Learn to Rectify the Bias of CLIP for Unsupervised Semantic Segmentation

Jingyun Wang
Beihang University
19231136@buaa.edu.cn

Guoliang Kang *
Beihang University, Zhongguancun Laboratory
kgl.prml@gmail.com

Abstract

Recent works utilize CLIP to perform the challenging unsupervised semantic segmentation task where only images without annotations are available. However, we observe that when adopting CLIP to such a pixel-level understanding task, unexpected bias occurs. Previous works don't explicitly model such bias, which largely constrains the segmentation performance. In this paper, we propose to explicitly model and rectify the bias existing in CLIP to facilitate the unsupervised semantic segmentation. Specifically, we design a learnable "Reference" prompt to encode class-preference bias and project the positional embedding of vision transformer to represent space-preference bias. Via a simple element-wise subtraction, we rectify the logits of CLIP classifier. Based on the rectified logits, we generate a segmentation mask via a Gumbel-Softmax operation. Then a contrastive loss between masked visual feature and the text features of different classes is imposed to facilitate the effective bias modeling. To further improve the segmentation, we distill the knowledge from the rectified CLIP to the advanced segmentation architecture via minimizing our designed mask-guided, feature-guided and text-guided loss terms. Extensive experiments on standard benchmarks demonstrate that our method performs favorably against previous state-of-the-arts. The implementation is available at https://github.com/dogehhh/ReCLIP.

1. Introduction

Semantic segmentation aims to attach a semantic label to each pixel of an image. Since the rising of deep learning [27, 28, 45, 54], semantic segmentation has been widely adopted in real-world applications, *e.g.*, autonomous driving, medical image segmentation, *etc*. Conventional approaches [6, 9, 10, 37, 38, 53, 60] for semantic segmentation have achieved remarkable performance. However, the superior performance of those methods relies heavily on large amounts of fully annotated masks. Collecting such high-quality pixel-level annotations can be both time-consuming and expensive, *e.g.*, some annotations for specialized tasks require massive expert knowledge, some are even inaccessible due to privacy reasons, *etc*. Therefore, it is necessary to explore unsupervised semantic segmentation where only images without annotations are available.

Unsupervised semantic segmentation (USS) has been studied for years. Many non-language-guided USS methods have been proposed, *e.g.*, clustering-based methods [11, 25, 29, 34, 42], contrastive-learning-based methods [21, 51], boundary-based methods [23], *etc*. Despite promising progress achieved, there still exhibits a large performance gap between USS and the supervised segmentation methods. Besides, these methods typically obtain class-agnostic masks and have to depend on additional processing (*e.g.*, Hungarian matching) to assign semantic labels to the masks, rendering them less practical in real scenarios.

Recently, large-scale visual-language pre-trained models, *e.g.*, CLIP [45], demonstrate superior zero-shot classification performance by comparing the degree of alignment between image feature and text features of different categories. A few CLIP-based USS approaches [22, 46, 49, 62] emerge and show remarkable performance improvement compared with the non-language-guided USS methods. These models require no access to any types of annotations, and directly assign a label to each pixel, benefiting from the aligned vision and text embedding space of CLIP. However, good alignment between image-level visual feature and textural feature doesn't necessarily mean good alignment between pixel-level visual feature and textural feature. Thus, for CLIP, unexpected bias may inevitably appear. Previous works didn't explicitly model such bias, which may largely constrain their segmentation performance.

We observe two kinds of bias existing in CLIP. From one aspect, as shown in Fig. 1(a), CLIP exhibits space-preference bias. CLIP performs apparently better for segmenting central objects than the ones distributed near the image boundary. It can be reflected by the fact that mIoU decreases as the distance between the centroids of object and the image increases. From the other aspect, as shown in Fig. 1(b), there exists class-preference bias between similar

