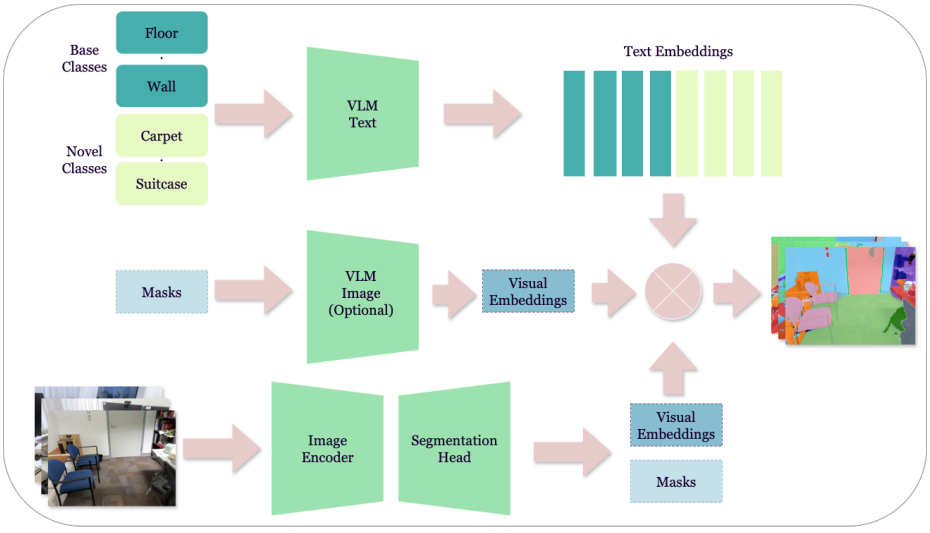


Fig. 3: The Common Pipeline for Open-Vocabulary 2D Segmentation. As described in the survey [37], the pipeline takes a set of images and generates masks with visual embeddings. Optionally, the generated masks are fed into a Visual Language Model (VLM) Image Encoder to generate visual embeddings, which are then combined with the remaining pipeline. In parallel, the VLM Text Encoder takes a set of base and novel classes and produces text embeddings. Subsequently, these visual and text embeddings are compared against each other to predict classes for each mask.

ability. In this section, we first explain the preliminaries required for our method (Sec. 3.1). Next, we propose a simple yet effective way for multi-modal prompt tuning (Sec. 3.2), and introduce a novel training strategy based on a triplet loss (Sec. 3.3). Finally, we discuss how to mine data for the triplet loss (Sec. 3.4).

3.1 Preliminaries

As shown in Fig. 3, the architecture of common open-vocabulary 2D segmentation pipelines typically comprises: a) a class-agnostic mask proposal component that generates potential masks along with their visual embeddings, and b) a pre-trained VLM text encoder to output text embedding for each class name, and c) optionally, a pre-trained VLM image encoder that also outputs visual embeddings given the mask proposals. Following most open-vocabulary segmentation models, in this paper, we use CLIP [30] with a Vision Transformer (ViT) backbone as the chosen VLM.

Now, to obtain the semantic label for each mask proposal, we compute the cosine similarity scores between the corresponding visual embedding $\mathbf{v}$ and text embeddings $\{\mathbf{t}_1, \cdots, \mathbf{t}_N\}$ of all N classes. The class with the highest similarity

score corresponds to the semantic prediction $\hat{\mathbf{y}}$ for this mask, formalized as:

$$\hat{\mathbf{y}} = \underset{n}{\operatorname{argmax}} \ \{\cos(\mathbf{v}, \mathbf{t}_n)\}. \tag{1}$$

We can obtain the semantic segmentation for the input image by repeating this process for all mask proposals.

Despite demonstrating promising results, open-vocabulary segmentation faces significant challenges, particularly due to the limited domain generalization capabilities of CLIP embeddings. These limitations hinder the segmentation precision across diverse domains. Drawing inspiration from the success of multi-modal prompt tuning in enhancing classification accuracy across various domains [16,18,22,26,48,49], we propose adopting a novel prompt tuning approach to specifically refine CLIP for improved domain-specific segmentation, as outlined in Sec. 3.2.

3.2 Multi-modal Prompt Tuning

Prompt tuning enables the adaptation of CLIP for target domains by learning the prompts instead of handcrafting them. The additional tokens appended provide contextual information on target domains while keeping the model as is. This way, it only introduces a small fraction of new learnable parameters and, hence, makes the learning process much more efficient.

More specifically, as illustrated in our architecture in Fig. 2e, we first concatenate a set of learnable prompts to image patch embeddings and text embeddings, respectively. After appending the prompt vectors, the enhanced tensors are formed for image visual embeddings, denoted as $\mathbf{v^{(0)}}$ and text embeddings, denoted as $\mathbf{t^{(0)}}$:

$$\mathbf{v^{(0)}} = [\mathrm{v}^{(0)}; \mathbf{e}_v^{(0)}; \mathbf{p}_v^{(0)}] \tag{2}$$

$$\mathbf{t^{(0)}} = [\mathrm{t}^{(0)}; \mathbf{e}_t^{(0)}; \mathbf{p}_t^{(0)}] \tag{3}$$

where $\mathrm{v}^{(0)}$ and $\mathrm{t}^{(0)}$ represent [CLS] and [EOS] special token embeddings, $\mathbf{e}_v^{(0)}$ and $\mathbf{e}_t^{(0)}$ are the visual and text embeddings. $\mathbf{p}_v^{(0)} = (\{p_i^v\}_{k=1}^K)^{(0)}$ and $\mathbf{p}_t^{(0)} = (\{p_i^t\}_{k=1}^K)^{(0)}$ correspond to the learnable prompts added in the input space, where K is the total number of learnable prompts and p_k^v, p_k^t are the k-th learnable prompt. It is worth mentioning that we initialize text prompts, $\mathbf{p}_t^{(0)}$ with the tokenization of "A photo of a" for the prompts added in the input space, while $\mathbf{p}_v^{(0)}$ is initialized from a random distribution [18]. Subsequently, we also append such K learnable prompts into deeper layers as the following:

$$\mathbf{v^{(j)}} = [\mathbf{e}_v^{(j-1)}; \mathbf{p}_v^{(j-1)}] \tag{4}$$

$$\mathbf{t^{(j)}} = [\mathbf{e}_t^{(j-1)}; \mathbf{p}_t^{(j-1)}] \tag{5}$$

where $\mathbf{v^{(j)}}, \mathbf{t^{(j)}}$ are the input tensors to the $(j+1)$-th layer, $1 < j \leq J$ and J is the prompt depth. If $J = 1$, the model defaults to combining CoOp [49] for the

text encoder and VPT-Shallow [16] for the visual encoder. J is by default always smaller than the total number of layers of the visual/text encoders. Therefore, for the remaining layers in the encoders after the J-th layer, we simply feed the preceding layer's prompt embedding through the remaining layers [18, 22].

3.3 Optimization

Here we introduce how we optimize the visual prompts $\mathbf{p}_v^{(j)}$ and text prompts $\mathbf{p}_t^{(j)}$ in each layer as discussed in Sec. 3.2. The process commences with the optimization of only visual prompts, and shifts to optimize only text prompts.

Optimization of Visual Prompts. In each iteration, we randomly sample a batch of 16 image segments with their class names, passing through the CLIP visual and text encoder to obtain their embedding $\mathbf{v}_i$ and $\mathbf{t}_i$ for each segment i. Following VPT [16], we use a cross-entropy loss $\mathcal{L}_{\text{ce}}(\mathbf{v}_i, \mathbf{t}_i)$ to optimize the visual prompts $\mathbf{p}_v^{(j)}$ for.

