Enhancing Open-Vocabulary Semantic Segmentation with Prototype Retrieval

Luca Barsellotti[1][0000−0001−8845−8523], Roberto Amoroso[1][0000−0002−1033−2485], Lorenzo Baraldi[1][0000−0001−5125−4957], and Rita Cucchiara[1][0000−0002−2239−283X]

University of Modena and Reggio Emilia, Modena, Italy
{name.surname}@unimore.it

Abstract. Large-scale pre-trained vision-language models like CLIP exhibit impressive zero-shot capabilities in classification and retrieval tasks. However, their application to open-vocabulary semantic segmentation remains challenging due to the gap between the global features extracted by CLIP for whole-image recognition and the requirement for semantically detailed pixel-level features. Recent two-stage methods have attempted to overcome these challenges by generating mask proposals that are agnostic to specific classes, thereby facilitating the identification of regions within images, which are subsequently classified using CLIP. However, this introduces a significant domain shift between the masked and cropped proposals and the images on which CLIP was trained. Fine-tuning CLIP on a limited annotated dataset can alleviate this bias but may compromise its generalization to unseen classes. In this paper, we present a method to address the domain shift without relying on fine-tuning. Our proposed approach utilizes weakly supervised region prototypes acquired from image-caption pairs. We construct a visual vocabulary by associating the words in the captions with region proposals using CLIP embeddings. Then, we cluster these embeddings to obtain prototypes that embed the same domain shift observed in conventional two-step methods. During inference, these prototypes can be retrieved alongside textual prompts. Our region classification incorporates both textual similarity with the class noun and similarity with prototypes from our vocabulary. Our experiments show the effectiveness of using retrieval to enhance vision-language architectures for open-vocabulary semantic segmentation.

Keywords: Open-Vocabulary · Semantic Segmentation · Retrieval

1 Introduction

Semantic segmentation is a widely studied Computer Vision task, involving the partitioning of an image into regions that correspond to specific object classes with semantic meaning. However, obtaining precise annotations for this task can be costly, hindering scalability to large datasets. In addition, conventional semantic segmentation models [5,7] are typically trained on a finite set of classes, making them unable to recognize novel or unexpected objects. To overcome these challenges, recent studies have focused on developing open-vocabulary semantic segmentation models [9,12,22,27] that can recognize a variable number of classes, including previously unseen or out-of-domain samples. These models offer greater flexibility and applicability to various real-world scenarios, such as robotics, autonomous driving, and medical image analysis [4,8].

The growing interest in open-vocabulary semantic segmentation can be attributed to the emergence of large-scale pre-trained vision-language models, such as CLIP [24] and ALIGN [16]. These models have been trained on billions of image-text training examples, enabling them to learn rich multi-modal features. Notably, they exhibit the ability to embed a vast vocabulary and, consequently, excellent zero-shot capabilities when applied to downstream tasks such as classification [10,3] and image retrieval. However, transferring this knowledge to dense prediction tasks presents challenges, as the model must not only identify the object classes within an image but also precisely localize them.

Two-stage approaches have emerged as effective methods for addressing the open-vocabulary segmentation task and tackling the localization problem. These methods involve two stages: first, a mask proposer generates class-agnostic mask proposals. Then, the image regions corresponding to the generated masks are extracted, and a CLIP model is used to perform open-vocabulary classification on each region. Although the class-agnostic proposer demonstrates strong generalization to arbitrary categories [23], the bottleneck in the performance is represented by the inability of CLIP in recognizing the masked and cropped image regions [18]. This limitation stems from the domain shift between the images provided to CLIP during training and those used in this setup. Resizing, masking and cropping the object image adversely affect its positioning in the feature space with respect to text embeddings. However, fine-tuning CLIP on a closed-vocabulary annotated dataset to compensate for this domain shift may interfere with its generalization capabilities on unseen classes.

To address the challenge introduced by the domain shift *without fine-tuning*, we propose a pre-processing step that involves creating a visual vocabulary that associates a given word with a series of reference CLIP visual feature embeddings. These embeddings are generated by collecting region proposals extracted from an image-caption dataset and by applying a clustering algorithm on top of them. Thus, the resulting cluster centroids incorporate the same domain shift while providing a rich variety of visual characteristics of the corresponding word. Alongside CLIP's open-vocabulary classification on each region, the vocabulary visual reference embeddings can be retrieved to augment the segmentation process, thereby improving its robustness and accuracy.

