# Training-Free Open-Vocabulary Segmentation with Offline Diffusion-Augmented Prototype Generation

Luca Barsellotti[1*] Roberto Amoroso[1*] Marcella Cornia[1] Lorenzo Baraldi[1] Rita Cucchiara[1,2]
[1]University of Modena and Reggio Emilia, Italy [2]IIT-CNR, Italy
{name.surname}@unimore.it

## Abstract

*Open-vocabulary semantic segmentation aims at segmenting arbitrary categories expressed in textual form. Previous works have trained over large amounts of image-caption pairs to enforce pixel-level multimodal alignments. However, captions provide global information about the semantics of a given image but lack direct localization of individual concepts. Further, training on large-scale datasets inevitably brings significant computational costs. In this paper, we propose FreeDA, a training-free diffusion-augmented method for open-vocabulary semantic segmentation, which leverages the ability of diffusion models to visually localize generated concepts and local-global similarities to match class-agnostic regions with semantic classes. Our approach involves an offline stage in which textual-visual reference embeddings are collected, starting from a large set of captions and leveraging visual and semantic contexts. At test time, these are queried to support the visual matching process, which is carried out by jointly considering class-agnostic regions and global semantic similarities. Extensive analyses demonstrate that FreeDA achieves state-of-the-art performance on five datasets, surpassing previous methods by more than 7.0 average points in terms of mIoU and without requiring any training. Our source code is available at aimagelab.github.io/freeda.*

## 1. Introduction

Semantic segmentation is a core problem in Computer Vision, which aims at partitioning an image into coherent regions according to a set of semantic categories [25, 32]. As manually annotating large-scale amounts of training data is expensive, scaling segmentation to large sets of concepts in a fully supervised manner is impracticable. This has recently moved the focus of the community towards open-vocabulary solutions [10, 13, 19, 24, 46, 49] that, learning from a narrow set of seen categories or weak forms of supervision, are able to segment novel and unseen categories.

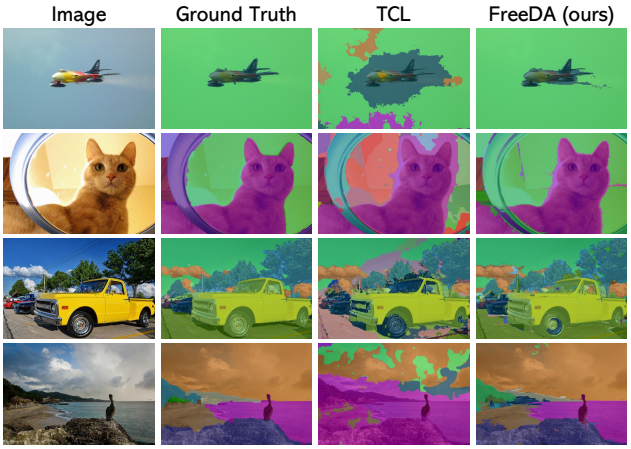


Figure 1. Open-vocabulary segmentation with: (a) TCL [6], which performs end-to-end learning of region-text alignment; (b) our FreeDA, which leverages generated textual-visual embeddings with global-local similarities and does not require any training.

One of the major challenges in this setting is how to transfer the ability to match texts and images of large-scale vision-language models (*e.g.*, CLIP [34] and ALIGN [16]) to a text-pixel alignment. Given a large-scale set of web-crawled image-caption pairs, previous approaches [6, 27, 35, 43, 44, 52] force the ability to localize textual concepts to emerge through contrastive learning techniques combined with grounding mechanisms [6, 43, 44]. However, captions often capture the global scene and might present ambiguities with respect to fine-grained elements, making this approach sub-optimal and computationally intensive.

On a different note, advances in diffusion models [14, 36] have shown remarkable results in text-to-image generation, and recent works have shown that their features encompass knowledge regarding the positioning of the generated objects [21, 39, 42]. This information can be exploited to generate large sets of attribution maps, which are more active in the area corresponding to a semantic class, thus providing a valuable source of information for semantic segmentation. We propose to explore this mechanism as

*Equal contribution.

an alternative to multimodal contrastive training, in a fully training-free methodology where no parameter is learned.

In contrast to previous works, our proposed approach follows an efficient two-step protocol: in an offline stage, we leverage a diffusion-augmented generation in which a collection of textual-visual reference vectors is generated. Then, at inference time, these references are retrieved to compute local and global similarities to segment the input image. In detail, we employ a large set of textual captions to generate synthetic images and corresponding attribution maps, through a localization mechanism based on cross-attention. Subsequently, we leverage a self-supervised visual backbone, DINOv2 [33], to build an offline set of visual prototypes associated with textual vectors, each representing the context of an instance in its synthetic scene.

At inference time, we extract both global features with a multimodal encoder (*i.e.*, CLIP) and local dense features with DINOv2, characterized by high semantic relatedness, and employ a superpixel algorithm to detect class-agnostic regions. By querying the input textual category in the set of textual-visual reference embeddings, we then assign each superpixel to the category that exhibits the highest combined similarity, between the global and local modalities. As our approach is training-*free* and relies on *D*iffusion-*A*ugmented generation, we name it FreeDA.

We validate the proposed framework by conducting extensive experiments on Pascal VOC [11], Pascal Context [30], COCO Stuff [4] and Object [23], Cityscapes [8], and ADE20K [50, 51]. Without requiring any form of training, FreeDA consistently outperforms previous approaches by a large margin, achieving state-of-the-art performance on all datasets. Overall, our work demonstrates that non-parametric approaches can provide a compelling and efficient alternative for open-vocabulary semantic segmentation, and opens up new opportunities for subsequent works. To sum up, the contributions of this paper are as follows:

- We introduce FreeDA, a novel training-free method for open-vocabulary semantic segmentation based on the generation of context-aware textual-visual reference embeddings through diffusion models.
- We present an inference pipeline that, leveraging the semantic correspondence of DINOv2, superpixel algorithms, and a combination of local and global similarities achieves precise and robust segmentation prediction.
- Our experiments show that our approach achieves state-of-the-art performance on five datasets, without requiring any form of training.

## 2. Related Work

**Open-Vocabulary Semantic Segmentation.** Building upon the success of large-scale vision-language models in zero-shot classification [16, 34], previous works on open-vocabulary segmentation have investigated strategies to transfer the multimodal image-text alignment toward finer granularity (*i.e.*, region or pixel level) [10, 13, 22, 43, 47].

A group of literature has been focusing on the supervision provided by dense annotations, available for a limited set of categories, to generalize on unseen classes. OpenSeg [13] decouples the task in a region proposer and a grounder that aligns regions to words from captions. Similarly, OVSeg [22] employs a two-stage method, in which class-agnostic regions are masked and provided to a CLIP encoder with learnable visual prompts. SAN [47] combines a side network with CLIP to propose regions while recognizing their corresponding semantic category. However, these approaches are affected by performance gaps between seen and unseen categories [10, 22] and, due to the costs of dense annotations, can be applied in limited domains.

Other works have instead exploited contrastive training over a large set of image-text pairs, without dense annotations. GroupViT [43] proposes a Transformer architecture that learns to group image regions progressively. MaskCLIP [52] adapts a frozen CLIP for dense predictions through modifications in the last attention layer. TCL [6] presents a grounding mechanism that learns to associate text to regions during contrastive learning. OVSegmentor [44] introduces a module based on slot attention to group tokens of a Transformer and aligns them to captions. Our approach falls into this research direction, since it relies only on a set of captions as support, without requiring dense annotations.

**Localization in Diffusion Models.** Diffusion models [36] have proven state-of-the-art performance in image generation. Few works tackle the task of localizing the concepts mentioned in the conditioning captions during the generation. DAAM [39] proposes exploiting the cross-attention mechanism that Stable Diffusion uses to extract attribution maps for the words mentioned in the prompt. DiffuMask [42] leverages the advances of DAAM to generate ground truth segmentation masks without human annotation and train a segmentation model on them. GroundedDiffusion [21] implements a grounding module to align textual and visual embeddings during the diffusion process.

Some works have investigated the usage of diffusion models for open-vocabulary segmentation. ODISE [45] employs Stable Diffusion as a feature extractor for its mask generator. OVDiff [18] generates a set of visual references at prediction time to support the segmentation process. Our approach also relies on the generation of images; however, this is done to collect visual prototypes during an offline stage, a choice that significantly reduces the computational load at prediction time.

**Superpixel Algorithms.** The concept of superpixel arises from the observation that pixels are not a natural representation of an image. A superpixel is a group of homogeneous pixels based on the visual characteristics of the image, such as shape, brightness, color, and texture. Over the years,

several extraction strategies have been developed with the goal of improving their quality and efficiency, such as watershed-based [15, 28, 31] and clustering-based [1, 2, 20] approaches. In this paper, we employ superpixels as a support for partitioning the image into class-agnostic regions, from which local visual similarities are computed.

