FLOSS: Free Lunch in Open-vocabulary Semantic Segmentation

Yasser Benigmim[1*] Mohammad Fahes[1] Tuan-Hung Vu[1,2] Andrei Bursuc[1,2] Raoul de Charette[1]
[1] Inria [2] Valeo.ai

Abstract

In this paper, we challenge the conventional practice in Open-Vocabulary Semantic Segmentation (OVSS) of using averaged class-wise text embeddings, which are typically obtained by encoding each class name with multiple templates (e.g., `a photo of <class>`, `a sketch of a <class>`). We investigate the impact of templates for OVSS, and find that for each class, there exist single-template classifiers—which we refer to as class-experts—that significantly outperform the conventional averaged classifier. First, to identify these class-experts, we introduce a novel approach that estimates them without any labeled data or training. By leveraging the class-wise prediction entropy of single-template classifiers, we select those yielding the lowest entropy as the most reliable class-experts. Second, we combine the outputs of class-experts in a new fusion process. Our plug-and-play method, coined FLOSS*, is orthogonal and complementary to existing OVSS methods, offering an improvement without the need for additional labels or training. Extensive experiments show that* FLOSS *consistently enhances state-of-the-art OVSS models, generalizes well across datasets with different distribution shifts, and delivers substantial improvements in low-data scenarios where only a few unlabeled images are available. Our code is available at* https://github.com/yasserben/FLOSS.

1. Introduction

In the past decade, advances in deep learning and the growing amount of training data have enabled the challenging task of semantic segmentation, which involves assigning semantic labels to each pixel in an image. Initially, this was limited to predefined categories [9, 10, 57, 69], but recent models rely on vision-language alignment [47] for open-vocabulary segmentation, where the semantic categories can be dynamically expressed at runtime in natural language.

In Open-Vocabulary Semantic Segmentation (OVSS), the CLIP model [47] is commonly used as the backbone, which is either retrained [43], fine-tuned [12, 38], or frozen and adjusted to extend global image-language alignment to denser pixel-language alignment. In this work, we study CLIP's original classification paradigm and focus on methods that keep all CLIP parameters frozen [23, 63, 79]. Among these, MaskCLIP [79] leverages CLIP's embedding layer for segmentation, CLIP-DINOiser [66] enhances features via distillation from an external DINO [6] network, while NACLIP [23] modifies CLIP's self-attention to improve spatial consistency. A common aspect of prior OVSS methods is their reliance on CLIP's default classification process, where image or patch embeddings are compared with text embeddings representing semantic categories. To represent a class, a common practice is to use multiple templates to encode class names (*e.g.*, "a photo of a car", "a sketch of a car", etc) and average their embeddings. Empirically, Radford *et al*. [47] engineered $M = 80$ templates to improve ImageNet zero-shot classification. For segmentation, OVSS methods [23, 66, 79] adopt by default the same templates to construct segmentation classifiers. While this "template-averaging"[1] strategy works well overall, it may not always provide the best classifier for each semantic concept.

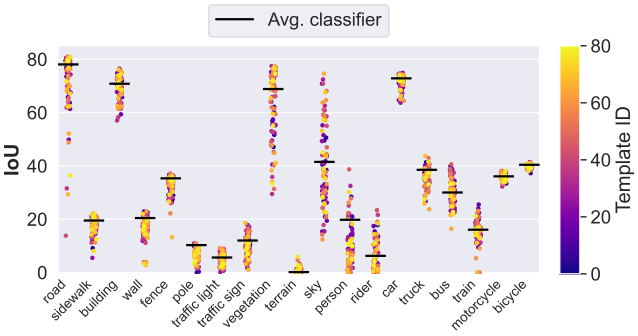


Figure 1. **Using individual vs average template classifiers.** Empirically, we observe that for each class there exist individual templates (colored dots) that lead to classifiers performing *better* than the popular CLIP classifier, which is built from the averaging of all 80 templates embeddings (—). The analysis was conducted on Cityscapes using the CLIP-DINOiser model.

*Work done during an internship at Inria.

[1]"Template-averaging" is an imprecise but convenient shorthand.

Our paper explores the impact of individual templates in OVSS – an aspect that has been overlooked in the literature. Fig. 1 shows an intriguing empirical observation: for each class, some individual templates outperform the averaged classifier that uses all templates. We call such templates **class-experts**, highlighting that they are trivial to identify using semantic ground truths.

However, without labels, two challenges arise: 1) *How to identify class-experts without ground truth?* 2) *How to fuse predictions from class-experts?*

We propose that entropy, an unsupervised metric, can help identify expert templates without labels. This leads to a new task—*training- and label-free template selection for OVSS*—which improves segmentation by selecting better text template subsets for a given dataset of unlabeled images. Our method shifts the attention to the text modality, unlike previous works that tweak the visual encoder, and also aims to improve the performance in an inductive setting, where the selected templates are applied to a different subset of images.

We summarize our contributions as follows:

- We analyze the current template-averaging practice for OVSS, showing that better class-wise classifiers can be constructed using only a subset of all templates, and that these expert templates can be identified without labels.
- We propose the novel task of training- and label-free template selection for OVSS. Given an unlabeled set of images, this task aims to identify class-wise experts that improve OVSS performance on unseen counterparts of the dataset.
- We introduce FLOSS, which leverages class-wise per-pixel entropy to select expert templates and employs a simple and effective fusion scheme to combine individual expert predictions.

With extensive experiments, we demonstrate that FLOSS consistently improves the state-of-the-art performance on OVSS benchmarks and exhibits generalization capabilities in the choice of experts. Furthermore, with only a few unlabeled images, our approach can select reasonably good expert templates, resulting in performance boost.

2. Related Works

Vision-Language Model (VLM) transferability. Since CLIP's release, its impressive zero-shot classification performance [47] and robustness [41, 58, 59] have made it a foundation for various adaptation and transfer learning tasks. Few-shot supervised fine-tuning, where only limited labeled data is available for downstream tasks like image classification, has gained significant attention. The goal is to leverage the vast knowledge of the VLM acquired through training on an immense image collection by steering it towards a task of interest via few labeled samples. Parameter efficiency is central to this setting. To ensure it, prompt learning [8, 33, 80–82], lightweight MLP training [19], and linear probing [29, 55] have been proposed. Another line of research explores training-free few-shot adaptation [64, 77, 83], where a classifier is built using few labeled samples and ensembled with the zero-shot classifier. Additionally, unsupervised adaptation has been explored [26, 28, 56], alongside test-time adaptation, where prompts are learned per test image [54].

While using VLMs for classification tasks aligns with their pre-training focus on global image recognition, applying them to dense prediction tasks like semantic segmentation is more challenging due to the gap between global image understanding and pixel-level prediction. We summarize related efforts in the following.

