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

Recent Open-Vocabulary Semantic Segmentation (OVSS) models extend the CLIP model to segmentation while maintaining the use of multiple templates (e.g., `a photo of <class>`, `a sketch of a <class>`, etc.) for constructing class-wise averaged text embeddings, acting as a classifier. In this paper, we challenge this status quo and investigate the impact of templates for OVSS. Empirically, we observe that for each class, there exist single-template classifiers significantly outperforming the conventional averaged classifier. We refer to them as class-experts. Given access to unlabeled images and without any training involved, we estimate these experts by leveraging the class-wise prediction entropy of single-template classifiers, selecting as class-wise experts those which yield the lowest entropy. All experts, each specializing in a specific class, collaborate in a newly proposed fusion method to generate more accurate OVSS predictions. Our plug-and-play method, coined FLOSS*, is orthogonal and complementary to existing OVSS methods, offering a "free lunch" to systematically improve OVSS without labels and additional training. Extensive experiments demonstrate that* FLOSS *consistently boosts state-of-the-art methods on various OVSS benchmarks. Moreover, the selected expert templates can generalize well from one dataset to others sharing the same semantic categories, yet exhibiting distribution shifts. Additionally, we obtain satisfactory improvements under a low-data regime, where only a few unlabeled images are available. Our code is available at* https://github.com/yasserben/FLOSS.

1. Introduction

Over the last decade, advances in deep neural networks, driven by the growing amount of training data, have made possible the exhaustive scene understanding task of semantic segmentation – assigning semantics to pixels of input images. Initially confined to a closed-set setup with predefined semantic categories [9, 10, 56, 68], recent models leverage breakthroughs in vision-language alignment [46] to overcome these constraints and advance toward segmentation with open-vocabulary semantics where the semantic categories of interest can be defined dynamically at runtime rather than being fixed in advance.

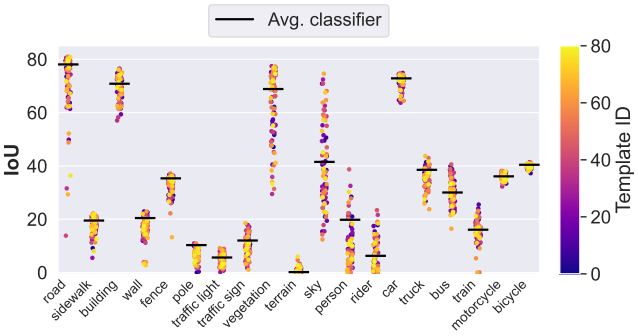


Figure 1. **Using individual vs average template classifiers.** Empirically, we observe that for each class there exist individual templates (colored dots) that lead to classifiers performing *better* than the popular CLIP classifier, which is built from the averaging of all 80 templates embeddings (—).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

For Open-Vocabulary Semantic Segmentation (OVSS), the CLIP model [46] serves as the *de facto* backbone, which is either retrained [42], fine-tuned [12, 37] or simply tweaked [22, 62, 78] to extend global image-language alignment to denser pixel-language alignment. In our work, we aim to study the fundamental classification paradigm of CLIP, which is adopted by default in OVSS. We therefore seek to examine the most faithful form of OVSS, staying as close as possible to the original CLIP, and we favor methods that keep all CLIP parameters untouched [22, 62, 78]. Without modifying CLIP, MaskCLIP [78] exploits the value-embedding layer and the last linear layer of the global attention pooling layer to obtain dense features for segmentation. CLIP-DINOiser [65] enhances MaskCLIP features by locality distillation from the external DINO [6] network. Differently, NACLIP [22] modifies the self-attention scheme to explicitly encourage spatial consistency in local neighborhoods. Overall, the open-vocabulary power fully stems from CLIP and previous OVSS methods mostly focused on improving the visual encoder while retaining the default classification paradigm with textual embeddings.

CLIP-based classification is done by assessing the cosine similarities of image embeddings, or pixel/patch-wise embeddings in OVSS, with text embeddings representing semantic categories. To improve performance, CLIP uses averaged text embeddings as semantic classifiers, constructed by averaging prompts in the embedding space. A prompt is formed by applying a predefined template (e.g., `a photo of a <class>`) on a class name (e.g., `road`), then passed through the text encoder to obtain a text embedding. Empirically, CLIP uses $M = 80$ well-engineered templates for image classification on ImageNet. For segmentation, the OVSS methods adopt by default the same template-averaging strategy to construct segmentation classifiers. While template-averaging results in an overall good classifier, there is no guarantee that it leads to the optimal solution for discriminating semantic concepts. Toward a better classification strategy, we advocate for a closer study of individual templates and their effects in OVSS – an aspect that has been overlooked in the literature.

Our paper revisits this premise and is motivated by an intriguing empirical observation, depicted in Fig. 1 and further detailed in Sec. 3: for each class, there exist individual templates which lead to better classifiers *on this class* than the M-average classifier using all templates. We call such templates **experts** of the class they excel on, highlighting that such experts are trivial to identify using semantic ground truths.

However, without access to labels, two challenges arise and construct the basis of our work: 1- *How to identify expert templates without ground truth?* 2- *How to fuse individual expert predictions?* To answer these questions, our study demonstrates that unsupervised metrics, particularly entropy, reliably enable label-free identification of experts. This insight leads us to introduce a new task that improves OVSS performance : *training- and label-free template selection for OVSS*. Orthogonal to existing *training-free* OVSS works which focus on tweaking the visual encoder to enhance the visual features for OVSS, our work shifts the attention to the text encoder and proposes a strategy to select better textual template subsets tailored to a specific dataset of unlabeled images. These templates are aimed at improving the performance in an inductive setting, that is, when using them to segment a different subset of the dataset than the one used for selection. 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

VLM transferability. Since the release of CLIP, the impressive zero-shot classification performance [46] and robustness [40, 57, 58] have made it a foundation for several adaptation/transfer learning tasks. Few-shot supervised fine-tuning has garnered considerable attention, where limited data is available for a downstream image classification task. 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 key in this setting, hence some methods address it using prompt learning [8, 32, 79–81], others train additional lightweight MLPs at the output level [18], while some recent methods [28, 54] show that a well-trained linear probing is a competitive baseline. Another line of research explores training-free few-shot adaptation for classification in the presence of labels [63, 76, 82]. The main idea is to build a classifier based on the few-shot data and their labels and then ensemble it with the zero-shot classifier. Unsupervised adaptation has also been explored [25, 27, 55], alongside test-time adaptation, where prompts are learned per test image [53].