Optimization of Text Prompts. Once optimized, we freeze the visual prompts and sorely optimize the text prompts $\mathbf{p}_t^{(j)}$ with an objective:

$$\mathcal{L}(\mathbf{v}_i, \mathbf{t}_i^+, \mathbf{t}_i^-) = \mathcal{L}_{\text{ce}}(\mathbf{v}_i, \mathbf{t}_i^+) + \lambda \mathcal{L}_{\text{triplet}}(\mathbf{v}_i, \mathbf{t}_i^+, \mathbf{t}_i^-) \tag{6}$$

where $\mathbf{v}_i, \mathbf{t}_i^+$ and $\mathbf{t}_i^-$ are the visual embedding, true class name, and negative class names corresponding to the segment i, and $\mathcal{L}_{\text{t}}$ is a triplet loss [2].

$$\mathcal{L}_{\text{triplet}}(\mathbf{v}_i, \mathbf{t}_i^+, \mathbf{t}_i^-) = \max\{\|\mathbf{v}_i - \mathbf{t}_i^+\|_2 - \|\mathbf{v}_i - \mathbf{t}_i^-\|_2 + \mu, 0\} \tag{7}$$

and the margin μ is set to 1.5. The triplet loss is applied to retain open-vocabulary segmentation capabilities while adapting to a specific domain, inspired by the contrastive objective from CLIP [30]. Note that, we gradually increase the λ in (6) from λ_{min} to λ_{max}.

3.4 Triplet Mining

Negative Sample Database. One of the challenges of employing triplet loss is to find proper negative samples to form triplets. Using triplets with randomly selected negative samples will get the optimization process stuck quickly, making it hard to converge. To address this issue, we instruct GPT-4 to generate 5 closely related class names for each category in the training dataset. These names should be difficult for a machine learning model to distinguish among, but easily recognizable and distinguishable by humans. For instance, distinguishing "`wall`" from "`room divider`" poses a subtle yet significant challenge for models, despite being straightforward for humans. Refer to supplements for detailed prompts for GPT-4 negative name generation.

Online Hard Negative Sample Mining. Once having a generated negative database for each class, we apply an online hard negative mining strategy to find

informative triplets with the hardest negative name during training, as discussed in [13,32,34,42]. This is pivotal in refining the model's ability to distinguish between closely related but distinct classes, thereby improving its precision. Specifically, in a dataset with N classes, we will on-the-fly find the hardest negative name for a segment's class name from candidate names consisting of the remaining $N-1$ class names and all generated negative names combined. This selection process involves encoding all candidate names into embeddings using the CLIP text encoder and then calculating their L_2 distances to the visual embedding $\mathbf{v}_i$ of the target segment i The "hardest" negative sample is simply the text embedding with the lowest distance. This embedding, alongside the segment's class name and visual embedding, forms the triplet to fine-tune the text prompts.

4 Experiments

We evaluate our approach in different settings. First, we test its open vocabulary and closed-set classification capabilities on the ScanNet++ Office subset, KITTI-360, and ADE20K-150 datasets. We also conduct ablation studies to exploit the full potential of the triplet loss.

Datasets.

ScanNet++ Offices [43]. Scannet++ is adataset with 450+ 3D indoor scenes containing sub-millimeter resolution laser scans, registered 33-megapixel DSLR images, and commodity RGB-D streams from iPhone. To test domain adaptation, we take a subset of 14 office scenes for training and 16 office scenes for validation. In total, we have 7989 annotated training images, as well as 11054 test images. Also, to test the open-vocabulary segmentation ability, we consider 156 training class names and 233 test class names, 108 of which are in common between training and test names. Most of the unseen test classes are rare classes like "bucket", "headphone case" and "glasses case". For our segment classification task, we use their posed iPhone RGB images and render 2D semantic images from the ground-truth 3D semantics.

KITTI-360 [27]. A dataset with 11 sequences collected in the suburbs of Karlsruhe, Germany, which comprises 37 semantic classes with 24 "instance" classes and 13 "stuff" classes.

ADE20K-150 [44,45]. It includes RGB images from both indoor and outdoor scenes, having 2000 images for validation with 150 distinct classes.

Baselines. We compare against state-of-the-art prompt learning methods CoCoOp [48], VPT [16], RPO [22], and MaPLe [18], as well as robust fine-tuning method WiSE-FT [36].

Metrics. To measure the adaptation performance with ground-truth masks, we use Accuracy (Acc), and a family of F1 Score, including weighted-F1 (W-F1) by weighing the number of occurrences for each class, base-F1 (B-F1) over base classes seen during training, and novel-F1 (N-F1) over novel classes seen

only during test. As for the experiments with predicted masks, we measure the commonly used metrics mean IoU (mIoU) as well as the mean Accuracy (mAcc).

Implementation Details. Following previous prompt learning approaches [18, 22, 48, 49], we use Dassl library [47, 50] to implement prompt tuning on CLIP with the triplet loss. We first optimize only the visual prompts for 5 epochs. During training, we have a warmup epoch with a learning rate of 10^{-5}, and then set the learning rate to 0.0025 with a cosine scheduler from the second epoch. After training visual prompts, we only optimized the text prompts for another 5 epochs with also a learning rate of 0.0025. We use a batch size of 16 and an SGD optimizer on a single NVIDIA A100 GPU, and the training time is around 10-15 hours in total depending on the datasets. The 2D segments excerpted from different datasets have filled the background with the average pixel value of CLIP training images as done in [26]. The λ_{min} and λ_{max} is set to be 2 and 5, respectively.

4.1 Main Results

Parameter-efficient Domain Adaptation for Visual-Language Models. We commence our analysis by integrating CLIP into the task of 2D segment classification, thereby augmenting the efficacy of the overall open-vocabulary segmentation process. The results of prompt learning on segment classification are detailed in Table 1. As can be noticed, prompt learning techniques in general improve CLIP's segment classification capabilities in the target domain significantly. However, we see that the baseline multi-modal approaches like RPO and MaPLE trained with the cross-entropy loss do not achieve necessarily better results than VPT, which only adapts in the visual domain. This can be attributed to the inherently greedy nature of cross-entropy loss, impacting CLIP's intra-class representation and open-vocabulary generalization capabilities.

We also present the qualitative results on ScanNet++ Office scenes, KITTI-360 [27], and ADE20K [44, 45] in Fig. 4. We can see that our method shows clear improvements upon the original CLIP predictions without adaptations. We can distinguish similar classes like "door frame" and "door" as a result of the triplet loss. The original CLIP model suffers from the proximity of classes that come from similar contexts as seen by the examples of confusion between "wall outlet" - "wall", "sidewalk" - "road", and "rug" - "floor". In some cases, there are ambiguities in the matching, such as "table" example in the second row being classified as "office desk" by the original CLIP. In contrast, our method can also learn the annotator's intention due to visual examples introduced during the adaptation process. More comparisons with other baseline methods can be found in the supplementary material.