*Corresponding author

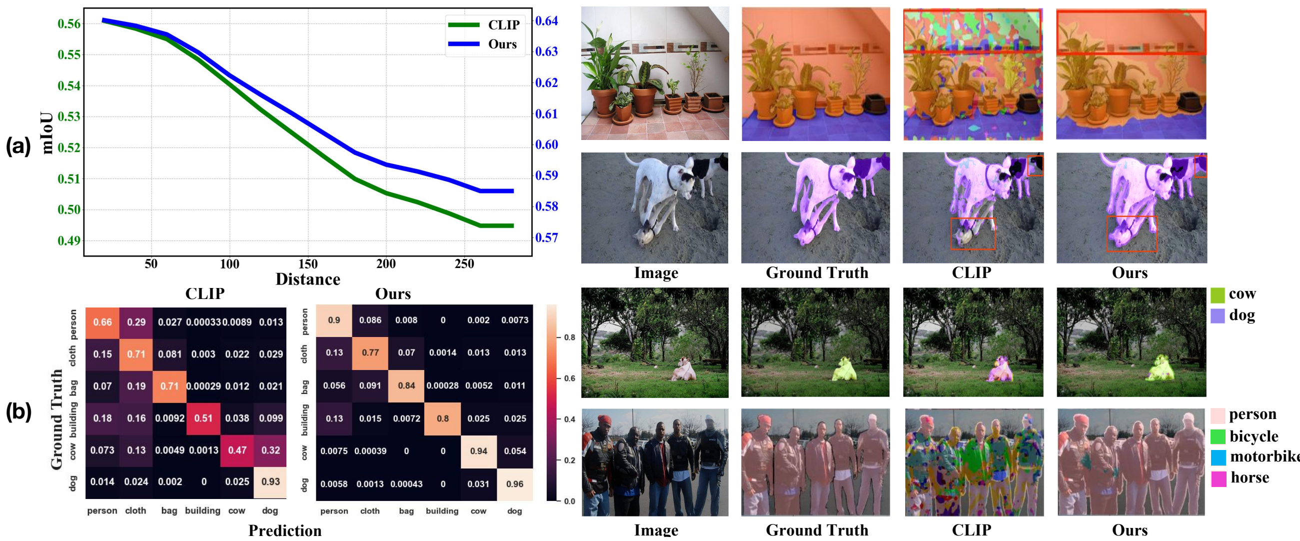


Figure 1. (a) **Space-preference bias.** (Left): The relationship between distance (x-axis) and mIoU (y-axis) is drawn on PASCAL VOC [18]. The distance means the spatial distance between the centroids of the object and the image and mIoU is computed based on predictions and ground truth. The curve shows that CLIP [45] (green) is apparently better at segmentation for central objects than boundary ones, but our method (blue) effectively mitigates this bias. More details about how we draw this figure has been shown in our supplementary material. (Right): Visualizations also show our improvement on space-preference bias qualitatively. (b) **Class-preference bias.** (Left): We randomly select 6 classes from PASCAL Context [40] and draw the confusion matrix of CLIP and our model. It shows that beside the ground truth, CLIP also prefers to assign an incorrect but relevant label to a pixel in quite a few cases, while our results show apparent improvement. (Right): The visualizations are consistent with what we observed in confusion matrix. For example, for a "cow", CLIP tends to classify it as "dog" incorrectly.

categories in CLIP. For example, according to visulization (right), when ground truth is "cow", CLIP tends to incorrectly classify it as "dog". We also show such trend between randomly selected classes by confusion matrix (left). Elements on diagonal line represent right classification, while others are false. We observe a wide range of class-preference bias introduced by CLIP.

In this paper, we propose to explicitly model and rectify the bias of CLIP to facilitate the USS. Specifically, we design two kinds of text inputs for each class, which are named as "Reference" and "Query" respectively. The Query is manually designed while the Reference contains learnable prompts. We adopt the text features of Query and Reference as classifiers to generate the Query and Reference logits respectively for each pixel of image. The Query logits represent the segmentation ability of original CLIP, while the Reference logits are expected to reflect the bias of CLIP preferring a specific class. Additionally, we project the positional embedding of CLIP's vision transformer to generate positional logits for each image. We expect the positional logits to represent space-preference bias of CLIP. Then, we remove the class-preference and the space-preference bias from original CLIP via a logit-subtraction mechanism, *i.e.,* subtracting the Reference logits and the positional logits from the Query logits. Based on the rectified logits, we generate a segmentation mask via a Gumbel-Softmax operation. Then the contrastive loss between masked visual feature and the text features of different categories is imposed to facilitate the effective bias modeling and rectification. To further improve the segmentation performance, we distill the knowledge from the rectified CLIP to the advanced segmentation architecture, via mask-guided distillation, feature-guided distillation and text-guided learning.

We conduct extensive experiments on standard semantic segmentation benchmarks, including PASCAL VOC [18], PASCAL Context [40] and ADE20K [61]. Experiment results demonstrate that our method performs favourably against previous state-of-the-arts. Notably, on PASCAL VOC, our method outperforms CLIP S4 [22] by 3.4%. Extensive ablation studies verify the effectiveness of each design in our framework.

Our contributions are summarized as follows.

- We observe that when applying CLIP to pixel-level understanding tasks, unexpected bias including space-preference bias and class-preference bias, occurs. Such bias may largely constrain the segmentation performance of CLIP-based segmentation models.
- We propose to explicitly model the class-preference and space-preference bias of CLIP via learnable Reference text inputs and projection of positional embedding. Through a simple logit-subtraction mechanism and the contrastive loss built on masked features, we effectively rectify the bias of CLIP.
- We conduct extensive experiments on segmentation

benchmarks under the unsupervised setting. Experiment results show superior performance of our method to previous state-of-the-arts.

2. Related Work

Pre-trained vision-language models. Pre-trained vision-language models (VLMs) [8, 14, 32, 33, 35] have developed rapidly with the help of large-scale image-text pairs available on the Internet. Recently, CLIP [45], ALIGN [26] and Slip [41] have made great progress on learning visual and textual representations jointly by using contrastive learning. With the image-level alignment with text, pre-trained VLMs have strong ability for zero-shot classification task and can be transferred to various downstream tasks, such as object detection [17, 54] and semantic segmentation [62].

Unsupervised semantic segmentation. While conventional approaches of semantic segmentation [37, 38, 53, 60] rely on pixel-level annotations and weakly-supervised methods [1, 16, 44, 56] still ask for image-level labels, unsupervised semantic segmentation (USS) methods [11, 20, 42, 50, 51] explores to train a segmentation model without any annotations. Models like [2, 7] adopt generative model [12] to separate foreground with background or generate corresponding masks. SegSort [23], HSG[29] and ACSeg [34] use clustering strategy, while IIC [25] uses mutual information maximization to perform unsupervised learning. MaskContrast [51] introduces contrastive learning into USS. Others like DSM [39] and LNE [13] exploit features extracted from self-supervised model, and combine it with spectral graph theory. However, the methods mentioned above either fail to segment images with multi-category objects or show a large performance gap with the supervised methods. Besides, they can only obtain class-agnostic masks and have to depend on additional strategy, such as Hungarian-matching algorithm [30], to match corresponding category with masks. Recently, pre-trained vision-language models are adopted in USS. MaskCLIP [62] modifies the image encoder of CLIP to generate patch-level features and directly performs segmentation with text features as classifiers. CLIP-py [46] performs contrastive learning between visual features from self-supervised ViT [4] and text features from CLIP. ReCo [49] performs image retrieval with CLIP and extracts class-wise embedding as classifier with co-segmentation. CLIP-S4 [22] learns pixel embeddings with pixel-segment contrastive learning and aligns such embeddings with CLIP in terms of embedding and semantic consistency. These methods can directly assign a category to each pixel, whose setting is named as language-guided unsupervised semantic segmentation. However, directly applying CLIP may bring prior bias, including space-preference bias and class-preference bias. As we know, there is no previous method trying to solve these bias and we manage to explicitly model the bias, and rectify them by element-wise subtraction.

