CLIP-DIY: CLIP Dense Inference Yields Open-Vocabulary Semantic Segmentation For-Free

Monika Wysoczańska[1*] Michaël Ramamonjisoa[2] Tomasz Trzciński[1,4,5]
Oriane Siméoni[3]
[1]Warsaw University of Technology, [2]Meta AI, [3]Valeo.ai, [4]Tooploox, [5]IDEAS NCBR

Abstract

The emergence of CLIP has opened the way for open-world *image perception. The zero-shot classification capabilities of the model are impressive but are harder to use for dense tasks such as image segmentation. Several methods have proposed different modifications and learning schemes to produce dense output. Instead, we propose in this work an open-vocabulary semantic segmentation method, dubbed* `CLIP-DIY`*, which does not require any additional training or annotations, but instead leverages existing unsupervised object localization approaches. In particular,* `CLIP-DIY` *is a multi-scale approach that directly exploits CLIP classification abilities on patches of different sizes and aggregates the decision in a single map. We further guide the segmentation using foreground/background scores obtained using unsupervised object localization methods. With our method, we obtain state-of-the-art zero-shot semantic segmentation results on PASCAL VOC and perform on par with the best methods on COCO.*

Figure 1. **Simple and accurate semantic segmentation for-free using** `CLIP-DIY`. Our method processes regions of an image *separately*, *in parallel* to produce per-patch-per-class scores, given a set of prompts that can take *any length*. `CLIP-DIY` does not require re-training, and can therefore immediately adapt to any new vocabulary.

1. Introduction

The task of *semantic segmentation*, which aims at predicting the class of every pixel in an image has been widely tackled using fully-supervised approaches [10, 16, 46], which require tedious and therefore expensive per-pixel annotations. Moreover, semantic segmentation has typically been performed with a *finite set of classes* [3, 13, 26] describing the types of objects that should be discovered in images. However, using a fixed number of classes is limiting for real-world applications as interesting object classes may vary in time and per application – having to perform annotation and re-training on new classes is expensive and sub-optimal. In this context, recent advances in Visual Language Models (VLMs) [1, 21, 23] have paved the way to *open-vocabulary* perception. Indeed, VLMs, trained with cheap and widely available image-text pairs, e.g. captions, offer new possibilities to describe images with a *large and open vocabulary*. Using such models can help alleviate both the problem of supervision and finite vocabulary.

In particular, the popular VLM Open-CLIP [21] has been exploited to perform *open-vocabulary perception*. While it has high performance on image classification, applying OpenCLIP on dense tasks is more challenging [59]. In order to improve CLIP segmentation abilities, different methods have been proposed to modify the architecture [37, 59], to add new modules [8, 27, 31] or to train new specifically designed models [37, 55] from scratch. Instead, we propose `CLIP-DIY`, a new *zero-shot open-vocabulary semantic segmentation* approach which makes direct use of the high-performance image classification properties of CLIP, *does not* need architecture changes or additional training. In particular, our method applies CLIP to a multi-scale grid of patches and aggregates the information into a single prediction map.

*Corresponding author: monika.wysoczanska.dokt@pw.edu.pl

Moreover, in order to further improve the quality of the localization of our predicted maps, we propose with `CLIP-DIY` to leverage the recent efforts in *unsupervised object localization*. This task aims at discovering every object depicted in images in a class-agnostic fashion and thus *without manual annotation*. Recent methods [4, 42, 44, 45, 50, 52] achieve impressive localization results by leveraging self-supervised features [7, 9]. While some methods discover one object per image [44, 52], more recent ones try to highlight all objects in an image [42, 45, 50].

We therefore propose to leverage those capabilities in `CLIP-DIY`, by guiding CLIP predictions with a very lightweight unsupervised foreground/background strategy which greatly improves the predictions' quality.

To summarize, our novel approach `CLIP-DIY` best leverages the open-world classification capabilities of CLIP and the high-quality of unsupervised object localization approaches yielding the following contributions:

- We introduce `CLIP-DIY`, a novel, simple technique for open-vocabulary semantic segmentation which *does not require additional training* or any *pixel-level annotation* but instead leverages strong self-supervised features with good localization properties combined with CLIP.
- Our multi-scale approach which uses simply CLIP as it was designed —for image classification— enables `CLIP-DIY` to produce well-localized predictions.
- We demonstrate that unsupervised foreground/background methods can be effectively used to provide spatial guidance to CLIP predictions.
- We achieve a new state-of-the-art zero-shot open-vocabulary semantic segmentation on PASCAL VOC dataset and perform on par with the best methods on COCO.
- We perform an extensive validation of the design of our method, and show that it is robust as it can be directly applied to in-the-wild open-world segmentation.

2. Related work

In this section, we discuss previous work related to ours, starting in Sec. 2.1 with zero-shot open-vocabulary semantic methods, then following with unsupervised object localization methods in Sec. 2.2. Finally, in Section 2.3 we focus more specifically on works that leverage a combination of self-supervised learned features and CLIP to perform open-world segmentation.

2.1. Zero-shot open-vocabulary semantic segmentation

With the aim to build generalizable models, *zero-shot* methods for semantic segmentation [5, 15, 20, 24, 25, 32, 54, 54, 58] propose to extend models trained in a fully-supervised fashion on a set of *seen* classes to new *unseen* classes. Many leverage relationships encoded in pre-trained word embeddings [30, 33] to discover new unseen concepts. Such methods require annotation for the seen classes while we aim to use no pixel-level annotation.