Open-Vocabulary Segment Classification. To further evaluate the capability for open-vocabulary segment classifications, we set up 125 unseen novel classes for ScanNet++ Office subset. As illustrated in Table 2, the original CLIP is indifferent to base and novel classes. Baseline methods all exhibit noticeable

Method	Modality	# Params	ScanNet++ Acc	ScanNet++ W-F1	KITTI-360 Acc	KITTI-360 W-F1	ADE20K-150 Acc	ADE20K-150 W-F1
No adaptation		0	9.0	11.2	19.1	23.4	27.8	32.7
CoCoOp [48]	🟦	∼ 77K	28.9	25.7	61.1	59.7	54.2	51.8
VPT [16]	🎨	∼ 786K	29.3	33.8	65.2	67.7	58.2	59.8
RPO [22]	🎨🟦	∼ 43K	35.4	30.6	66.0	63.6	58.1	55.2
MaPLe [18]	🎨🟦	∼ 2300K	36.7	31.4	64.5	62.2	61.6	59.6
OpenDAS (**Ours**)	🎨🟦	∼ 135K	**40.1**	**36.5**	**72.9**	**70.9**	**65.7**	**63.6**

Table 1: Quantitative Comparisons on Adaptation for Segment Classification. We evaluate our method for segment classification on the indoor dataset (ScanNet++ Office), outdoor (KITTI-360), and a combination of both domains (ADE20K-150). Baseline methods adapt either the text encoder, 🟦, the visual encoder, 🎨, or both. Params denote the number of parameters learned during training. Our approach can achieve significant improvements over all datasets demonstrating its ability to adapt to a new domain.

Method	Modality	# Params	W-F1	B-F1	N-F1
No adaptation		0	11.2	11.0	12.0
CoCoOp [48]	🟦	∼ 77K	25.7	34.3	12.7
VPT [16]	🎨	∼ 786K	33.8	37.6	12.8
RPO [22]	🎨🟦	∼ 43K	30.6	40.9	14.9
MaPLe [18]	🎨🟦	∼ 2300K	31.4	41.9	15.4
OpenDAS (**Ours**)	🎨🟦	∼ 135K	**36.5** +5.1	**46.8** +4.9	**20.9** +5.5

Table 2: Quantitative Comparisons on Open-Vocabulary Segment Classification. We evaluate the ScanNet++ Office subset. Besides the weighted-F1 (W-F1) metric, we also evaluate the performance on base and novel classes with base-F1 (B-F1) and novel-F1 (N-F1).

performance boosts over the original CLIP on the W-F1 and B-F1, but only marginally improve upon the N-F1 metric. In contrast, we demonstrate superior performance on all three metrics. We attribute the improvement in novel classes to the visual adaptation, closing the domain gap between CLIP's training images and segments. In the meantime, triplet loss preserves the structured embedding space. This indicates that our method can adapt to the target domain while still preserving strong open-vocabulary capability.

Comparison Against Robust Fine-Tuning Method WiSE-FT [36]. We further compare our approach against a robust fine-tuning method that fine-tunes the entire CLIP text and visual encoder, respectively. We show comparisons in Table 3 on KITTI-360 and ADE20K on the closed-vocabulary setting, as well as ScanNet++ Office dataset on the open-vocabulary setting. As can be observed, when compared to fine-tuning on the CLIP text encoder, we can

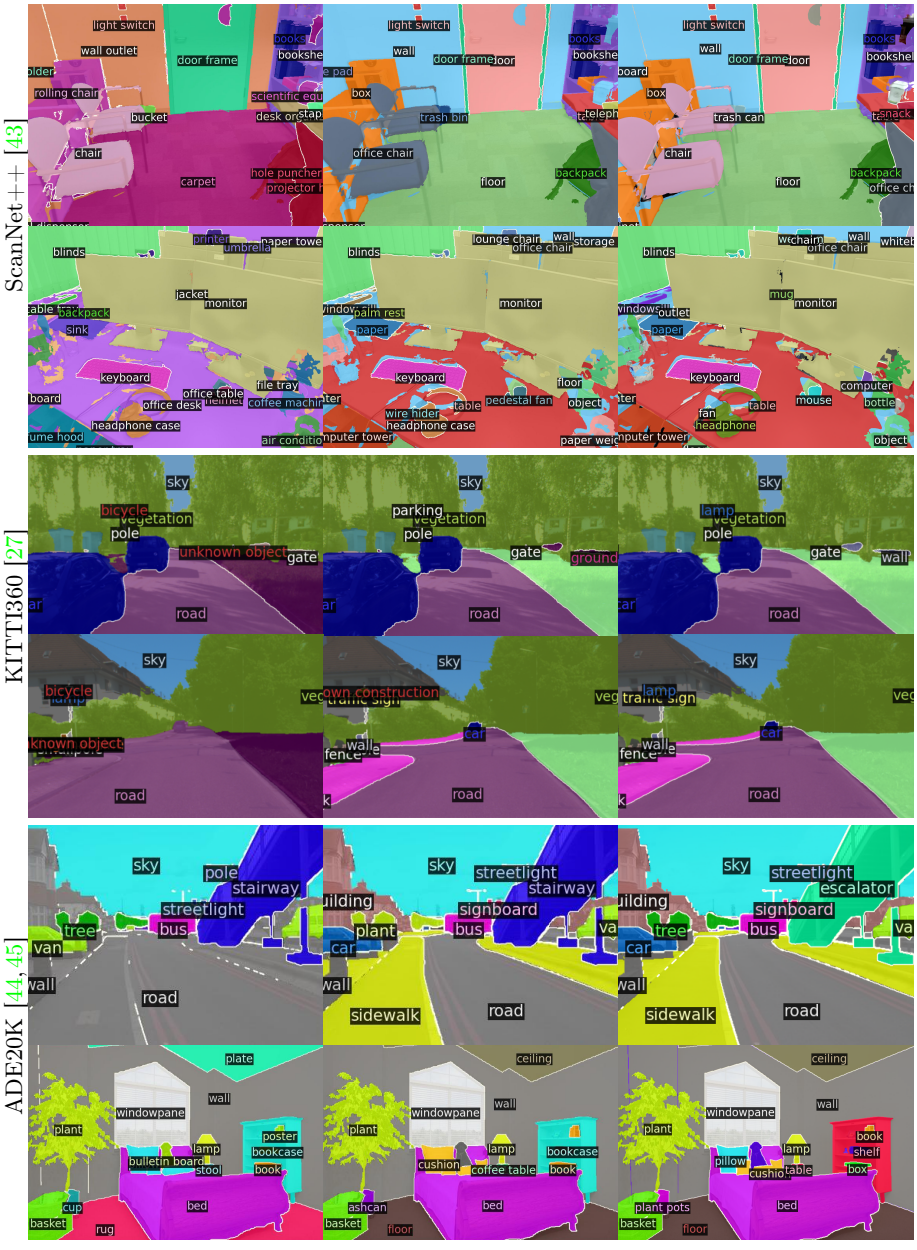


Fig. 4: Qualitative Comparison on Segment Classification. We show the predicted object classes with the ground truth masks given on three datasets.

Method	Modality	# Params	time/iter	KITTI-360 Acc	KITTI-360 W-F1	ADE20K-150 Acc	ADE20K-150 W-F1	ScanNet++ W-F1	ScanNet++ B-F1	ScanNet++ N-F1
No adaptation		0		19.1	23.4	27.8	32.7	11.2	11.0	12.0
WiSE-FT [36]	🟦	∼ 123M	0.85 s	53.9	58.8	47.9	51.0	25.4	34.8	10.2
WiSE-FT [36]	🎨	∼ 304M	1.05 s	**78.8**	**80.5**	**73.9**	**74.8**	**44.6**	**51.9**	12.9
OpenDAS (**Ours**)	🎨🟦	∼ 135K	**0.17 s**	72.9	70.9	65.7	63.6	36.5	46.8	**20.9**

Table 3: Comparison with Robust Fine-Tuning [36] on KITTI-360, ADE20K-150, and ScanNet++ Office. We can see that we significantly outperform the text-encoder fine-tuning setting on all three datasets with about 1000× fewer parameters. We also present competitive results over their image-encoder fine-tuning setting that uses over 2000× more parameters, and show stronger N-F1 results on ScanNet++, while being 5× faster to train.