Language-guided semantic segmentation. Recently, many works are exploring semantic segmentation guided by language under different settings. Zero-shot works [3, 31, 43, 55] split classes into seen and unseen set. During the training period, only masks of seen classes are provided. For inference, models are tested on both seen and unseen classes, but the test data is still in the same distribution with the training data. Open-vocabulary works [5, 19, 36, 47, 57–59] are trained in one scenario with extra annotations including class-agnostic masks or image captions, but are used for predicating segmentation masks of novel classes in other scenarios. From the technical view, our method also falls into the category of language-guided semantic segmentation. However, we consider the unsupervised setting. In this setting, we have access to images without any annotations during training. The training and inference images are sampled from the same distributions and the same set of categories. Such a setting is different to the typical zero-shot or open-vocabulary setting.

3. Method

Background In this work, we aim to rectify the bias of CLIP for unsupervised semantic segmentation. In USS, we only have access to images without any types of annotations to train the segmentation model. For training and inference, the same set of categories are considered and the data distributions are assumed to be the same.

Overview The general framework of our method is illustrated in Fig. 2. We aim to rectify the bias of CLIP including the class-preference bias and the space-preference bias, to facilitate unsupervised semantic segmentation. From a high level, class-preference bias reflects the shift of CLIP predictions towards specific classes, while space-preference bias reflects the shift of CLIP predictions towards specific spatial locations. Both biases will be finally reflected in the predicted logit maps. A reasonable way to rectify the bias is to subtract its logit maps from the normal logit maps predicted via original CLIP.

To realize our goal, we firstly forward the image $I \in \mathbb{R}^{3\times H\times W}$ through the image encoder of CLIP to obtain the patch-level image features Z. We design two kinds of text inputs for each class. One is the manually designed Query text input Q and the other one is the Reference text input R which consists of a learnable prompt and the name of class. For each class, passing Q and R through the text encoder of CLIP, we obtain two text embeddings W_q and W_r, which serve as the weights of query segmentation head and reference segmentation head respectively. Taking the same visual feature Z as input, the query and reference segmentation heads output a query logit map M_q (Sec. 3.1) and Reference logit map M_r (Sec. 3.2) respectively. Meanwhile, positional embedding p is sent into a learnable convolutional network to generate positional logit map M_p (Sec. 3.3). Then the bias logit map M_b can be obtained by adding Reference logit map M_r and positional logit map M_p. We then subtract M_b from M_q to generate the rectified logits M. Based on the rectified logits, we generate masks for different classes. Then a contrastive loss is imposed between the masked visual features and the text features of different categories to encourage the bias modeling and rectification (Sec. 3.4). In order to enhance the segmentation performance, we distill the knowledge of rectified CLIP to the advanced segmentation architecture with specifically designed mask-guided, feature-guided and text-guided loss terms (Sec. 3.5). In both the rectification and distillation stages, we keep CLIP frozen.

3.1. Baseline: Directly Segment with CLIP

Following MaskCLIP [62], we adapt pre-trained CLIP [45] (ViT-B/16) to the semantic segmentation task. We remove the query and key embedding layers of last attention but reformulate the value embedding layer and the last linear projection layer into two respective 1×1 convolutional layers. Therefore, the image encoder can not only generate local features for dense predictions, but also keep the visual-language association in CLIP by freezing its pre-trained weights. We forward image I through the image encoder and obtain patch-level features $Z \in \mathbb{R}^{n \times D}$ (n is the number of patches and D is the dimension of features in CLIP).

Each text in Query $Q = \{Q_1, Q_2, \cdots, Q_C\}$ (C is the number of classes) is an ensemble of several manually designed prompts, *e.g.*, "a good/large/bad photo of a $[CLS]$" where $[CLS]$ denotes the a specific class name. Passing Q through the text encoder, we obtain its text embeddings W_q. We treat text embeddings W_q as the weight of segmentation head to perform 1×1 convolution. By sending features Z to the segmentation head, we get a Query logit map $M_q \in \mathbb{R}^{n \times C}$. Then the segmentation mask can be predicted by $\arg\max$ operation on M_q.

3.2. Learn Class-Preference Bias

In order to explicitly model the class-preference bias brought by pre-trained CLIP for specific datasets, we design a Reference R_i, $i = \{1, 2, \cdots, C\}$, as additional text input for each class. Inspired by CoOp [63], R_i consists of a learnable prompt which is shared across all the classes for efficiency, and a class name $[CLS]$, which can be formed as

$$R_i = [v_1][v_2]...[v_l]...[v_L][CLS], \tag{1}$$

where each $[v_l] (l \in \{1, ..., L\})$ is a vector with dimension D and serves as a word embedding. The L is a constant representing the number of word embeddings to learn. Totally, there are 77 word embeddings for a text of CLIP. As two word embeddings are used for class name and two for indicating start and end of a text, we set L to 73.

Passing the Reference R through text encoder, we obtain reference text embedding $W_r \in \mathbb{R}^{C \times D}$. The text embedding W_r is then directly utilized as the weights of segmentation head to perform 1×1 convolution on visual feature Z and output a Reference logit map $M_r \in \mathbb{R}^{n \times C}$.