Alternatively, *open-vocabulary* approaches [53] exploit image-text alignment without needing to pre-define vocabulary. Several [28, 28, 31, 38, 59] build on top of the popularCLIP [21, 35] model which showed impressive *global* text-image alignment properties but lacks localization quality [22]. Using class-agnostic object masks, it is possible to learn to align the embeddings of selected pixels with text [11, 14, 38], but at the cost of pixel-level annotations. Without extra supervision, MaskCLIP [59] alters the last pooling layer of CLIP to produce dense predictions and use them as pseudo-labels to train a segmentation model, forming MaskCLIP+. Using only image captions—cheap to acquire and widely available—as supervision, [27, 28, 55, 56] learn local alignment between image *regions* and paired text with contrastive objectives. Regions are formed using a learnt hierarchical mechanism [55], cross-attention based clustering [28], using a clustering head trained with diverse view [27] or slot-attention [56]. PACL [31] adds an embedder module that learns affinity between patches and the global text token, TCL [8] builds a new local contrastive objective which directly aligns captions with pre-selected patches and ViewCO [39] proposes a multi-view consistent learning approach. CLIPpy [37] proposes to fully re-train CLIP with a few well-designed modifications to directly obtained denser features. Alternatively, ReCO [43] builds prototypes of the desired vocabulary (using CLIP-based retrieval) which are then used for co-segmentation.

Rather than modifying the architecture of CLIP or training a new module specifically designed to densify its outputs, we propose to directly use *as is* the good classification ability of the model. Indeed, we perform a dense multi-scale patch classification. By doing so, our method can easily be adapted to any new dataset, with any new vocabulary.

2.2. Unsupervised object localization

Interestingly, recent works have shown that ViT features trained in a self-supervised fashion [6, 7, 9] on images–with no human-made annotations–have good localization properties [44, 52]. Such properties have been exploited to tackle the problem of *unsupervised object localization* [47, 48] which requires to localize objects–any object–depicted in images, and such without any cue. A set of methods [29, 42, 44, 50, 52] exploit the good correlation properties

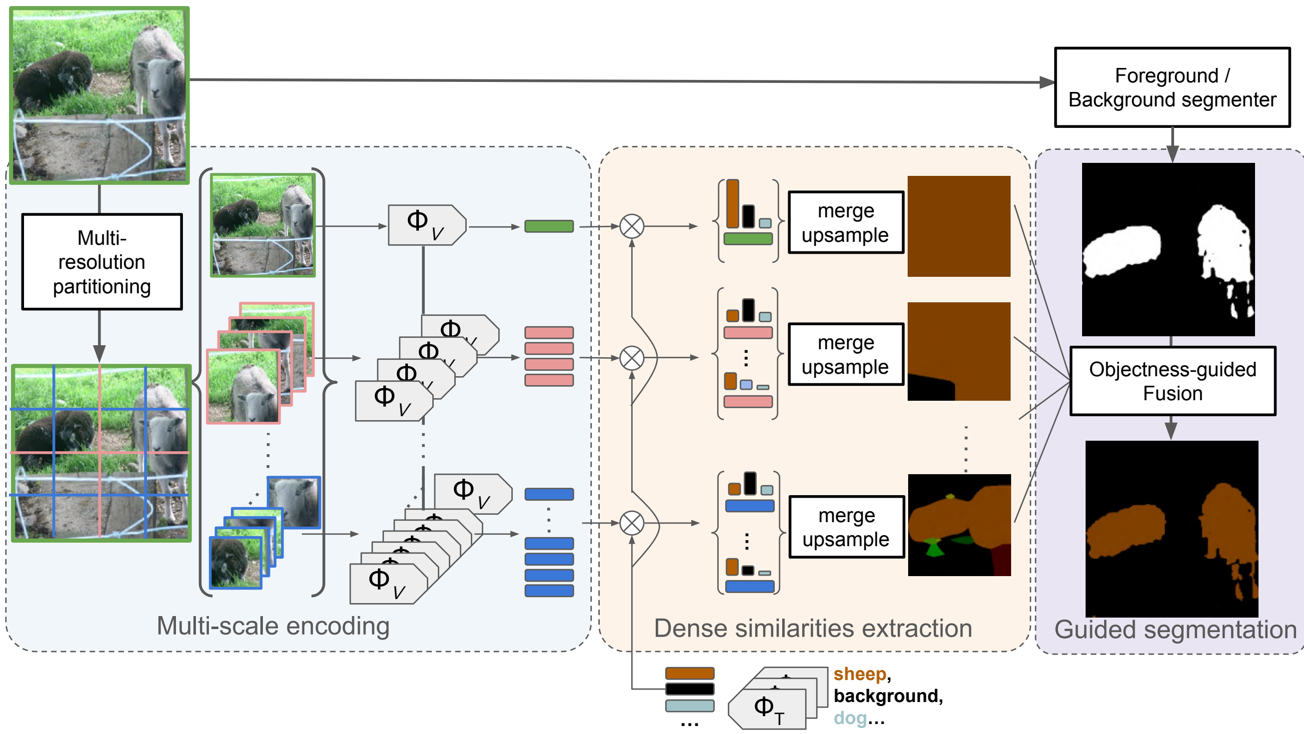

Figure 2. **Overview of `CLIP-DIY`**, our two-step pipeline for segmentation map extraction. We note Φ_V and Φ_T the CLIP encoders for image and text respectively. (1) An input image is partitioned into smaller patches, and each of them is fed *indepently* to the image encoder, yielding a vector of per-class similarity to an *arbitrary-length* vocabulary of classes. (2) Patches are then aggregated back before upsampling to produce dense similarity maps. (3) An objectness score obtained by an off-the-shelf foreground-background segmentation method such as FOUND [45] is used to guide the prediction of the final segmentation.

of the feature and find an object as the set of patches which highly differs to the other patches. Alternatively, [51] exploits the attention mechanisms with different queries and produces maps that are ranked and filtered and [45] proposes to look for the background instead of the objects in order to avoid single object discovery and to need priors about objects. The coarse object localization results obtained using those methods can be used as pseudo-labels to train large instance or segmentation models in a class-agnostic fashion [44, 50–52, 57]. Recent FOUND [45] is a very light model–a single conv1x1–trained to produce foreground/background segmentation of good quality. When self-trained FOUND achieves even better results and discovers more objects per image [44, 50]. In this work, we propose to leverage the good object localization properties of unsupervised object localization models, which make no hypotheses about object classes and remain therefore *open*. In particular, we take advantage of the efficiency of FOUND [45] to guide our zero-shot segmentation.