Our experiments demonstrate the effectiveness of integrating retrieval methods to enhance the two-stage architecture without the need for further fine-tuning. The combination of the visual vocabulary reference embeddings and the two-step segmentation approach yields enhanced performance, highlighting the potential of utilizing pre-existing knowledge and domain adaptation techniques to address the domain shift challenge in open-vocabulary segmentation.

2 Related Works

Semantic Segmentation is a fundamental dense prediction task in Computer Vision that aims to assign a label to each pixel of an image. The field is primarily driven by two main lines of research: one that treats it as a pixel-level classification problem [1,2,5], and another that decouples it into a two-subtask problem [7], involving the proposal of regions of interest and the subsequent classification of these proposed regions. Both approaches have shown excellent performance in closed-vocabulary scenarios under the supervised learning paradigm.

Zero-Shot Semantic Segmentation has gained significant attention in recent years, driven by the high costs associated with annotating masks for a wide range of categories. In this setting, models are trained on a set of seen classes and are then expected to generalize their knowledge to unseen classes. While early works predominantly relied on discriminative [26] and generative methods [14], recent advancements have shifted towards the decoupling paradigm [9,28]. These methods aim to enhance the generalization capabilities of the class-agnostic mask proposer, enabling it to accurately identify novel objects. Additionally, they leverage the power of large-scale pre-trained vision-language models to assign appropriate labels to each region proposal, further improving the overall performance of zero-shot semantic segmentation. Our proposed method is closely related to zero-shot semantic segmentation as it harnesses pre-existing knowledge encoded in the visual vocabulary and employs reference embeddings to enhance the segmentation performance for previously unseen categories.

Open-Vocabulary Segmentation is a generalized zero-shot learning task that aims to establish a method for arbitrary recognition of an unlimited number of object classes, even with the use of additional training data. LSeg [17] aligns dense per-pixel and textual embeddings in the same semantic space, whereas OpenSeg [12] and GroupViT [27] propose to group pixels before learning visual-semantic alignments. Some methods, such as MaskCLIP [30] and PACL [22], investigate the capability of CLIP itself in producing dense predictions already aligned with text embeddings. Two-stage approaches have proven remarkable performance in open-vocabulary segmentation, compensating for the poor localization ability of CLIP. Their main bottleneck is given by the domain shift between the masked regions and the images on which CLIP has been trained. To bridge this gap, ZSSeg [28] proposes a textual prompt-learning approach, whereas OVSeg [18] exploits the usage of learnable tokens to replace blank areas of the masked regions. In our proposed method, we tackle the domain shift issue by constructing a visual vocabulary that aligns with the preprocessing steps applied to the input images. This alignment effectively incorporates the domain shift and improves the robustness of the model.

3 Method

Open vocabulary semantic segmentation involves the task of assigning a label from a set of arbitrary categories to each pixel in an image. In two-stage meth-

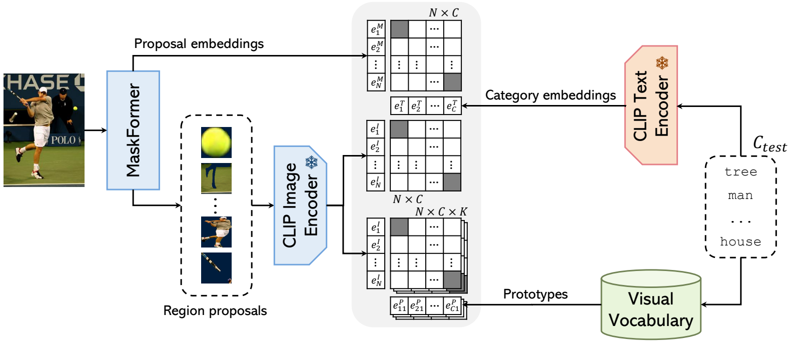

Fig. 1. Overview of our proposed method, VOCSeg, for two-stage open-vocabulary semantic segmentation enhanced by visual prototype retrieval.

ods [9,18,28], this task is reformulated into dividing the image into coherent regions and assigning each region a category.