Leveraging VLMs for classification-like tasks is a natural direction, as they typically focus on global image recognition with text, which aligns with the way VLMs are pretrained. In contrast, using VLMs for dense prediction tasks like semantic segmentation is more challenging due to the gap between global and pixel-wise understanding. We summarize related efforts in the following.

Open-vocabulary Semantic Segmentation (OVSS). OVSS aims at performing text-based segmentation using VLMs. CLIP-like VLMs appear as a natural choice for this task, however their specific global pooling layers hinder their ability to produce dense pixel-level language-aligned features [30, 78]. A number of approaches propose to densify CLIP without updating its parameters and thus preserving the image-language alignment. They are categorized as *training-free*. MaskCLIP [78] was the first attempt to do this, by approximating the attention pooling layer with a convolutional layer. Subsequent works build upon MaskCLIP and densify CLIP by aggregating

multiple views of the same image [30, 65], integrating priors from other vision foundation models for better pooling [34, 65], extracting different information from the self-attention layers [3, 22, 36], or removing anomaly tokens [2]. Another line of research focuses on fine-tuning CLIP for segmentation tasks. Some approaches use weak supervision through image-level tags, captions [20, 47], or class-agnostic object masks [20, 48]. Others employ full supervision [12, 35, 37, 73] on densely annotated datasets like COCO Stuff [5], following strategies similar to early inductive zero-shot semantic segmentation methods [4, 66]. The latter approaches take various paths: some fine-tune CLIP's image encoder [35, 37, 73], others combine this with additional backbones and segmentation heads [37], while some learn side networks while keeping CLIP frozen [73]. Many other approaches have been proposed in this active research area [23, 67, 71, 72, 75]. While these methods effectively boost segmentation performance, they introduce significant computational and data requirements. Such approaches demand substantial labeled data, computational resources, and careful parameter optimization, limiting their practical deployment in resource-constrained scenarios.

In this work, we consider training-free approaches for preserving their unaltered image-language features from pre-training. We evaluate FLOSS on three approaches: the seminal MaskCLIP [78], the recent CLIP-DINOiser [65] and NACLIP [22].

Prompt in VLMs. CLIP's seamless interface between the image and language domains has inspired the adoption of multiple prompt engineering practices from LLMs to better reveal the knowledge encapsulated in the vision encoder and boost performance. The seminal CLIP work [46] introduced several dataset-specific templates, e.g., `a photo of <class>`, to bridge the distribution gap between training captions and test prompts in zero-shot classification. Subsequent works crowdsourced prompt templates [1], manually changed the class names [20, 59] or prompt templates [21] to better fit a certain task, e.g., object detection [20, 21], occupancy prediction [59]. While effective, manual prompt tuning is a tedious task leading to the emergence of different automated strategies, such as adding random characters or words to the original template [49], enriching class names with WordNet related concepts [19] or synonyms [38], and learning more informative class names [26]. Publicly available LLMs can be used to produce more expressive visual descriptions starting from simple class names or retrieve them from a dataset [50] and further use them as prompts. Prompt embeddings can be learned for fitting a specific task [44] or a test sample [53], integrating connections between training and test distribution. They can also be mined with the help of an external LLM [15]. Our approach is orthogonal to most of these strategies. Starting from a pre-defined set of templates, we select in an unsupervised manner template experts for each class to improve performance. Most of the previous approaches have been proposed for image classification. In this work, we deal with semantic segmentation and study the original ImageNet templates proposed in CLIP which have already been validated 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, to increase robustness and improve overall performance, practitioners average the embeddings of prompts obtained using all available templates [39, 45, 46, 49], 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 [22, 65, 67, 78] typically rely on the M=80 ImageNet templates of CLIP [46] 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 [46, 78], the templates were originally hand-crafted for zero-shot classification on ImageNet [46], 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 the CLIP classifier *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.2 for similar observations across datasets and models.

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

4. Method

Given access to a dataset $\mathcal{D}$ with 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 pre-defined ImageNet templates of CLIP (Sec. 4.1) by leveraging unsupervised metrics and 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.*, $\Omega(\cdot)$) and therefore propose relying on unsupervised metrics which act as performance proxy [61, 69]. We leverage entropy as it is an established measure of uncertainty [17, 29], commonly adopted in deep learning literature [31, 33, 60]. 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 ablation 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 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 $\texttt{softmax}(\cdot)$ 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:

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

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 $\mathcal{H}_{m,k}$ is in ascending order. Finally, $\texttt{Top-N}(\cdot)$ thus returns N template indices corresponding to 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 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,

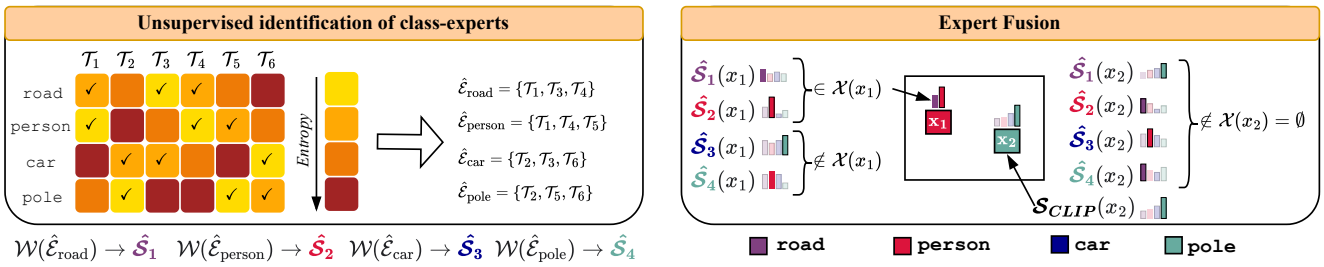


Figure 2. **Overview of FLOSS. Left:** for each class, selected expert templates (✓) construct the estimated set $\hat{\mathcal{E}}$, which classifies the input image and yield the soft-segmentation map $\mathbf{S}$. In this example we set $N = 3$ for the `Top-N(...)` template indices corresponding to 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 $\mathbf{S}_{\text{CLIP}}(x_2)$.

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 fusion strategy. Overall, our method incurs minimal computational overhead since the vision encoder is shared across all experts, and the additional cosine similarity computations of the K class-experts are efficiently performed on GPU.