2.3. Combining self-supervised features & CLIP

Combining self-supervised learning [7, 9, 17, 18] with VLMs has been previously explored by different open-vocabulary segmentation methods [37, 40, 43]. Correlation qualities are used to perform co-segmentation [43], when pre-training properties are directly leveraged to initialize the visual encoder backbone [37, 56]. Related to our work, ZGS [40] builds potential masks using clustering strategies on self-supervised features and assigns them a class. Contrary to all previous approaches, it explores the task without predefined text prompts. Although ZGS currently obtains lower results than other baselines on the task, it opens an interesting direction for future work.

In this work, we do not perform co-segmentation (which expects to know classes of interest), nor retrain a model from scratch. Instead, we propose to guide CLIP prediction with an unsupervised foreground/background segmentation method, which to the best of our knowledge has not yet been explored.

3. `CLIP-DIY`

We tackle the problem of open-vocabulary semantic segmentation with no supervision. Let us consider a set of queries $t_j \in T$ formulated in natural language. Our goal is to localize each query if present in the image, yielding one mask per query, i.e. a segmentation map. Our approach,

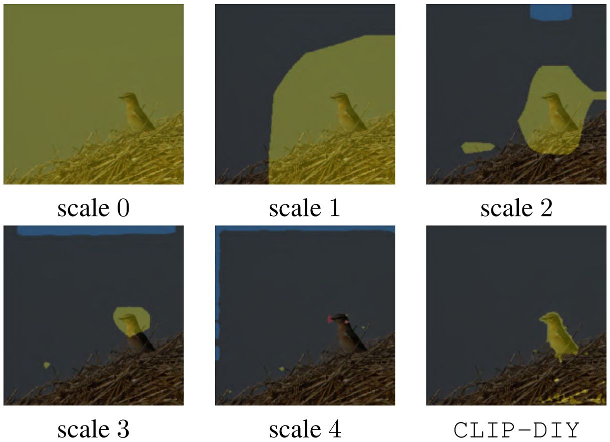

Figure 3. **Multi-scale maps ρ_s of `CLIP-DIY`.** Our method produces multi-scale predictions of dense logits that capture information at different levels of granularity. While the coarsest ($s = 0$) scale pools global information from the image, finer scales induce more localized , and potentially discover different object classes. While our method can incorporate as many scales as needed, very fine resolutions such as scale 4 and above do not generate meaningful information.

summarized in Fig. 2, consists of two stages. In Sec. 3.1, we describe our first step, where we run our proposed multi-scale dense inference to obtain coarse semantic maps by running CLIP on image patches at different scales. Our second step, which we cover in Sec. 3.2, consists of refining the initial segmentation using an off-the-shelf foreground-background extractor.

3.1. Dense inference with CLIP

Our method leverages CLIP [35] as a backbone. Contrary to most CLIP-based approaches for zero-shot semantic segmentation (as discussed in Sec. 2.1), we do not rely on patch tokens within the image encoder. Instead, we leverage CLIP's zero-shot capabilities by running the model densely on image partitions. Thus, we calculate the alignment between each of the multi-scale patches and the considered textual queries.

Encoding prompts Given a set of textual queries $\mathcal{T}$, we encode a prompt $t_j \in \mathcal{T}$ with $j \in [0, |\mathcal{T}|]$ using CLIP text encoder $\Phi_T(t_j) \in \mathbb{R}^d$, where $d = 512$ in CLIP. Following previous works [8], we formulate a prompt as: `a photo of a` $\{t_j\}$. In what follows we use the notation $\Phi_T(\mathcal{T}) \in \mathbb{R}^{d \times |\mathcal{T}|}$ as the set of encodings of all text queries $\mathcal{T}$. Note that we always consider a `background` class, thus we always assume $t_0 =$ `background`.

Multi-scale image partitioning Given an input image $x \in \mathbb{R}^{H \times W \times 3}$, we first reshape it into a sequence of 2D patches $\mathcal{X} = \{x_i \in \mathbb{R}^{P \times P \times 3}\}_{i=1..N}$, where $N = \lceil \frac{H}{P} \rceil \cdot \lceil \frac{W}{P} \rceil$ and $P \times P$ is the patch size. We then extract visual embeddings $\Phi_V(x_i) \in \mathbb{R}^d$ for each patch. In practice, since we use a ViT [12] as our visual encoder Φ_V, and take the [CLS] output as our visual embedding $\Phi_V(x_i)$[1]. We can then run the same partitioning for different scales $\mathcal{S}$ using different patch sizes, yielding a set of partitions $\{\mathcal{X}_s\}_{s \in \mathcal{S}}$.

Dense similarities extraction Given our multi-scale partitions, and a text query $t \in \mathcal{T}$, we build a dense similarity map ρ_s^t for each scale $s \in \mathcal{S}$, such that:

$$\rho_s^t = \underset{H,W}{\text{Upsample}} \left(\bigcup_{x \in \mathcal{X}_s} [\Phi_V(x) \otimes \Phi_T(t)] \right), \tag{1}$$

where $\bigcup$ is a merging operator that puts patches back onto a 2D grid, $\otimes$ denotes the inner product computed between the visual and text embeddings and $\underset{H,W}{\text{Upsample}}$ is a bilinear up-sampling operator which upsamples its input to the resolution $H \times W$.