Open-Vocabulary Semantic Segmentation (OVSS). OVSS uses VLMs like CLIP for text-based segmentation. However, CLIP's global pooling layers limit its ability to generate dense pixel-level, language-aligned features [31, 79]. To address this, several *training-free* methods densify CLIP without altering its parameters, preserving its image-language alignment. MaskCLIP [79] was the first to replace the attention pooling layer with a convolutional layer. Later works further densify CLIP by aggregating multiple image views [31, 66], integrating priors from other vision foundation models for better pooling [35, 66], extracting different information from the self-attention layers [3, 23, 37], or removing anomaly tokens [2]. Seg-TTO [14] leverages an LLM to generate discriminative attributes for rare classes in specialized domains (*e.g.*, medical sciences, earth monitoring) that use technical terminology (*e.g.*, *mediastinum*), operating in a test-time optimization framework while maintaining zero-shot capabilities. Other approaches fine-tune CLIP for segmentation tasks using weak supervision through image-level tags, captions [21, 48], or class-agnostic object masks [21, 49]. Others employ full supervision [12, 36, 38, 74] on densely annotated datasets like COCO Stuff [5], following strategies similar to early inductive zero-shot semantic segmentation methods [4, 67]. Among them, some fine-tune CLIP's image encoder [36, 38, 74], others combine this with additional backbones and segmentation heads [38], while some learn side networks while keeping CLIP frozen [74]. Many other approaches have been proposed in this active research area [24, 68, 72, 73, 76]. While effective, these methods require labeled data and computational resources, limiting their practicality in resource-constrained scenarios.

Our work focuses on training-free approaches, evaluating FLOSS on three key methods: MaskCLIP [79], CLIP-DINOiser [66] and NACLIP [23], while being applicable to any text-based segmentation framework.

Prompting in VLMs. CLIP's image-language interface has inspired the adoption of various prompt engineering practices from LLMs to better reveal the knowledge encapsulated in the vision encoder and boost performance. The seminal CLIP work [47] introduced dataset-specific templates (*e.g.*, "a photo of <class>") to bridge the gap between training captions and test prompts for zero-shot classification. Subsequent methods have improved prompts by crowdsourcing templates [1], adjusting class names [21, 60], or modifying templates for tasks like object detection [21, 22] and occupancy prediction [60]. While effective, manual prompt tuning is labor-intensive, leading to automated strategies like adding random words or characters to the original template [50], enriching class names with WordNet concepts [20] or synonyms [39], learning more informative class names [27], and using LLMs to generate more expressive descriptions starting from simple class names or retrieve them from a dataset [51] and further use them as prompts. Some approaches also learn prompts for task-specific fitting [45] or test-time adaptation [54], and others mine them with external LLMs [16].

Our approach differs by selecting expert templates for each class in an unsupervised manner, starting from predefined templates. While previous methods have focused on image classification, we apply this strategy to semantic segmentation using CLIP's original ImageNet templates, which have been used by default in prior works.

3. Preliminaries

OVSS aims to produce a soft-segmentation map $\hat{\mathbf{S}} \in [0,1]^{H \times W \times K}$ given an image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ and a set of classes $\{\mathcal{C}_k\}_{k \in [1,K]}$ expressed in natural language. The final segmentation map $\hat{\mathbf{P}}$ is derived by applying $\arg\max(\cdot)$ on $\hat{\mathbf{S}}$ along the K-dimension. Leveraging the aligned vision encoder $f(\cdot)$ and text encoder $g(\cdot)$ of CLIP, the standard practice consists in classifying an image patch by comparing its vision embedding to the text embedding of class-specific prompts constructed using predefined templates $\{\mathcal{T}_m\}_{m \in [1,M]}$. An example of prompt is "$\underbrace{\text{a bright photo of a}}_{\mathcal{T}_m}$ $\underbrace{\text{car}}_{\mathcal{C}_k}$", denoted $\mathcal{P}_{m,k}$ for compactness. Instead of a single template, practitioners use "template-averaging" to increase robustness and improve overall performance by averaging the embeddings of prompts obtained using all available templates [40, 46, 47, 50], which is a form of ensembling on the text encoder side. This writes $\bar{\boldsymbol{c}}_k = \frac{1}{M} \sum_{m=1}^{M} g(\mathcal{P}_{m,k})$, where $\bar{\boldsymbol{c}}_k$ is the resulting representation of class $\mathcal{C}_k$. Subsequently, for each patch, the predicted class is the one that maximizes the cosine similarity between the visual and the text embeddings:

$$\hat{k} = \arg\max_k f(\mathcal{V})^{\mathrm{T}} \bar{\boldsymbol{c}}_k \,, \tag{1}$$

where $\mathcal{V}$ represents the visual patch. All embeddings are ℓ_2-normalized. The text embeddings of K classes define a classifier that we denote:

$$\boldsymbol{W}(\{\mathcal{T}_m\}_{m \in [1,M]}) = \{\bar{\boldsymbol{c}}_k\}_{k \in [1,K]} \,. \tag{2}$$

As in the above equation, the classifier can be represented as a function of the templates used to compute the average embeddings of all classes.

Empirical observations on single templates. OVSS models [23, 66, 68, 79] typically rely on the M=80 ImageNet templates of CLIP [47] to compute the average classifier of Eq. (2), which we refer to as $\boldsymbol{W}_{\text{CLIP}}$ for brevity. While using $\boldsymbol{W}_{\text{CLIP}}$ was shown to be robust [47, 79], the templates were originally hand-crafted for zero-shot classification on ImageNet [47], and their individual effect on class-wise performance remains unexplored for OVSS. Rather than $\boldsymbol{W}_{\text{CLIP}}$ that uses all M templates, one can construct M distinct classifiers, each utilizing a single template $\mathcal{T}_m$. This writes:

$$\boldsymbol{W}(\mathcal{T}_m) = \{g(\mathcal{P}_{m,k})\}_{k \in [1,K]} \,. \tag{3}$$

We design a systematic experiment to evaluate the performance of single-template classifiers $\boldsymbol{W}(\mathcal{T}_m)$ from Eq. (3). Fig. 1 shows the per-class performance on Cityscapes [13] of each of the 80 single-template classifiers (plotted as colored dots, *e.g.*, ●), where colors encode the template identifier (*i.e.*, m), along with the average $\boldsymbol{W}_{\text{CLIP}}$ classifier (plotted as —). This leads to two key observations:

a) For each class, there exist single-template classifiers outperforming $\boldsymbol{W}_{\text{CLIP}}$ *on this class*; we refer to the corresponding templates as **class-experts**. More precisely, the set of class-experts is defined as:

$$\mathcal{E}_k = \{\mathcal{T}_m \mid \Omega_k(\boldsymbol{W}(\mathcal{T}_m)) > \Omega_k(\boldsymbol{W}_{\text{CLIP}}), m \in [1,M]\} \,, \tag{4}$$

where $\Omega_k(\cdot)$ denotes the IoU performance on class k.

b) A single-template classifier excelling on one class may exhibit suboptimal performance on others, implying that for $k, l \in [1,K]$ the expert sets $\mathcal{E}_k$ and $\mathcal{E}_l$ *may* differ.