## 3. Proposed Method

The goal of open-vocabulary segmentation is to segment an image according to an arbitrary set of categories represented through free-form texts. Our training-free approach decouples the task into two phases: a *diffusion-augmented prototype generation* phase, which is carried out in an off-line manner (visually represented in Figure 2), and a *semantic correspondence-based inference* stage, which is employed at test time to perform prediction over an input image. This second stage is visually depicted in Figure 3.

### 3.1. Diffusion-Augmented Prototype Generation

During the pre-processing phase, we collect a large set of visual prototypes and corresponding textual key embedding vectors, which describe semantic instances along with their textual and visual contexts. A textual key represents a semantic category and its textual context as described in a caption. A visual prototype, instead, describes an instance of that semantic category contextualized in an image. Collections of prototypes belonging to the same semantic class, thus, represent examples of the visual variety of that class.

**Extracting Localized Masks with Diffusion Models.** As prototypes will be employed to predict semantic classes in a non-parametric way, it is crucial to build a large collection of prototypes with high semantic variance. To this aim, we generate a large set of real-world scenes using Stable Diffusion [36] starting from a large set of captions. Generating images rather than collecting real images from web-scale datasets allows us to control the resulting semantic distribution and its variance. Most importantly, also, latent-based diffusion models can predict the location of objects in the generated scene [39].

Diffusion models, indeed, map word embeddings of the conditioning text to the activations of their denoising subnetwork (*e.g.*, U-Net [36, 37]) through cross-attention layers applied at different scales. Cross-attention activations, therefore, relate each word of the conditioning caption to a portion of the image and can be employed to generate weak localization masks. As each layer of the denoising network produces cross-attention maps at a different scale, we upscale all intermediate maps at the original image size. Then, we collapse across heads, layers, and diffusion time steps to obtain a single object mask.

Formally, the attribution map of a word $w$ from the conditioning caption over a generated image $I$ is expressed as

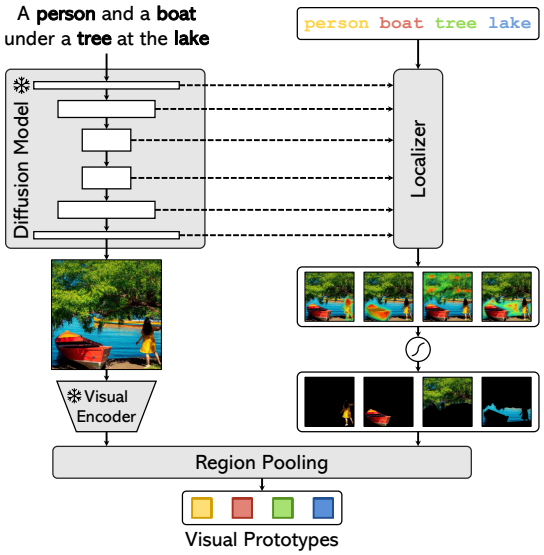


Figure 2. Overview of the diffusion-augmented prototype generation phase of FreeDA. Visual prototypes are generated by pooling self-supervised visual features on weak localization masks extracted from Stable Diffusion.

$$A(I, w) = \frac{1}{TLH} \sum_{t,l,h} \text{upsample}(\mathcal{A}(I, w)_{t,l,h}), \quad (1)$$

where $\mathcal{A}(I, w)$ indicates the collection of cross-attention maps with respect to the tokens of word $w$, and $t$, $l$, and $h$ index diffusion time steps, denoising layers, cross-attention heads respectively. Finally, usample$(\cdot)$ denotes a bilinear interpolation operator.

With the aforementioned approach for building localized masks, we employ a set of captions, designed to describe real images, to condition Stable Diffusion [36] and generate the corresponding set of synthetic images. Through a noun parser [26], from each caption we also extract mentioned nouns $\{w_1, \dots, w_N\}$ and obtain their corresponding attribution maps $A(I, w_i) \in \mathbb{R}^{H \times W}$ over the generated image. Then, we normalize the scores of the attribution maps in the range $[-1, 1]$, apply a sigmoid function, and binarize the result by thresholding it to a constant value $\gamma$. The output of this process is a weak localization mask $M(I, w_i) \in \{0, 1\}^{H \times W}$ for each noun $w_i$ mentioned in the input caption.

**Visual Prototypes Extraction.** To encode the content of the aforementioned weak localization masks, we adopt DINOv2 [33], which showcases good localization and semantic matching capabilities. Given a generated image $I \in \mathbb{R}^{H \times W \times 3}$, we extract its dense features $v(I) \in \mathbb{R}^{\frac{H}{P} \times \frac{W}{P} \times d_v}$, where $P$ is the input patch size of the backbone and $d_v$ is the dimensionality of its embedding space. For every noun $w_i$ in the sentence, we interpolate the weak localization mask $M(I, w_i)$ to the size of the dense features, obtaining a re-

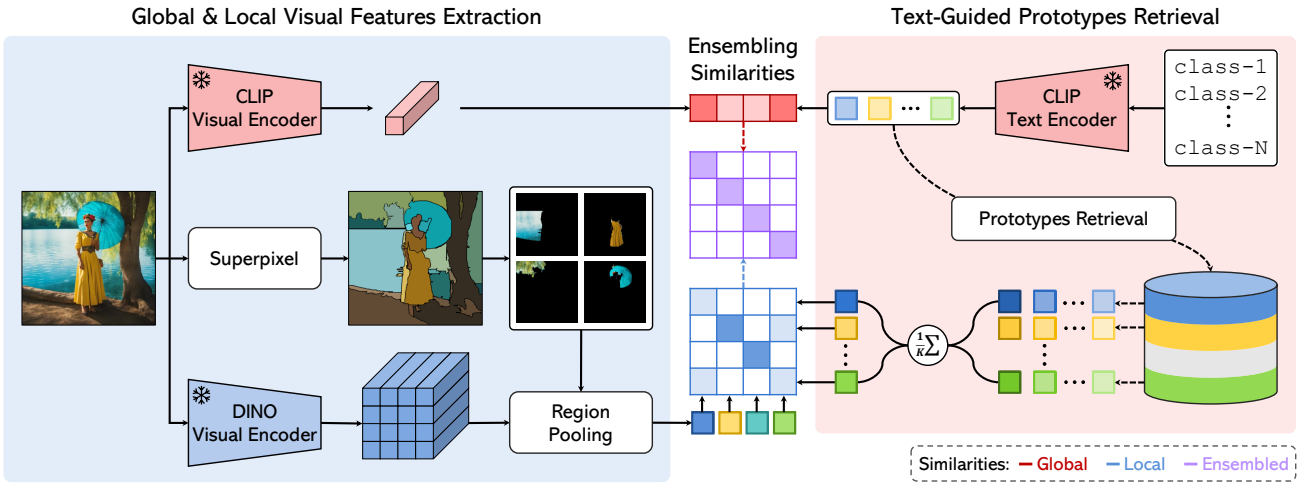


Figure 3. Overview of the inference process in FreeDA. Local (region-level) and global similarities are computed by employing, respectively, visual self-supervised and multimodal contrastive embedding spaces, and by comparing them with input texts and prototypes, built during the off-line stage.

sized version of the localization mask $\hat{M}(I, w_i) \in \mathbb{R}^{\frac{H}{P} \times \frac{W}{P}}$. Then, we perform a region pooling operation to aggregate visual features over the localization mask, as follows:

$$p(I, w_i) = \frac{\sum_{h=0}^{\frac{H}{P}} \sum_{w=0}^{\frac{W}{P}} v(I)[h, w] \hat{M}(I, w_i)[h, w]}{\sum_{h=0}^{\frac{H}{P}} \sum_{w=0}^{\frac{W}{P}} \hat{M}(I, w_i)[h, w]}, \quad (2)$$

where square brackets indicate indexing over spatial axes. The resulting vector $p(I, w_i) \in \mathbb{R}^{D_v}$ is the *visual prototype* for the noun $w_i$ extracted from the input image $I$, and is defined as the mean of the dense features covered by the corresponding binary mask. Prototypes built with this approach embed a visual descriptor of the corresponding word localized in a synthetic context, obtained from a real description.