The resulting map $\rho_s^t \in \mathbb{R}^{H \times W}$ yields an estimate of the similarity between each pixel in the input image and a text query $t \in \mathcal{T}$. In Fig. 3 we show aggregated similarity maps $\rho_s = \{\rho_s^t\}_{t=1..|\mathcal{T}|} \in \mathbb{R}^{H \times W \times |\mathcal{T}|}$ obtained at different scales. We can see that while the coarsest scale $s = 0$ is responsible for pooling global information on the objects to segment, finer scales $s > 1$ result in more localized maps.

Having obtained similarity maps ρ_s^t for each text query $t \in \mathcal{T}$ and each scale $s \in \mathcal{S}$ we aggregate the multi-scale predictions into a map M_{CLIP}^t such that:

$$M_{CLIP}^t = \frac{1}{|S|} \sum_{s \in \mathcal{S}} \rho_s^t. \tag{2}$$

3.2. Guided segmentation

Finally, we propose to refine the multi-scale segmentation maps M_{CLIP}^t of Eq. 2 using an objectness map produced by an off-the-shelf *unsupervised* foreground-background segmentation method [42, 45], noted Θ. As discussed in related work, such methods exploit self-supervised features, e.g. [7], to discover the pixels likely depicting objects.

When fed with an image $x \in \mathbb{R}^{H \times W}$, the foreground-background segmentation method produces an output $\Theta(x) \in [0, 1]^{H \times W}$ with a per-pixel confidence score close to 1 for a pixel in a foreground object. We use this proxy for objectness estimation to refine our segmentation masks M_{CLIP}^t. In particular, we refine M_{CLIP}^t using the output of Θ for each text query $t \in \mathcal{T}$ except for $t_0 =$ `background`,

[1] For convolutional network backbones such as Resnet [19], we could use the output of AvgPool.

where we take the complement of $\Theta(x)$ following:

$$\hat{M}^t = \begin{cases} 1 - \Theta(x) & \text{if } t = \texttt{background}, \\ \Theta(x) & \text{otherwise}, \end{cases} \tag{3}$$

such that similarities with the background class are down-weighted for pixels deemed as *salient* by Θ.

Finally, we compute the output of `CLIP-DIY` as:

$$M = \underset{t \in \mathcal{T}}{\text{SoftMax}} \left(M^t_{CLIP} \odot \hat{M}^t \right) \in \mathbb{R}^{H \times W \times |\mathcal{T}|}, \tag{4}$$

where $\odot$ denotes the Hadamard product and SoftMax is the softmax operator computed over text queries. In Fig. 3 we show how the aggregation of all scales paired with the guidance of Θ results in accurate object segmentation.

4. Experiments

In this section, we present the experiments conducted to evaluate our method and justify particular design choices. First, in Sec 4.1 we give details about our experimental setup. In Sec. 4.2, we discuss how our method compares against other open-vocabulary semantic segmentation approaches, both quantitatively and qualitatively. We then give more insight into our method with a series of ablations (Sec. 4.3), failure mode analysis (Sec. 4.4) and finally real open-world evaluation (Sec. 4.5).

4.1. Experimental setup

Datasets & metric We evaluate our method on two common semantic segmentation benchmarks: PASCAL VOC 2012 [13] and COCO [26], comprising of 20 and 80 foreground classes respectively. PASCAL VOC has an additional `background` class, and we adopt a unified protocol [8,55] considering a `background` class in all datasets. We evaluate results with the mean Intersection-over-Union (mIoU) metric. For evaluation, we resize input images to have the shorter side of length 448 following [8].

Implementation details If not otherwise specified, we use the CLIP ViT-B/32 model OpenCLIP version [21] trained with LAION [41]. The input images are resized to 224×224 and the patch size is 32×32. We empirically find that running our model on 3 different scales i.e. $|\mathcal{S}| = 3$ with patch sizes of $P_0 = 256, P_1 = 128, P_2 = 64$ gives the best results for both evaluated datasets. We discuss this later in Sec. 4.3.

Baselines We compare our method with existing state-of-the-art zero-shot open-vocabulary methods. In particular, we evaluate against methods including self-trained MaskCLIP+ [59], learning grouping strategies: GroupViT [55], SegCLIP [28], ViL-Seg [27], OVSegmentor [56], ViewCo [39] , using class prototypes: ReCo† [43], text-grounding strategy: TCL [8] and with CLIPpy [37] which uses a T-5 backbone and improves dense abilities of CLIP. [2] We detail in Tab. 1 the VLM backbones used per method and the if additional training data was. Every method (except for MaskCLIP) requires training a specific module/model used to get denser predictions; instead, we use the vanila CLIP model.

A note on fair comparison Following TCL [8], we use a unified evaluation protocol corresponding to an open-world scenario where prior access to the target data before evaluation is not allowed. In particular, we do not consider query expansion, e.g. class name expansion or rephrasing. As discussed in [8] exploring language biases can greatly improve the overall segmentation performance. However, we only use the original class names from the compared datasets. We also report best-reported scores for all methods. It is to be noted that TCL uses a post-processing technique, namely PAMR [2] , when other methods do not.

4.2. Results

In this section, we compare our method to previous work both quantitatively and qualitatively.

Quantitative results We summarize in Tab. 1, the comparison of our `CLIP-DIY` to baselines. We report results of `CLIP-DIY` with both CLIP ViT-B/32 and CLIP ViT-B/16, given that most methods use the latter.