As we obtain different Reference logit maps for different classes, we expect the Reference logit map to encode the class-preference bias to facilitate the following bias rectification process. It is worth noting that we make Reference R learnable but keep Query Q fixed. It is because when we make them both learnable, Query may also capture some class-preference bias and technically we cannot guarantee which text embedding should encode the class-preference bias, resulting in an implicit bias modeling. Thus, in our framework, we choose not to make the Q learnable to just encourage R encode the bias. Such a design cooperates with the following logit subtraction mechanism to make the bias modeling and rectification more meaningful and effective.

3.3. Learn Space-Preference Bias

In ViT [15], positional embeddings (PE) are important for encoding spatial information to features. Thus, we assume the space-preference bias should depend on the positional embeddings (PE) and choose to learn a projection of PE to represent the space-preference bias.

We design a 3-layer 3×3 convolutional network, and each convolutional layer is followed by a batch normalization. Specifically, we project PE p by the designed convolutional network to obtain positional logits $M_p \in \mathbb{R}^{n \times 1}$. During the training process, the projection network is optimized to model the space-preference bias in positional logits M_p. Since PE is shared across all categories, the learned space-preference bias is also shared across all the categories.

3.4. Rectify Bias with Contrastive Learning Loss

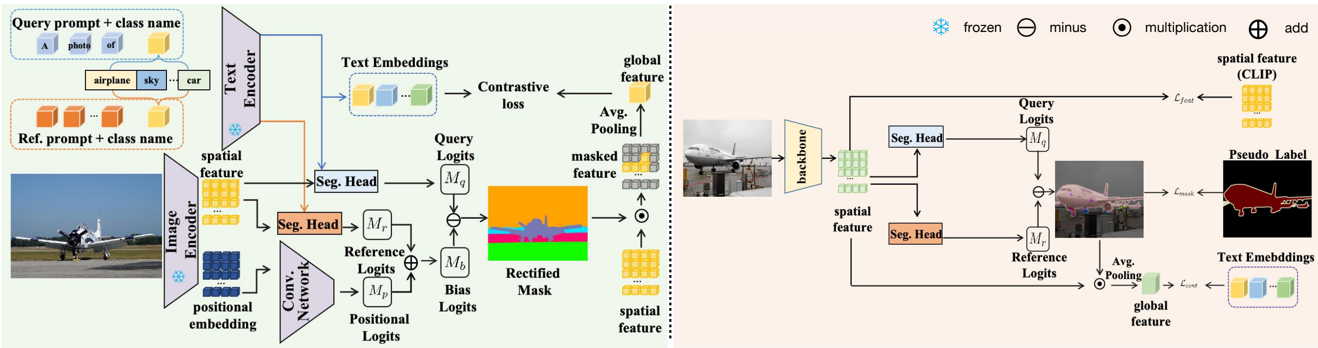
By keeping Query prompt Q fixed, the Query logits M_q represents the natural prediction ability of CLIP model, which may contain class-preference bias and space-preference bias. With learnable Reference prompt and projection of PE, we explicitly encode the class-preference bias and space-preference bias into the Reference logit map M_r and positional logit map M_p respectively. We then add M_r and M_p together to depict the final bias M_b, *i.e.*,

$$M_b = M_r + M_p^*, \tag{2}$$

where we simply expand M_p to $M_p^* \in \mathbb{R}^{n \times C}$ which means we repeat each M_p for C times.

Then we perform a simple element-wise subtraction between M_q and M_b to generate the rectified logit map M,

$$M = M_q - M_b. \tag{3}$$



(a) Rectification Stage

(b) Distillation

Figure 2. **Method overview.** We propose a new framework for language-guided unsupervised semantic segmentation. **(a) Rectification Stage**: At this stage, we aim to rectify the class-preference and space-preference bias of CLIP. **(b) Distillation**: We distill knowledge from the rectified CLIP to the advanced segmentation architecture with the mask-guided, feature-guided and text-guided loss terms.

From a high-level, this operation can be interpreted as subtracting the "bias" from the predictions with text features of Query as segmentation head.

Since the $\arg\max$ operation is not differentiable, we utilize a Gumbel-Softmax trick [24] to generate the candidate masks for each class with τ_1 as the temperature, *i.e.,*

$$\hat{M} = \textbf{Gumbel-Softmax}(M, \tau_1). \tag{4}$$

We apply $\hat{M}$ to the feature Z to get the masked features Z_g, which are expected to encode the features of objects with different classes. We then design a contrastive loss to supervise the learning via aligning masked features with text features of different classes. As the masked features represent the objects of interest without context, we utilize W_q generated by text input Q for alignment.

Additionally, as it is usually impossible for an image to contain all the classes of interest, we need to infer what classes exist in the current image. Our strategy is as follows. Firstly, we choose to extract global visual feature from image encoder of CLIP. Then we calculate the similarity scores between the text input of a specific class and the global visual feature of the image. We only choose the classes whose similarity scores are higher than a threshold t as the potential classes exist in the current image.

Then we may select K pseudo labels $\{c_1, c_2, \cdots, c_K\}$ for each image. We perform a global average pooling of Z_g and compute the similarities between pooled visual features of pseudo labels and text features of all the categories. The contrastive loss can be defined as

$$\mathcal{L} = -\frac{1}{K}\sum_{k=1}^{K} \log \frac{\exp\{S_{c_k,c_k}/\tau\}}{\sum_{j=1}^{C}\exp\{S_{c_k,j}/\tau\}} \tag{5}$$

where $S_{c_k,j}$ denotes the similarity between visual feature of pseudo label c_k and text feature of j-th ($j \in \{1, 2, \cdots, C\}$ category and the τ is a constant.