5. Experiments

Experimental Setup. We evaluate our method on three state-of-the-art open-vocabulary semantic segmentation models: the seminal MaskCLIP [78] and two very recent models CLIP-DINOiser [65] and NACLIP [22]. CLIP-DINOiser trains a convolution layer using only 1K unlabeled images from ImageNet [14] 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. All models utilize CLIP ViT-B/16 backbone. CLIP-DINOiser and MaskCLIP are based on OpenCLIP [11], while NACLIP employs OpenAI's original CLIP [46]. For all models – CLIP-DINOiser, NACLIP, and MaskCLIP– we follow the authors' evaluation protocol. For CLIP-DINOiser, we resize input images to 448 pixels on the shortest side with a sliding window of size 448 and stride 224. For NACLIP and MaskCLIP, images are resized to 336 pixels (except 560 for Cityscapes) with a 224 window and stride 112.

Datasets and Metric. We evaluate across multiple semantic segmentation benchmarks: PASCAL CONTEXT 59 [41] (PC59), PASCAL VOC 20 [16] (VOC20), COCO-Stuff [5] (Stuff), ADE20K [77] (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 [74] and MAPILLARY [43] as well as ACDC [51] 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	CS	VOC20	PC59	ADE	Stuff	Avg
CLIP [46]	6.7	49.1	11.2	3.2	5.7	15.2
GroupViT [70]	11.1	79.7	23.4	9.2	15.3	27.7
CLIP-Surgery [36]	31.4	–	–	12.9	21.9	–
GEM [3]	–	–	32.6	15.7	–	–
SCLIP [62]	32.2	80.4	34.2	16.1	22.4	37.1
CLIP-DIY [64]	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 [52]	21.1	57.8	22.3	11.2	14.8	25.4
MaskCLIP [78]	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 [22]	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 [65]	31.1	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). We highlight **best** performance.

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 by CLIP to engineer the set of 80 templates used in all models. The benefit of our method 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. Furthermore, 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 $\mathcal{E}_k$:

$$\hat{\rho}_k = 100 \times \frac{|\hat{\mathcal{E}}_k \cap \mathcal{E}_k|}{|\hat{\mathcal{E}}_k|} \,. \tag{8}$$

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.3, showing some variability.

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. Bold highlights the **best** performance.

Generalization to unseen datasets. We now evaluate the generalization of experts to other unseen datasets. 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.

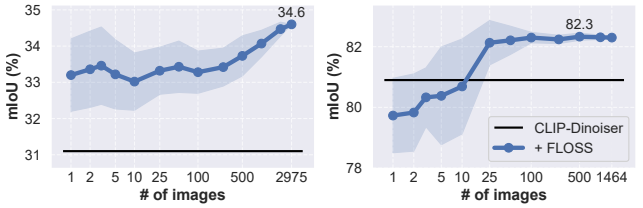
5.2. Ablation Study

Effect of the number of images. 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 de-

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 4. **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.

fault 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.



(a) Cityscapes (b) PASCAL VOC 20

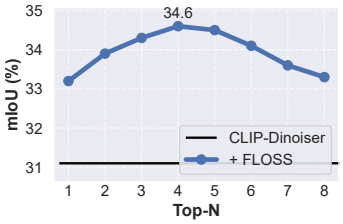
Figure 3. **Effect of the number of images.** We evaluate CLIP-DINOiser with FLOSS on urban scenes (a) and general object segmentation (b), when varying the number of images used to identify class-experts. Results show the average of 10 seeds, with standard deviation. Despite larger variation, we note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

Fusion strategies for expert predictions. We investigate alternative strategies to fuse the expert predictions (*cf*., Sec. 4.2), reporting performance on Cityscapes in Tab. 4. 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

Metric	mIoU
Avg. Probability	34.4
MaNo [69]	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 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.

in Fig. 4a.

Avg. Probability computes the average probability assigned to class k across all pixels predicted as class k. This writes:

$$\text{Avg.Prob}_k(\mathcal{T}_m) = \frac{1}{C_{m,k}} \sum_{i=1}^{C_{m,k}} \text{softmax}(\mathbf{q}_i)_k \tag{9}$$

Higher values indicate better performance in this case.

MaNo [69] is adapted as it was originally designed for assessing general model performance in image classification, to evaluate class-wise performance in semantic segmentation. This approach leverages the low density separation (LDS) assumption, which posits that optimal decision boundaries should pass through regions of low data density. The LDS principle suggests a direct correlation between the magnitude of logit values and a model's generalization capabilities, so the higher the MaNo score, the better it is. For each class k, we compute the average L_p norm of the probability vectors for pixels predicted as that class:

$$\text{MaNo}_k(\mathcal{T}_m) = \left(\frac{1}{C_{m,k} K} \sum_{i=1}^{C_{m,k}} \sum_{k=1}^{K} |\text{softrun}(\mathbf{q}_i)_k|^p \right)^{\frac{1}{p}} \tag{10}$$

Where softrun [69] is the normalization function used to avoid error accumulation, and we use $p = 4$ in practice, as done in the original paper.

"ITI" [24] 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. For this metric, higher scores represent better results. A detailed formulation of *"ITI"* is provided in Appendix A.4.

Overall, our entropy measure (*cf*. Eq. (5)) is the most effective, although *Avg. Probability* performs on par.

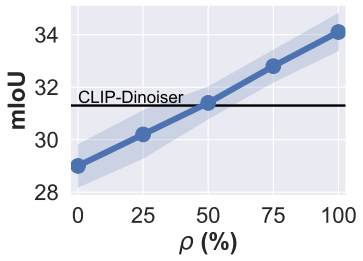


Figure 5. **Effect of class-expert quality.** We report the performance of FLOSS in the oracle-based setting, where each class-wise set of 4 experts contains a ratio (ρ) of true experts. The experiment is conducted on Cityscapes using the CLIP-DINOiser model.

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 labels 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 Cityscapes 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. An interesting observation is that 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 of our method using only the *best true class-expert* (i.e., $\{\mathcal{T}_{\tilde{m}} \mid \tilde{m} \in \texttt{Top-N}\big(\Omega_k(\boldsymbol{W}(\mathcal{T}_m))\big)\}$). On Cityscapes, the oracle variant with $N \in \{1, 2, 3, 4\}$ achieves respectively 37.1/37.0/37.3/37.5% mIoU, showing the potential for further improvement assuming the identification of better experts.

Impact of different backbones. Tab. 5 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.