Method	ADE20K-150 mIoU (%)	ADE20K-150 mAcc (%)
OVSeg [26]	29.8	48.1
OVSeg [26] + OpenDAS	**35.8** +6.0	**51.2** +3.1

Table 4: Benefit of Semantic Understanding in Open-Vocabulary Segmentation. We apply our prompting tuning method to a recent open-vocabulary 2D semantic segmentation method OVSeg [26]. Our approach boosts their performance in understanding semantic labels even with only their predicted masks.

achieve significant improvements in all metrics and all three datasets by only training ∼0.1% of CLIP-text encoder parameters. Looking into the comparison over fine-tuning the visual encoder, we show competitive results with only ∼0.05% of the parameters, and achieve significant improvements on novel classes in the ScanNet++ Office dataset.

Benefit of Semantic Understanding in Open-Vocabulary Segmentation. Moreover, besides evaluating our segment classification performance given GT masks, we also apply our prompt tuning method to OVSeg [26]'s predicted masks, assessing whether our method can help OVSeg better understand the semantics of each segment. As shown in Table 4, we observe the performance boost in both mIoU and mAcc when applying our method for OVSeg. This demonstrates that OpenDAS can be incorporated into open-vocabulary segmentation pipelines using only the predicted masks instead of ground truth ones.

4.2 Ablation Studies

We conduct ablation studies on $\lambda_{\max}$, which determines the weight of triplet loss during the adaptation of the text encoder, and the prompt depth, which determines the number of layers we add prompts.

$\lambda_{\max}$	ScanNet++ Acc	ScanNet++ W-F1	KITTI-360 Acc	KITTI-360 W-F1
0	38.9	34.8	66.4	64.4
2	39.8	35.8	71.9	70.2
5	**40.1**	**36.5**	**72.9**	**70.9**
10	39.3	35.7	72.1	70.8

Prompt Depth J	ScanNet++ Acc	ScanNet++ W-F1	KITTI-360 Acc	KITTI-360 W-F1
1	30.5	27.4	64.4	61.9
6	30.5	28.2	65.8	63.4
9	34.8	31.2	68.9	67.3
12	**40.1**	**36.5**	**72.9**	**70.9**

Table 5: Ablation Study. We conduct ablation studies on the hyperparameter of $\lambda_{\max}$ that controls the triplet loss's contribution increasing speed, as well as the prompt depth, which determines the number of layers we insert prompts.

$\lambda_{\max}$. Setting $\lambda_{\max} = 0$ indicates that the training objective in Eq. (6) for text prompts defaults to only the cross-entropy loss. As $\lambda_{\max}$ increases, the weight for triplet loss also increases during training. We can see on the left side of Table 5 that the performance has a peak when we set $\lambda_{\max} = 5$ for both datasets, so we choose it as our default value.

Prompt Depth. On the right side of Table 5, we present an ablation study on different prompt depths for the text and visual encoder. When prompt depth J is set to 1, we only add prompts in the input space, and the larger J is, the more layers we add learnable prompts to. We observe that OpenDAS has an improved performance over the increment of J.

5 Conclusion

In this paper, we show that prompt tuning methods can significantly enhance VLMs' performance for open-vocabulary segmentation tasks in a parameter-efficient way. We demonstrate that combining prompt tuning methods with triplet loss can improve the adaptation even further achieving improvements not only in base classes but also in novel classes. We apply our model with ground-truth masks into different datasets and gain a significant improvement over previous prompt learning methods, as well as robust fine-tuning methods.

Although we show promising results in this new paradigm of domain adaptation with class generalization, this work also poses some limitations. One limitation is that, when given all the training classes during inference, the adapted models all favor base classes over novel classes impacting their performance on novel classes significantly. Another limitation in this novel field is the lack of benchmarks where we can have visually close domains with a novel set of classes. Despite the absence of these benchmarks, our work shows promising results in the adaptation of VLMs and we hope that future work can bring more applications with the prompt tuning strategy.

Supplementary Material for OpenDAS: Domain Adaptation for Open-Vocabulary Segmentation

Anonymous ECCV 2024 Submission

Paper ID #2410

In the supplementary material, we first compare our method with others using three different datasets to see how well it does and give more details on how we put our method into practice. Next, we explain in detail how we create our negative database, which is key for the triplet loss optimization. We also include extra studies to dig deeper into how our method works, which we could not fit into the main paper. Lastly, we show results for each class using OpenDAS on all datasets, discussing the benefits and shortcomings of our method.

1 Qualitative Comparisons over Baselines

In the provided figures (see Fig. 1, Fig. 2, Fig. 3), we compare our method's segment classification capabilities, OpenDAS, over the other baseline methods, as well as the ground truth labels for reference.

Looking more into Fig. 1, for qualitative comparison on ScanNet++ [43], our method, OpenDAS, displays robustness in classifying basic elements like 'wall', 'floor', and 'whiteboard' across various viewpoints. Due to its triplet loss, OpenDAS adeptly differentiates between conceptually similar objects, for instance, 'ceiling' versus 'ceiling beam' and 'wall' versus 'wall beam' – a distinction that poses a challenge for other methods. However, it sometimes also fails, choosing 'object' instead of 'kettle' in the first column, and instead of 'window frame' in the second column. Despite these occasional errors, OpenDAS seems to be the closest to ground truth classes given these qualitative examples.

For KITTI-360 [27], we present the comparison in Fig. 2. Here, the task is relatively simpler as we have only 37 semantic classes. Among all of these, separating 'road' from 'sidewalk' could play a vital role in autonomous driving applications. Several multi-modal models struggle to separate these two classes, especially in instances where they share similar coloration. Our method OpenDAS, leveraging the nuanced capabilities provided by triplet loss during training, successfully identifies and segregates these analogous classes.

In Fig. 3, we present some qualitative comparison on ADE20K-150 Dataset [44, 45]. As the task is closed-set with 150 classes, all prompt learning methods perform generally accurately on this dataset. However, we see that in some cases, multi-modal joint prompt tuning as done in RPO [22] and MaPLe [18] can result in a degradation in CLIP's original representations, leading to occasional misclassification between 'sky' and other entities, unlike VPT [16] and OpenDAS, which employ isolated visual prompt tuning. Similarly, we observe some other failures with the classification of 'table' and 'chair' in the second column correctly