In our proposed open-vocabulary semantic segmentation architecture depicted in Figure 1, we introduce a novel approach to tackle these challenges. The architecture comprises three main components: a mask proposer, an enhanced CLIP model with retrieval capabilities, and a visual vocabulary. The mask proposer generates region proposals within the image, while the CLIP model extracts embeddings for these proposed regions. These embeddings serve as representations for independent open vocabulary classification of each region. However, it is essential to consider the domain shift introduced by cropping and masking regions, as it deviates from the training images of CLIP. To mitigate this domain shift, we introduce the concept of *visual prototypes*. Firstly, we employ a two-stage segmentation method on a dataset consisting of image-text pairs to obtain region proposals for a diverse range of words. These proposals collectively form the visual vocabulary, which encapsulates the domain shift resulting from the cropping and masking process. Subsequently, we generate visual prototypes for each word by clustering the corresponding set of collected regions. These prototypes serve as representative embeddings within the feature space.

At inference time, we leverage textual category embeddings and retrieved prototypes for each category. These prototypes reside in the same feature space as the embeddings and allow us to incorporate both textual and visual similarities using only the CLIP model, avoiding an increase in computational effort.

3.1 Prototype Extraction from Image-Caption Pairs

Collecting a visual vocabulary. In our approach to open-vocabulary segmentation, it is crucial to utilize prototypes that capture both the distinctive features of each category and the domain shift resulting from masking the regions. These prototypes play a pivotal role in classifying the proposed regions by identifying visually similar correspondences. However, collecting regions for a large vocabulary represents a challenge, making the use of pre-annotated segmentation datasets in-

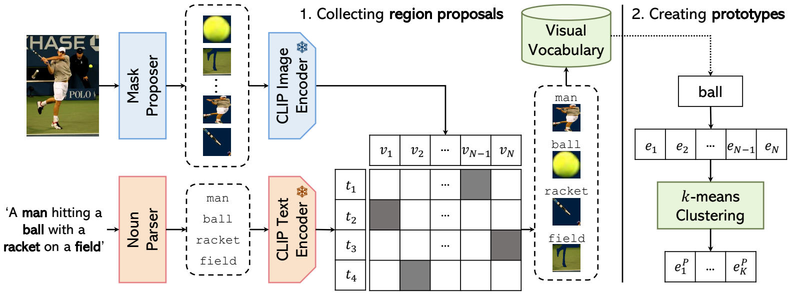

Fig. 2. Overview of the approach for collecting region proposals starting from image-caption pairs and of the clustering process used to generate prototypes.

feasible due to their limited category coverage. To tackle this challenge, we adopt a *self-labeling* strategy for constructing an open-vocabulary collection of regions. This strategy involves extracting regions from a dataset of image-caption pairs, associating them with a vocabulary based on their corresponding captions, and subsequently generating prototypes through the clustering of similar embeddings, as shown in Figure 2.

Specifically, we extract nouns from each caption, incorporate them into a text prompt, and provide them as input to the Text Encoder of a CLIP model. Subsequently, we obtain mask proposal embeddings using the Image Encoder of the same CLIP model and match the mask proposals with each noun using their respective computed embeddings. Although this matching process may introduce some noise, the presence of the noun in the caption ensures that one of the masks must be related to the corresponding object. Finally, we singularize the extracted nouns and store the CLIP embeddings of each match in a visual vocabulary.

Generating prototypes. Finally, we perform a k-means clustering on the set of collected region embeddings for each noun in the vocabulary to generate a set of prototypes, represented by the cluster centroids. The k-means algorithm groups similar features, forming representative prototypes for each noun category. In this way, we ensure that our prototypes capture a wide range of visual characteristics.

Handling rare nouns. There are cases where the number of collected embeddings may not be sufficient to perform k-means clustering effectively, either due to a limited correspondence in the captions or arbitrary test categories that do not match entries in the visual vocabulary. For these rare nouns, we employ a k-nearest neighbors algorithm. This algorithm matches the textual embeddings extracted using CLIP with the most similar words present in the vocabulary. Subsequently, we perform k-means clustering on the embeddings of the N neighbors to generate prototypes. We increment the value of N until we have an adequate number of embeddings to perform the k-means clustering effectively.