Method	ViT-B/16			ViT-L/14		
	CS	PC59	Avg.	CS	PC59	Avg.
SCLIP [62]	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 [22]	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 5. **Effect of backbone.** Results (mIoU) show that NACLIP + FLOSS outperforms NACLIP across ViT-B/16 and ViT-L/14 CLIP backbones.

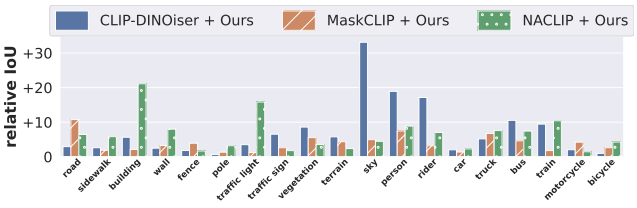


Figure 6. **Performance of best-class experts.** Performance gains (IoU) when using best class-experts versus standard template-averaging. Results are shown for CLIP-DINOiser, MaskCLIP and NACLIP. The bars represent the IoU difference between using best class-experts (selected with ground-truth labels)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

6. Perspectives and Future Work

Upon testing different unsupervised metrics for class-experts identification, we find that simply using entropy is a strong estimator candidate. Future research might construct stronger estimators that are better proxies of prediction accuracy. We believe that the avenue of templates is extremely promising, and that considerable performance gains for OVSS lie in their identification. The gap is yet far from being filled, as seen in Figure 6, which shows for each class the maximum improvement *w.r.t.* the averaged-templates model that can be achieved, only using the **best class-experts**. For instance, selecting the best sky-expert would results in more than 30 percentage points IoU! Even these "upper-bounds" can be pushed further by harnessing other templates than the 80 ones we use, or augmented versions of them.

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 intrigu-

ing 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. Further the oracle performance shows the potential of the class-experts, paving the way for future research. Our research contributes to the community's work in utilizing large-scale pre-trained foundation models for essential downstream tasks.

Acknowledgment. This research was partially funded by the French Agence Nationale de la Recherche (ANR) with the project SIGHT (ANR-20-CE23-0016). We sincerely thank Telecom Paris for providing the resources necessary to run our experiments and Nacereddine Laddaoui for his invaluable 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. 3

[3] Walid Bousselham, Felix Petersen, Vittorio Ferrari, and Hilde Kuehne. Grounding everything: Emerging localization properties in vision-language transformers. In *CVPR*, 2024. 3, 6, 8

[4] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. In *NeurIPS*, 2019. 3

[5] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, 2018. 3, 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. 6

[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, 3

[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. 4, 5

[14] 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

[15] 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

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

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

[18] 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

[19] 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

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

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

[22] Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. Pay attention to your neighbours: Training-free open-vocabulary semantic segmentation. In *WACV*, 2025. 1, 3, 5, 6, 8

[23] Cong Han, Yujie Zhong, Dengjie Li, Kai Han, and Lin Ma. Open-vocabulary semantic segmentation with decoupled one-pass network. In *ICCV*, 2023. 3

[24] 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

[25] 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

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

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

[28] 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

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

[30] 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, 3

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

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

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

[34] 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. 3

[35] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 3

[36] 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. 3, 6

[37] 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, 3

[38] 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

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

[40] 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

[41] 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

[42] 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

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

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

[45] 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

[46] 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, 6, 12

[47] Kanchana Ranasinghe, Brandon McKinzie, Sachin Ravi, Yinfei Yang, Alexander Toshev, and Jonathon Shlens. Perceptual grouping in contrastive vision-language models. In *ICCV*, 2023. 3

[48] 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. 3

[49] 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

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

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

[52] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. In *NeurIPS*, 2022. 6

[53] 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

[54] 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

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

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

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

[58] 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

[59] 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

[60] 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

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

[62] Feng Wang, Jieru Mei, and Alan Yuille. Sclip: Rethinking self-attention for dense vision-language inference. In *ECCV*, 2024. 1, 6, 8

[63] 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

[64] Monika Wysoczańska, Michaël Ramamonjisoa, Tomasz Trzciński, and Oriane Siméoni. Clip-diy: Clip dense inference yields open-vocabulary semantic segmentation for-free. In *WACV*, 2024. 6

[65] 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, 3, 5, 6

[66] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *CVPR*, 2019. 3

[67] 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. 3

[68] 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

[69] 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

[70] 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. 6

[71] 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. 3

[72] 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. 3

[73] Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. Side adapter network for open-vocabulary semantic segmentation. In *CVPR*, 2023. 3

[74] 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

[75] Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. *Advances in Neural Information Processing Systems*, 2023. 3

[76] 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

[77] 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

[78] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *ECCV*, 2022. 1, 2, 3, 5, 6

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

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

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

[82] 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

Appendix

A. Additional experiments

In this section, we provide additional experimental results and analyses that complement our main findings. First, we showcase the consistent improvements brought by our method across different models and datasets (Section A.1). Then we extend our empirical analysis of class-expert templates to additional datasets and models (Section A.2). We follow with a detailed assessment of our expert identification method's accuracy (Section A.3) and an in-depth explanation of the different unsupervised metrics we investigated (Section A.4). Finally, we present a comprehensive study of the relationship between entropy and IoU across different settings (Section A.5).

A.1. Performance improvements across models

Fig. 7 provides a graphical view across datasets of the improvement in mIoU when plugging FLOSS on existing OVSS models. This demonstrates the general applicability of our approach, which can be seamlessly integrated with existing models without requiring any additional training.

A.2. Empirical observations on other settings

The foundation of our work is the observation that there exist class-wise expert templates. We extend here our initial analysis to 4 datasets, for both CLIP-DINOiser in Fig. 8 and NACLIP in Fig. 9, showing that our initial observations hold in these settings. Interestingly, we note that for PASCAL CONTEXT 59 averaging all CLIP templates performs already very well and most individual template underperform the averaging CLIP templates. Arguably, this might be related to the proximity of PASCAL VOC 20 with ImageNet for which the templates of CLIP were highly engineered [46].

A.3. Quality of experts

We report the quality of our estimated top-4 experts using entropy as unsupervised metric for both CLIP-DINOiser in Fig. 10 and NACLIP in Fig. 11. In details, for each class $k \in [1, K]$ the quality is computed as the normalized intersection between the set of estimated experts $\hat{\mathcal{E}}_k$ and the true set of experts $\mathcal{E}_k$, as detailed in Eq. (8). For each dataset, we also report the average accuracy over all classes, *i.e.*, $\frac{1}{K}\sum_k \hat{\rho}_k$, shown as a horizontal line.