First, comparing our results with the two different backbones, we remark that better scores are obtained with CLIP ViT-B/32 which patch inputs are larger, and are therefore less expensive at inference time. We believe this could be explained by an existing upper bound on CLIP accuracy w.r.t the level of granularity; patches too small might induce noisy classification—as we observed in Fig. 3

Comparing to baselines, we observe that our method achieves the best mIoU result on PASCAL VOC dataset and outperforms all previous works by more than 4 mIoU pts, and such without post-processing. This result is particularly interesting given that our method does not require dedicated training to improve CLIP segmentation abilities but instead leverages the unsupervised object localization method.

Moreover, we obtain 31.0 mIoU on COCO when the best-performing method CLIPpy achieves 32.0, so just 1 mIoU pt better than our approach. We can observe in Fig.4 that CLIPpy discovers more queries per image than us, even though its segmentation outputs appear to be noisier. Such results might suit a better COCO benchmark and less PASCAL VOC.

[2] We do not compare against [31] due to lack of comparable results and open source code.

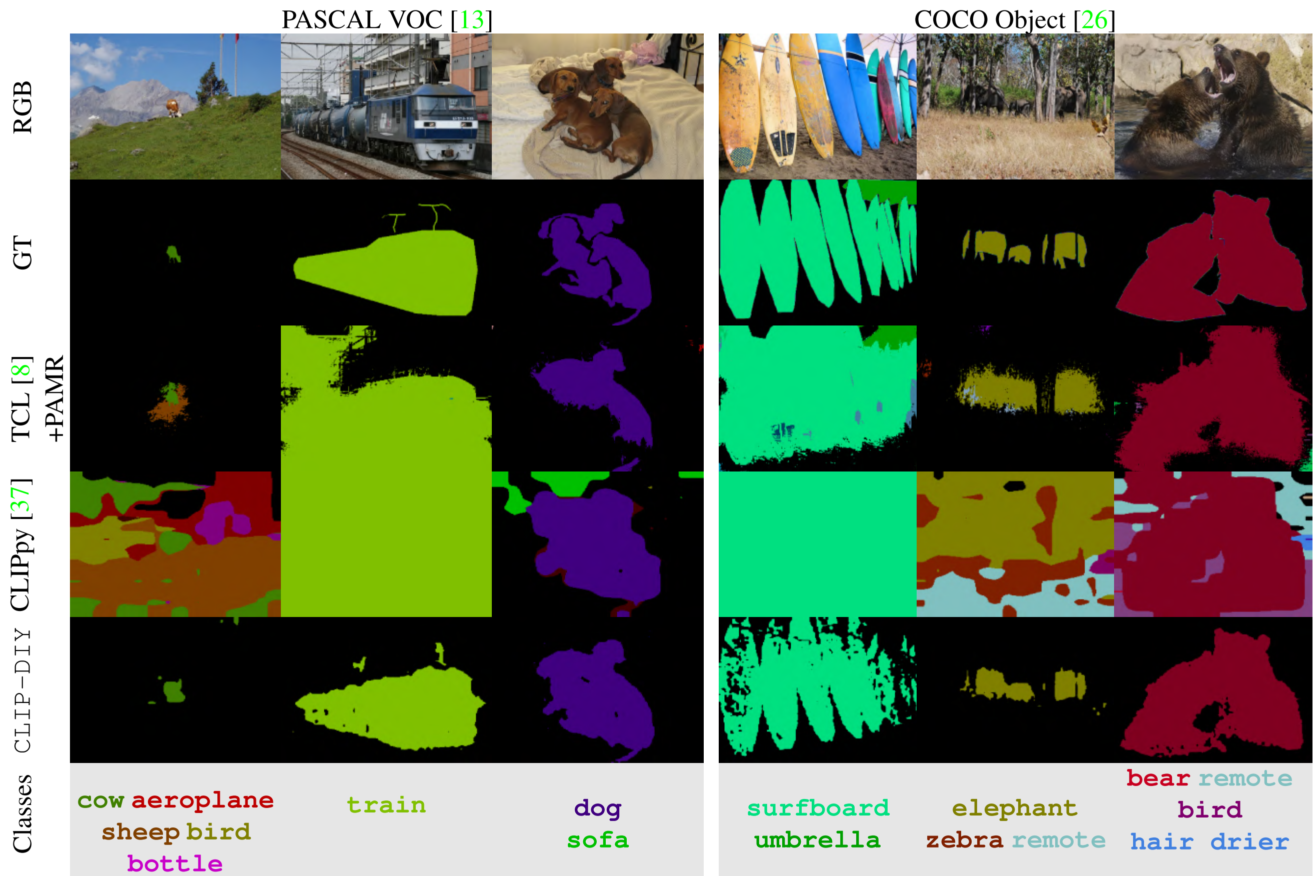

Figure 4. **Qualitative open-vocabulary segmentation results**. We compare our method against CLIPpy [37] and TCL (with PAMR post-processing) [8]. Our method consistently outperforms two other methods by producing accurate segmentation masks. TCL and CLIPpy also both suffer from hallucinating classes based on context, such as the `aeroplane` and `sheep` in 1st column, or `zebra` in 5th column. All pixels annotated in black are from the `background` class.