Following the above observations, class-experts are readily identified when ground truth labels are accessible, through direct performance comparison. We refer readers to Appendix A.3 for similar observations across datasets and models.

However, we are interested in the case where an unlabeled dataset is provided. Therefore, two questions stem from the previous empirical observations: (i) *How to identify class-experts in an unsupervised, training-free manner?* (ii) *Given these identified class-experts, how to derive the final semantic predictions?*

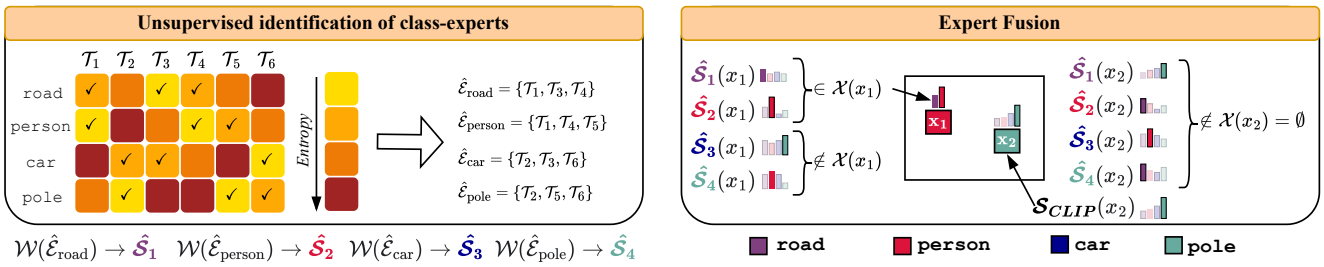


Figure 2. **Overview of FLOSS. Left:** for each class, selected expert templates (✓) construct the estimated set $\hat{\mathcal{E}}$, which classifies the input image and yield the soft-segmentation map $\mathbf{S}$. In this example we set $N = 3$ for the `Top-N(...)` template indices corresponding to lowest entropy entries to select. **Right:** we explain the fusion strategy for two cases. For the pixel x_1, the set of experts that predict their own class of expertise $\mathcal{X}(x_1)$ contains 2 experts; consequently, the decision is taken by looking at the maximum softmax scores of those two, leading to the decision of classifying x_1 as "person" with the higher score. For the pixel x_2, there is no expert predicting its own class of expertise, i.e., $\mathcal{X}(x_2) = \varnothing$, and the decision is taken by the default classifier $\boldsymbol{W}_{\text{CLIP}}$; the pixel x_2 thus takes the scores from $\mathbf{S}_{\text{CLIP}}(x_2)$.

4. Method

Given access to a dataset $\mathcal{D}$ of unlabeled images, our goal is to improve the performance of OVSS using the aforementioned class-experts, *without training or access to any labels*. To do so, for each class $k \in [1, K]$ our method first identifies a small set of class-experts among the M predefined ImageNet templates of CLIP (Sec. 4.1) by leveraging unsupervised metrics. It then builds on a simple scheme to fuse class-experts' predictions into a single OVSS semantic output (Sec. 4.2).

4.1. Unsupervised identification of class-experts

Referring to our definition of class-experts in Sec. 3, for each of the K classes, our goal is to estimate a set of class-experts $\hat{\mathcal{E}}_k$, among the M pre-defined CLIP templates. Being in an unsupervised setting, we have no access to the true classifier performance (*i.e.*, $\Omega(\cdot)$ in Eq. (4)) and therefore propose relying on unsupervised metrics which act as performance proxy [62, 70]. We leverage entropy as it is an established measure of uncertainty [18, 30], commonly adopted in deep learning literature [32, 34, 61]. The entropy value of a softmax prediction indicates a classifier's confidence, which empirically provides a reliable indicator of prediction quality. In our setting, entropy measures the confidence of single-template classifiers for each class, helping identify class-experts. We also study other unsupervised metrics in the ablations (Sec. 5.2).

In detail, we estimate class-experts as the set of individual templates whose associated classifiers (*cf.*, Eq. (3)) exhibit low entropy. Specifically, for each template m and class k, we compute the average entropy $\mathcal{H}_{m,k}$ of the $C_{m,k}$ pixels predicted as class k by $\boldsymbol{W}(\mathcal{T}_m)$ in all images of $\mathcal{D}$:

$$\mathcal{H}_{m,k} = -\frac{1}{C_{m,k}} \sum_{i=1}^{C_{m,k}} \mathbf{q}_i^{\mathrm{T}} \log(\mathbf{q}_i)\,, \tag{5}$$

where $\mathbf{q}_i \in \Delta^{K-1}$ is pixel-wise probability prediction obtained by applying $\mathtt{softmax}(\cdot)$ over K class-wise cosine similarities, and Δ^{K-1} is the $(K-1)$-simplex in $\mathbb{R}^K$. Subsequently, for class k, the estimated set of experts is obtained as the N templates yielding the lowest entropy:

$$\hat{\mathcal{E}}_k = \{\mathcal{T}_{\hat{m}} \mid \hat{m} \in \mathtt{Top\text{-}N}(\arg\operatorname*{sort}_m \mathcal{H}_{m,k})\}\,, \tag{6}$$

where $\hat{\mathcal{E}}_k$ is expected to be an estimation of $\mathcal{E}_k$ (Eq. (4)); the $\arg\operatorname*{sort}_m$ operator returns the sorted list of M template indices so that $\mathcal{H}_{m,k}$ are in ascending order. Finally, $\mathtt{Top\text{-}N}(\cdot)$ thus returns N template indices corresponding to the lowest entropy entries. In practice, we use $N = 4$ and validate this choice through an ablation study in our experimental section. Following Eq. (2), the expert classifier of class k, denoted $\boldsymbol{W}(\hat{\mathcal{E}}_k)$, is constructed by averaging the embeddings obtained from the N templates in $\hat{\mathcal{E}}_k$ for each of the K class names.