**Textual Keys Extraction.** In addition to representing visual prototypes, we employ a text encoder to represent nouns in their lexical context. To this aim, we define a set of textual templates $\mathcal{T}$ (*e.g.*, `A photo of a [NOUN]`), and embed each noun in all templates. This results in a textual embedding for each template, $t_i(w) \in \mathbb{R}^{D_t}, i = 1 \dots, T$, where $T$ is the number of templates. We define $\hat{t}(w) = \frac{\sum_{i=1}^{T} t_i(w)}{T}$ as the mean noun embedding, and then linearly interpolate with the full caption embedding $\hat{c}$ to also capture the global context of the entire scene. Specifically, the resulting textual key vector $k(c, w)$ for a word $w$ taken from a caption $c$ is then defined as

$$k(c, w) = \alpha \hat{t}(w) + (1 - \alpha)\hat{c}, \quad (3)$$

where $\alpha \in (0, 1)$ is a scalar weight. Similar to prototypes, keys obtained through this process represent nouns contextualized in the caption in which they have been extracted. As each textual key is associated with a visual prototype, the set of textual keys extracted from a dataset can be indexed via an approximate nearest neighbor search to efficiently retrieve visual prototypes given a textual query.

### 3.2. Training-Free Mask Prediction

At inference time, our goal is to query the keys of the pre-built collection index to retrieve their corresponding prototypes. Then, we employ these prototypes as references to segment the input image through semantic correspondence with both local and global features.

**Retrieving Prototypes.** Given a set of textual categories $\{c_1, \dots, c_S\}$, we consider the same set of templates $\mathcal{T}$ employed during textual keys computation and embed each category as $\hat{t}(c_i) = \frac{\sum_{j=1}^{T} t_j(c_i)}{T}$, where $t_j(c_i)$ is the text embedding of a template applied on a category. For each category $c_i$, we leverage $\hat{t}(c_i)$ to query the key embeddings of the pre-built collection index and retrieve the $K$ most similar ones according to cosine similarity. Each key embedding corresponds to the combination of the text embeddings of both a noun and the caption in which the noun is mentioned, and is uniquely linked with a visual prototype. Hence, we compute a representative visual prototype for each category as the mean of retrieved prototypes. Formally,

$$\overline{p}(c_i) = \frac{\sum_{k=1}^{K} p_{ik}}{K}, \quad (4)$$

where $\{p_{ik}\}_{k=1}^{K}$ is the set of retrieved prototypes for the given category $c_i$.

**Superpixel-based Local Regions.** Once a visual representation of a class has been obtained through the aforementioned procedure, a straightforward solution to predict a segmentation mask for an image $I$ would be computing the semantic correspondences (*i.e.*, cosine similarities) for

each of its dense feature $v(I)$ against the representative prototypes of input categories $\overline{p}(I, c_i), \quad i = 1, \ldots, S$, and interpolate the result to the original image size. However, such an approach would lead to noisy segmentation masks.

In particular, it has been observed that DINOv2 shows good matching properties across objects from different images, but lacks in recognizing shapes and boundaries [48]. Hence, we propose to exploit a superpixel algorithm (*i.e.*, the Felzenszwalb's algorithm [12]) to partition the image by grouping pixels into class-agnostic non-overlapping regions according to their visual appearances and positions.

Each superpixel can be interpreted as a binary mask $R \in \{0, 1\}^{H \times W}$ that is active on pixels belonging to it. Similar to the construction of visual prototypes, we interpolate each superpixel at the size of the dense features and perform a region pooling stage as defined in Eq. 2 to produce superpixel embeddings $r_i \in \mathbb{R}^{D_v}, \quad i = 1, \ldots, |R|$. Then, for each superpixel embedding, we compute the cosine similarity against the representative prototypes of the categories. We associate each pixel with the unique region that includes it and we refer to this similarity in the unimodal space of the visual backbone as *local similarity*.

**Combining Local and Global Similarities.** While retrieved prototypes are linked with text, their feature vectors show good local matching properties but weaker global semantic capabilities. As correctly classifying pixels from a semantic point of view is crucial in segmentation, we propose to combine the local similarities obtained at the superpixel level with a global similarity measure which refers to the entire image. We compute this in the multimodal space of a vision-language model (*i.e.*, CLIP [34]), which instead has good semantic classification capabilities.

Specifically, we embed the input image using the image encoder of CLIP to produce an image embedding $i(I) \in \mathbb{R}^{D_t}$. Then, we compute cosine similarities between the image embedding and all the category embeddings $\hat{t}(c_i), \quad i = 1, \ldots, c_S$. Finally, we combine this global similarity with the single local similarities associated with class-agnostic regions. The final similarity between a local region and a semantic class is therefore computed as

$$s(r_j, c_i) = \beta l(r_j, c_i) + (1 - \beta) g(I, c_i), \tag{5}$$

where $r_j$ indicates the local region, $c_i$ the semantic class, and $I$ the input image. Further, $l(r_j, c_i)$ is the local similarity between the region of interest and the class, and $g(I, c_i)$ is the global similarity extracted from CLIP space. To obtain the final segmentation mask, each region is then associated with the semantic class with the highest similarity.

# 4. Experiments

## 4.1. Experimental Setup

**Datasets.** We evaluate FreeDA on the validation splits of traditional semantic segmentation benchmarks, namely Pascal VOC 2012 [11], Pascal Context [30], COCO Stuff [4], Cityscapes [8], and ADE20K [50, 51]. In particular, the validation sets of these datasets respectively contain 20, 59, 171, 150, and 19 semantic categories and 1,449, 5,104, 5,000, 2,000, and 500 images. In addition to these datasets for which we do not consider pixels not belonging to any category, we also validate our method when considering them as part of an additional "unknown" class (also referred to as "background" class in the literature). For these experiments, we again employ Pascal VOC 2012 and Pascal Context, and also include the COCO Objects dataset [4] which is a variant of COCO-Stuff with 80 foreground categories on the same validation split. To assess the segmentation performance, we employ the mean Intersection-over-Union (mIoU) on all the classes of each dataset.

**Implementation Details.** Textual sentences used as input in our diffusion-augmented prototype generation pipeline are taken from the COCO Captions dataset [7, 23]. We consider all five captions available for each image, thus obtaining a large set of captions describing natural images that can be used as input for a diffusion-based generative architecture. It is worth noting that we do not utilize the images associated with these captions. To generate the collection of visual prototypes, we employ Stable Diffusion v2.1 [36] with 50 diffusion steps and a threshold $\gamma$ equal to 0.45. The scalar weight $\alpha$ that combines the mean noun embeddings and caption embeddings to form keys is equal to 0.9.

We use DINOv2 [33] pre-trained on the LVD-142M dataset as the self-supervised visual backbone, using both the ViT-B/14 and the ViT/L-14 versions, with an input image size of $518 \times 518$. This leads to dense features with size corresponding to $37 \times 37$. We also employ CLIP [34] as the multimodal encoder using the original OpenAI weights, on top of the ViT-B/16 and ViT-L/14 architectures. We use the same CLIP model for both key embeddings and global similarity computation, so that *(i)* we embed the arbitrary categories at inference time just one time and *(ii)* we do not need to load two different text encoders into memory.

To extract superpixels, we use the Felzenszwalb's algorithm [12]. We build and leverage an efficient exact retrieval index through the `faiss` library [17] based on cosine similarity. We consider the number of retrieved prototypes $K$ equal to 350 for all datasets and the ensembling weight $\beta$ between local and global similarities equal to 0.8 for all benchmarks except for Pascal VOC for which we use $\beta$ equal to 0.7. More details are in the supplementary.

**Evaluation Protocol.** To perform all experiments, we follow the unified evaluation protocol for unsupervised open-

| Model | PAMR | Dataset | Parameters (M) | | Similarity | | mIoU | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Total | Trainable | Textual | Visual | VOC | Context | Stuff | Cityscapes | ADE |
| ReCo [38] | ✗ | ImageNet1k★ | 313.0 | 0.0 | ✗ | ✓ | 57.7 | 22.3 | 14.8 | 21.1 | 11.2 |
| GroupViT [43] | ✗ | CC12M+RedCaps♦ | 55.8 | 55.8 | ✓ | ✗ | 79.7 | 23.4 | 15.3 | 11.1 | 9.2 |
| MaskCLIP [52] | ✗ | - | 291.0 | 0.0 | ✓ | ✗ | 74.9 | 26.4 | 16.4 | 12.6 | 9.8 |
| TCL [6] | ✗ | CC3M+CC12M♦ | 178.3 | 21.7 | ✓ | ✗ | 77.5 | 30.3 | 19.6 | 23.1 | 14.9 |
| OVDiff [18] | ✗ | - | 1,226.4 | 0.0 | ✗ | ✓ | 81.7 | 33.7 | - | - | 14.9 |
| MaskCLIP [52] | ✓ | - | 291.0 | 0.0 | ✓ | ✗ | 72.1 | 25.3 | 15.1 | 11.2 | 9.0 |
| ReCo [38] | ✓ | ImageNet1k★ | 313.0 | 0.0 | ✗ | ✓ | 62.4 | 24.7 | 16.3 | 22.8 | 12.4 |
| GroupViT [43] | ✓ | CC12M+YFCC♦ | 55.8 | 55.8 | ✓ | ✗ | 81.5 | 23.8 | 15.4 | 11.6 | 9.4 |
| TCL [6] | ✓ | CC3M+CC12M♦ | 178.3 | 21.7 | ✓ | ✗ | 83.2 | 33.9 | 22.4 | 24.0 | 17.1 |
| **FreeDA (ViT-B)** | ✗ | COCO Captions★ | 236.1 | 0.0 | ✗ | ✓ | **85.6** (+2.4) | **43.1** (+9.2) | **27.8** (+5.4) | **36.7** (+12.7) | **22.4** (+5.3) |
| **FreeDA (ViT-L)** | ✗ | COCO Captions★ | 732.0 | 0.0 | ✗ | ✓ | **87.9** (+4.7) | **43.5** (+9.6) | **28.8** (+6.4) | **36.7** (+12.7) | **23.2** (+6.1) |