Method	extra training ?	Backbones Visual	Backbones Text	**PASCAL VOC**	**COCO Object**
ReCo† [43]	✓	ViT-L/14*	CLIP-ViT-L/14*	25.1	15.7
ViL-Seg [27]	✓	ViT-B/16		37.3	-
MaskCLIP+† [59]	✓	ResNet101 [19]		38.8	20.6
CLIPpy [37]	✓	ViT-B/16	T-5 [36]	52.2	**32.0**
GroupViT [55]	✓	ViT-S/16	12T	52.3	-
ViewCo [39]	✓	ViT-S/16	12T	52.4	23.5
SegCLIP [28]	✓	ViT-B/16	CLIP-ViT-B/16	52.6	26.5
OVSegmentor [56]	✓	ViT-B/16	BERT-ViT-B/16	53.8	25.1
TCL [8] + PAMR [2]	✓	ViT-B/16	CLIP-ViT-B/16	<u>55.0</u>	<u>31.6</u>
`CLIP-DIY` (ours)		ViT-B/16	CLIP-ViT-B/16	59.0	30.4
`CLIP-DIY` (ours)		ViT-B/32	CLIP-ViT-B/32	**59.9**	31.0

Table 1. **Zero-shot open-vocabulary segmentation**. Comparison of our approach to the state of the art (under the mIoU metric). While our method does not need any additional training it performs significantly better than the current SOTA on PASCAL VOC **(+4.9)** and performs on par with its competitors on COCO object, ranking 3rd on COCO. We mark with † results from [8]. All methods are evaluated considering that `background` is a class of the dataset. We note with $*$ when more than one backbone was used, we refer here to CLIP-like backbones. GroupViT and ViewCo use a 12 Transformer layers backbone following [34], noted 12T.

Method	PASCAL VOC	COCO
CLIP-DIY	59.9	31.0
w/o multi-scale	56.0	25.9
w/o objectness	24.1	15.5

Table 2. **Ablation studies.** We find that while all components are improving the performance of our method, objectness is particularly critical. Numbers are reported using a ViT-B/32 backbone.

Qualitative results We qualitatively compare here our method with best performing TCL [8] and CLIPpy [37] in Fig. 4. Our method consistently produces better masks with better object boundaries, which we attribute to the high-quality saliency maps. Moreover, we observe that CLIP-DIY produces correct semantic results on the foreground objects, with fewer artefacts than CLIPpy. Our method seems also less sensitive to biases, for instance, both TCL and CLIPpy hallucinate the sheep class on the grass (on the very left image) and CLIPpy also predicts aeroplane in the sky (in most left image) and zebra next to the elephants (middle image in COCO dataset).

4.3. Ablations

In this section, we perform ablation studies to validate the individual choices in the design of CLIP-DIY.

First, we study in Tab. 2 the impact of the different elements of our method. In particular, we investigate the impact of using a multi-scale mechanism and leveraging the objectness produced by the foreground/background segmenter. We notice that by removing our multi-scale scheme we drop results by 3.9 and 5.1 mIou pts on Pascal and COCO respectively, showing the benefit of considering patches of different sizes. Additionally, the largest drop is observed when removing the foreground/background saliency guidance showing the effectiveness of combining CLIP with the current lightest unsupervised object localization model, FOUND.

Multi-scale approach We conduct an ablation study on the scales used in our multi-scale scheme to generate the predictions. We progressively add inner scales in Eq. 2 and report the resulting accuracy on all datasets. The results, detailed in Tab. 3 show an optimum when using three fine scales. Adding more does not appear to improve results nor downgrade them, showing the stability of our method. Interestingly, we also see that removing information from the global scale greatly reduces performance on all datasets (-5.7/-3.5 mIoU pts on PASCAL VOC/COCO respectively). In Fig. 5 we visualize the individual contribution of each scale to our final prediction. As in Fig. 3, we observe that coarse scales capture the global context, while finer scales capture more local one, such that objects can be separated.

Scales used					PASCAL	COCO
0	1	2	3	4	VOC	Object
✓					56.0	25.9
✓	✓				58.8	28.7
✓	✓	✓			**59.9**	**31.0**
✓	✓	✓	✓		59.5	29.9
✓	✓	✓	✓	✓	59.3	29.4
	✓	✓	✓		53.8	26.4

Table 3. **Ablation of our multi-scale approach.** To validate our multi-scale design, we report the results of our method with different sets of scales, by progressively adding more fine-grained scales. We find empirically that after scale 3, adding more scales gives similar or worse results. We also report results without the first scale $s = 0$ (global scale), which leads to worse results. Numbers are reported using a ViT-B/32 backbone.

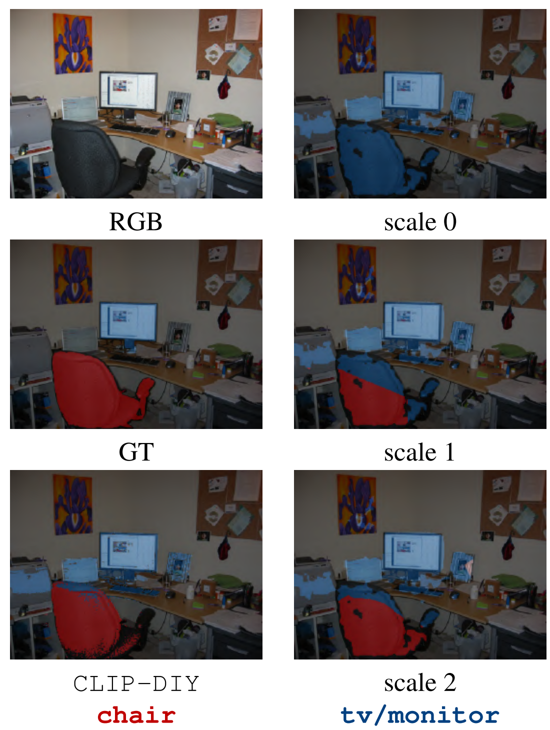

Figure 5. **Qualitative ablation study.** In the right column we show the results when progressively adding more scales in the multi-scale approach. Running with 3 scales enables segmenting most of the chair.