The above process leads to K classifiers $\{\boldsymbol{W}(\hat{\mathcal{E}}_k)\}_{k\in[1,K]}$, all obtained in a fully unsupervised manner, and each is expected to excel on one specific class. However, it is important to note that although a classifier is expert of a single class, it outputs a full semantic map containing all K classes. Therefore, the above formulation produces K soft-segmentation maps, $\{\hat{\mathbf{S}}_k\}_{k\in[1,K]}$. To output a single OVSS map, we introduce a simple scheme to fuse all expert predictions.

4.2. Fusion of expert predictions

To fuse all predicted soft-segmentation maps $\{\hat{\mathbf{S}}_k\}_{k\in[1,K]}$ into a single output $\hat{\mathbf{S}}$, we rely on the fact that each map $\hat{\mathbf{S}}_k$ excels at segmenting class k. Therefore, for each pixel, the class assigned is the one having the highest probability among the expert classifiers *which predicted their own class of expertise*. Occasionally, when no expert predicts its own

class of expertise for a given pixel, we simply revert to using the $\boldsymbol{W}_{\text{CLIP}}$ classifier. In practice, we observe that only $\approx 2\%$ of pixels fall under the latter case. Therefore, the vast majority of decisions are carried out by experts.

Formally, for a given pixel x, $\hat{\mathbf{S}}_k(x)$ is the $K-$dimensional softmax probability of the expert k at the position of x. Let $\mathcal{X}(x) = \{k \mid \arg\max_{i \in [1,K]} \hat{\mathbf{S}}_k(x)[i] = k\}$ be the index set of experts that predicted their own class of expertise for x. The prediction of x is determined as:

$$\hat{\mathbf{P}}(x) = \begin{cases} k^* = \arg\max_{k \in \mathcal{X}(x)} \hat{\mathbf{S}}_k(x)[k], & \text{if } \mathcal{X}(x) \neq \varnothing, \\ k^\dagger = \arg\max_{k \in [1,K]} \mathbf{S}_{\text{CLIP}}(x)[k], & \text{otherwise.} \end{cases} \tag{7}$$

Fig. 2 illustrates the unsupervised identification of class-experts as well as the fusion strategy. Overall, our method incurs computational overhead primarily from computing cosine similarities for the K class-experts, which scales with the number of classes and becomes more significant when K is very large ($>$100). However, this overhead is constrained to the similarity computations since the vision encoder is shared across all experts, requiring only a single forward pass regardless of the number of classes. We provide computational details in Appendix A.2.

5. Experiments

Experimental setup. We evaluate our method on three state-of-the-art open-vocabulary semantic segmentation models: the seminal MaskCLIP [79] and two recent models CLIP-DINOiser [66] and NACLIP [23]. CLIP-DINOiser trains a convolution layer using only 1K unlabeled images from ImageNet [15] for fast inference on pixel-level segmentation tasks, effectively imitating the DINO priors for pooling guidance and removing the need for DINO encoder at runtime. MaskCLIP and NACLIP are training-free adaptation methods of CLIP.

For a fair comparison, we report the version of NACLIP without refinement. We primarily select training-free models that keep the image encoder frozen and produce unaltered, dense image-language features. This enables us to assess the impact of the class-experts identification strategy by using the original CLIP pre-trained features and ImageNet templates. Unless otherwise stated, all models utilize CLIP ViT-B/16 backbone. CLIP-DINOiser and MaskCLIP are based on OpenCLIP [11], while NACLIP employs OpenAI's original CLIP [47]. For all models – CLIP-DINOiser, NACLIP, and MaskCLIP– we follow the authors' evaluation protocol. For CLIP-DINOiser and MaskCLIP, we resize images to 448 pixels on the shortest side with a sliding window of size 448 and stride 224. For NACLIP, images are resized to 336 pixels (except for Cityscapes which are resized to 560 pixels) with a 224 window and stride 112.

Method	CS	VOC20	PC59	ADE	Stuff	Avg
CLIP [47]	6.7	49.1	11.2	3.2	5.7	15.2
GroupViT [71]	11.1	79.7	23.4	9.2	15.3	27.7
CLIP-Surgery [37]	31.4	–	–	12.9	21.9	–
GEM [3]	–	–	32.6	15.7	–	–
SCLIP [63]	32.2	80.4	34.2	16.1	22.4	37.1
CLIP-DIY [65]	11.6	79.7	19.8	9.9	13.3	26.9
TCL [7]	23.1	77.5	30.3	14.9	19.6	33.1
ReCo [53]	21.1	57.8	22.3	11.2	14.8	25.4
MaskCLIP [79]	25.0	**61.8**	25.5	14.2	17.5	28.7
+ FLOSS	**25.8**	**61.8**	**26.2**	**14.9**	**17.8**	**29.3**
NACLIP [23]	35.5	79.7	35.2	17.4	23.3	38.2
+ FLOSS	**37.0**	**80.2**	**35.9**	**18.4**	**23.6**	**39.0**
CLIP-DINOiser [66]	31.3	80.9	35.9	20.0	24.6	38.5
+ FLOSS	**34.6**	**82.3**	**36.2**	**20.7**	**24.7**	**39.7**

Table 1. **OVSS across datasets and models.** We report mIoU metric for eight OVSS baselines and three models either using the average templates (*i.e.*, MaskCLIP, NACLIP, and CLIP-DINOiser) or with our method (*i.e.*, +FLOSS). FLOSS consistently improves OVSS models across datasets of varying complexity, from urban scenes (Cityscapes, 19 classes) to general objects (COCO-Stuff, 171 classes). Bold highlights the **best** performance.

Datasets and Metric. We evaluate FLOSS across multiple semantic segmentation benchmarks: PASCAL CONTEXT 59 [42] (PC59), PASCAL VOC 20 [17] (VOC20), COCO-Stuff [5] (Stuff), ADE20K [78] (ADE), and Cityscapes [13] (CS). Note that these datasets exhibit different ontology and semantic granularity, ranging from 19 classes (CS) to 171 classes (Stuff). Additionally, we demonstrate the generalization capabilities of our class-experts identified on Cityscapes to other driving datasets such as BDD-100K [75] and MAPILLARY [44] as well as ACDC [52] under Night, Fog, Rain and Snow conditions. Importantly, our approach is training-free and uses no labels. Performance is measured using mean Intersection over Union (mIoU).

Implementation details. The only hyperparameter of FLOSS, being the number of class-experts to identify in Eq. (6), is set to $N = 4$ in all experiments. While our method is training-free, class-experts are estimated using the training set of each dataset, while performance is always reported on the corresponding validation set.