Table 1. Comparison with state-of-the-art unsupervised open-vocabulary semantic segmentation models on Pascal VOC [11], Pascal Context [30], COCO Stuff [4], Cityscapes [8], and ADE20K [50, 51], without considering the unknown category. The markers ♦ and ★ refer, respectively, to datasets used for training and support only.

vocabulary semantic segmentation established by Cha *et al*. [6]. Specifically, we evaluate the model considering the class names from the default version of the `MMSegmentation` toolbox. We resize the images to have a shorter side equal to 448 and employ a sliding window approach with a stride of 224 pixels.

## 4.2. Comparison with the State of the Art

We first compare FreeDA with recent state-of-the-art approaches for unsupervised open-vocabulary semantic segmentation. Specifically, we include ReCo [38] and OVDiff [18] that, similarly to our approach, exploit the arbitrary input categories to obtain a set of visual references. While ReCo curates an archive based on ImageNet1k [9], OVDiff generates a set of synthetic references at inference time by conditioning on a fixed prompt template, without necessitating external support data. Also, we compare with MaskCLIP [52], which introduces some modifications to the CLIP architecture to exploit its multimodal embedding space, and GroupViT [43] and TCL [6] that rely on extensive contrastive training on large-scale datasets to learn a textual-visual alignment. When considering segmentation benchmarks with the background class, we also include ViewCo [35], SegCLIP [27], and OVSegmentor [44] that, analogously to GroupViT and TCL, are based on natural language supervision via contrastive learning paradigms.

Table 1 shows the results on the five benchmarks without the unknown category (*i.e*., Pascal VOC, Pascal Context, COCO Stuff, Cityscapes, and ADE20K). We report the performance of two variants of our approach: one based on DINOv2 ViT-B/14 and CLIP ViT-B/16 and the other based on DINOv2 ViT-L/14 and CLIP ViT-L/14, respectively denoted as FreeDA (ViT-B) and FreeDA (ViT-L). For this comparison, since the usage of superpixels to improve the adherence of predictions on the image can be interpreted as a mask refinement step, we also report the performance of considered competitors when using the Pixel-Adaptive Mask Refinement (PAMR) proposed in [3] to refine the final predictions. As it can be seen, both variants of our solution achieve the best results on all datasets, surpassing all the competitors by a consistent margin. Specifically, when comparing with methods without PAMR, FreeDA achieves an average improvement of 10.0 and 10.9 mIoU points with respect to TCL [6], respectively for the ViT-B and ViT-L variants. This performance improvement is confirmed also when comparing FreeDA with PAMR-based approaches, leading to an average increase of 7.0 and 7.9 mIoU points compared to the best-performing method.

| Model | PAMR | Training Dataset | mIoU | | |
|---|---|---|---|---|---|
| | | | VOC | Context | Object |
| GroupViT [43] | - | CC12M+RedCaps | 50.4 | 18.7 | 27.5 |
| MaskCLIP [52] | - | - | 38.8 | 23.6 | 20.6 |
| ReCo [38] | - | - | 25.1 | 19.9 | 15.7 |
| ViewCo [35] | - | CC12M+YFCC | 52.4 | 23.0 | 23.5 |
| SegCLIP [27] | - | CC3M+COCO Captions | 52.6 | 24.7 | 26.5 |
| TCL [6] | - | CC3M+CC12M | 51.2 | 24.3 | 30.4 |
| OVSegmentor [44] | - | CC4M | 53.8 | 20.4 | 25.1 |
| GroupViT [43] | ✓ | CC12M+YFCC | 51.1 | 19.0 | 27.9 |
| MaskCLIP [52] | ✓ | - | 37.2 | 22.6 | 18.9 |
| TCL [6] | ✓ | CC3M+CC12M | 55.0 | 30.4 | 31.6 |
| **FreeDA (ViT-L)** | - | - | **55.4** (+0.4) | **38.3** (+7.9) | **37.4** (+5.8) |

Table 2. Comparison with state-of-the-art unsupervised open-vocabulary semantic segmentation models on the validation sets of Pascal VOC [11], Pascal Context [30], and COCO Object [4], when considering the additional unknown category.

In Table 2, we instead report the results on the three segmentation datasets, namely Pascal VOC, Pascal Context, and COCO Object, used to validate the effectiveness of segmentation methods when also considering the additional "unknown" category. Following [43], we apply a threshold on the final similarities to detect pixels that do not belong to any of the provided input categories. In particular, we apply the threshold on the similarity values obtained after ensembling local and global similarities. For this experiment, we restrain the comparison to methods that do not employ specific techniques to take into account the background of the scene but instead perform a threshold as done in our case. Notably, FreeDA achieves the best results on

| Backbone | Global Similarity | Superpixels | mIoU | | |
|---|---|---|---|---|---|
| | | | VOC | Cityscapes | ADE |
| CLIP (ViT-B/16) | ✗ | ✗ | 61.3 | 21.3 | 13.4 |
| DINO (ViT-B/16) | ✗ | ✗ | 34.2 | 26.0 | 9.5 |
| DINOv2 (ViT-B/14) | ✗ | ✗ | 75.6 | 34.4 | 20.7 |
| DeiT-III (ViT-L/16) | ✗ | ✗ | 54.8 | 21.8 | 11.4 |
| CLIP (ViT-L/14) | ✗ | ✗ | 45.9 | 20.0 | 11.4 |
| DINOv2 (ViT-L/14) | ✗ | ✗ | 70.2 | 33.2 | 19.5 |
| DINO (ViT-B/16) | ✓ | ✗ | 80.4 | 27.8 | 16.5 |
| DINOv2 (ViT-B/14) | ✓ | ✗ | 86.2 | 35.0 | 21.9 |
| DINOv2 (ViT-L/14) | ✓ | ✗ | 87.2 | 34.5 | 21.6 |
| DINO (ViT-B/16) | ✓ | ✓ | 81.1 | 29.8 | 17.3 |
| DINOv2 (ViT-B/14) | ✓ | ✓ | 87.0 | 36.6 | **23.2** |
| DINOv2 (ViT-L/14) | ✓ | ✓ | **87.9** | **36.7** | **23.2** |

Table 3. Ablation study results using different visual backbones and validating the contribution of the key components of our solution. Results are reported on the validation sets of Pascal VOC [11], Cityscapes [8], and ADE20K [50, 51].

all three benchmarks, surpassing both methods that do not employ any mask refinement stages and approaches that instead refine their predictions using PAMR [3]. In particular, FreeDA reaches 55.4, 38.3, and 37.4 mIoU points respectively on Pascal VOC, Pascal Context, and COCO Object, which correspond to an improvement of 0.4, 7.9, and 5.8 points with respect to the best method (*i.e.*, TCL [6] using PAMR as mask refinement technique).

These results highlight the effectiveness of our solution which, despite being completely training-free, achieves a new state of the art for unsupervised open-vocabulary semantic segmentation on all eight considered benchmarks. Some qualitative results are shown in Figure 4.

### 4.3. Ablation Studies and Analyses

We then evaluate the contribution of each component employed in our final solution and the effectiveness of different backbones to extract visual and textual features.