3.2 Two-Stage Open-Vocabulary with Prototype Retrieval

The objective of two-stage open-vocabulary semantic segmentation is to identify a pair of mappings $(\mathcal{S}, \mathcal{L})$ for an input image $I \in \mathbb{R}^{H\times W\times 3}$ across C_{test} arbitrary categories. In this task, $\mathcal{S}$ partitions I into a set P of T regions, defined as follows:

$$P = \{P_i,\}_{i=1}^{T} \quad \text{with} \quad P_i \subseteq I, \cup_{i=1}^{T} P_i = I, \forall i, j : i \neq j, P_i \cap P_j = \emptyset \ , \tag{1}$$

whereas $\mathcal{L}$ assigns a category $c \in C_{\text{test}}$ to each region $P_i \subseteq I$, where $i = 1, \ldots, T$.

Extracting Mask Proposals Embeddings. To obtain class-agnostic mask proposals, we utilize MaskFormer [7]. This model is trained on a set of classes C_{train}, nevertheless, as reported by Xu *et al.* [28], it can generate T high-quality mask proposals $\{M_i\}_{i=1}^{T}$ and their corresponding mask embeddings, even for unseen classes. Each mask proposal $M_i \in \mathbb{R}^{H \times W}$ is converted into a binary mask $M_i^B \in 0, 1^{H \times W}$ by applying a sigmoid function followed by thresholding. The binary mask indicates the location of the object in the input image.

In the original MaskFormer [7] architecture, the mask embedding is a C_{train}-dimensional distribution that represents the probability of each training class. To extend the model to an open-vocabulary setting, inspired by [18,28], we modify MaskFormer in such a way that each mask generates an F-dimensional embedding, where F is the embedding dimension of a CLIP model. This adaptation ensures compatibility between the mask embeddings and the CLIP textual embeddings, which are extracted from the nouns of various semantic classes, thus enabling open-vocabulary capabilities. We include an additional F-dimensional learnable embedding for `no-object`.

Further, we also employ the CLIP image encoder to extract an additional set of embeddings from the proposed regions, which complements the ones generated for each region by MaskFormer. In particular, for each binary mask M_i^B, we erase the unused background, crop around a bounding box, that incorporates entirely the foreground area, and resize to the input resolution of CLIP. Then, the region is fed to CLIP to produce an embedding that can be used to compute similarity against the textual category embeddings.

Assigning proposals to classes. For each category in C_{test}, we retrieve a set of K reference prototype embeddings from a visual vocabulary. To compute the final similarities between region proposals and categories, we combine two terms: one which exploits textual category labels and one that exploits the reference prototype embeddings. In particular, for each category $c_j \in C_{\text{test}}$ we extract an embedding e_j^T with CLIP using the Textual Encoder, we retrieve a set of prototypes $\{e_{jk}^P\}_{k=1 \ldots K}$, and for each region P_i we extract an embedding e_i^I with the Image Encoder of CLIP and an embedding e_i^M with MaskFormer. First, we aggregate the prototype similarities by considering the average of the maximum similarity with the K prototypes assigned to c_j and the mean similarity with all of them. This is a trade-off between considering the nearest reference embedding which is the most significant for the current region and the robustness offered by a single average embedding representative for the whole concept:

$$s_{i,j}^P = \frac{1}{2} \max_k \text{sim}(e_i^I, e_{jk}^P) + \frac{1}{2K} \sum_{k=1}^{K} \text{sim}(e_i^I, e_{jk}^P), \tag{2}$$

where $i = 1 \ldots T$, $j = 1 \ldots |C_{\text{test}}|$, $k = 1 \ldots K$ and $\text{sim}(\cdot, \cdot)$ is the cosine similarity.

Then, since both the prototype similarities and the textual similarities are computed in the same feature space, we fuse them using a linear combination with weights α and $(1-\alpha)$. This ensembling strategy rewards the situations in which the textual and prototype similarities agree, whereas penalizes cases of disagreement. Formally, the resulting aggregated similarity is defined as

$$\tilde{s}_{i,j} = \alpha s^{P}_{i,j} + (1-\alpha) \cdot \mathrm{sim}(e^{I}_{i}, e^{T}_{j}). \tag{3}$$

The probability vector over classes $\tilde{p}$ is computed through the softmax function with a temperature τ.