5.1. Main results

Tab. 1 reports the mIoU of OVSS baselines across five datasets showing FLOSS plugged into three different models: MaskCLIP, NACLIP and CLIP-DINOiser. The performance gains vary with dataset complexity, showing smaller improvements on VOC20 $[+0.05, +0.5, +1.4]$ or PC59 $[+0.7, +0.7, +0.3]$ than on CS $[+0.8, +1.5, +3.3]$. We attribute this to VOC20 and PC59 being closer to ImageNet which was leveraged in CLIP [47] to engineer the set of 80 templates used in all models. The benefit of our method

Method	CS	VOC20	PC59	ADE	Stuff	Avg
MaskCLIP + FLOSS	51%	54%	56%	57%	56%	55%
NACLIP + FLOSS	49%	20%	57%	57%	52%	47%
CLIP-DINOiser + FLOSS	62%	41%	57%	45%	44%	50%

Table 2. **Quality of our estimated class-experts.** We measure the quality of our estimated experts by evaluating their intersection with the true experts, *cf*. Eq. (8).

Method	in domain	out of domain						
	CS	Night	Fog	Rain	Snow	BDD	MAP	Avg
MaskCLIP	24.5	13.3	20.3	21.0	19.9	22.7	25.8	20.5
+ FLOSS	**25.8**	**13.9**	**21.7**	**22.3**	**21.3**	**22.9**	**26.8**	**21.5**
NACLIP	35.5	23.3	**32.9**	30.7	32.6	31.4	35.3	31.0
+ FLOSS	**37.0**	**23.6**	32.5	**31.0**	**33.6**	**32.3**	**36.5**	**31.6**
CLIP-DINOiser	31.3	12.7	24.1	27.9	27.2	31.3	35.2	26.4
+ FLOSS	**34.6**	**16.6**	**29.0**	**29.8**	**29.4**	**32.2**	**36.6**	**28.9**

Table 3. **Generalization of experts across datasets.** We evaluate the transferability of experts identified on the Cityscapes (CS) dataset to unseen datasets that share the same semantic classes and exhibit changes in lighting (Night), weather (Fog, Rain, Snow), or location and scenery (BDD, MAP). Results show a consistent improvement when using FLOSS, across almost all settings.

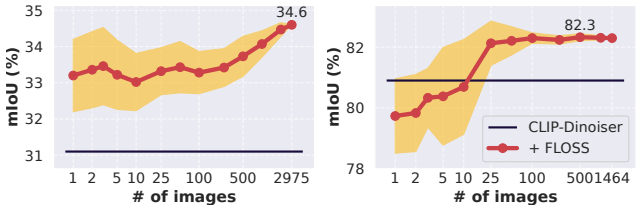
is evident on challenging benchmarks like ADE and Stuff, which have a large number of semantic categories making accurate segmentation particularly hard.

Quality of experts. We evaluate the quality $\hat{\rho}_k$ of our experts by measuring, for each class k, the intersection of the estimated class-expert $\hat{\mathcal{E}}_k$ with the true set of experts $\mathcal{E}_k$:

$$\hat{\rho}_k = 100 \times \frac{|\hat{\mathcal{E}}_k \cap \mathcal{E}_k|}{|\hat{\mathcal{E}}_k|} = 100 \times \frac{|\hat{\mathcal{E}}_k \cap \mathcal{E}_k|}{N}. \quad (8)$$

Where $|.|$ denotes the cardinality of a set. Quality averaged over classes is reported in Tab. 2 and shows that regardless of the model used, we correctly estimate approximately half of the true experts. In some scenarios, such as PASCAL VOC 20 with NACLIP, even when only 20% of predicted experts are true experts, it is sufficient to bring some improvements (+0.5 mIoU in Tab. 1). Per-class quality is reported in Appendix A.4, showing some variability.

Generalization to unseen datasets. We now evaluate the generalization of experts to other unseen datasets sharing the same label set but exhibiting distribution shifts. In Tab. 3, we report performance of FLOSS when experts are identified on Cityscapes and used to evaluate performance on six unseen datasets which include four ACDC conditions (Night, Fog, Rain, Snow), BDD-100K, and MAPILLARY. Overall, we consistently improve the performance with notable gains of up to +4.9 mIoU for CLIP-DINOiser on ACDC Fog. This demonstrates that our class-wise experts are readily usable on other datasets sharing the same semantic classes, despite the presence of distribution shifts.



(a) Cityscapes (b) PASCAL VOC 20

Figure 3. **Effect of the number of images.** We evaluate CLIP-DINOiser with FLOSS on urban scenes (a) and general object segmentation (b), when varying the number of images used to identify class-experts. Results show the average of 10 seeds, with standard deviation. Despite larger variation, we note the robustness of our method, even when having access to very few unlabeled images.

Low-data regimes. To investigate the data efficiency of FLOSS, we study its performance in lower data regimes. Fig. 3 reports the mIoU of CLIP-DINOiser on Cityscapes and PASCAL VOC 20, as the number of images sampled from their respective training sets (containing 2,975 and 1,464 images, respectively) varies. The figure shows the average and standard deviation over 10 seeds (*i.e.*, randomly sampling 10 different image sets). On both datasets, the general trend shows that our method benefits from accessing more images, which stems from the better estimation of class-experts. Remarkably, our method performs well in a very low-data regime, outperforming the default CLIP-DINOiser with only 1 image on Cityscapes and around 25 images on PASCAL VOC 20. We conjecture that the latter requires more images due to its lower per-image class density compared to Cityscapes, which exhibits higher variability and density per image.

Low-data regimes & domain shift. To further corroborate its practicality, we evaluate FLOSS under stricter conditions combining both low-data regimes and domain shift. Specifically, we use only a single CS image to identify experts that are tested on 6 shifted domains. Tab. 4 provides evidence on the effectiveness of FLOSS in this strict setting, leading to an average improvement of 1.9% across OOD datasets.

Expert transferability. We investigate the cross-dataset transferability of class-experts by evaluating whether experts identified on one dataset can effectively improve performance on semantically related but distinct datasets. We employ COCO-Stuff (171 classes) to identify class-experts for CLIP-DINOiser and apply them to Cityscapes, PASCAL VOC 20, and PASCAL CONTEXT 59, which share 10, 16, and 37 common classes with COCO-Stuff, respectively. For non-overlapping classes, we use the default average classifier $\boldsymbol{W}_{\text{CLIP}}$. As reported in Tab. 5, FLOSS shows consistent improvements, confirming expert transferability across datasets.

Method	in domain	out of domain						
	CS	Night	Fog	Rain	Snow	BDD	MAP	Avg
CLIP-DINOiser	31.3	12.7	24.1	27.9	27.2	31.3	35.2	26.4
+ FLOSS	**33.2**	**16.2**	**28.2**	**29.3**	**28.7**	**31.7**	**35.6**	**28.3**

Table 4. **FLOSS with 1 image & domain shift.** Experts are identified using **a single Cityscapes (CS) image** and evaluated on unseen datasets that share the same semantic classes and exhibit changes in lighting (Night), weather (Fog, Rain, Snow), or location and scenery (BDD, MAP). We observe a consistent boost.

