Learning Open-vocabulary Semantic Segmentation Models From Natural Language Supervision

Jilan Xu[1,2] Junlin Hou[1] Yuejie Zhang[1*] Rui Feng[1*] Yi Wang[2] Yu Qiao[2] Weidi Xie[2,3]
[1]School of Computer Science, Shanghai Key Lab of Intelligent Information Processing,
Shanghai Collaborative Innovation Center of Intelligent Visual Computing, Fudan University
[2]Shanghai AI Laboratory [3]CMIC, Shanghai Jiao Tong University

Abstract

This paper considers the problem of open-vocabulary semantic segmentation (OVS), that aims to segment objects of arbitrary classes beyond a pre-defined, closed-set categories. The main contributions are as follows: ***First****, we propose a transformer-based model for OVS, termed as OVSegmentor, which only exploits web-crawled image-text pairs for pre-training without using any mask annotations. OVSegmentor assembles the image pixels into a set of learnable group tokens via a slot-attention based binding module, then aligns the group tokens to corresponding caption embeddings.* ***Second****, we propose two proxy tasks for training, namely masked entity completion and cross-image mask consistency. The former aims to infer all masked entities in the caption given group tokens, that enables the model to learn fine-grained alignment between visual groups and text entities. The latter enforces consistent mask predictions between images that contain shared entities, encouraging the model to learn visual invariance.* ***Third****, we construct CC4M dataset for pre-training by filtering CC12M with frequently appeared entities, which significantly improves training efficiency.* ***Fourth****, we perform zero-shot transfer on four benchmark datasets, PASCAL VOC, PASCAL Context, COCO Object, and ADE20K. OVSegmentor achieves superior results over state-of-the-art approaches on PASCAL VOC using only 3% data (4M vs 134M) for pre-training.*

1. Introduction

Semantic segmentation considers the problem of assigning class labels to each pixel in the image. It plays critical roles in a wide range of real-world scenarios, including autonomous driving, computer-aided diagnosis and satellite image analysis, to name a few. Generally speaking, two lines of research dominate semantic segmentation, one way is to cluster the pixels into different groups and assign a semantic label to each group; the other idea treats segmentation as pixel-wise classification, casting each pixel into one category. Despite tremendous progress, the scalability of existing approaches that rely on supervised training has been fundamentally limited: (1) costly annotation procedure. Extensive manual pixel-wise annotations are required for training segmentation models; (2) closed-set segmentation. The model is restricted to segmenting objects from a closed-set of categories. Whenever a new dataset comes, the model requires re-training.

*Corresponding author

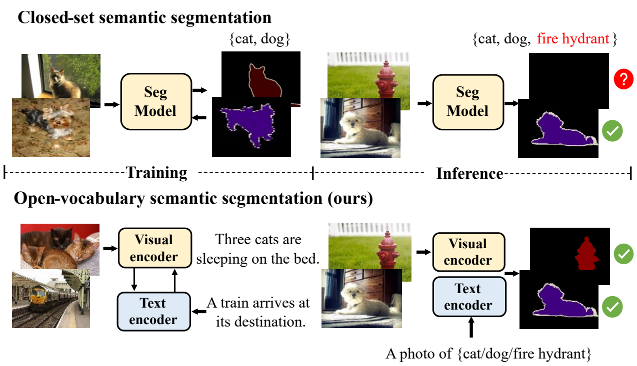

Figure 1. **An illustration of open-vocabulary semantic segmentation.** Models trained on closed-set classes (cat and dog) fail to segment novel class (fire hydrant). We train a visual encoder and a text encoder on web-collected image-text pairs without using any mask labels, and our model can segment arbitrary object classes.

In this paper, our goal is to train an open-vocabulary semantic segmentation (OVS) model, by exploiting freely available image-caption pairs on Internet, as illustrated in Fig. 1. The recent work, for example, CLIP [41] and ALIGN [23] have demonstrated that a combination of large-scale image-caption pairs and simple noise contrastive estimation can learn powerful image-text embeddings from scratch, and show strong "zero-shot" generalization abilities for open-vocabulary classification. Additionally, GroupViT [53] extends the idea towards semantic segmentation by training a segmentation model with text supervision only. They perform hierarchical grouping of visual tokens, which are then aligned to the corresponding text embeddings via a contrastive loss. However, the following issues remain challenging and unsolved: *First*, the captions only provide coarse, image-level descriptions, which are insufficient for training semantic segmentation models where fine-grained, pixel-wise supervision is usually needed. *Second*, the diversity of web-collected data is large, that requires the model to learn visual invariance on objects of interest, with only weak supervision provided. For instance, the visual appearance of two images with similar captions can be drastically different.