**Effect of Changing the Visual Backbone.** We first consider the performance of our approach when using different visual backbones to compute local similarities. In particular, we evaluate DeiT-III [40] pre-trained for image classification on ImageNet1k and based on ViT-L/16, CLIP [34] in both its ViT-B/16 and ViT-L/14 versions, DINO [5] based on the ViT-B/16 architecture, and our final choice DINOv2 [33] using both the variant based on ViT-B/14 and the one based on ViT-L/14. Given that different input and patch sizes can lead to different output feature sizes, we resize all images to $518 \times 518$ when using visual backbones with a patch size of 14 and $592 \times 592$ when employing visual backbones with a path size of 16, thus always having features with a spatial size equal to $37 \times 37$. To validate only the role of different visual backbones, we apply them without global similarities and without superpixels to extract mask proposals. When considering the variant without superpixels, we directly compute the local similarities on the dense features and we interpolate them to the original image size.

| Local Backbone | Textual/Global Backbone | mIoU | | |
|---|---|---|---|---|
| | | VOC | Cityscapes | ADE |
| DINO (ViT-B/16) | CLIP (ViT-B/16) | 80.8 | 30.6 | 17.0 |
| DINOv2 (ViT-B/14) | CLIP (ViT-B/16) | 85.6 | **36.7** | 22.4 |
| DINOv2 (ViT-L/14) | CLIP (ViT-B/16) | 86.9 | 36.3 | 22.3 |
| DINOv2 (ViT-L/14) | CLIP (ViT-L/14) | **87.9** | **36.7** | **23.2** |

Table 4. Performance analysis when employing visual and textual backbones of different sizes.

Results are reported in the upper part of Table 3, using the CLIP ViT-L/14 model to extract textual features. As it can be noticed, DINOv2 exhibits the best performance among both architectures based on ViT-B and ViT-L, confirming the power of self-supervised features in this setting.

**Adding Global Similarities and Superpixels.** To evaluate the contribution of global features and superpixel-based mask proposals, we report in the lower part of Table 3 the performance of FreeDA first adding only global similarities and then also including superpixels to extract mask proposals. Both strategies give a consistent contribution to the final performance, also when considering different visual backbones to compute local similarities. For example, when using DINOv2, global features bring an improvement of 0.9 mIoU points on the ADE20K dataset, while superpixels further enhance the final performance by an additional 1.6 mIoU points. Additionally, it is worth noting that the contribution of global similarities is more significant in Pascal VOC where images are characterized by the presence of a single or few objects occupying large areas of the scene, thus favoring global features instead of local ones.

**Impact of Backbone Size.** In Table 4 we investigate how much using a ViT-Large architecture to extract both visual and textual features increases the performance compared to a ViT-Base model. As also demonstrated by the complete results of the two variants of FreeDA reported in Table 1, this corresponds to around 2.3 mIoU points on Pascal VOC when employing DINOv2 to extract local features, while obtaining similar performance on Cityscapes and ADE20K.

**Superpixel Algorithms and Prototype Aggregation Strategies.** In Table 5, we instead validate the choice of employing Felzenszwalb's algorithm [12] to extract superpixels by comparing it with three widely adopted superpixel proposal algorithms, namely Watershed [15], SLIC [2], and SEEDS [41]. While different versions of superpixel algorithms lead to similar performance, the usage of Felzenszwalb's algorithm helps to further improve the results on all three datasets considered. In addition to comparing different superpixel extraction strategies, we also include the results obtained using PAMR [3] as a mask refinement method. For this experiment, we first compute local similarities for dense features and ensemble them with the global similarity, then we apply PAMR to refine the resulting seg-

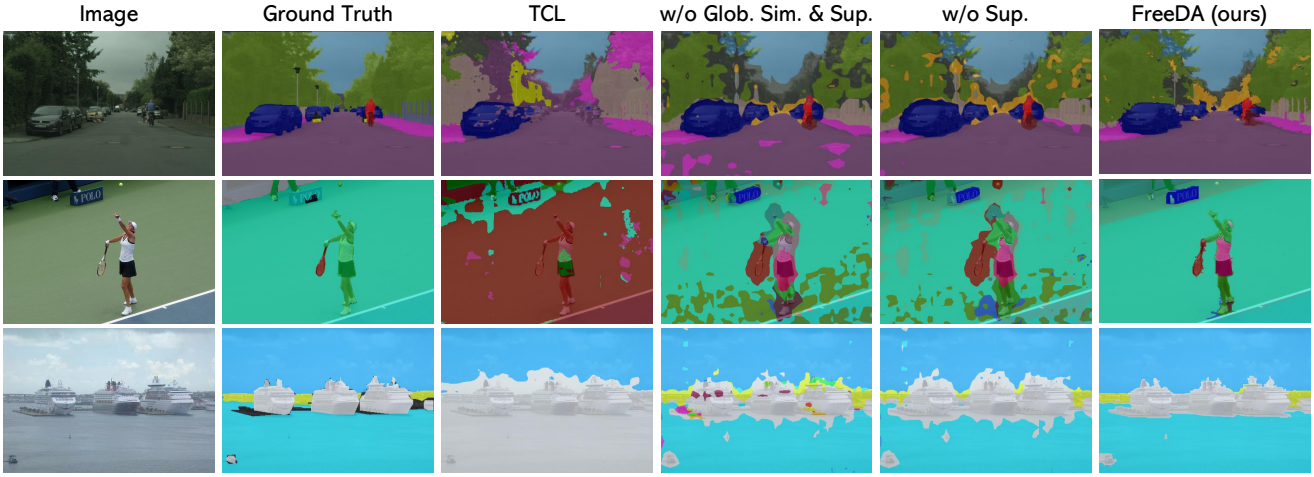


Figure 4. Qualitative results of FreeDA in comparison with TCL [6], with and without global similarities and superpixels.

| Model | Superpixels | mIoU | | |
|---|---|---|---|---|
| | | VOC | Cityscapes | ADE |
| w/ mean embedding (PAMR) | - | 87.0 | 34.4 | 23.0 |
| w/ mean embedding | Watershed | 87.0 | 32.7 | 21.8 |
| w/ mean embedding | SLIC | 87.3 | 33.5 | 21.8 |
| w/ mean embedding | SEEDS | 87.5 | 32.3 | 22.4 |
| w/ mean similarity | Felzenszwalb | 79.5 | 29.3 | 18.8 |
| w/ max similarity | Felzenszwalb | 82.0 | 26.2 | 17.6 |
| **FreeDA** (w/ mean embedding) | Felzenszwalb | **87.9** | **36.7** | **23.2** |

Table 5. Performance analysis using different algorithms to compute superpixels and different prototypes aggregation strategies.

mentation masks. Notably, employing superpixels to extract mask proposals leads to improved final results.

To validate the aggregation strategy used in FreeDA, in which we aggregate retrieved prototypes by computing their average embedding (*i.e.*, "mean embedding" in Table 5), we compare it with two different approaches based on first computing local similarities for all retrieved prototypes and then aggregating them by considering the mean or the maximum (*i.e.*, "mean similarity" and "max similarity"). Computing the average embedding of all retrieved prototypes brings the best results across all datasets.

**Retrieval Performance Analysis.** Finally, we analyze the performance when varying the retrieval parameters. Since our method leverages an exact retrieval index, we first validate how much using an approximate search impacts the performance. Specifically, the left plot of Figure 5 shows the trade-off between speed and performance when using a graph-based HNSW (Hierarchical Navigable Small World) index [29]. We report the CPU times to search the most similar $K = 350$ key embeddings when changing the depth of exploration in the index, and their corresponding mIoU scores. This parameter controls the size of the dynamic list of candidate nearest neighbors that are explored during the search process. On the right plot of Figure 5, we instead show the performance variation when changing the number

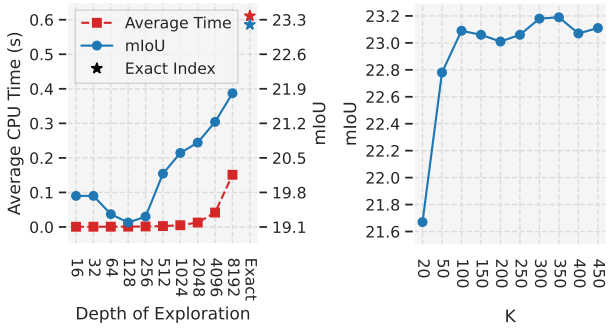


Figure 5. Retrieval results when using an approximate index (left) and varying the number of retrieved key-prototype pairs (right).

$K$ of searched keys. Results are reported on the ADE20K dataset. As it can be seen, using an approximate index only partially deteriorates the performance while consistently reducing time computation. On the same line, increasing the number of retrieved key embeddings does not improve the final performance, while retrieving a reduced number of items partially leads to lower results.

## 5. Conclusion

We presented FreeDA, a training-free approach for unsupervised open-vocabulary segmentation. Our approach leverages visual prototypes and textual keys extracted offline with diffusion-augmented generation and exploits local-global similarities at inference time. Experimentally, we achieve state-of-the-art results on five different datasets.