While we observe a high variability of the quality across classes, our average top-4 quality is around 50%, meaning that half of our estimated expert templates are usually actual experts of the class. We also note that the identification of experts on PASCAL VOC 20 is less accurate, which may stem from the rare class-experts on said dataset and the high performance of the average CLIP templates.

A.4. Unsupervised metrics for expert identification

Given a template classifier $\boldsymbol{W}(\mathcal{T}_m)$ making predictions on unlabeled images, we detail the computation of Inter-to-Intra (ITI) :

Inter-to-Intra (ITI) is a class ratio that quantifies feature separability by examining the relationship between inter-class separation and intra-class compactness for each class k. The intra-class compactness measure $D_{\text{intra,k}}(\mathcal{T}_m)$ calculates the average squared Euclidean distance between the global class feature centroid $\bar{f}_k$ and the class-specific features $\bar{f}_k^i$ (denoting the average feature for class k in image i) across all N training images, capturing feature variability within the class.

Conversely, the inter-class separability measure $D_{\text{inter,k}}(\mathcal{T}_m)$ computes the average squared distance between the target class centroid $\bar{f}_k$ and all other class centroids $\bar{f}_i$, measuring how distinctly a class is represented in feature space. The ITI ratio $\text{ITI}_k(\mathcal{T}_m)$ then provides a normalized metric where higher values indicate well-separated and compact feature representations, with $\bar{f}_k$ representing the class k's feature centroid across the entire dataset, N representing the training dataset size, and K signifying the total number of semantic classes.

$D_{\text{intra,k}}(\mathcal{T}_m)$, $D_{\text{inter,k}}(\mathcal{T}_m)$ and $\text{ITI}_k(\mathcal{T}_m)$ write as:

$$
\begin{aligned}
D_{\text{intra,k}}(\mathcal{T}_m) &= \frac{1}{N}\sum_{i=1}^{N} \|\bar{f}_k - \bar{f}_k^i\|_2^2 \\
D_{\text{inter,k}}(\mathcal{T}_m) &= \frac{1}{N-1}\sum_{i=1}^{N-1} \|\bar{f}_k - \bar{f}_i\|_2^2 \\
\text{ITI}_k(\mathcal{T}_m) &= \frac{D_{\text{inter,k}}(\mathcal{T}_m)}{D_{\text{intra,k}}(\mathcal{T}_m)}
\end{aligned}
\tag{11}
$$

A.5. Studying the relation of entropy and IoU

We analyze the relationship between entropy and IoU across different models and datasets. Figs. 12 and 13 show this relationship for CLIP-DINOiser, while Figs. 14 and 15 present the same analysis for NACLIP. For each class, we plot the entropy and IoU of individual templates (colored dots) against the performance of the original CLIP averaged-templates (dotted line). The number of valid templates (those predicting at least one pixel for the class) is indicated in parentheses next to each class name.

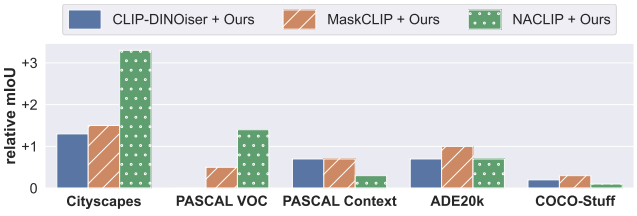
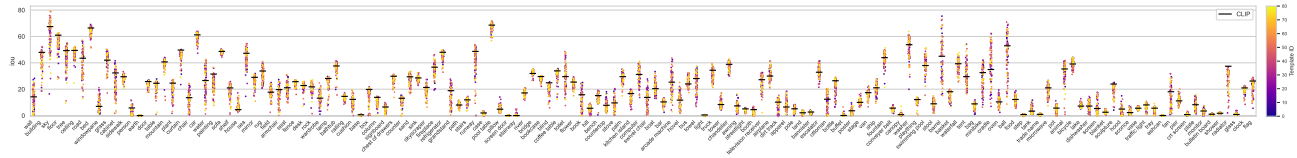
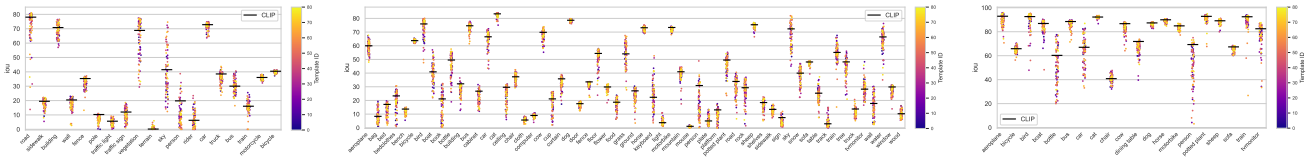


Figure 7. **Performance boost with FLOSS**. We report the mIoU difference when using FLOSS on top of an existing OVSS model. Our training-free method consistently improves all OVSS models across different datasets through class-expert template identification.