To tackle the above challenges, (i) we propose a transformer-based model for open-vocabulary semantic segmentation, dubbed as OVSegmentor, that can segment objects of arbitrary categories via zero-shot transfer, with only image-caption pairs for pre-training. Specifically, we introduce learnable group tokens to cluster image patches via a slot-attention [33] based binding module, and align the group tokens with corresponding caption embeddings. Note that our model neither requires ground-truth masks for training nor additional re-training on target segmentation datasets, substantially alleviating the annotation efforts and improving transfer efficiency; (ii) As for training on the image-caption dataset, we propose two proxy tasks, namely masked entity completion and cross-image mask consistency, the former trains the model to infer *all* the masked entities in the sentence given the group tokens, and the latter enforces consistent mask prediction for images with the common entity. Both tasks have shown to be beneficial in learning entity-specific, fine-grained and visually invariant group semantics; (iii) We construct an image-caption dataset, termed as CC4M, by designing an automatic approach to filter CC12M [7] with frequently appeared visual entities, significantly improving the training efficiency.

We pre-train the proposed OVSegmentor on our filtered image-caption dataset (CC4M), without using any manual segmentation masks whatsoever. The model is evaluated on four segmentation benchmarks, PASCAL VOC 2012 [18], PASCAL Context [38], COCO Object [31], and ADE20K [59] in a zero-shot manner, *i.e.*, the model is directly evaluated on target datasets without any finetuning. Extensive experiments demonstrate that our model surpasses existing models that are trained with full supervision and outperforms state-of-the-art self-supervised approaches on PASCAL VOC by using only 3% data (4M vs 134M) for pre-training, significantly improving the training efficiency.

2. Related Work

Vision-Language Pre-training. Vision-language pre-training (VLP) aims to learn joint visual-textual representations for a variety of multimodal downstream tasks. Existing works either learn unimodal encoders by distinguishing the positive pair(s) from the unpaired samples [3,28,41] or focus on one multimodal encoder for joint feature learning with masked image/language modeling and image-text matching losses [12,27,29]. Additionally, some approaches seek fine-grained supervision for cross-modal interaction [20, 30, 55–57]. For example, GLIP [30] proposed to align the bounding boxes with corresponding phrases in the text. However, they still rely on ground-truth grounding annotations. In contrast, our work explores fine-grained information with only weak supervision provided. Despite remarkable performance on multimodal downstream tasks, few of these vision-language models have been designed for fundamental vision tasks (*e.g.*, semantic segmentation).

Zero-shot/Open-vocabulary Semantic Segmentation. The goal of zero-shot semantic segmentation is to segment objects of interest that are not seen in the training set. Prior works [22, 40, 51] mainly transfer the knowledge from the training set (seen) to the testing set (unseen) via visual-semantic mapping. Inspired by the open-vocabulary nature of language, current approaches [17, 21, 26, 37, 44, 45, 60] exploit vision-language models (*e.g.*, CLIP [41]) pre-trained on large-scale image-caption pairs. They need either finetuning [17, 21, 26] or self-training [60] on the target segmentation dataset, which is less efficient and flexible than zero-shot transfer. VLP for open-vocabulary segmentation combines the merits of both zero-shot transfer and open-vocabulary recognition. GroupViT [53] designed a grouping vision transformer and learned the alignment between groups and text via the contrastive loss. ViL-Seg [32] combined image-text contrastive loss with online pixel clustering for segmentation. A concurrent work CLIPpy [42] explored different aggregation operations for training spatial-aware vision-language models for segmentation. Beyond the global image-text matching, we further design masked entity completion and cross-image mask consistency to enrich the group semantics.