Fusing with MaskFormer predictions. Since MaskFormer is trained on C_{train}, its performance is biased towards categories belonging to this set. When the object contained in the region P_i is not recognized as a category of C_{train}, MaskFormer produces an embedding similar to the `no-object` embedding. Hence, when the softmax is applied to its similarities, all the resulting probabilities corresponding to the categories of C_{test} are small, and the one corresponding to `no-object` is large, which is removed after the softmax. So, the final prediction of P_i and c_j is obtained through the weighted geometric mean, with weights β and $(1-\beta)$, between the probability $\tilde{p}$ of the visual-text branch and the probability $\hat{p}$ resulting from MaskFormer, in such a way that the prediction of MaskFormer is enhanced only when it is confident about it (*i.e.*, when c_j belongs to C_{train} too):

$$p_{i,j} = \tilde{p}^{\beta}_{i,j} \cdot \hat{p}^{(1-\beta)}_{i,j}. \tag{4}$$

Computing Semantic Segmentation. Finally, mask predictions and probabilities are aggregated to compute the semantic segmentation. Specifically, the score $z_j(q)$ of a category $c_j \in C_{\text{test}}$ in a pixel q is computed as the sum of each mask activation M_i multiplied for the corresponding probability $p_{i,j}$:

$$z_j(q) = \sum_{i=1}^{T} M_i(q) p_{i,j}. \tag{5}$$

4 Experimental Evaluation

4.1 Datasets

Following Liang *et al.* [18], we train our MaskFormer backbone on COCO-Stuff [6] using the all available 171 categories. We conduct experiments on five sets of test categories, obtained upon three datasets: PASCAL-VOC 2012 [11], ADE20k [29] (150 and 847 categories), and PASCAL-Context [21] (59 and 459 categories).

COCO-Stuff is an extension of the MS COCO [19] dataset for semantic segmentation. It contains annotations for 171 classes on 118,287 training images and 5,000 validation images. Due to its high-quality annotations, we use it as the training dataset for the mask proposer. As reported in [28,18], MaskFormer trained on a set of seen classes can produce high-quality masks on unseen classes.

PASCAL-VOC 2012 contains annotations for 20 classes on 11,185 training images and 1,449 validation images. Its classes exhibit significant overlapping with COCO-Stuff categories (95% overlap). This overlap makes it interesting to evaluate performance on known objects sampled from a distribution that differs from the distribution of the training dataset.

ADE20k is a challenging segmentation dataset containing several indoor and outdoor scenes. It is partitioned into 20,000 training images, 3,000 test images, and 2,000 validation images. In the original setting, it contains 150 classes ($\sim$ 45% overlap with COCO-Stuff), but its full version comprises more than 3,000 classes. Following [7], we evaluate the performance on the set containing 847 classes.

PASCAL-Context is an extension of the PASCAL-VOC 2010 dataset. It contains 4,998 training images and 5,005 validation images in two settings, one with the most frequently used 59 classes ($\sim$ 83% overlap with COCO-Stuff) and one with the whole 459 classes.

4.2 Experimental Setup

We train the modified MaskFormer model on the COCO-Stuff dataset, according to [18], with the Swin-B [20] backbone. We follow the original training settings of MaskFormer [7]. We use the OpenCLIP [15] implementation of CLIP with ViT-L/14 backbone trained on LAION2B [25]. To embed the category names with CLIP, we surround them with the text prompts proposed in the original CLIP [24] and in ViLD [13]. To obtain a diverse set of prototypes, we utilize COCO Captions [6]. We collect 15,000 unique nouns from the dataset. To extract binary masks we apply a threshold of 0.4 after the sigmoid.