## Acknowledgment

This work has been conducted under a research grant co-funded by Leonardo S.p.A., and supported by the PNRRM4C2 project "FAIR - Future Artificial Intelligence Research" and by the EU Horizon project "ELIAS - European Lighthouse of AI for Sustainability" (No. 101120237), both funded by the European Commission.

## References


[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels. Technical report, EPFL Technical Report, 2010. 3

[2] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. PAMI*, 34(11):2274–2282, 2012. 3, 7

[3] Nikita Araslanov and Stefan Roth. Single-Stage Semantic Segmentation from Image Labels. In *CVPR*, 2020. 6, 7

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and Stuff Classes in Context. In *CVPR*, 2018. 2, 5, 6, 12

[5] 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. 7

[6] 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. 1, 2, 6, 7, 8, 15

[7] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO Captions: Data Collection and Evaluation Server. *arXiv preprint arXiv:1504.00325*, 2015. 5

[8] 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. 2, 5, 6, 7, 12

[9] 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. 6

[10] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *CVPR*, 2022. 1, 2

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results, 2012. 2, 5, 6, 7, 12

[12] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59:167–181, 2004. 5, 7, 11

[13] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 1, 2

[14] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *JMLR*, 23 (1):2249–2281, 2022. 1

[15] Zhongwen Hu, Qin Zou, and Qingquan Li. Watershed superpixel. In *ICIP*, 2015. 3, 7

[16] 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 *ICML*, 2021. 1, 2

[17] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Trans. on Big Data*, 7(3):535–547, 2019. 5

[18] Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. Diffusion Models for Zero-Shot Open-Vocabulary Segmentation. *arXiv preprint arXiv:2306.09316*, 2023. 2, 6

[19] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 1

[20] Zhengqin Li and Jiansheng Chen. Superpixel segmentation using linear spectral clustering. In *CVPR*, 2015. 3

[21] Ziyi Li, Qinye Zhou, Xiaoyun Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Open-vocabulary object segmentation with diffusion models. In *ICCV*, 2023. 1, 2

[22] 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. 2

[23] 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 *ECCV*, 2014. 2, 5

[24] Quande Liu, Youpeng Wen, Jianhua Han, Chunjing Xu, Hang Xu, and Xiaodan Liang. Open-world semantic segmentation via contrasting and clustering vision-language embedding. In *ECCV*, 2022. 1

[25] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1

[26] Edward Loper and Steven Bird. NLTK: The Natural Language Toolkit. *arXiv preprint cs/0205028*, 2002. 3

[27] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. SegCLIP: Patch Aggregation with Learnable Centers for Open-Vocabulary Semantic Segmentation. In *ICML*, 2023. 1, 6

[28] Vaïa Machairas, Etienne Decencière, and Thomas Walter. Waterpixels: Superpixels based on the watershed transformation. In *ICIP*, 2014. 3

[29] Yu A Malkov and Dmitry A Yashunin. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. *IEEE Trans. PAMI*, 42(4): 824–836, 2018. 8

[30] 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. 2, 5, 6, 12

[31] Peer Neubert and Peter Protzel. Compact watershed and preemptive slic: On improving trade-offs of superpixel segmentation algorithms. In *ICPR*, 2014. 3

[32] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 1

[33] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al.

DINOv2: Learning Robust Visual Features without Supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 3, 5, 7

[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 *ICML*, 2021. 1, 2, 5, 7, 11

[35] 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. In *ICLR*, 2023. 1, 6

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis With Latent Diffusion Models. In *CVPR*, 2022. 1, 2, 3, 5

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 3

[38] Gyungin Shin, Weidi Xie, and Samuel Albanie. ReCo: Retrieve and Co-segment for Zero-shot Transfer. In *NeurIPS*, 2022. 6

[39] Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. What the DAAM: Interpreting Stable Diffusion Using Cross Attention. In *ACL*, 2022. 1, 2, 3

[40] Hugo Touvron, Matthieu Cord, and Hervé Jégou. DeiT III: Revenge of the ViT. In *ECCV*, 2022. 7

[41] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin De Capitani, and Luc Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *ECCV*, 2012. 7

[42] Weijia Wu, Yuzhong Zhao, Mike Zheng Shou, Hong Zhou, and Chunhua Shen. DiffuMask: Synthesizing Images with Pixel-level Annotations for Semantic Segmentation Using Diffusion Models. *arXiv preprint arXiv:2303.11681*, 2023. 1, 2

[43] 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. 1, 2, 6

[44] 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. 1, 2, 6

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

[46] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A Simple Baseline for Open-Vocabulary Semantic Segmentation with Pre-trained Vision-Language Model. In *ECCV*, 2022. 1

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

[48] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A Tale of Two Features: Stable Diffusion Complements DINO for Zero-Shot Semantic Correspondence. In *NeurIPS*, 2023. 5

[49] Hang Zhao, Xavier Puig, Bolei Zhou, Sanja Fidler, and Antonio Torralba. Open vocabulary scene parsing. In *CVPR*, 2017. 1

[50] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene Parsing Through ADE20K Dataset. In *CVPR*, 2017. 2, 5, 6, 7, 11, 12

[51] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic Understanding of Scenes Through the ADE20K Dataset. *IJCV*, 127(3):302–321, 2019. 2, 5, 6, 7, 11, 12

[52] Chong Zhou, Chen Change Loy, and Bo Dai. Extract Free Dense Labels from CLIP. In *ECCV*, 2022. 1, 2, 6

In this supplementary material, we delve into additional implementation details pertaining to our prototype generation process, offering information to facilitate reproducibility. A comprehensive list of the used textual prompts is presented to clarify the experimental setup. We systematically explore the impact of varying superpixel hyperparameters on the overall performance of our proposed model. We examine the combined influence of entire caption contexts and word embeddings during prototype generation. Our findings highlight the effectiveness of this approach, particularly for categories consisting of multiple words. We also investigate the impact of employing the unimodal backbone for both local and global matching. Our results demonstrate the advantage of leveraging a multimodal feature extractor like CLIP for global matching. To enhance interpretability, we include visual examples showcasing captions, generated images, and their corresponding attributions and binary masks. Additionally, we include qualitative results across all the considered benchmark datasets. We conduct a thorough examination of both successful cases and instances of failure, supplementing our analysis with "into the wild" examples—segmentation results obtained by prompting our model with diverse free-form textual inputs.

## A. Additional Implementations Details

**Textual Templates.** To encode through the CLIP text encoder both the nouns extracted during prototype generation and the input categories utilized at inference time, we employ the following set of templates $\mathcal{T}$, introduced in [34]:

```
itap of a {}.
a bad photo of the {}.
a origami {}.
a photo of the large {}.
a {} in a video game.
art of the {}.
a photo of the small {}.
```

As discussed in [34], these templates provide a powerful means of contextualizing textual input, making them particularly well-suited for our application in the context of prototype generation and inference.

**Prototypes generation.** The foundation of our prototype generation lies in the utilization of a dataset of images paired with captions. To ensure the reproducibility of our results, we detail the negative prompts employed during the generation of images with Stable Diffusion in Table 6. These negative prompts play a crucial role in guiding the generation process, aiming to produce prototypes that are realistic and high-quality. The prototypes generation is performed offline and requires around 5.2 sec for each COCO caption. During inference, computing a category embedding and performing prototypes retrieval takes around 10.8 ms and 12.9 ms for the Base and Large versions of FreeDA.

| | | |
|---|---|---|
| *3d* | *abstract* | *art* |
| *asymmetric* | *bad anatomy* | *bad art* |
| *bad proportions* | *blurry* | *canvas frame* |
| *cartoon* | *cartoonish* | *cgi* |
| *cloned face* | *colorless* | *computer graphic* |
| *cropped* | *cut off* | *deformed* |
| *dehydrated* | *digital* | *digital art* |
| *disfigured* | *doll* | *duplicate* |
| *error* | *extra arms* | *extra fingers* |
| *extra legs* | *extra limbs* | *fused fingers* |
| *fuzzy* | *grainy* | *graphic* |
| *gross proportions* | *inaccurate* | *jpeg artifacts* |
| *long neck* | *low quality* | *low-resolution* |
| *lowres* | *malformed limbs* | *misshaped* |
| *missing arms* | *missing legs* | *morbid* |
| *mutant* | *mutated* | *mutated hands* |
| *mutation* | *mutilated* | *octane* |
| *out of focus* | *out of frame* | *oversaturated* |
| *photoshop* | *poorly drawn face* | *poorly drawn hands* |
| *render* | *retro* | *signature* |
| *text* | *too many fingers* | *ugly* |
| *unreal* | *unreal engine* | *unrealistic* |
| *username* | *video game* | *watermark* |
| *weird colors* | *worst quality* | |

Table 6. Negative prompts employed in Stable Diffusion during prototypes generation.