Fully-/Weakly-/Semi-Supervised Semantic Segmentation. Fully-supervised semantic segmentation emerges from per-pixel classification [8, 34] to mask classification [13, 48]. To relieve the laborious annotations, extensive efforts have been made to address semantic segmentation with less supervision. In the family of weakly-supervised object localization [24,50,54] and semantic segmentation [1,9,52], only class labels are available for supervision. Generally, the class activation maps [58] derived from the classification network serve as the initial segmentation results. Another line of research focuses on semi-supervised semantic segmentation [11, 19, 36, 39, 49] where a few samples have dense per-pixel labels and the remaining samples are unlabeled. These works mainly perform supervised training on labeled samples with additional consistency regularizations posed on unlabeled samples. Despite promising results, these approaches are still limited to closed-set object categories.

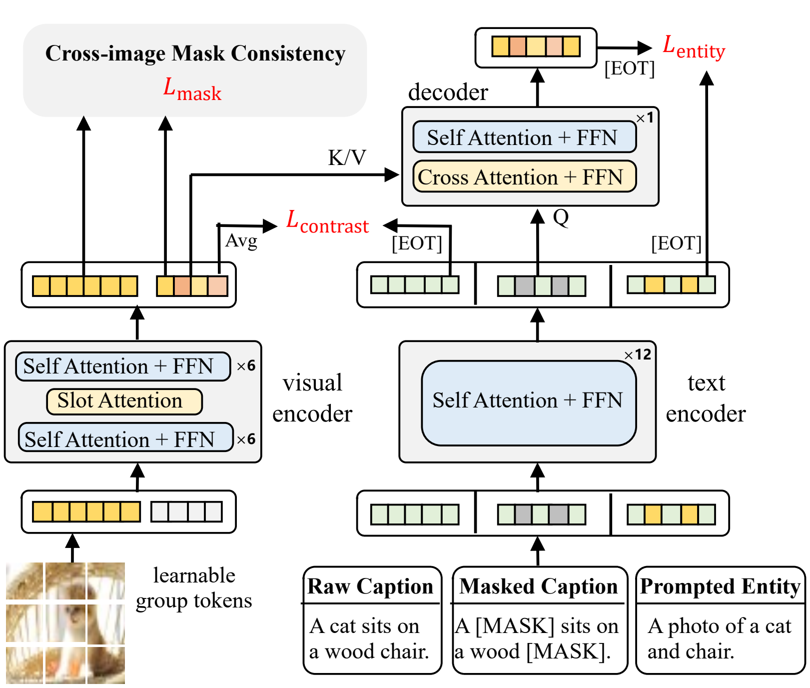

Figure 2. **Overall framework.** The model takes an image-text pair as input, and outputs the image features and a set of learnable group tokens, which is aligned to the caption embedding via $\mathcal{L}_{\text{contrast}}$. The visual encoder consists of Transformer encoder layers, with a slot-attention based binding module in between to cluster the image tokens into groups. To enrich group semantics, (1) $\mathcal{L}_{\text{entity}}$ trains the model to infer *all* masked entities in the caption given the group tokens. (2) $\mathcal{L}_{\text{mask}}$ enforces consistent mask predictions of images with the same entity (right figure). During inference, only visual and text encoders and learned group tokens are used to generate the segmentation mask.

3. Methodology

Assuming we are given a dataset of image-caption pairs, *i.e.*, $\mathcal{D}_{\text{train}} = \{(I_1, T_1), \ldots, (I_N, T_N)\}$, where $I_i \in \mathbb{R}^{H \times W \times 3}$ denotes the image, and T_i refers to its corresponding alt-text downloaded from the Internet, the goal is to train an open-vocabulary semantic segmentation (OVS) model only with these (very) weak labels, that can segment objects of arbitrary classes. For each pair (I, T), we first split the image into $L = HW/P^2$ non-overlapping patches with patch size P. These patches are then transformed into a sequence of image tokens with MLPs, we reuse $I \in \mathbb{R}^{L \times D}$ to represent them for ease of understanding. Additionally, we introduce K learnable tokens $G \in \mathbb{R}^{K \times D}$, that aims to group the image tokens into clusters.