A better modeling of bias yields more accurate estimations of object masks. Then the masked features of objects are more aligned with corresponding text features. As a result, the contrastive loss (Eq. (5))will be lower. In contrast, a worse modeling of bias results in higher contrastive loss. Thus, minimizing Eq. (5) will drive the model to update towards making more accurate mask predictions, *i.e.*, rectifying the bias of CLIP when adapted to the downstream USS task.

3.5. Distillation for Enhanced Results

As CLIP is not specifically designed for segmentation tasks, we choose to distill the knowledge from the rectified CLIP (teacher) to the advanced segmentation architecture (student) to enhance the feature representations and finally improve the segmentation performance. In our paper, we choose DeepLab V2 [6] as the student network which directly inherits and fixes the Query and Reference segmentation heads, and utilizes logit subtraction mechanism to generate the final segmentation masks. In our design, the rectified CLIP works as a teacher to guide the feature learning of DeepLab V2 with three designed loss terms.

Mask-guided loss. We directly exploit segmentation masks generated from the bias rectification stage as pseudo labels. We compute a cross-entropy loss between the pseudo labels $\tilde{M}$ and predictions M^D from DeepLab V2, *i.e.,*

$$\mathcal{L}_{mask} = -\frac{1}{HW}\sum_{i}^{H}\sum_{j}^{W} \log P_{ij}(\tilde{M}_{ij}) \tag{6}$$

where H and W denote the height and width of an image. The $P_{ij}(\tilde{M}_{ij}) = \text{Softmax}(M_{ij}^D)_{\tilde{M}_{ij}}$, which means the predicted probability with respect to the pseudo label $\tilde{M}_{ij}$.

Feature-guided loss. After obtaining features Z from visual backbone of rectified CLIP and $\hat{Z}$ from student visual backbone, we resize two features to the same shape of original image by bilinear interpolation. We then perform a L1 loss between two features

$$\mathcal{L}_{feat} = ||Z - \hat{Z}||_1 \tag{7}$$

Therefore, visual features from student network is aligned with visual and textual features of rectified CLIP, providing an important basis for the following text-guided loss.

Text-guided loss. We adopt the same strategy as the Rectification stage (Sec. 3.4) to compute contrastive loss as our text-guided loss $\mathcal{L}_{text}$. The only difference is that we use feature $\hat{Z}$ and masks generated by the student network to obtain the corresponding masked features.

Total distillation loss. Total distillation loss $\mathcal{L}_{distill}$ is calculated as follows, where α and β are both constants and set to 0.5. Effect of each loss term is studied in Sec. 4.3.

$$\mathcal{L}_{distill} = \mathcal{L}_{mask} + \alpha\mathcal{L}_{feat} + \beta\mathcal{L}_{text}. \tag{8}$$

4. Experiments

4.1. Setup

Datasets. We conduct experiments on three standard benchmarks for semantic segmentation, including PASCAL VOC 2012 [18], PASCAL Context [40] and ADE20K [61]. PASCAL VOC 2012 (1,464/1,449 train/validation) contains 20 object classes, while PASCAL Context (4,998/5,105 train/validation) is an extension of PASCAL VOC 2010 and we consider 59 most common classes in our experiments. ADE20K (20,210/2,000 train/validation) is a segmentation dataset with various scenes and 150 most common categories are considered.

Implementation details. For the image encoder of CLIP, we adopt ViT-B/16 as visual backbone. For the text encoder of CLIP, we adopt Transformer [52]. During the whole training period, we keep both of the encoders frozen. We use conventional data augmentations including random cropping and random flipping. Relevant hyper-parameters for each dataset, including number of rectification epochs, number of distillation iterations and parameters of data augmentation are shown in our supplementary material. For both stages, we use a SGD [48] optimizer with a learning rate of 0.01 and a weight decay of 0.0005. We adopt the poly strategy with the power of 0.9 for learning ratte. In our experiment, we report the mean intersection over union (mIoU) as evaluation metric.

Baselines. We compare with previous USS methods to verify the superiority of our method, including MaskCLIP(+) [62], CLIP-S4 [22], ReCo [49], CLIPpy [46], *etc.* From the technical view, although our method falls into the category of text-guided segmentation, we choose not to compare with text-guided methods which are designed for zero-shot or open-vocabulary setting (*e.g.,* GroupViT [57], TCL [5] and ViewCo [47]). Two main differences exist between the zero-shot/open-vocabulary setting and the unsupervised setting: 1) in zero-shot/open-vocabulary, the category sets for training and inference are non-overlapped, but in USS the category sets are shared; 2) text-guided zero-shot/open-vocabulary methods usually rely on large-scale image-text pairs or class-agnostic masks to supervise the training, but in USS we only utilize the images without any types of annotations to train the model.

Table 1. **Comparison with non-language-guided unsupervised semantic segmentation methods on PASCAL VOC.**

Model	LC	mIoU
IIC [25]	✓	9.8
SegSort [23]	✓	11.7
Deep Spectral Methods [39]	✗	37.2
HSG [29]	✗	41.9
ACSeg [34]	✗	47.1
MaskContrast [51]	✓	49.6
Ours (rectification)	✗	**58.5**
Ours (distill)	✗	**75.4**
Ours (distill)	✓	**76.1**

4.2. Comparison with SOTA methods

We compare our method with both previous non-language-guided USS methods (Table 1) and CLIP-based USS methods (Table 2). In all tables, we use “Ours (rectification)” to denote the segmentation results after the rectification stage and “Ours (distill)” to denote the segmentation results after distillation. Unless otherwise stated, results from previous methods are directly cited from the original papers.