Foreground segmenters We compare different foreground-background segmenters and the overall performance of our method using each one of them. The results, summarized in Tab. 4, show that our method performs the best when using FOUND, more specifically the FOUND model that has been re-trained with self-training in the original work. We also experiment with CutLER [50], which performs unsupervised instance segmentation. We use the predicted instance masks or compute a saliency. In

Saliency Method	w. training	PASCAL VOC
FOUND-bkg [45]	✗	48.4
FOUND [45]	✓	59.9
CutLER saliency [50]	✓	55.4
CutLER mask [50]	✓	50.8

Table 4. **Comparison of different objectness methods** used for objectness-guided fusion. We find that the version of FOUND that was trained *in the original paper* performs the best, and therefore keep this method as our objectness guide.

both cases, we obtain slightly worse results. We give more details in Sec. 2.2 of Supp. material.

4.4. Failure cases

Failure cases We qualitatively analyse failure cases of our method by showing a couple of examples in Fig. 6. We observe that some of the failures are due to inaccurate annotations: in (a) `bear` is only partially annotated and in (b) a mask for an `elephant` is annotated too coarsely. Our method, benefiting from FOUND's accurate saliency predictions is able to produce better segmentation masks. We also observe that our method is limited by the quality of the saliency (c, d), which we comment more on below.

Ambiguities In the examples (c) and (d) of Fig. 6, we observe that our method can fail to segment objects with significant overlap, resulting in ambiguity regarding the foreground class, which is especially harmful when annotations are too coarse or incomplete. In (c), only the `bench` is annotated, while our method segments only the orange, which would result in a low IoU score. In (d), the opposite behaviour occurs, where `CLIP-DIY` discovers more classes than the annotated ones. Finally, similarly to most of the open-vocabulary methods based on CLIP, our method suffers from sensitivity to text ambiguities. We show more failure cases in Sec. A.3.3 of the Supplementary material.

4.5. Our method in the wild

We also test our method in the wild. We randomly download a set of images and provide textual queries we find most suitable. We show in Fig. 1 that our method exhibits off-the-shelf open-world segmentation capabilities, being able to produce masks for specific prompts. More results, including comparisons against other methods of in-the-wild open-world segmentation, are presented in Sec. A.3.2 of the Supplementary material.

5. Conclusions

We introduce a new method for open-vocabulary semantic segmentation, namely `CLIP-DIY`, which exploits CLIP's open-vocabulary classification abilities. As opposed to recent approaches we run CLIP densely at multiple scales to obtain coarse semantic mask proposals. When further guided by the quick fully unsupervised object localization method FOUND , which estimates foreground saliency, our model obtains state-of-the-art results on PASCAL VOC and performs on-par with baselines on COCO dataset. Since our method does not require any specific training, it can be used as an off-the-shelf method for open-world segmentation, and could therefore serve as a tool to help dataset annotators. While `CLIP-DIY` already yields competitive results, we believe future work could make it more diverse and efficient.

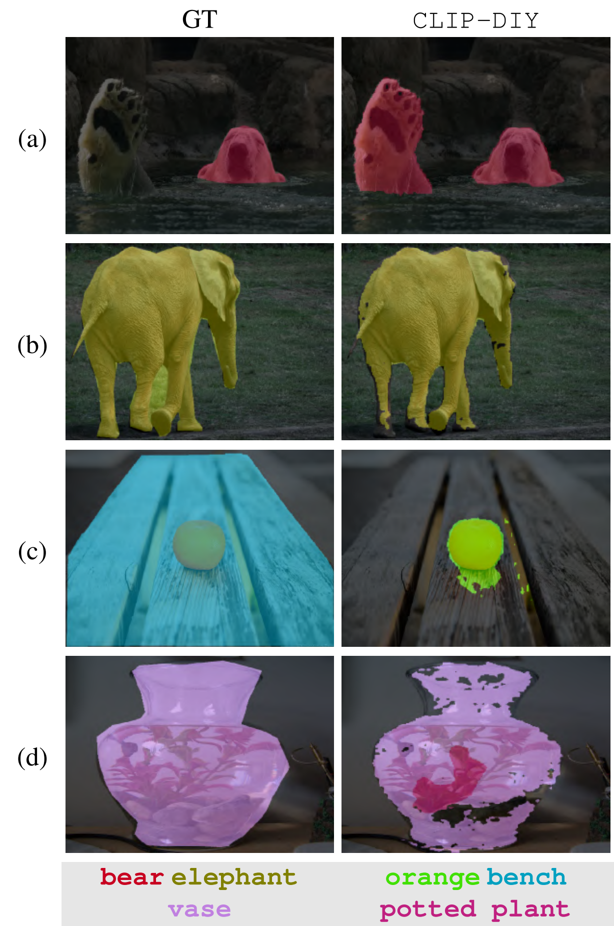

Figure 6. **Failure cases.** Our method is not robust to a few cases such as (a) incomplete or (b) coarse labelling. Another failure mode of our method is when there is an ambiguity in the foreground class (c-d).