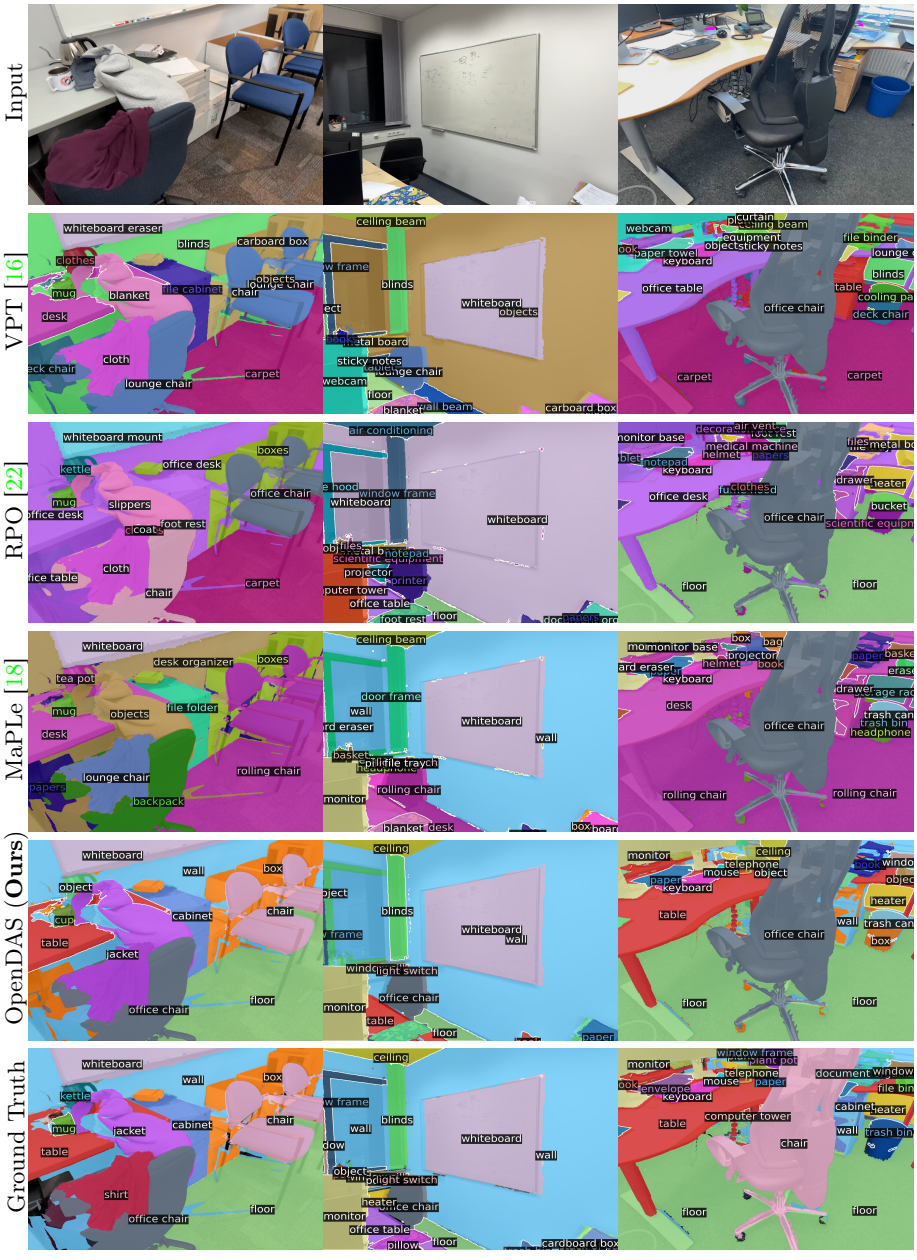


Fig. 1: Qualitative Comparison on Segment Classification on ScanNet++ Office [43]. We show the object classes with the ground truth masks predicted by baselines and OpenDAS, as well as ground truth labels. The masks are colorized based on the class ID.

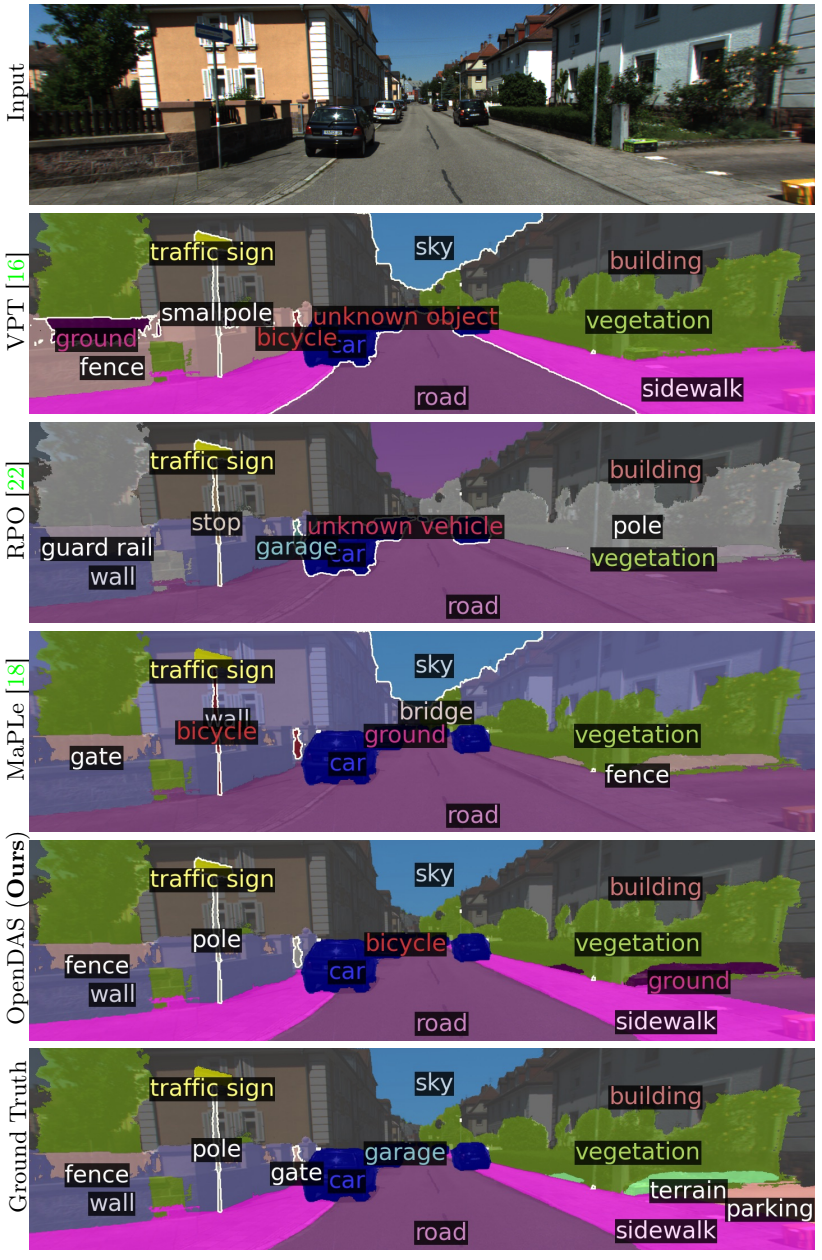


Fig. 2: Qualitative Comparison on Segment Classification on KITTI-360 Dataset [27]. We show the predicted object classes with the ground truth masks given on three datasets. The masks are colorized based on the class ID.

by other methods as well as distinguishing 'grass' from other classes. OpenDAS, while generally proficient, is not without its faults, as evidenced by the occasional inability to discriminate 'grass' from 'earth' or the misidentification of a glass door as a 'mirror'.

2 Negative Database for Triplet Loss

When triplet loss is trained with easy negatives, the learned latent space is likely to be suboptimal, not being able to distinguish similar classes and not generalizing well to unseen classes. By creating a negative database, we augment the set of negative classes to distinguish from.

For this purpose, we use GPT-4 [1] to generate 5 negatives for each class. We give the following instructions.

"Your task is to produce five distinct examples for each class provided in the list, ensuring that the examples are not subcategories of each other but rather represent clear and separate entities within the same class. This means that each example should not be a subset or type of another example within the same category. The objective is to create similar examples that might be confused by a machine learning model but remain discernible to a human observer to be used as clear negative examples for triplet loss training. The output format should be a Python dictionary for easy integration."

Some examples of the generated classes are as follows.

- `'wall': ['room divider', 'partition', 'divider screen', 'privacy screen', 'decorative panel']`
- `'ceiling': ['chandelier', 'pendant light', 'skylight', 'light fixture', 'ceiling fan']`
- `'folder organizer': ['bedside table', 'end table', 'chest of drawers', 'bar stool', 'storage ottoman']`

During training, we choose the hardest negative for each sample among all the training and negative classes combined to optimize the triplet loss.

3 Further Implementation Details

For the OpenDAS training pipeline, we first prepare segmentation datasets for the training of segment classification. Assuming that we have the 2D images as well as semantic annotations, we perform pre-processing on the dataset to adapt the semantic annotations to the classification task. As illustrated in Fig. 4, ground truth segmentation masks are applied, and the background is filled with the mean pixel values from CLIP's original training images, mirroring the approach adopted by a prior open-vocabulary segmentation work, OVSeg [26]. Each segment is annotated with a unique ID to facilitate the classification task.