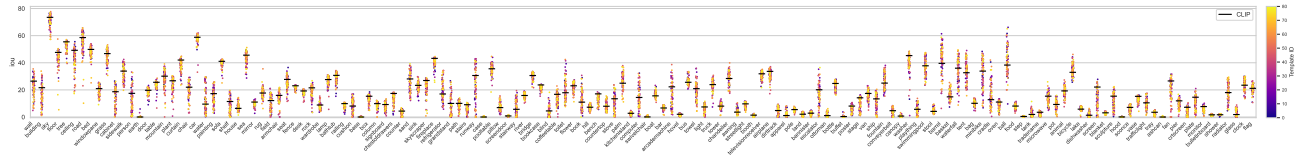


(a) ADE20K

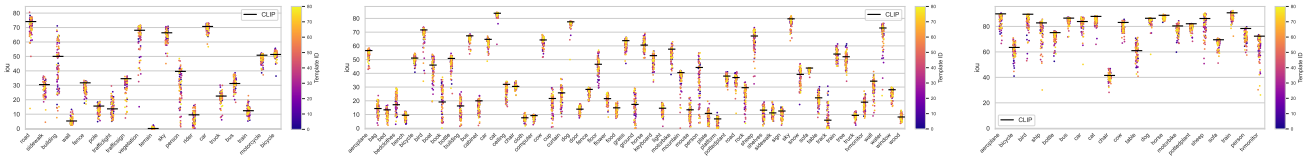


(b) Cityscapes (c) PASCAL CONTEXT 59 (d) PASCAL VOC 20

Figure 8. **Individual template vs average templates (CLIP-DINOiser).**

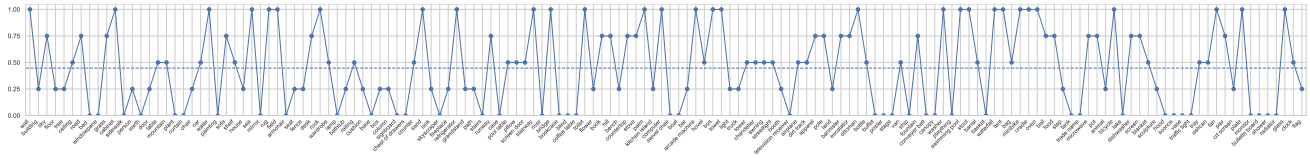


(a) ADE20K

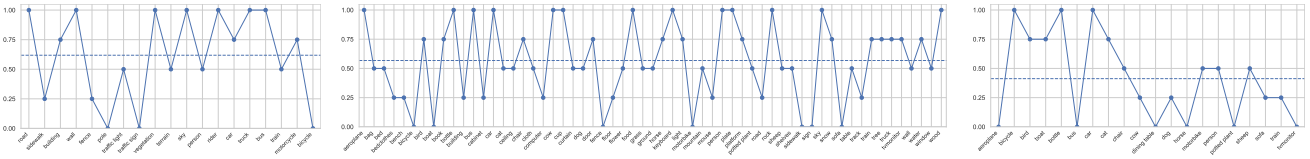


(b) Cityscapes (c) PASCAL CONTEXT 59 (d) PASCAL VOC 20

Figure 9. **Individual template vs average templates (NACLIP).**

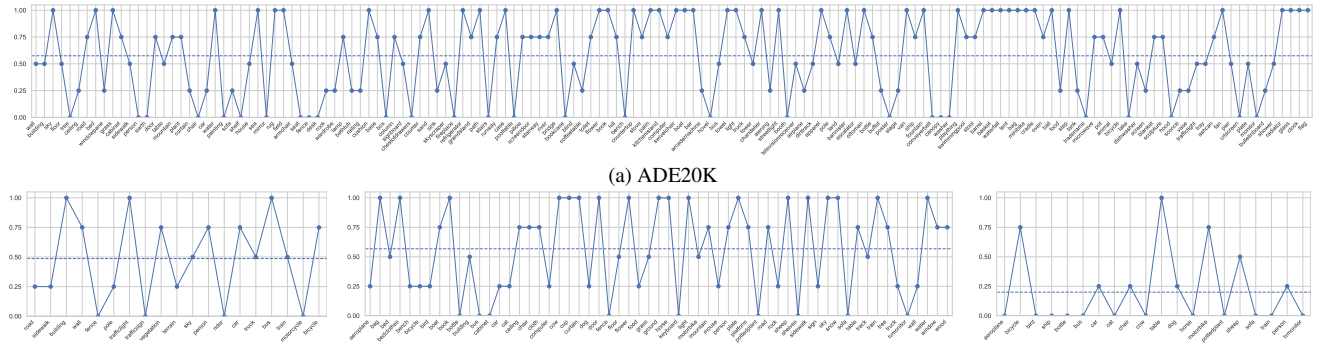


(a) ADE20K



(b) Cityscapes (c) PASCAL CONTEXT 59 (d) PASCAL VOC 20

Figure 10. **Quality of the estimated templates (CLIP-DINOiser).** For each class k, we report the accuracy of the estimated top-4 experts $\hat{\mathcal{E}}_k$ as the normalized intersection with the set of true experts from that class, i.e., $\mathcal{E}_k$. The dash line indicates the average across classes.



(a) ADE20K

(b) Cityscapes (c) PASCAL CONTEXT 59 (d) PASCAL VOC 20

Figure 11. **Quality of the estimated templates (NACLIP).** For each class k, we report the accuracy of the estimated top-4 experts $\hat{\mathcal{E}}_k$ as the normalized intersection with the set of true experts from that class, i.e., $\mathcal{E}_k$. The dash line indicates the average across classes.