Method	in-domain	out of domain					
	Stuff	CS		VOC20		PC59	
		common	all	common	all	common	all
CLIP-DINOiser	24.6	34.5	31.1	80.5	80.8	39.3	36.0
+ FLOSS	**24.7**	**35.8**	**32.2**	**82.3**	**82.4**	**40.0**	**36.2**

Table 5. **Cross-dataset expert transferability.** Class-experts identified on COCO-Stuff are applied to CS, VOC20, and PC59. Results are reported for **common** classes shared with COCO-Stuff and **all** classes. FLOSS demonstrates consistent improvements across all target datasets.

5.2. Ablation Study

Fusion strategies for expert predictions. We investigate alternative strategies to fuse the expert predictions (*cf*., Sec. 4.2), reporting performance on Cityscapes in Tab. 6. Simply averaging all soft segmentation maps ("Average-all") irrespective of the templates class of expertise leads to suboptimal results across all models. When accounting for templates' expertise in fusion, conflicts arise often as *several* experts may predict their own class of expertise. We therefore explore three strategies to resolve conflicts: "Default" which falls back to $\boldsymbol{W}_{\text{CLIP}}$ predictions, "Average" which averages the probability vectors of conflicting experts, and "Highest" which selects the most confident expert as described in Sec. 4.2. The latter consistently yields the best performance across all models, validating our fusion scheme.

Metrics for expert identification. We replace entropy in Eq. (6) with other unsupervised metrics to act as proxies of the template class-wise performance, reporting performance on Cityscapes using the CLIP-DINOiser model in Fig. 4a.

(i) *Avg. Probability* computes the average probability assigned to class k across all pixels predicted as class k.

(ii) *MaNo* [70] is adapted as it was originally designed for assessing general model performance in image classification, to evaluate class-wise performance in semantic segmentation.

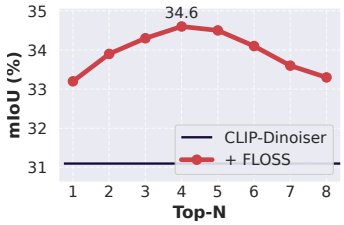
(iii) *"ITI"* [25] is the Inter-class separability to Intra-class similarity ratio, and captures both the separability of the classes in the feature space as well as the ability to keep samples of the same class close together.

Fusion	Conflict	MaskCLIP +FLOSS	NACLIP +FLOSS	CLIP-DINOiser +FLOSS
Average-all		24.7	36.6	31.2
Expert	Default	24.8	36.6	32.0
Expert	Average	25.6	36.7	34.4
Expert	Highest	**25.8**	**37.0**	**34.6**

Table 6. **Strategies for the fusion of expert predictions.** Exploring various fusion strategies on Cityscapes shows the benefit of our strategy, which outperforms all others across models.

Metric	mIoU
Avg. Probability	34.4
MaNo [70]	29.9
ITI	30.0
Entropy	**34.6**

(a) Expert metrics



(b) Effect of Top-N

Figure 4. **Ablation of experts.** We ablate our choices of metrics to identify experts as well as the number of experts per class. (a) shows unsupervised metrics to identify the experts, showing that our choice of entropy yields the best performance, although 'Avg. Probability' performs on par. (b) reports performance of FLOSS when varying the number of N experts. Results are reported on the Cityscapes dataset using the CLIP-DINOiser model.

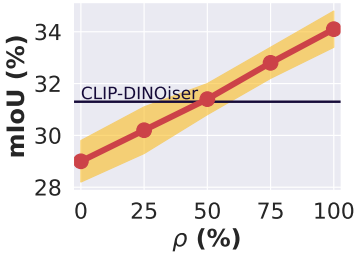


Figure 5. **Effect of class-expert quality.** We report the performance of FLOSS in the oracle-based setting, where each class-wise set of 4 experts contains a ratio (ρ) of true experts. The experiment is conducted on Cityscapes using the CLIP-DINOiser model.

These metrics are detailed in Appendix A.5.
Overall, our entropy measure (*cf*. Eq. (5)) is the most effective, although *Avg. Probability* performs on par.

Impact of class-expert identification quality. At the core of our method is the identification of class-experts. While we demonstrated there are templates which are class-expert, for each class there exist also under-performing templates (*cf*. Fig. 1) which, if selected, could harm the performance.

To measure the impact of experts quality on FLOSS, we devise *an oracle-based* experiment using $N = 4$ templates per-class , where we have access to ground truth la-

Method	ViT-B/16			ViT-L/14		
	CS	PC59	Avg.	CS	PC59	Avg.
SCLIP [63]	32.2	34.2	33.2	21.3	25.2	23.3
GEM [3]	32.6	35.6	34.1	27.1	28.1	27.6
NACLIP [23]	35.5	35.2	35.4	31.4	32.1	31.8
+ FLOSS	**37.0**	**35.9**	**36.5**	**32.4**	**32.4**	**32.4**

Table 7. **Effect of backbone.** Results (mIoU) show that NACLIP + FLOSS outperforms NACLIP across ViT-B/16 and ViT-L/14 CLIP backbones.

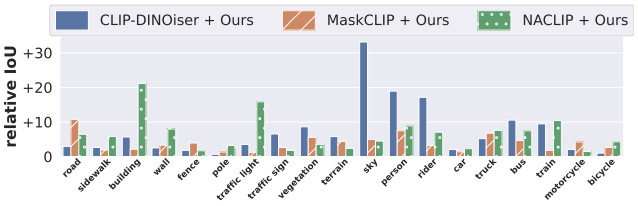


Figure 6. **Performance of best-class experts.** Performance gains (IoU) when using best class-experts versus standard template-averaging. Results are shown for CLIP-DINOiser, MaskCLIP and NACLIP. The bars represent the IoU difference between using best class-experts (selected with ground-truth labels) and the default template-averaging approach.

bels to identify the true class-experts. We randomly sample a fixed ratio ρ of templates among true experts (*i.e.*, $\mathcal{E}_k$ from Eq. (4)) and the remaining ones among non-experts (*i.e.*, $\{\mathcal{T}_m\}\backslash\mathcal{E}_k$). For example, a ratio of $\rho = 75\%$ indicates that 3 templates are experts and 1 is non-expert. The outcome of this experiment on CS using the CLIP-DINOiser model is reported in Fig. 5 and shows an expected performance boost when the ratio of true experts increases. It also highlights that 50% of true experts is sufficient for FLOSS to surpass the original CLIP-DINOiser. Interestingly, the performance of CLIP-DINOiser + FLOSS in Tab. 1 (34.6) slightly surpasses the performance at $\rho = 100\%$ shown in Fig. 5 (34.1). This however does not indicate that our estimated templates in Tab. 1 are all true experts – as reflected by Tab. 2 – but rather that our correctly estimated experts are on average better than random true experts.