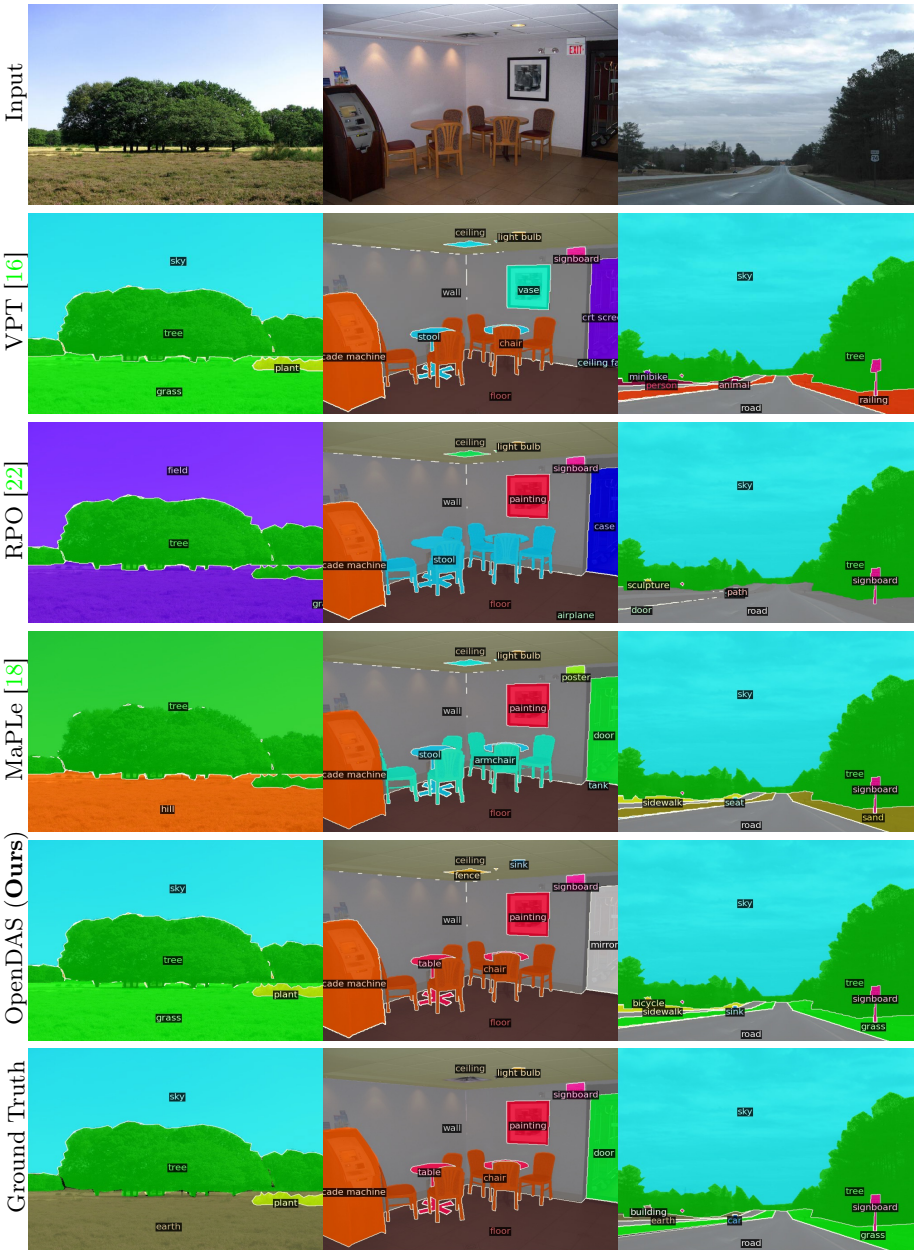


Fig. 3: Qualitative Comparison on Segment Classification on ADE20K Dataset [44, 45]. We show the object classes with the ground truth masks predicted by baselines and OpenDAS, as well as ground truth labels. The masks are colorized based on the class ID.

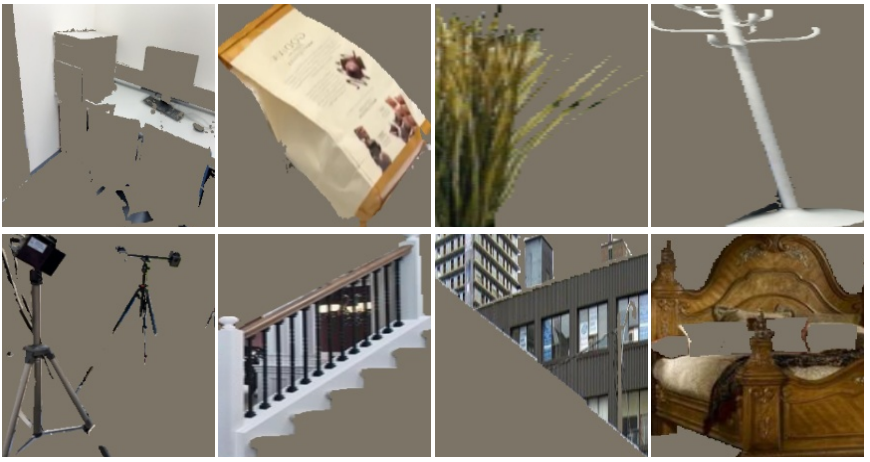


Fig. 4: Training Images. We prepare classification annotations for the segment classification task by applying ground truth masks on images, where the background is filled with the mean pixel values from the original CLIP training images. Each segment is annotated with a unique ID for the classification task.

Having prepared the training set, we employ the Dassl library [47,50] to train our prompt learning approach, adhering to the conventions established by prior prompt learning methodologies [18,22,48,49]. This way, we ensure compatibility with other baselines. We perform the inference with the same library and report the classification results measured with this implementation.

For class prediction with ground truth masks on a given image, we seamlessly integrate our tailored CLIP model with learned prompts into a segmentation framework built upon Detectron2 [39]. This segmentation pipeline is based on OVSeg [26], thereby enabling the integration of our customized CLIP model into the OVSeg pipeline. This integration allows us to use class-agnostic mask predictions that come from MaskFormer [4], as well as ground truth masks.

4 Further Ablation Studies

In this section, we extend the existing ablation studies to ADE20K dataset, while adding new ablations on the number of learnable prompts on all three datasets.

Maximum Triplet Loss Weight $\lambda_{\max}$. This parameter determines the maximum weight of the triplet loss when it is combined with the cross-entropy loss for textual prompt tuning. If $\lambda_{\max} = 0$, then we only use cross-entropy loss during our training. As presented in Table 1, we observe that the optimal performance is achieved when $\lambda_{\max} = 5$.

Prompt Depth (J). We conduct further ablations on the prompt depth parameter, J. This determines how many layers we inject prompts into. When $J = 1$,

$\lambda_{\max}$	ScanNet++ Acc	ScanNet++ W-F1	KITTI-360 Acc	KITTI-360 W-F1	ADE20K-150 Acc	ADE20K-150 W-F1
0	38.9	34.8	66.4	64.4	51.7	48.4
2	39.8	35.8	71.9	70.2	55.8	54.0
5	**40.1**	**36.5**	**72.9**	**70.9**	**65.7**	**63.6**
10	39.3	35.7	72.1	70.8	58.9	56.5

Table 1: Maximum Triplet Loss Weight ($\lambda_{\max}$) Comparison. Comparison of OpenDAS with Different $\lambda_{\max}$ Values on the Scannet++ Office subset, KITTI-360 and ADE20K Datasets.