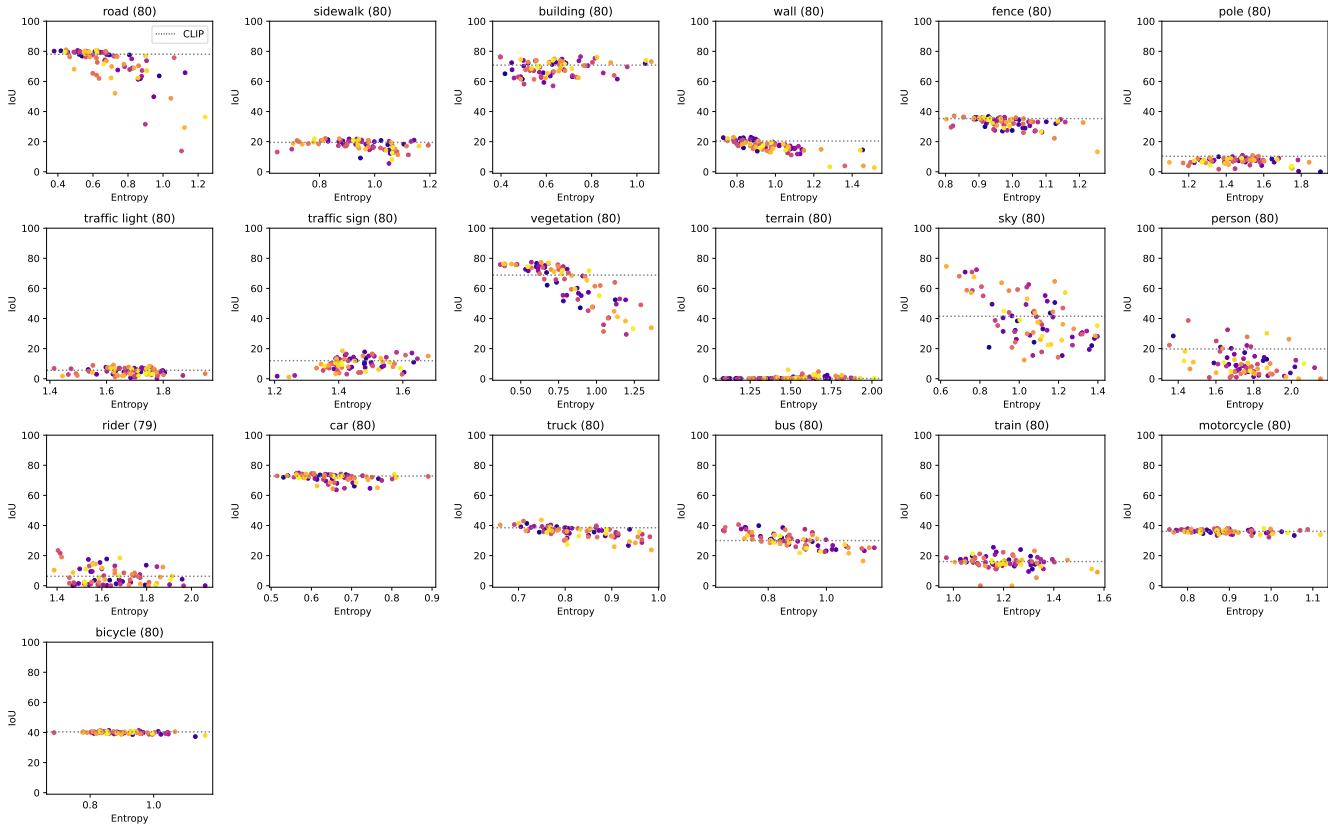


Figure 12. **Entropy and IoU per template (CLIP-DINOiser, Cityscapes).** Each plot reports, for a given class, the entropy and IoU of all individual template (colored dots) as well as the original CLIP averaged-templates performance (dotted line). Note that we consider templates valid only if they predict more than 0 pixels for that class, and therefore report the number of valid templates in parenthesis next to the class name.

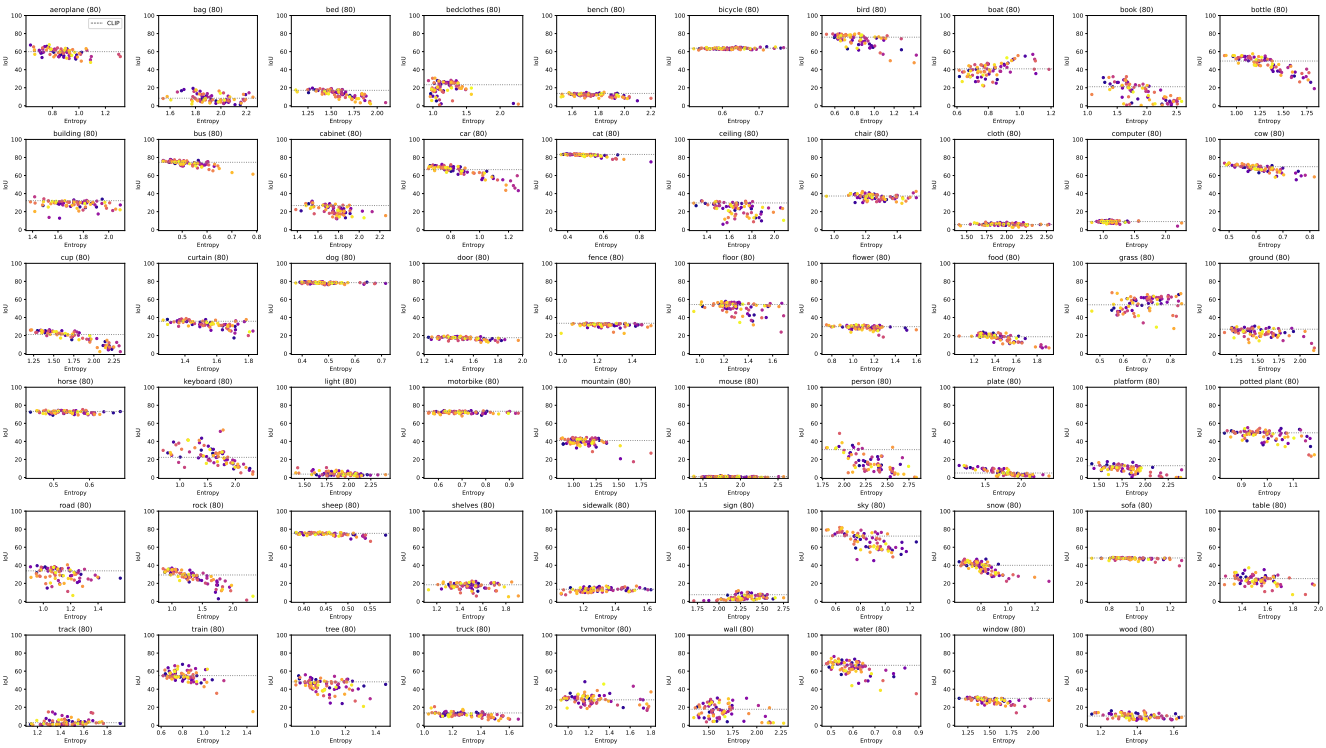


Figure 13. **Entropy and IoU per template (CLIP-DINOiser, PASCAL CONTEXT 59).** Each plot reports, for a given class, the entropy and IoU of all individual template (colored dots) as well as the original CLIP averaged-templates performance (dotted line). Note that we consider templates valid only if they predict more than 0 pixels for that class, and therefore report the number of valid templates in parenthesis next to the class name.