## B. Additional Experiments and Analyses

**Effect of Superpixel Parameters.** Felzenszwalb *et al*. [12] introduced an efficient superpixel algorithm that employs a graph-based approach. The algorithm initiates by constructing a graph representation of the image, where each pixel serves as a node, and edges connect neighboring pixels. Edge weights are determined based on the RGB color space differences between adjacent pixels. Consequently, connected components, initially established as individual components for each pixel, are progressively merged. The growth of each component is regulated by the scale of observation parameter $k$. The algorithm also incorporates two additional parameters: the diameter of the Gaussian filter used for pre-processing to enhance image smoothness and counter artifacts ($\sigma$), and the enforced minimum size of superpixels, $\mu$. We employ the implementation of the `skimage`[1] library.

In Table 7, we report the parameter values employed on the examined datasets. Figure 6 further shows the performance variations obtained when altering these parameters on the ADE20K dataset [50, 51]. Notably, minor variations in these parameters have negligible effects on final performance. However, imposing large superpixels through minimum size or scale of observation can significantly degrade the results.

[1] https://scikit-image.org/

| Dataset | $\mu$ | $\sigma$ | $k$ |
|---|---|---|---|
| Pascal VOC | 100 | 0.7 | 20 |
| Pascal Context | 100 | 1.0 | 20 |
| COCO Stuff | 100 | 1.0 | 100 |
| Cityscapes | 50 | 0.5 | 20 |
| ADE20K | 100 | 1.0 | 20 |

Table 7. Parameters employed for Felzenszwalb's algorithm on each dataset.

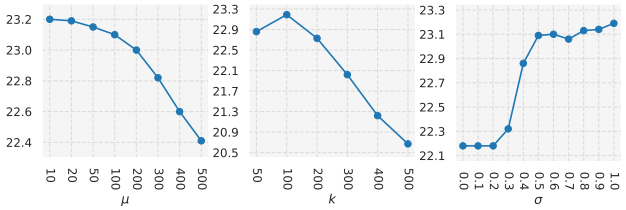


Figure 6. Effect of the variation of superpixel hyperparameters on ADE20K, measured in terms of mIoU.

**Impact of caption context.** In Section 3.1 of the main paper, we outline our methodology for extracting textual key embeddings. Specifically, we employ a linear combination of the word embedding $\hat{t}$ and the caption embedding $\hat{c}$, controlled by a parameter $\alpha$. In our main results, we set $\alpha$ to 0.9 to effectively incorporate the textual context into the key embedding.

In Table 8, we conduct an ablation study on this choice. The case without caption context corresponds to setting $\alpha$ to 1. It is noteworthy that the inclusion of textual context proves to be particularly beneficial for input categories that consist of more than one word, such as `chest of drawers`. This scenario is prevalent in in-the-wild situations, thus emphasizing the practical utility of our approach in diverse and real-world settings.

**Impact of unimodal global matching.** In Table 9, we investigate the impact of employing DINOv2 for local and global matching. Since DINOv2 embeddings are not aligned with text, we compute global matching by using the similarity between the `CLS` token of DINOv2 and the representative visual prototypes of the categories. As can be observed, the usage of a text-aligned CLIP backbone improves performance w.r.t. the unimodal DINOv2 global features.

## C. Explainability

A notable advantage of our prototype-based approach lies in its inherent explainability, as the set of referring images used to generate prototypes can be visualized a posteriori. In our approach, in particular, we can visualize the generated images associated with the retrieved prototypes for a given input category, along with the corresponding attribution maps and binary masks.

Figure 9 illustrates the explainability capabilities of our solution, showcasing examples of retrieved prototypes for a specified category, highlighted within the captions in which the corresponding noun was mentioned. We further include the corresponding generated images, attribution maps, and binarized masks, providing a comprehensive view of the explainability achieved by our approach.

| | Caption Context | mIoU Context | mIoU Stuff | mIoU ADE |
|---|---|---|---|---|
| | ✗ | 43.1 | 27.4 | 22.2 |
| **FreeDA** | ✓ | **43.5** | **28.8** | **23.2** |

Table 8. Effect of full caption embeddings on the performance of key embeddings.

| Local Backbone | Global Backbone | VOC | Cityscapes | ADE |
|---|---|---|---|---|
| DINOv2 (ViT-B/14) | DINOv2 (ViT-B/14) | 78.4 | 30.7 | 17.8 |
| DINOv2 (ViT-L/14) | DINOv2 (ViT-L/14) | 74.4 | 33.5 | 20.3 |
| DINOv2 (ViT-B/14) | CLIP (ViT-B/16) | 85.6 | **36.7** | 22.4 |
| DINOv2 (ViT-L/14) | CLIP (ViT-L/14) | **87.9** | **36.7** | **23.2** |

Table 9. mIoU results with DINOv2 for local/global matching.

## D. Additional Qualitative Results

**Results on benchmark datasets.** Figure 10 shwocases additional qualitative results on Pascal VOC [11], Pascal Context [30], COCO Stuff [4], Cityscapes [8], and ADE20K [50, 51]. These qualitative samples offer a comprehensive view of the performance of our approach, and highlight the versatility and effectiveness of our method across a range of scenes and categories, reinforcing its applicability in various real-world scenarios.

**In-the-wild results.** Additionally, in Figure 7 we report a collection of in-the-wild examples obtained by prompting our model with diverse free-form textual inputs. Specifically, we extract noun chunks from sample captions of the COCO Captions validation set using the `spaCy`[2] NLP library. After removing stop-words, the noun chunks are utilized as input categories for segmenting the corresponding images. These results extend our analysis beyond curated datasets and demonstrate the adaptability and robustness of our approach in handling real-world scenarios with varied and unstructured textual descriptions.

**Failure cases.** Finally, in Figure 8 we report sample scenarios in which our model encounters challenges and exhibits failure cases. The first row illustrates an image of a TV displaying a video game. Owing to the strong semantic correspondence properties at the token-level of DINOv2, our model tends to segment individual elements shown on the TV screen, thereby impacting the overall segmentation performance for the TV class. The second row of the figure instead presents another failure case featuring an image

[2] https://spacy.io/

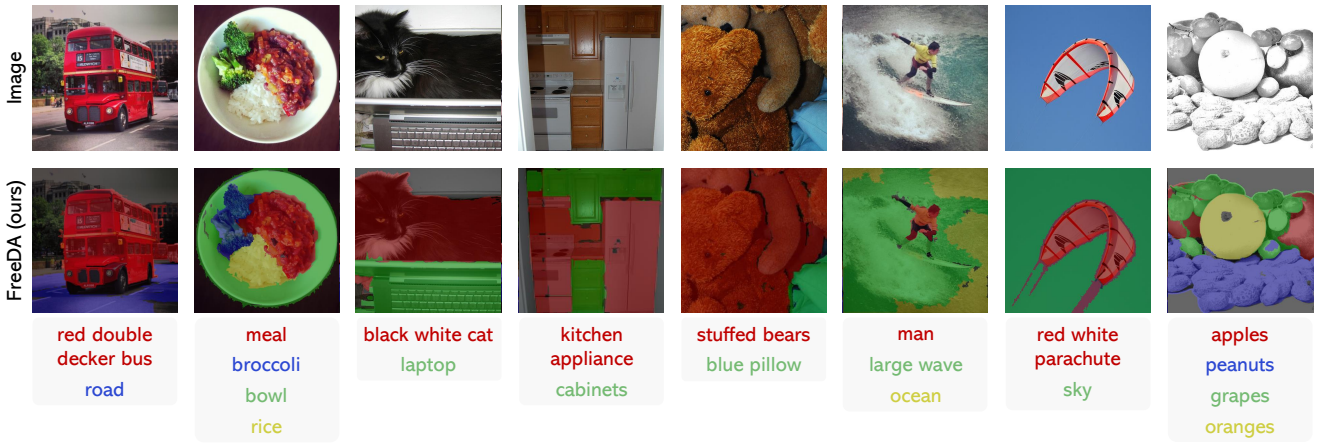


Figure 7. In-the-wild segmentation results obtained by prompting our model with diverse free-form textual inputs.