Prompt Depth (J)	ScanNet++ Acc	ScanNet++ W-F1	KITTI-360 Acc	KITTI-360 W-F1	ADE20K Acc	ADE20K W-F1
1	30.5	27.4	64.4	61.9	56.4	53.7
6	30.5	28.2	65.8	63.4	56.0	54.0
9	34.8	31.2	68.9	67.3	62.0	60.1
12	40.1	36.5	72.9	70.9	65.7	63.6
16	41.5	38.2	74.2	73.2	70.5	69.1
24	**43.1**	**40.2**	**75.7**	**75.2**	**73.1**	**71.9**

Table 2: Comparison of Different Prompt Depth Values. We compare different prompt depth values on the Scannet++ Office subset, KITTI-360, and ADE20K-150 validation splits. In the main paper, we use J=12 for all the experiments. Our further analysis reveals that the more layers we add prompts into, the better OpenDAS performs.

we only inject prompts in the input space. As we use the ViT-L/14 backbone, J has to be ≤ 24. As shown in Table 2, the larger the prompt depth is, the better the performance on all three datasets. As these results were obtained after the paper submission, we will update results for the baselines and our method with $J = 24$ as it gives the best results for domain adaptation.

Number of Learnable Prompts (K). This determines the number of prompts injected at each layer, or in other words, the 'width' of the prompts. During prompt length ablation, we use prompt depth $J = 12$ for all experiments as in the main paper. In Table 3, we observe that we can gain additional improvement by adding more learnable prompts to the visual encoder for the ScanNet++ Office subset and ADE20K-150 validation set. We also see from these numbers that it is better to have larger prompts in the visual encoder for adaptation and less prompts injected into the textual encoder. Based on the ablation, we can conclude that number of learnable prompts should be tuned based on the domain, into which we are adapting.

K 🎨	K 📘	# Params	ScanNet++ Acc	ScanNet++ W-F1	KITTI-360 Acc	KITTI-360 W-F1	ADE20K-150 Acc	ADE20K-150 W-F1
8	4	∼ 135K	40.1	36.5	**72.9**	**70.9**	65.7	63.6
8	8	∼ 172K	39.6	35.5	72.0	70.4	59.3	57.2
8	12	∼ 209K	39.7	37.0	72.3	70.9	57.5	55.2
12	4	∼ 184K	**40.3**	**37.4**	72.4	70.7	70.1	68.5
12	8	∼ 221K	39.6	36.7	72.7	71.5	**70.9**	**69.4**
12	12	∼ 258K	39.6	37.2	72.3	71.1	70.4	68.4

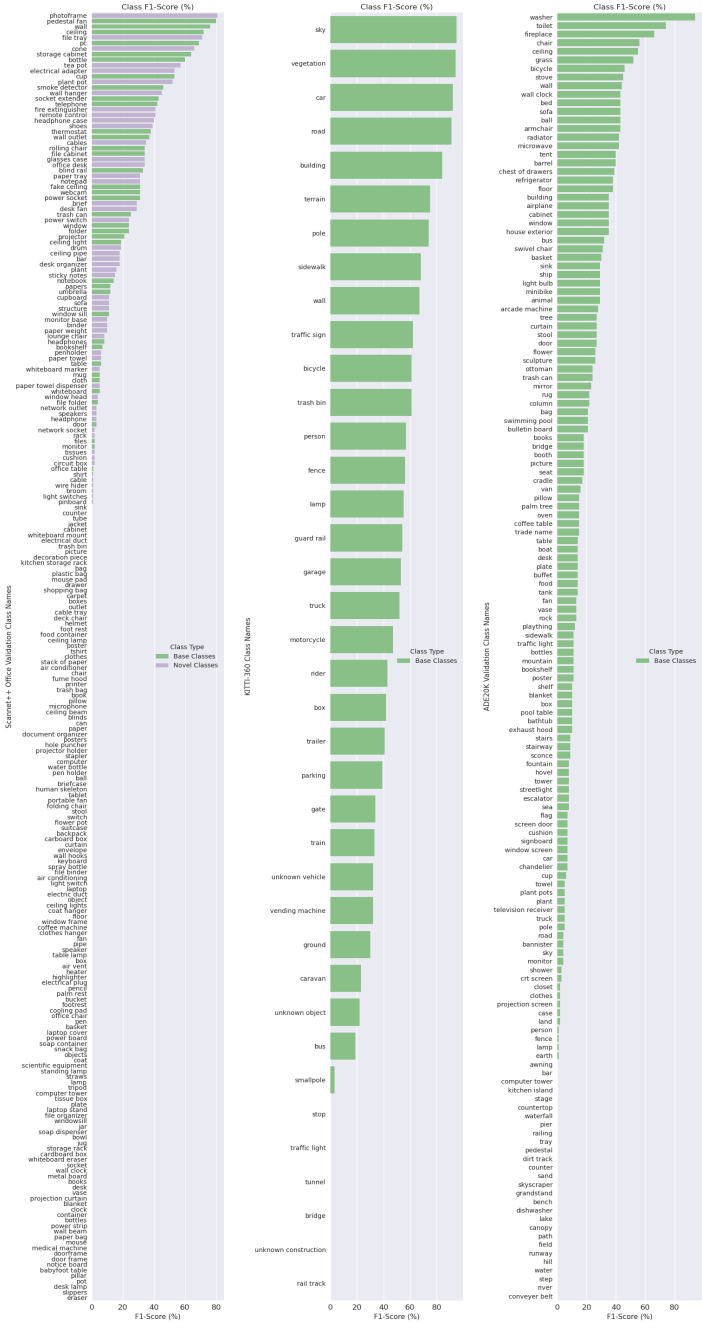
Table 3: Number of Learnable Prompts (K) Comparison on Different Datasets. We compare OpenDAS with different numbers of learnable prompt values on the Scannet++ Office subset, KITTI-360, and ADE20K-150. We denote the prompt length for the visual encoder as K 🎨 and for the textual encoder as K 📘.

5 Class F1-Scores (%)

In Fig. 5, we present the class-wise F1-Scores for all three datasets using OpenDAS for segment classification. In Fig. 5a, focusing on the ScanNet++ Office subset, which features 233 classes including both base and novel categories, we note high F1-Scores for items like 'photo frame', 'pedestal fan', and 'wall', suggesting accurate model recognition. Conversely, items such as 'eraser', 'paper', and 'notebook' show low F1-Scores, potentially due to insufficient training data. Novel labels like 'desk lamp' and 't-shirt' also score low, possibly because the model confuses them with similar base class items like 'lamp' and 'shirt', leading to base class bias during inference.

In Fig. 5b, we present class-wise F1-scores on KITTI-360. We see that most occurring and distinct items like 'sky', 'vegetation' and 'car' have very high F1-scores, whereas underrepresented items like 'stop', 'traffic light', 'tunnel' and 'bridge' cannot be distinguished very well.

In Fig. 5c, we present the class-wise F1-scores on ADE20K-150 classes. There is a clear pattern observable where the model performs well in recognizing large and distinct objects such as 'washer', 'toilet', and 'fireplace', as indicated by the longest bars. These items usually have distinct shapes and sizes that can be consistently identified across different scenes. Moving down the graph, we see items with more complex structures or that may be more challenging to distinguish from like 'skyscraper', 'grandstand', and 'bench'. This is likely due to similar items with more occurrences in the dataset.