As shown in Table 1, our model shows remarkably better segmentation results compared with the conventional non-language-guided USS methods. In order to further evaluate the strength of features extracted from our distillation model, we also report our results of linear classification (LC). Following MaskContrast [51], we fix our model to generate pixel embeddings and train a linear classifier to generate semantic segmentation masks. The results of LC also prove that our method extracts strong features and achieves significant gains.

In Table 2, we also make a comparison with typical CLIP-based methods, including MaskCLIP, CLIPpy, ReCo and CLIP-S4. These methods all consider the same CLIP-based USS setting with ours. The results show our method performs favorably against previous CLIP-based methods. For example, after the rectification stage, our method outperforms MaskCLIP by 9.0%, 4.1% and 1.6% respectively on the three datasets, while after distillation, our method outperforms MaskCLIP+ by 5.4%, 2.7% and 2.1%. Our method also shows better results than CLIPpy by 20.8% and 0.8% on PASCAL VOC and ADE20K and outperforms CLIP-S4 by 3.4% and 0.2% on PASCAL VOC and PASCAL Context. Since ReCo employs a “context elimination” (CE) trick which may introduce prior knowledge, we also report the results of ReCo by removing this trick (ReCo w/o CE in Table 2). Our method outperforms ReCo obviously, *e.g.*, on PASCAL VOC, ours (rectification) outper-

Table 2. **Comparison with CLIP-based unsuperivsed semantic segmentation methods on various benchmarks.**

Method	VOC	Context	ADE
CLIPpy [46]	54.6	/	13.5
MaskCLIP [62]	49.5	21.7	9.5
MaskCLIP+ [62]	70.0	31.1	12.2
ReCo [49]	55.2	26.2	/
ReCo (w/o CE)	54.8	23.1	/
CLIP-S4 [22]	72.0	33.6	/
Ours (rectification)	**58.5**	**25.8**	**11.1**
Ours (distill)	**75.4**	**33.8**	**14.3**

Table 3. **Ablation on whether Query should be learnable.** Results show that fixing Query performs favorably against making Query learnable.

Query	Reference	VOC	Context	ADE
✓	✓	56.7	23.7	9.4
✗	✓	**56.7**	**24.4**	**10.5**

forms ReCo and ReCo (w/o CE) by 3.3% and 3.7% respectively.

4.3. Ablation study

Should Query be learnable? We study on whether the prompt of Query should be learnable. As shown in Table 3, we conduct experiments with learnable and fixed Query respectively. As we aim to show the effect of fixing Query, we don't utilize the projected positional logits to calculate the numbers shown in the table. The numbers show that fixing Query performs favorably against making Query learnable. This verifies that we should only learn Reference but keep Query fixed as we discuss in Sec. 3.2.

Effect of different bias modeling. As shown in Table 4, compared with the baseline without any bias rectification (numbers with only "Query"), utilizing learbable Reference to model the class-preference bias brings remarkable performance improvement, *e.g.,* 7.2% mIoU on PASCAL VOC, 2.7% mIoU on PASCAL Context and 1.0% mIoU on ADE20K. While introducing projection of positional embedding to model the space-preference bias, the numbers are further improved, *i.e.,* around 1% better than modeling the class-preference bias only. Those results verify the effectiveness of bias modeling by introducing learnable Reference and projection of positional embedding.

Effect of element-wise subtraction on bias rectification. In order to validate the effect of our element-wise subtraction, we conduct experiments in Table 5. We compare our subtraction mechanism with an alternative solution, *i.e.,* instead of subtracting logits encoding bias (*i.e.,* Reference logits plus positional logits), we add all the logits. Comparisons shown in the table verify the effectiveness of our

Table 4. **Effect of bias modeling.** Results show effect of both bias modeling quantitatively and each contributes to better results.

Query	Reference	PE	VOC	Context	ADE
✓			49.5	21.7	9.5
✓	✓		56.7	24.4	10.5
✓	✓	✓	**58.5**	**25.8**	**11.1**

Table 5. **Effectiveness of element-wise subtraction.** Results show that the element-wise subtraction removes bias from CLIP.

Query	Ref+PE	sub	add	VOC	Context
✓	✗	✗	✗	49.5	21.7
✓	✓	✗	✓	57.8	24.9
✓	✓	✓	✗	**58.5**	**25.8**

subtraction way. As we aim to remove the bias from the original CLIP, we speculate that the subtraction operation may work as a strong prior which regularizes the training to facilitate meaningful and effective bias modeling. The effect of bias rectification by element-wise subtraction is further visualized in Fig. 3.

Visualization of Bias Modeling. In Fig. 3 (b), we illustrate how we explicitly model and rectify the class-preference bias. As shown in the first column of Fig. 3 (b), the original CLIP (see "Query" mask) tends to misclassify part of a "person" (see the ground-truth mask denoted as "GT") into a "boat". Such a mistake is reflected in the comparison between the logit heatmaps for the "person" channel and the "boat" channel: in the area misclassified as "boat", the logits for "boat" are relatively higher than those for "person". In contrast, for Reference logit map, the person-channel logits are quite low, while the boat-channel logits are generally very high, especially for the misclassified area. Consequently, via the proposed logit subtraction operation, we obtain the rectified logits (see the last column), where the boat channel is largely suppressed. Finally, we obtain a much better mask (see "Ours" in the first column).

In Fig. 3 (c), we aim to show that the projection of positional embedding (PE) effectively models the space-preference bias. By comparing the results with and without rectifying logits projected by PE (see the dashed boxes of the last two columns), we find that rectification with PE largely improves the segmentation performance in the boundary areas. The results illustrate that the projection of PE does encode the space-preference bias and rectifying such bias may largely improve the segmentation results.