4.3 Ablation Studies

Masking Strategy. We investigate the impact of three different masking strategies for extracting the regions detected by the mask proposer. In particular, MaskFormer generates N mask proposals denoted as $M_i \in \mathbb{R}^{H \times W}$. These proposals indicate the activation level of each position in the image with respect to the detected region. In our main pipeline, referred to as *binary* strategy, we consider the binarized masks $\{M_i^B\}_{i=1}^N$. In order to isolate the foreground object and eliminate the potential interference of surrounding context noise on the open-vocabulary classification of the region through CLIP, we erase the background information, keeping solely the foreground object. However, we also acknowledge that in certain cases, the background can provide crucial information for accurately recognizing the object. To address this, we explore two alternative strategies: one in which we crop the region without erasing the background (which we name *none*), and one, instead, in which we attenuate the background by multiplying the image pixels with a normalized heatmap derived from the originally proposed mask (termed *heatmap*). This allows us to retain some contextual information while still emphasizing the foreground object of interest.

Our experimental results, as reported in Table 1, demonstrate that the *binary strategy* provides the best mIoU scores. We argue that the noise introduced by the background overwhelms any potential advantage gained from the contextual information when it comes to clarifying the foreground object.

Table 1. Ablation on three different masking strategies, in terms of mIoU score.

	Masking Strategy		
Dataset	**None**	**Heatmap**	**Binary**
ADE-150	17.7	17.7	**22.5**
PAS-20	82.51	85.0	**93.4**

Table 2. Ablation on similarity ensembling, in terms of mIoU score.

	Similarity		
Dataset	**Text**	**Visual**	**Ensembling**
ADE-150	21.0	20.1	**22.5**
PAS-20	92.6	93.2	**93.4**

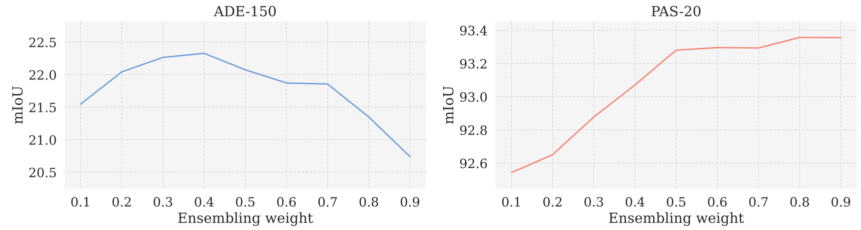

Fig. 3. Ablation on different values of the ensembling weight α.

Ensembling. In our method, we introduce the usage of CLIP for both image-to-text and image-to-image similarities to leverage their benefits concurrently. In Table 2, we present a comparison between the individual usage of these similarities, as well as their ensembling. The results show a significant improvement of $+1.5$ mIoU on the ADE-150 dataset and $+0.2$ on the PAS-20 dataset compared to the baseline that considers only visual similarity. We argue that the reason behind this observed improvement is the complementary nature of the two types of similarities provided by CLIP. Image-to-text similarity captures the semantic understanding of the textual information associated with the images, while image-to-image similarity focuses on the shared visual content between images.

In Figure 3, we present the trend of the mIoU as a function of the ensemble weight, for both ADE-150 and PAS-20 datasets. Notably, we observe that the performance trends differ between the two datasets, with ADE-150 performing better when assigning a larger weight to the text similarity, while PAS-20 performs better with a larger weight assigned to the visual similarity. We hypothesize that this discrepancy is influenced by the number of arbitrary categories in (C_{test}) and the quality of the vocabulary employed. Factors such as the number of samples collected for a specific word, the accuracy of matching region with words, the distribution of the embeddings in the feature space, and their representativeness of the semantic concept all play significant roles. These observations emphasize the need for an adaptation phase specific to the set of arbitrary classes, by tuning the value of the ensemble weight to obtain the best performance.

Number of Reference Prototypes. Figure 4 illustrates the trend of the mIoU as the number of clusters k in the k-means algorithm increases. We observe that the mIoU reaches its peak at $k = 10$ for both datasets and shows a tendency to stabilize as k further increases. The variation in mIoU can be attributed to the frequency of word occurrences in the captions. We theorize that as k increases,

[1] OpenSeg uses ALIGN as the pre-trained vision-language model instead of CLIP.

Table 3. Comparison with other state-of-the-art two-stage models.