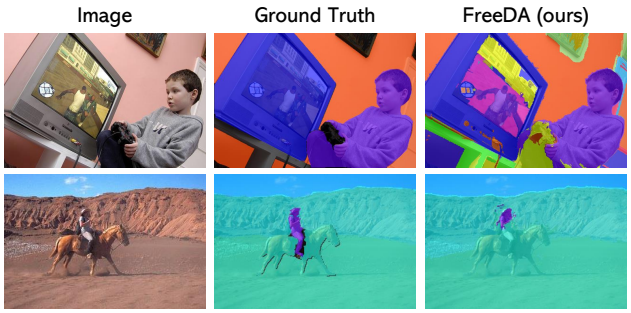


Figure 8. Sample failure cases.

of a person atop a horse. However, the segmentation is incomplete and only partially captures the person. This limitation can be attributed to the prototypes corresponding to horses ridden by persons, whose noisy binarized masks include their legs. Overall, these failure cases shed light on areas where our model may struggle, emphasizing the need for further refinement and consideration of complex visual contexts.

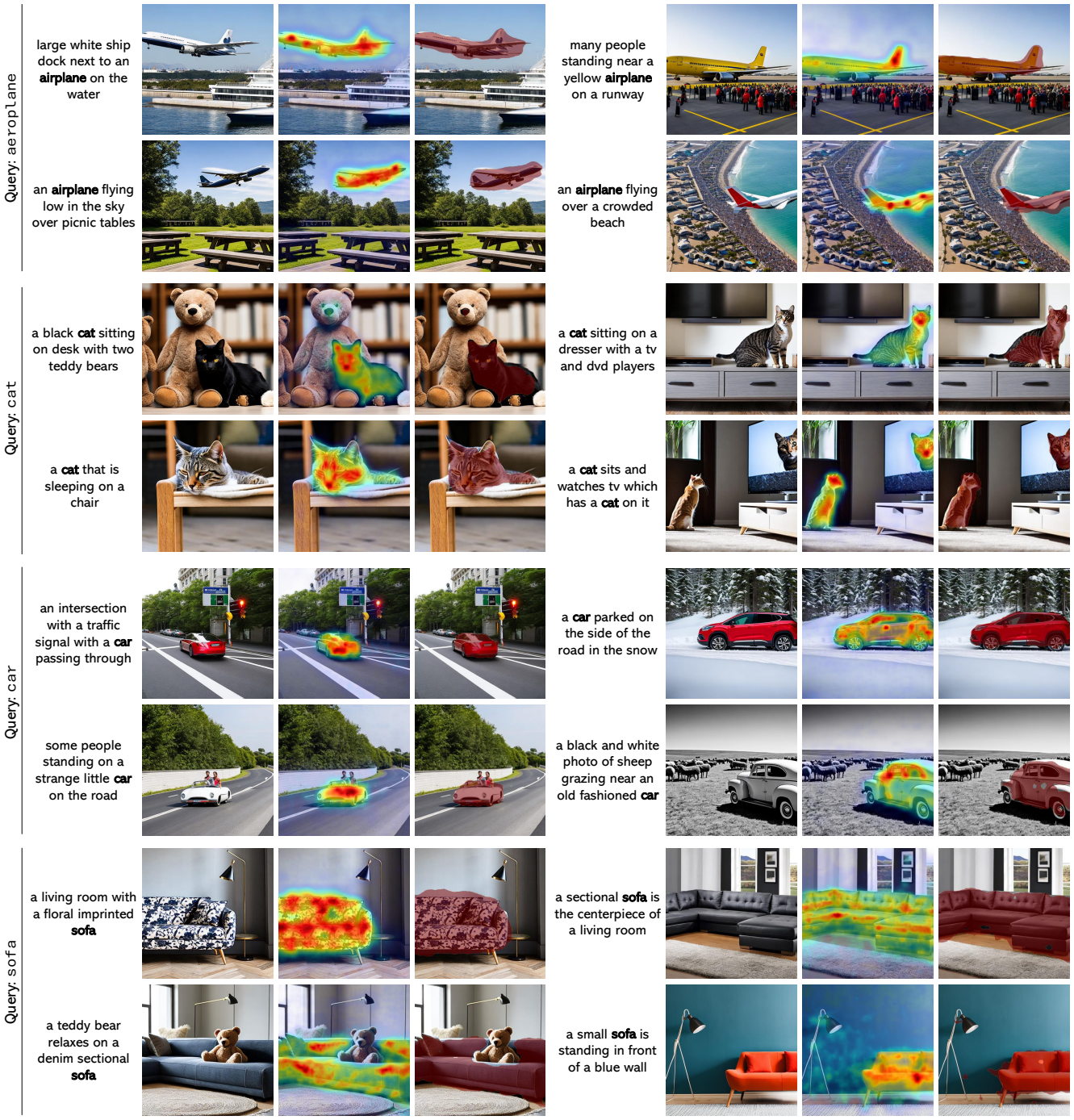


Figure 9. Examples of retrieved prototypes for a specified textual category. From left to right, we show the original COCO caption, the corresponding generated image, the attribution map, and the binarized mask (area highlighted in red).

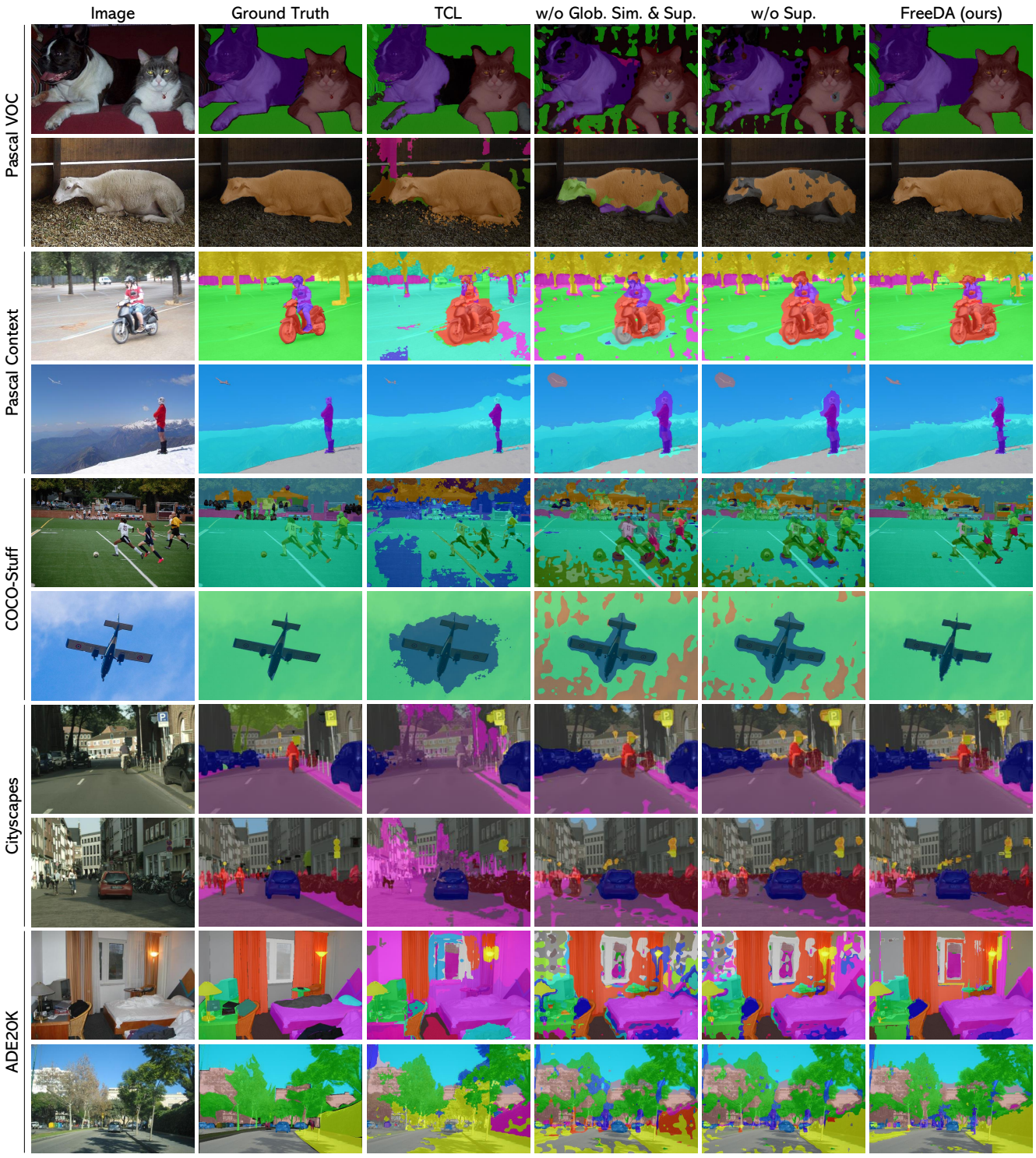


Figure 10. Additional qualitative results of FreeDA in comparison with TCL [6], with and without global similarities and superpixels.