Effect of different distillation loss terms. We conduct experiments on PASCAL VOC to validate the effect of each loss term of our distillation framework, including the mask-guided, feature-guided and text-guided loss terms. From Table 7, all the loss terms contribute to better performance.

Effect of segmentation head for distillation. As illustrated

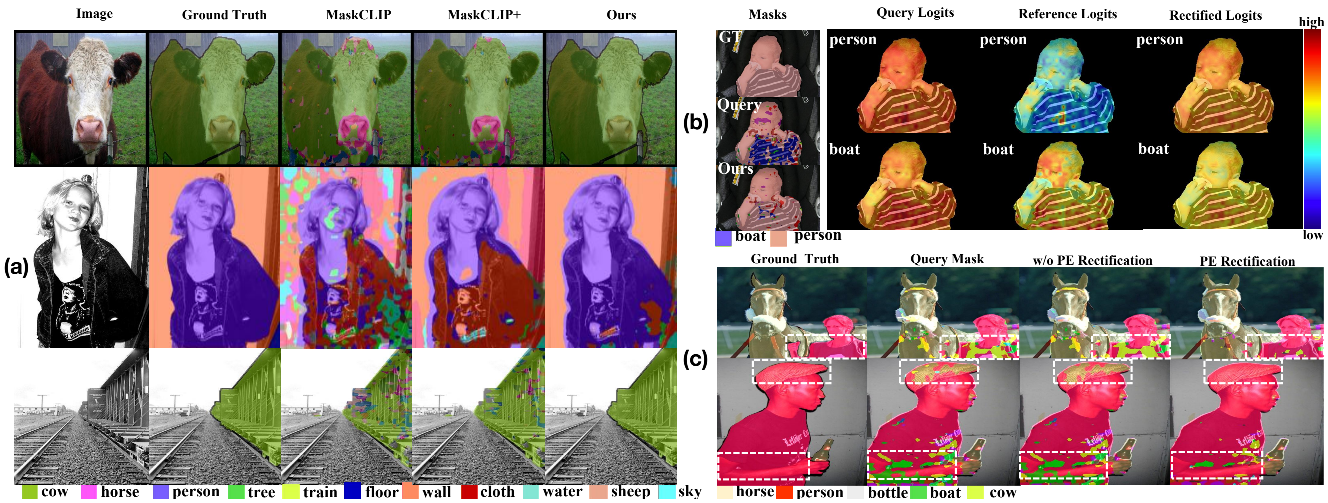


Figure 3. **(a) Qualitative Results**: We visualize segmentation results on PASCAL Context. From the visualization, we observe that our model outperforms MaskCLIP(+) obviously by rectifying both class-preference bias and space-preference bias. **(b) Class-preference bias**: In order to explain how Reference explicitly models the class-preference bias, we show the heatmap of Reference logits. **(c) Space-preference bias**: The segmentation within dashed boxes shows the effectiveness of PE projection on rectifying space-preference bias.

Table 6. **Effect of segmentation head for distillation.** We perform distillation with both original classification head (Ori. Head) and structure from our rectification stage

Ori. Head	Query	Reference	PE Proj.	mIoU
✓	✗	✗	✗	73.8
✗	✓	✗	✗	74.1
✗	✓	✓	✗	**75.4**
✗	✓	✓	✓	75.0

Table 7. **Ablation results on distillation loss.** According to the results, each loss item contributes to better distillation result.

Mask	Feature	Text	mIoU
✓			73.0
✓	✓		74.1
✓	✓	✓	**75.4**

in Fig 2 (b), instead of directly using original classification head at the distillation stage, the student network (*i.e.,* DeepLab V2 [6]) inherits and fixes the Query and Reference segmentation heads from our rectification stage, and utilizes logit subtraction mechanism to generate the final segmentation masks. In Table 6, we conduct experiments with original head ("Ori. Head") of DeepLab V2 and other types of segmentation heads which are directly inherited from our rectification stage on PASCAL VOC. From the results, we observe that it is better to inherit the components from our rectification stage than adopting the original classification head. However, introducing projection of PE in segmentation head cannot bring further improvement.

Qualitative Results. We visualize our segmentation results in Fig. 3 (a). It can be observed that there exists apparent space-preference bias and class-preference bias in the segmentation results of original CLIP (MaskCLIP) and these bias still cannot be removed even after distillation (MaskCLIP+). However, our model outperforms MaskCLIP(+) obviously by rectifying the bias of CLIP for unsupervised semantic segmentation.

5. Conclusion

In this paper, we propose a new framework for language-guided unsupervised semantic segmentation. We observe bias, including space-preference bias and class-preference bias, exists in CLIP when directly applying CLIP to segmentation task. We propose using additional Reference to learn class-preference bias and projecting positional embedding to represent space-preference bias, and then manage to rectify them by a simple element-wise logit subtraction mechanism. For further improving the segmentation performance, we distill the knowledge from rectified CLIP to advanced segmentation backbone with specifically designed losses. Extensive experiments demonstrate that our method achieve superior segmentation performance compared to previous state-of-the-arts. We hope our work may inspire future research to investigate how to better adapt CLIP to complex visual understanding tasks.

Acknowledgement This research project is supported by National Natural Science Foundation (NNSF) of China under Grant 92370114 and National Key R&D Program of China under Grant 2022YFB3104900.