To further measure the upper bound of CLIP-DINOiser + FLOSS, we implement an oracle version using only the *best true class-expert* (i.e., $\{\mathcal{T}_{\tilde{m}} \mid \tilde{m} \in \texttt{Top-N}(\Omega_k(\boldsymbol{W}(\mathcal{T}_m)))\}$). On CS, the oracle with $N \in \{1, 2, 3, 4\}$ achieves respectively 37.1/37.0/37.3/37.5%, showing the potential for improvement assuming the identification of better experts.

FLOSS using the validation set. First, we investigate the effectiveness of FLOSS in a transductive setting. Using the validation set of CS for expert selection, CLIP-DINOiser + FLOSS achieves 34.4% mIoU on the same set, compared to 31.3% for CLIP-DINOiser. Second, using a portion of the validation set to select experts, *i.e.*, 5/25/50/75 val images, we achieve 32.8/33.0/33.0/33.3% on the rest of the validation images, compared to 30.9% for CLIP-DINOiser alone.

Impact of different backbones. Tab. 7 demonstrates the effectiveness of our proposed FLOSS when integrated with NACLIP using the ViT-L/14 backbone. Our approach achieves consistent improvements, reaching 32.4% mIoU on both Cityscapes and PC59 datasets, leading to a 0.6 gain in average performance over NACLIP. These results validate the effectiveness of our method when using large-scale vision transformers.

6. Perspectives and Future Work

Our experiments show that entropy is a strong estimator of class-experts. However, future research could develop even better proxies for prediction accuracy. Template-based methods appear especially promising, with significant potential for improving OVSS performance. As illustrated in Fig. 6, there is still a large gap to close—for example, selecting the **best expert** for the "sky" class could improve IoU by over 30 percentage points. These "upper-bounds" could be further increased by using more or augmented templates beyond the 80 currently used.

7. Conclusion

We propose a new task for improving OVSS models without having access to labels and without training, from a prompt template perspective. Our work is motivated by an intriguing observation: for each class, some templates excel in segmenting it. We refer to these templates as class-experts. Given a set of unlabeled images, our FLOSS uses the entropy of predictions as an unsupervised metric to identify these class-experts, thus not requiring any training or labels. Once selected, we propose a simple scheme for fusing the predictions of the class-experts, resulting in an improved overall OVSS performance in the inductive setting, even in the presence of distribution shifts. Additionally, we corroborate the effectiveness of FLOSS, which benefits from being plug-and-play, and further show that only few unlabeled images can be sufficient for expert selection. Furthermore, the oracle performance shows the potential of the class-experts, paving the way for future research.

Acknowledgment. This research was partially funded by the French Agence Nationale de la Recherche (ANR) with the project SIGHT (ANR-20-CE23-0016) and supported by ELSA - European Lighthouse on Secure and Safe AI funded by the European Union under grant agreement No. 101070617. We sincerely thank Telecom Paris for providing the resources necessary to run our experiments and Nacereddine Laddaoui for his help with infrastructure. We are also grateful to Ivan Lopes for proofreading.

References

[1] Stephen H Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, et al. Promptsource: An integrated development environment and repository for natural language prompts. In *ACL*, 2022. 3

[2] Sule Bai, Yong Liu, Yifei Han, Haoji Zhang, and Yansong Tang. Self-calibrated clip for training-free open-vocabulary segmentation. *arXiv preprint arXiv:2411.15869*, 2024. 2

[3] Walid Bousselham, Felix Petersen, Vittorio Ferrari, and Hilde Kuehne. Grounding everything: Emerging localization properties in vision-language transformers. In *CVPR*, 2024. 2, 5, 8

[4] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. In *NeurIPS*, 2019. 2

[5] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, 2018. 2, 5

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 1

[7] Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *CVPR*, 2023. 5

[8] Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. Plot: Prompt learning with optimal transport for vision-language models. In *ICLR*, 2023. 2

[9] 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. *TPAMI*, 2017. 1

[10] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 1

[11] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *CVPR*, 2023. 5

[12] Seokju Cho, Heeseong Shin, Sunghwan Hong, Anurag Arnab, Paul Hongsuck Seo, and Seungryong Kim. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. In *CVPR*, 2024. 1, 2

[13] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 3, 5

[14] Ulindu De Silva, Didula Samaraweera, Sasini Wanigathunga, Kavindu Kariyawasam, Kanchana Ranasinghe, Muzammal Naseer, and Ranga Rodrigo. Test-time optimization for domain adaptive open vocabulary segmentation. *arXiv preprint arXiv:2501.04696*, 2025. 2

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[16] Reza Esfandiarpoor, Cristina Menghini, and Stephen H Bach. If clip could talk: Understanding vision-language model representations through their preferred concept descriptions. In *EMNLP*, 2024. 3

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results, 2012. 5

[18] Yarin Gal. Uncertainty in deep learning. *PhD Thesis, University of Cambridge*, 2016. 4

[19] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *IJCV*, 2024. 2

[20] Yunhao Ge, Jie Ren, Andrew Gallagher, Yuxiao Wang, Ming-Hsuan Yang, Hartwig Adam, Laurent Itti, Balaji Lakshminarayanan, and Jiaping Zhao. Improving zero-shot generalization and robustness of multi-modal models. In *CVPR*, 2023. 3

[21] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 2, 3

[22] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*, 2022. 3

[23] Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. Pay attention to your neighbours: Training-free open-vocabulary semantic segmentation. In *WACV*, 2025. 1, 2, 3, 5, 8

[24] Cong Han, Yujie Zhong, Dengjie Li, Kai Han, and Lin Ma. Open-vocabulary semantic segmentation with decoupled one-pass network. In *ICCV*, 2023. 2

[25] Dapeng Hu, Jian Liang, Jun Hao Liew, Chuhui Xue, Song Bai, and Xinchao Wang. Mixed samples as probes for unsupervised model selection in domain adaptation. In *NeurIPS*, 2023. 7

[26] Xuefeng Hu, Ke Zhang, Lu Xia, Albert Chen, Jiajia Luo, Yuyin Sun, Ken Wang, Nan Qiao, Xiao Zeng, Min Sun, et al. Reclip: Refine contrastive language image pre-training with source free domain adaptation. In *WACV*, 2024. 2