Method	Training Dataset	Frozen CLIP	Similarity Text	Similarity Visual	PAS 20	ADE 150	ADE 847	PC 59	PC 459
GroupViT [27]	GCC+YFCC	✓	✓	✗	52.3	-	-	22.4	-
ZegFormer [9]	COCO-Stuff-156	✓	✓	✗	80.7	16.4	-	-	-
OpenSeg [16] (R-101) [1]	COCO Panoptic	✗	✓	✗	60.0	15.3	4.0	36.9	6.5
ZSSeg [28] (R-101)	COCO-Stuff-171	✗	✓	✗	88.4	20.5	7.0	47.7	-
OVSeg [18] (R-101)	COCO-Stuff-171	✗	✓	✗	89.2	24.8	7.1	53.3	11.0
OVSeg [18] (Swin-B)	COCO-Stuff-171	✗	✓	✗	94.5	29.6	9.0	55.7	12.4
VOCSeg	COCO-Stuff-171	✓	✓	✓	93.4	22.5	8.1	47.3	10.8

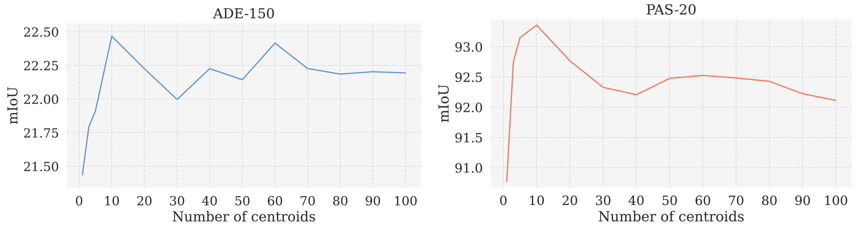

Fig. 4. Ablation on the number of clusters used in the k-means algorithm.

the noise incorporated in the reference embeddings also increases. On the other hand, when using a small value of k, the variety of representations offered by the vocabulary becomes limited. This limitation hampers the ability to embed different visual concepts under the same word, leading to decreased performance in capturing the multitude of nuances in the objects.

4.4 Comparison with state-of-the-art methods

We conduct a comparison with other open-vocabulary architectures based on a two-stage approach: GroupViT [27], ZegFormer [9], OpenSeg [16], ZSSeg [28] and OVSeg [18]. The results can be observed in Table 3. The "Similarity" column highlights the uniqueness of our approach in leveraging the similarities between image embeddings to bridge the gap between the images used to train CLIP and the regions extracted in two-stage approaches. Despite introducing a preprocessing step without additional parameters or fine-tuning CLIP, our method outperforms ZSSeg, which utilizes learnable tokens in the textual prompts, on both the ADE-150 and ADE-847 settings by +2 and +1.1 mIoU respectively and on PAS-20 by 5 mIoU. It also surpasses OpenSeg on all benchmark datasets, obtaining a +7.2 on ADE-150, +4.1 on ADE-847, +23.4 on PAS-20, +10.4 on P-59 and +4.3 on P-459. Furthermore, it outperforms OVSeg with a ResNet-101 backbone on ADE-150 by +4.2 and ADE-847 by +1.0. These architectures achieve high performance through fine-tuning or learnable tokens on a limited set of annotated segmentation data, which limits their generalization ability. In contrast, our method provides comparable results while allowing the extension of the visual vocabulary without compromising the quality of previously collected prototypes. Moreover, our VOCSeg largely outperforms ZegFormer and GroupViT, which operate in the same setting (*i.e.*, without fine-tuning CLIP). Our best performance is achieved using $k = 10$ in the k-means algorithm, $N = 10$ in the k-nearest neighbors algorithm, α equal to 0.8, 0.35, 0.2, 0.9, and 0.1 on, respectively, PAS-20, ADE-150, ADE-847, PAS-59 and PAS-459, and β equal to 0.7 on ADE-150 and ADE-847, and 0.6 on PAS-20, PAS-59, and PAS-459.

5 Conclusions

Our solution introduces the concepts of visual vocabulary and visual prototypes. These prototypes, extracted through clustering techniques, are a collection of reference embeddings in the vision-language space containing visual features common to the object they refer to. Through extensive experiments, we have shown that it is possible to retrieve these prototypes at inference time to enhance the recognition of the proposed regions without additional learnable parameters and without fine-tuning the large-scale vision-language model.

Acknowledgments

Research partly funded by PNRR - M4C2 - Investimento 1.3, Partenariato Esteso PE00000013 - "FAIR - Future Artificial Intelligence Research" - Spoke 8 "Pervasive AI", funded by the European Commission under the NextGeneration EU programme.