References

[1] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4981–4990, 2018. 3

[2] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[3] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 3

[5] Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11165–11174, 2023. 3, 6, 1

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018. 1, 5, 8

[7] Mickaël Chen, Thierry Artières, and Ludovic Denoyer. Unsupervised object segmentation by redrawing. *Advances in neural information processing systems*, 32, 2019. 3

[8] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European Conference on Computer Vision*, pages 104–120, 2020. 3

[9] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 1

[10] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1290–1299, 2022. 1

[11] Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16794–16804, 2021. 1, 3

[12] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018. 3

[13] Zhijie Deng and Yucen Luo. Learning neural eigenfunctions for unsupervised semantic segmentation. *arXiv preprint arXiv:2304.02841*, 2023. 3

[14] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11162–11173, 2021. 3

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, and S. Gelly. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 4

[16] Ye Du, Zehua Fu, Qingjie Liu, and Yunhong Wang. Weakly supervised semantic segmentation by pixel-to-prototype contrast. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4320–4329, 2022. 3

[17] Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. Learning to prompt for open-vocabulary object detection with vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14084–14093, 2022. 3

[18] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010. 2, 6

[19] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI*, pages 540–557. Springer, 2022. 3

[20] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. 3

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 1

[22] Wenbin He, Suphanut Jamonnak, Liang Gou, and Liu Ren. Clip-s4: Language-guided self-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11207–11216, 2023. 1, 2, 3, 6, 7

[23] Jyh-Jing Hwang, Stella X Yu, Jianbo Shi, Maxwell D Collins, Tien-Ju Yang, Xiao Zhang, and Liang-Chieh Chen. Segsort: Segmentation by discriminative sorting of segments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7334–7344, 2019. 1, 3, 6

[24] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017. 5

[25] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9865–9874, 2019. 1, 3, 6

[26] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. 3

[27] Guoliang Kang, Lu Jiang, Yunchao Wei, Yi Yang, and Alexander Hauptmann. Contrastive adaptation network for single-and multi-source domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 44(4): 1793–1804, 2020. 1

[28] Guoliang Kang, Yunchao Wei, Yi Yang, Yueting Zhuang, and Alexander Hauptmann. Pixel-level cycle association: A new perspective for domain adaptive semantic segmentation. *Advances in neural information processing systems*, 33: 3569–3580, 2020. 1

[29] Tsung-Wei Ke, Jyh-Jing Hwang, Yunhui Guo, Xudong Wang, and Stella X Yu. Unsupervised hierarchical semantic segmentation with multiview cosegmentation and clustering transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2571–2581, 2022. 1, 3, 6

[30] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 3

[31] Boyi Li, Kilian Q. Weinberger, Serge J. Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, 2022. 3

[32] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11336–11344, 2020. 3

[33] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 3

[34] Kehan Li, Zhennan Wang, Zesen Cheng, Runyi Yu, Yian Zhao, Guoli Song, Chang Liu, Li Yuan, and Jie Chen. Acseg: Adaptive conceptualization for unsupervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7162–7172, 2023. 1, 3, 6

[35] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2046–2065, 2020. 3

[36] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7061–7070, 2023. 3

[37] Chen Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, et al. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *International Conference on Learning Representations*, 2015. 1, 3

[38] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 3

[39] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Deep spectral methods: A surprisingly strong baseline for unsupervised semantic segmentation and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8364–8375, 2022. 3, 6

[40] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 891–898, 2014. 2, 6

[41] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pretraining. In *European Conference on Computer Vision*, pages 529–544. Springer, 2022. 3

[42] Yassine Ouali, Céline Hudelot, and Myriam Tami. Autoregressive unsupervised image segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 142–158. Springer, 2020. 1, 3

[43] Giuseppe Pastore, Fabio Cermelli, Yongqin Xian, Massimiliano Mancini, Zeynep Akata, and Barbara Caputo. A closer look at self-training for zero-label semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2693–2702, 2021. 3

[44] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1713–1721, 2015. 3

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 4

[46] Kanchana Ranasinghe, Brandon McKinzie, Sachin Ravi, Yinfei Yang, Alexander Toshev, and Jonathon Shlens. Perceptual grouping in vision-language models. *arXiv preprint arXiv:2210.09996*, 2022. 1, 3, 6, 7

[47] 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. *arXiv preprint arXiv:2302.10307*, 2023. 3, 6

[48] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951. 6

[49] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. In *Advances in Neural Information Processing Systems*. 1, 3, 6, 7

[50] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X*, pages 268–285. Springer, 2020. 3

[51] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10052–10062, 2021. 1, 3, 6

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 6

[53] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Eca-net: Efficient channel attention for deep convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11534–11542, 2020. 1, 3

[54] Zhenyu Wang, Yali Li, Xi Chen, Ser-Nam Lim, Antonio Torralba, Hengshuang Zhao, and Shengjin Wang. Detecting everything in the open world: Towards universal object detection. *arXiv preprint arXiv:2303.11749*, 2023. 1, 3

[55] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8256–8265, 2019. 3

[56] Jinheng Xie, Jianfeng Xiang, Junliang Chen, Xianxu Hou, Xiaodong Zhao, and Linlin Shen. C2am: contrastive learning of class-agnostic activation map for weakly supervised object localization and semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 989–998, 2022. 3

[57] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18134–18144, 2022. 3, 6, 1

[58] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Yi Wang, Yu Qiao, and Weidi Xie. Learning open-vocabulary semantic segmentation models from natural language supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2935–2944, 2023.

[59] Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. Side adapter network for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2945–2954, 2023. 3

[60] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 1, 3

[61] B. Zhou, Z. Hang, Francesco Xavier Puig Fernandez, S. Fidler, and A. Torralba. Scene parsing through ade20k dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 6

[62] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, pages 696–712. Springer, 2022. 1, 3, 4, 6, 7

[63] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 4