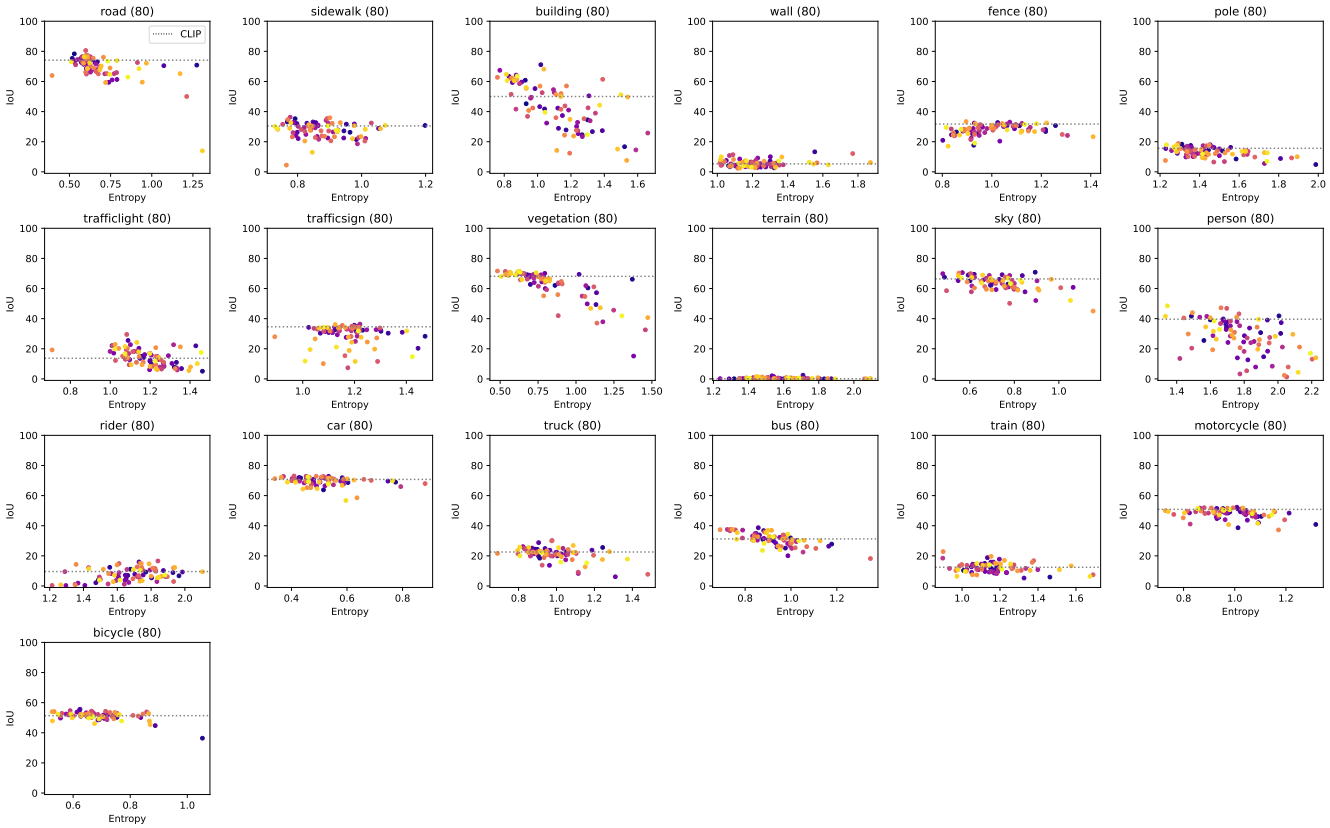


Figure 14. **Entropy and IoU per template (NACLIP, Cityscapes).** Each plot reports, for a given class, the entropy and IoU of all individual template (colored dots) as well as the original CLIP averaged-templates performance (dotted line). Note that we consider templates valid only if they predict more than 0 pixels for that class, and therefore report the number of valid templates in parenthesis next to the class name.

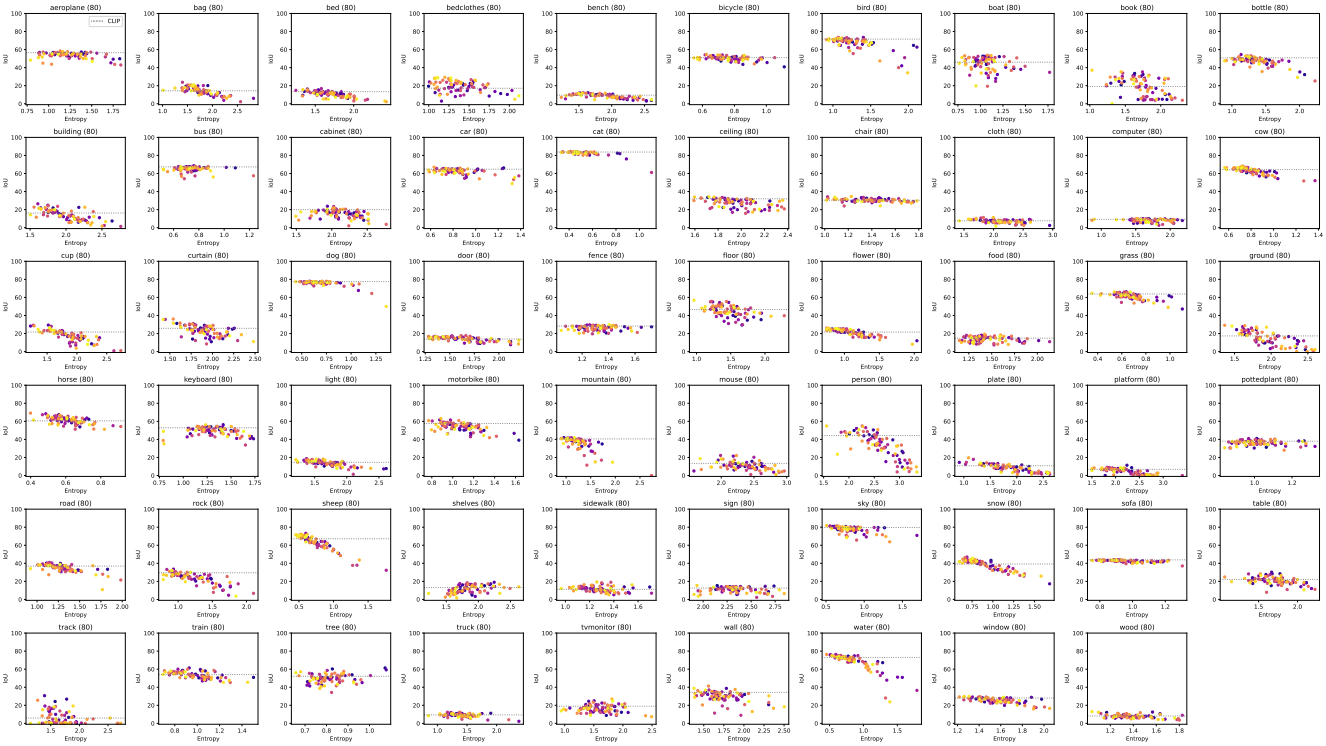


Figure 15. **Entropy and IoU per template (NACLIP, PASCAL CONTEXT 59).** Each plot reports, for a given class, the entropy and IoU of all individual template (colored dots) as well as the original CLIP averaged-templates performance (dotted line). Note that we consider templates valid only if they predict more than 0 pixels for that class, and therefore report the number of valid templates in parenthesis next to the class name.