References

1. Amoroso, R., Baraldi, L., Cucchiara, R.: Assessing the role of boundary-level objectives in indoor semantic segmentation. In: CAIP (2021)
2. Amoroso, R., Baraldi, L., Cucchiara, R.: Improving indoor semantic segmentation with boundary-level objectives. In: IWANN (2021)
3. Bruno, P., Amoroso, R., Cornia, M., Cascianelli, S., Baraldi, L., Cucchiara, R.: Investigating bidimensional downsampling in vision transformer models. In: ICIAP (2022)
4. Cancilla, M., Canalini, L., Bolelli, F., Allegretti, S., Carrión, S., Paredes, R., Gómez, J.A., Leo, S., Piras, M.E., Pireddu, L., et al.: The deephealth toolkit: a unified framework to boost biomedical applications. In: ICPR (2021)
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. TPAMI (2017)
6. Chen, X., Fang, H., Lin, T.Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft coco captions: Data collection and evaluation server. arXiv (2015)
7. Cheng, B., Schwing, A., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. NeurIPS (2021)
8. Cipriano, M., Allegretti, S., Bolelli, F., Di Bartolomeo, M., Pollastri, F., Pellacani, A., Minafra, P., Anesi, A., Grana, C.: Deep segmentation of the mandibular canal: a new 3d annotated dataset of cbct volumes. IEEE Access (2022)
9. Ding, J., Xue, N., Xia, G.S., Dai, D.: Decoupling zero-shot semantic segmentation. In: CVPR (2022)

10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (2021)
11. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results
12. Ghiasi, G., Gu, X., Cui, Y., Lin, T.Y.: Open-vocabulary image segmentation. In: ECCV (2022)
13. Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-vocabulary object detection via vision and language knowledge distillation. arXiv (2021)
14. Gu, Z., Zhou, S., Niu, L., Zhao, Z., Zhang, L.: Context-aware feature generation for zero-shot semantic segmentation. In: ACM Multimedia (2020)
15. Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., Schmidt, L.: Openclip. Zenodo (2021)
16. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: ICML (2021)
17. Li, B., Weinberger, K.Q., Belongie, S., Koltun, V., Ranftl, R.: Language-driven semantic segmentation. In: ICLR (2022)
18. Liang, F., Wu, B., Dai, X., Li, K., Zhao, Y., Zhang, H., Zhang, P., Vajda, P., Marculescu, D.: Open-Vocabulary Semantic Segmentation with Mask-adapted CLIP. CVPR (2023)
19. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
20. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: CVPR (2021)
21. Mottaghi, R., Chen, X., Liu, X., Cho, N.G., Lee, S.W., Fidler, S., Urtasun, R., Yuille, A.: The role of context for object detection and semantic segmentation in the wild. In: CVPR (2014)
22. Mukhoti, J., Lin, T.Y., Poursaeed, O., Wang, R., Shah, A., Torr, P.H., Lim, S.N.: Open vocabulary semantic segmentation with patch aligned contrastive learning. In: CVPR 2023 (2022)
23. Qi, L., Kuen, J., Wang, Y., Gu, J., Zhao, H., Torr, P., Lin, Z., Jia, J.: Open world entity segmentation. TPAMI (2022)
24. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
25. Schuhmann, C., et al.: LAION-5b: An open large-scale dataset for training next generation image-text models. In: NeurIPS Datasets and Benchmarks Track (2022)
26. Xian, Y., Choudhury, S., He, Y., Schiele, B., Akata, Z.: Semantic projection network for zero-and few-label semantic segmentation. In: CVPR (2019)
27. Xu, J., De Mello, S., Liu, S., Byeon, W., Breuel, T., Kautz, J., Wang, X.: Groupvit: Semantic segmentation emerges from text supervision. In: CVPR (2022)
28. Xu, M., Zhang, Z., Wei, F., Lin, Y., Cao, Y., Hu, H., Bai, X.: A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In: ECCV (2022)
29. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: CVPR (2017)
30. Zhou, C., Loy, C.C., Dai, B.: Extract free dense labels from clip. In: ECCV (2022)