We present a VLP framework, termed OVSegmentor, it assembles the image patches into groups and aligns the group to categories, by only exploiting weak supervisions from image-caption pairs. Conceptually, the architecture consists of two stages, namely, a **patch-to-group binding** that assigns all patches with same semantics into one group, and **group-to-category alignment** that computes matching scores between each of these group tokens with semantic categories, the segmentation mask m can be computed as:

$$m = \arg\max_{C} \mathbb{A} \cdot \mathcal{G} \cdot \mathcal{T}^{\top}, \tag{1}$$

where $\mathbb{A} \in \mathbb{R}^{L \times K}$ characterises a soft binding procedure, denoting the likelihood of each image patch being assigned into the output group embeddings ($\mathcal{G} \in \mathbb{R}^{K \times D}$); and $\mathcal{T} \in \mathbb{R}^{C \times D}$ denotes the embeddings with dimension D for a total of C categories, computed from a text encoder. **Note that**, at training time, manual segmentation masks are not present, and the intermediate binding and assignment can only be learnt from the image-caption pairs implicitly.

At inference stage, the patch-group affinity $\mathbb{A}$ can be computed by the product between the image features $\mathcal{I} \in \mathbb{R}^{L \times D}$ and the learned group tokens $\mathcal{G}$, and $\mathcal{T}$ can be derived by encoding the prompted class labels (*e.g.*, A photo of {class}). Thus the patch-level mask prediction can be calculated via Eq. 1. The pixel-level segmentation mask can then be obtained by up-sampling the patch-level mask (m). In the following sections, we detail the overall architecture and training procedure of each component. The overall framework is illustrated in Fig. 2.

3.1. Architecture

In this section, we will detail the proposed architecture at training stage, including a visual encoder for learning group tokens and the binding procedure, a text encoder that computes variants of caption embeddings. We describe the encoding procedure for image and caption pair, the subscripts are thus ignored for simplicity.

3.1.1 Visual Encoder

The visual encoder consists of two components, namely, Transformer encoders and binding modules. Specifically, the image tokens and learnable group tokens are concatenated, and iteratively processed by the Transformer encoders, and a binding module with slot-attention [33] is adopted for grouping. The visual encoder is defined as:

$$[\mathcal{G};\mathcal{I}] = \Phi^2_{\text{enc}}(\cdot) \circ \Phi_{\text{bind}}(\cdot) \circ \Phi^1_{\text{enc}}([G;I]), \tag{2}$$

where $\Phi^1_{\text{enc}}(\cdot)$ and $\Phi^2_{\text{enc}}(\cdot)$ refer to modules with Transformer encoder layers, and $\Phi_{\text{bind}}(\cdot)$ is the binding module. The output $\mathcal{G} \in \mathbb{R}^{K\times D}$ is the encoded group tokens, and $\mathcal{I} \in \mathbb{R}^{L\times D}$ refers to the output image tokens.

Transformer Encoder. Both Φ^1_{enc} and Φ^2_{enc} consist of 6 Transformer encoder layers [47], where each layer is composed of a multi-head self-attention (MHSA) layer followed by layer normalisation (LN) and a feed-forward network (FFN). Φ^1_{enc} takes the concatenation of image patches and randomly initialised group tokens as input, and outputs intermediately encoded group and image tokens (G' and I'); Φ^2_{enc} processes the output from the binding module.

Binding Module. The binding module uses slot-attention to cluster image tokens into groups in a data-dependent manner, *i.e.*, image patches with similar appearance and semantics are encouraged to be grouped together. Formally, the binding module accepts the output from the Transformer encoder Φ^1_{enc}, and transforms them into query, key and values with linear transformations:

$$\mathbf{Q} = W^{\text{bind}}_q G', \quad \mathbf{K}, \mathbf{V} = W^{\text{bind}}_k I',\ W^{\text{bind}}_v I'. \tag{3}$$

In contrast to the standard cross attention in Transformer Decoders [47], slot-attention performs normalisation over queries, encouraging each image token to be claimed by one of the group tokens. The binding process can be defined as:

$$\mathbb{A}_{j,k} = \frac{\exp(\mathbf{K}_j \cdot \mathbf{Q}_k)}{\sum_l \exp(\mathbf{K}_j \cdot \mathbf{Q}_l)}, \tag{4}$$

where $\mathbb{A}_{j,k}$ refers to the probability of the j-th image token being assigned to the k-th group. Then, each updated group token $\tilde{G}'_k$ is computed by taking the weighted average of the image tokens that are assigned to it. The output of the binding module G^{bind} is defined as:

$$\tilde{G}'_k = \frac{\sum_{j=1}^{L} \mathbb{A}_{j,k}\mathbf{V}_j}{\sum_{j=1}^{L} \mathbb{A}_{j,k}}, \tag{5}$$

$$G^{\text{bind}} = G' + W^{\text{bind}}_o \tilde{G}', \tag{6}$$

where W^{bind}_o is a linear transformation. Now we have obtained the correspondence between each patch and groups, next we describe the procedure for encoding captions.