[27] Haiwen Huang, Songyou Peng, Dan Zhang, and Andreas Geiger. Renovating names in open-vocabulary segmentation benchmarks. In *NeurIPS*, 2024. 3

[28] Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022. 2

[29] Yunshi Huang, Fereshteh Shakeri, Jose Dolz, Malik Boudiaf, Houda Bahig, and Ismail Ben Ayed. Lp++: A surprisingly strong linear probe for few-shot clip. In *CVPR*, 2024. 2

[30] Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *ML*, 2021. 4

[31] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, et al. ConceptFusion: Open-set multimodal 3d mapping. In *RSS*, 2023. 2

[32] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*, 2017. 4

[33] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *CVPR*, 2023. 2

[34] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NeurIPS*, 2017. 4

[35] Mengcheng Lan, Chaofeng Chen, Yiping Ke, Xinjiang Wang, Litong Feng, and Wayne Zhang. Proxyclip: Proxy attention improves clip for open-vocabulary segmentation. In *ECCV*, 2024. 2

[36] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 2

[37] Yi Li, Hualiang Wang, Yiqun Duan, Jiheng Zhang, and Xiaomeng Li. A closer look at the explainability of contrastive language-image pre-training. *PR*, 2025. 2, 5

[38] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *CVPR*, 2023. 1, 2

[39] Yuqi Lin, Minghao Chen, Wenxiao Wang, Boxi Wu, Ke Li, Binbin Lin, Haifeng Liu, and Xiaofei He. Clip is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation. In *CVPR*, 2023. 3

[40] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. In *ICLR*, 2023. 3

[41] Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. Revisiting the calibration of modern neural networks. In *NeurIPS*, 2021. 2

[42] 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 *CVPR*, 2014. 5

[43] Jishnu Mukhoti, Tsung-Yu Lin, Omid Poursaeed, Rui Wang, Ashish Shah, Philip HS Torr, and Ser-Nam Lim. Open vocabulary semantic segmentation with patch aligned contrastive learning. In *CVPR*, 2023. 1

[44] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*, 2017. 5

[45] Sarah Parisot, Yongxin Yang, and Steven McDonagh. Learning to name classes for vision and language models. In *CVPR*, 2023. 3

[46] Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. In *ICCV*, 2023. 3

[47] 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 *ICML*, 2021. 1, 2, 3, 5

[48] Kanchana Ranasinghe, Brandon McKinzie, Sachin Ravi, Yinfei Yang, Alexander Toshev, and Jonathon Shlens. Perceptual grouping in contrastive vision-language models. In *ICCV*, 2023. 2

[49] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *CVPR*, 2022. 2

[50] Karsten Roth, Jae Myung Kim, A Koepke, Oriol Vinyals, Cordelia Schmid, and Zeynep Akata. Waffling around for performance: Visual classification with random words and broad concepts. In *ICCV*, 2023. 3

[51] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. Learning to retrieve prompts for in-context learning. In *NAACL-HLT*, 2022. 3

[52] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *CVPR*, 2021. 5

[53] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. In *NeurIPS*, 2022. 5

[54] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In *NeurIPS*, 2022. 2, 3

[55] Julio Silva-Rodriguez, Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. A closer look at the few-shot adaptation of large vision-language models. In *CVPR*, 2024. 2

[56] Vladan Stojnic, Yannis Kalantidis, and Giorgos Tolias. Label propagation for zero-shot classification with vision-language models. In *CVPR*, 2024. 2

[57] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *ICCV*, 2021. 1

[58] Weijie Tu, Weijian Deng, and Tom Gedeon. A closer look at the robustness of contrastive language-image pre-training (clip). In *NeurIPS*, 2023. 2

[59] Weijie Tu, Weijian Deng, Dylan Campbell, Stephen Gould, and Tom Gedeon. An empirical study into what matters for calibrating vision-language models. In *ICML*, 2024. 2

[60] Antonin Vobecky, Oriane Siméoni, David Hurych, Spyridon Gidaris, Andrei Bursuc, Patrick Pérez, and Josef Sivic. Pop-3d: Open-vocabulary 3d occupancy prediction from images. In *NeurIPS*, 2023. 3

[61] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019. 4

[62] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021. 4

[63] Feng Wang, Jieru Mei, and Alan Yuille. Sclip: Rethinking self-attention for dense vision-language inference. In *ECCV*, 2024. 1, 5, 8

[64] Zhengbo Wang, Jian Liang, Lijun Sheng, Ran He, Zilei Wang, and Tieniu Tan. A hard-to-beat baseline for training-free CLIP-based adaptation. In *ICLR*, 2024. 2

[65] Monika Wysoczańska, Michaël Ramamonjisoa, Tomasz Trzciński, and Oriane Siméoni. Clip-diy: Clip dense infer-

ence yields open-vocabulary semantic segmentation for-free. In *WACV*, 2024. 5

[66] Monika Wysoczańska, Oriane Siméoni, Michaël Ramamonjisoa, Andrei Bursuc, Tomasz Trzciński, and Patrick Pérez. Clip-dinoiser: Teaching clip a few dino tricks for open-vocabulary semantic segmentation. In *ECCV*, 2024. 1, 2, 3, 5

[67] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *CVPR*, 2019. 2

[68] Bin Xie, Jiale Cao, Jin Xie, Fahad Shahbaz Khan, and Yanwei Pang. Sed: A simple encoder-decoder for open-vocabulary semantic segmentation. In *CVPR*, 2024. 2, 3

[69] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021. 1

[70] Renchunzi Xie, Ambroise Odonnat, Vasilii Feofanov, Weijian Deng, Jianfeng Zhang, and Bo An. Mano: Exploiting matrix norm for unsupervised accuracy estimation under distribution shifts. In *NeurIPS*, 2024. 4, 7

[71] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *CVPR*, 2022. 5

[72] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Yi Wang, Yu Qiao, and Weidi Xie. Learning open-vocabulary semantic segmentation models from natural language supervision. In *CVPR*, 2023. 2

[73] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *CVPR*, 2023. 2

[74] Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. Side adapter network for open-vocabulary semantic segmentation. In *CVPR*, 2023. 2

[75] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020. 5

[76] Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. In *NeurIPS*, 2023. 2

[77] Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of clip for few-shot classification. In *ECCV*, 2022. 2

[78] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *IJCV*, 2019. 5

[79] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *ECCV*, 2022. 1, 2, 3, 5

[80] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022. 2

[81] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022.

[82] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *ICCV*, 2023. 2

[83] Xiangyang Zhu, Renrui Zhang, Bowei He, Aojun Zhou, Dong Wang, Bin Zhao, and Peng Gao. Not all features matter: Enhancing few-shot clip with adaptive prior refinement. In *ICCV*, 2023. 2