(a) Scannet++ Office **(b) KITTI-360** **(c) ADE20K**

Fig. 5: Class F1-Score (%). We report class F1-scores on all three datasets when we perform segment classification with OpenDAS. Green bars refer to the base classes and violet bars refer to the novel classes.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023) 18
2. Balntas, V., Riba, E., Ponsa, D., Mikolajczyk, K.: Learning local feature descriptors with triplets and shallow convolutional neural networks. In: BMVC (2016) 8
3. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. pp. 1877–1901 (2020) 4
4. Cheng, B., Schwing, A.G., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation (2021) 20
5. Cho, S., Shin, H., Hong, S., An, S., Lee, S., Arnab, A., Seo, P.H., Kim, S.: Cat-seg: Cost aggregation for open-vocabulary semantic segmentation (2023) 4
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (2019) 4
7. Ding, J., Xue, N., Xia, G.S., Dai, D.: Decoupling zero-shot semantic segmentation. In: CVPR (2022) 4
8. Ding, Z., Wang, J., Tu, Z.: Open-vocabulary universal image segmentation with maskclip (2023) 4
9. Ghiasi, G., Gu, X., Cui, Y., Lin, T.Y.: Scaling open-vocabulary image segmentation with image-level labels. In: ECCV (2022) 2, 4
10. Goyal, S., Kumar, A., Garg, S., Kolter, Z., Raghunathan, A.: Finetune like you pretrain: Improved finetuning of zero-shot vision models. In: CVPR (2023) 4
11. Gu, Y., Han, X., Liu, Z., Huang, M.: PPT: Pre-trained prompt tuning for few-shot learning. In: Muresan, S., Nakov, P., Villavicencio, A. (eds.) Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (2022) 4
12. He, S., Ding, H., Jiang, W.: Primitive generation and semantic-related alignment for universal zero-shot segmentation. In: CVPR (2023) 4
13. Hermans, A., Beyer, L., Leibe, B.: In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737 (2017) 9
14. Huang, H., Chu, J., Wei, F.: Unsupervised prompt learning for vision-language models. ArXiv **abs/2204.03649** (2022) 4
15. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision (2021) 1, 3
16. Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N.: Visual prompt tuning. In: ECCV (2022) 3, 4, 5, 7, 8, 9, 11, 15, 16, 17, 19
17. khattak, M.U., Ferjad, M., Muzzamal, N., Gool, L.V., Tombari, F.: Learning to prompt with text only supervision for vision-language models. arXiv:2401.02418 (2024) 3, 4
18. Khattak, M.U., Rasheed, H., Maaz, M., Khan, S., Khan, F.S.: Maple: Multi-modal prompt learning. In: CVPR (2023) 3, 4, 5, 7, 8, 9, 10, 11, 15, 16, 17, 19, 20

19. Khattak, M.U., Wasim, S.T., Naseer, M., Khan, S., Yang, M.H., Khan, F.S.: Self-regulating prompts: Foundational model adaptation without forgetting. In: ICCV (2023) 3, 4
20. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. arXiv preprint arXiv:2304.02643 (2023) 2
21. Kumar, A., Raghunathan, A., Jones, R.M., Ma, T., Liang, P.: Fine-tuning can distort pretrained features and underperform out-of-distribution. In: ICLR (2022) 4
22. Lee, D., Song, S., Suh, J., Choi, J., Lee, S., Kim, H.J.: Read-only prompt optimization for vision-language few-shot learning. In: ICCV (2023) 3, 4, 5, 7, 8, 9, 10, 11, 15, 16, 17, 19, 20
23. Lester, B., Al-Rfou, R., Constant, N.: The power of scale for parameter-efficient prompt tuning. In: Moens, M.F., Huang, X., Specia, L., Yih, S.W.t. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (2021) 4
24. Li, B., Weinberger, K.Q., Belongie, S., Koltun, V., Ranftl, R.: Language-driven semantic segmentation. In: ICLR (2022) 3
25. Li, X.L., Liang, P.: Prefix-tuning: Optimizing continuous prompts for generation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (2021) 4
26. Liang, F., Wu, B., Dai, X., Li, K., Zhao, Y., Zhang, H., Zhang, P., Vajda, P., Marculescu, D.: Open-vocabulary semantic segmentation with mask-adapted clip. In: CVPR (2023) 2, 4, 7, 10, 13, 18, 20
27. Liao, Y., Xie, J., Geiger, A.: KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. IEEE TPAMI (2022) 9, 10, 12, 15, 17
28. Liu, P., Orru, Y., Paxton, C., Shafiullah, N.M.M., Pinto, L.: Ok-robot: What really matters in integrating open-knowledge models for robotics. arXiv preprint arXiv:2401.12202 (2024) 1
29. Liu, X., Zheng, Y., Du, Z., Ding, M., Qian, Y., Yang, Z., Tang, J.: Gpt understands, too. AI Open (2023) 4
30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision (2021) 1, 2, 3, 4, 6, 8
31. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners (2019) 4
32. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: CVPR (2015) 9
33. Shah, D., Equi, M.R., Osiński, B., Xia, F., Ichter, B., Levine, S.: Navigation with large language models: Semantic guesswork as a heuristic for planning. In: Conference on Robot Learning (2023) 1
34. Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P., Moreno-Noguer, F.: Discriminative learning of deep convolutional feature point descriptors. In: ICCV (2015) 9
35. Wei, Y., Hu, H., Xie, Z., Liu, Z., Zhang, Z., Cao, Y., Bao, J., Chen, D., Guo, B.: Improving clip fine-tuning performance. In: ICCV (2023) 4
36. Wortsman, M., Ilharco, G., Kim, J.W., Li, M., Kornblith, S., Roelofs, R., Lopes, R.G., Hajishirzi, H., Farhadi, A., Namkoong, H., Schmidt, L.: Robust fine-tuning of zero-shot models. In: CVPR (2022) 4, 9, 11, 13

37. Wu, J., Li, X., Xu, S., Yuan, H., Ding, H., Yang, Y., Li, X., Zhang, J., Tong, Y., Jiang, X., Ghanem, B., Tao, D.: Towards open vocabulary learning: A survey. IEEE TPAMI (2024) 2, 3, 6
38. Wu, J., Antonova, R., Kan, A., Lepert, M., Zeng, A., Song, S., Bohg, J., Rusinkiewicz, S., Funkhouser, T.: Tidybot: Personalized robot assistance with large language models. Autonomous Robots (2023) 1
39. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. https://github.com/facebookresearch/detectron2 (2019) 20
40. Xia, Y., Shi, L., Ding, Z., Henriques, J.F., Cremers, D.: Text2loc: 3d point cloud localization from natural language. arXiv preprint arXiv:2311.15977 (2023) 1
41. Xu, J., Liu, S., Vahdat, A., Byeon, W., Wang, X., De Mello, S.: Open-Vocabulary Panoptic Segmentation with Text-to-Image Diffusion Models. arXiv preprint arXiv:2303.04803 (2023) 4
42. Xuan, H., Stylianou, A., Pless, R.: Improved embeddings with easy positive triplet mining (2020) 9
43. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: ICCV (2023) 9, 12, 15, 16
44. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: CVPR (2017) 9, 10, 12, 15, 19
45. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ade20k dataset. IJCV (2019) 9, 10, 12, 15, 19
46. Zhou, C., Loy, C.C., Dai, B.: Extract free dense labels from clip. In: ECCV (2022) 4
47. Zhou, K., Liu, Z., Qiao, Y., Xiang, T., Loy, C.C.: Domain generalization: A survey. IEEE TPAMI (2022) 10, 20
48. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Conditional prompt learning for vision-language models. In: CVPR (2022) 3, 4, 5, 7, 9, 10, 11, 20
49. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. IJCV (2022) 3, 4, 7, 10, 20
50. Zhou, K., Yang, Y., Qiao, Y., Xiang, T.: Domain adaptive ensemble learning. IEEE Transactions on Image Processing (2021) 10, 20