Acknowledgments

We would like to thank Georgy Ponimatkin for the interesting discussions. This work was supported by the National Centre of Science (Poland) Grant No.2022/45/B/ST6/02817 and by the grant from NVIDIA providing one RTX A5000 24GB used for this project.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 1

[2] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4253–4262, June 2020. 5, 6, 14, 15

[3] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019. 1

[4] Adam Bielski and Paolo Favaro. MOVE: unsupervised movable object segmentation and detection. *CoRR*, abs/2210.07920, 2022. 2

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

[6] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020. 2

[7] 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. 2, 3, 4

[8] Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2, 4, 5, 6, 7, 12, 13, 14, 15

[9] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 3

[10] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 17864–17875. Curran Associates, Inc., 2021. 1

[11] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. 2022. 2

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4

[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 1, 5, 6, 12

[14] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision*, pages 540–557. Springer, 2022. 2

[15] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. Context-aware feature generation for zero-shot semantic segmentation. In *ACM MM*, 2020. 2

[16] Shijie Hao, Yuan Zhou, and Yanrong Guo. A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406:302–321, 2020. 1

[17] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 3

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 3

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 6

[20] Ping Hu, Stan Sclaroff, and Kate Saenko. Uncertainty-aware learning for zero-shot semantic segmentation. *NeurIPS*, 2020. 2

[21] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. If you use this software, please cite it as below. 1, 2, 5

[22] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. If you use this software, please cite it as below. 2

[23] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1

[24] Naoki Kato, Toshihiko Yamasaki, and Kiyoharu Aizawa. Zero-shot semantic segmentation via variational mapping. In *ICCVW*, 2019. 2

[25] Peike Li, Yunchao Wei, and Yi Yang. Consistent structural relation learning for zero-shot segmentation. *NeurIPS*, 2020. 2

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1, 5, 6, 12

[27] Quande Liu, Youpeng Wen, Jianhua Han, Chunjing Xu, Hang Xu, and Xiaodan Liang. Open-world semantic segmentation via contrasting and clustering vision-language embedding, 2022. 1, 2, 5, 6

[28] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. SegCLIP: Patch aggregation with learnable centers for open-vocabulary semantic segmentation. *ICML*, 2023. 2, 5, 6

[29] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Deep spectral methods: A surprisingly strong baseline for unsupervised semantic segmentation and localization. In *CVPR*, 2022. 2

[30] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 2

[31] 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*, pages 19413–19423, 2023. 1, 2, 5

[32] Giuseppe Pastore, Fabio Cermelli, Yongqin Xian, Massimiliano Mancini, Zeynep Akata, and Barbara Caputo. A closer look at self-training for zero-label semantic segmentation. In *CVPR*, 2021. 2

[33] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. 2

[34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 6

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 4

[36] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020. 6

[37] Kanchana Ranasinghe, Brandon McKinzie, Sachin Ravi, Yinfei Yang, Alexander Toshev, and Jonathon Shlens. Perceptual grouping in contrastive vision-language models. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 1, 2, 3, 5, 6, 7, 12, 13, 14

[38] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[39] Pengzhen Ren, Changlin Li, Hang Xu, Yi Zhu, Guangrun Wang, Jianzhuang Liu, Xiaojun Chang, and Xiaodan Liang. Viewco: Discovering text-supervised segmentation masks via multi-view semantic consistency, 2023. 2, 5, 6

[40] Pitchaporn Rewatbowornwong, Nattanat Chatthee, Ekapol Chuangsuwanich, and Supasorn Suwajanakorn. Zero-guidance segmentation using zero segment labels. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 3

[41] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 5

[42] Gyungin Shin, Samuel Albanie, and Weidi Xie. Unsupervised salient object detection with spectral cluster voting. In *CVPRW*, 2022. 2, 4

[43] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 3, 5, 6

[44] Oriane Siméoni, Gilles Puy, Huy V. Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. Localizing objects with self-supervised transformers and no labels. November 2021. 2, 3

[45] Oriane Siméoni, Chloé Sekkat, Gilles Puy, Antonin Vobecky, Éloi Zablocki, and Patrick Pérez. Unsupervised object localization: Observing the background to discover objects. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2023. 2, 3, 4, 8, 12

[46] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7262–7272, 2021. 1

[47] Huy V. Vo, Patrick Pérez, and Jean Ponce. Toward unsupervised, multi-object discovery in large-scale image collections. In *ECCV*, 2020. 2

[48] Van Huy Vo, Elena Sizikova, Cordelia Schmid, Patrick Pérez, and Jean Ponce. Large-scale unsupervised object discovery. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 16764–16778. Curran Associates, Inc., 2021. 2

[49] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 136–145, 2017. 12

[50] Xudong Wang, Rohit Girdhar, Stella X Yu, and Ishan Misra. Cut and learn for unsupervised object detection and instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3124–3134, 2023. 2, 3, 7, 8, 12

[51] Xinlong Wang, Zhiding Yu, Shalini De Mello, Jan Kautz, Anima Anandkumar, Chunhua Shen, and Jose M. Alvarez. Freesolo: Learning to segment objects without annotations. In *CVPR*, 2022. 3

[52] Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L. Crowley, and Dominique Vaufreydaz. Self-supervised transformers for unsupervised object discovery using normalized cut. In *Conference on Computer Vision and Pattern Recognition*, 2022. 2, 3

[53] Jianzong Wu, Xiangtai Li, Shilin Xu, Haobo Yuan, Henghui Ding, Yibo Yang, Xia Li, Jiangning Zhang, Yunhai Tong, Xudong Jiang, Bernard Ghanem, and Dacheng Tao. Towards open vocabulary learning: A survey. *arXiv pre-print*, 2023. 2

[54] 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

[55] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. *arXiv preprint arXiv:2202.11094*, 2022. 1, 2, 5, 6

[56] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2935–2944, 2023. 2, 3, 5, 6

[57] Andrii Zadaianchuk, Matthaeus Kleindessner, Yi Zhu, Francesco Locatello, and Thomas Brox. Unsupervised semantic segmentation with self-supervised object-centric representations. *CoRR*, abs/2207.05027, 2022. 3

[58] Hang Zhao, Xavier Puig, Bolei Zhou, Sanja Fidler, and Antonio Torralba. Open vocabulary scene parsing. In *ICCV*, 2017. 2

[59] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 5, 6