3.1.2 Text Encoder

Till this end, we filter all the captions and only keep the ones with informative entities, followed by exploiting three variants of the caption encoding, namely, the entire caption embedding, the masked caption embedding and the prompted entity embedding. In all cases we use a pre-trained BERT [16] as the text encoder Φ_{text}.

Constructing Entity Set. We adopt the nltk toolkit [4] to extract entities from all the captions, and construct an entity set $\Omega = \Phi_{\text{entity}}(\{T_1, \ldots, T_N\})$ that only maintains the frequently appeared entities (*e.g.*, people, cat, shirt, etc.), and exclude the abstract nouns (*e.g.*, art, view, etc) as they usually do not correspond to any specific region in the image. For each image-caption pair, we can thus obtain an image-caption-entity triplet (I, T, E), where $E = \{e | e \in T \ \cap \ e \in \Omega\}$ includes *all* frequent entities in the caption.

Caption Embedding. Here, for each caption T, it is tokenised using the BERT tokeniser [16], with the start-of-token [SOT] and end-of-token [EOT] added to the beginning/ending. The caption embedding is computed as $\mathcal{T}^{\text{cap}} = \Phi_{\text{text}}(T) \in \mathbb{R}^{M\times D}$, where M is length after tokenisation.

Masked Caption Embedding. In the second variant, we mask *all* the frequent entities in the caption, and compute the masked caption embedding as $\mathcal{T}^{\text{m-cap}} = \Phi_{\text{text}}(g(T)) \in \mathbb{R}^{M\times D}$, where $g(\cdot)$ denotes an masking operation, that replaces the entity with a special [MASK] token.

Prompted Entity Embedding. We construct manual prompts for each of the entities in the triplets, and compute its textual embedding as $\mathcal{T}^{\text{entity}} = \Phi_{\text{text}}(h(E)) \in \mathbb{R}^{M\times D}$, where $h(\cdot)$ refers to the procedure for constructing manual prompts, for example, we randomly sample a prompt template provided in [41], and fill in the entities: *A painting of a* $\{entity_1\}$ *and* $\{entity_2\}$ *and* $\{entity_3\}$. This sentence is padded to the same length as the caption after tokenisation.

3.2. Training

As for training, we aim to learn the alignment between group tokens and caption embeddings via three proxy tasks, namely, image-caption alignment, masked entity completion, and cross-image mask consistency.

Image-caption Alignment. For each image-text pair, the objective is to align their visual and textual embeddings. The visual embedding z^{I} is the average of group tokens, and the textual embedding z^{T} is obtained by taking the [EOT] token feature of the caption embedding $\mathcal{T}^{\text{cap}}$, both projected to a 256-d joint feature space followed by normalisation. The image-caption contrastive loss $\mathcal{L}_{\text{contrast}}$ is formulated as:

$$\mathcal{L}_{\text{contrast}} = \frac{1}{2}\left(\frac{\exp(z^{\text{I}}_i \cdot z^{\text{T}}_i)}{\sum_l \exp(z^{\text{I}}_i \cdot z^{\text{T}}_l)} + \frac{\exp(z^{\text{I}}_i \cdot z^{\text{T}}_i)}{\sum_l \exp(z^{\text{I}}_l \cdot z^{\text{T}}_i)}\right). \tag{7}$$

Here, we omit the temperature parameter for simplicity.

Masked Entity Completion. The goal of masked entity completion is to infer all the masked entities in the sentence given the group tokens. In specific, we adopt a Transformer decoder layer, where a projection of the masked caption embedding is treated as query, and two linear transformations of group tokens are treated as key and values, respectively.

$$\tilde{\mathcal{T}}^{\text{m-cap}} = \Phi_{\text{dec}}(W_q^{\text{dec}}\mathcal{T}^{\text{m-cap}}, W_k^{\text{dec}}\mathcal{G}, W_v^{\text{dec}}\mathcal{G}) \in \mathbb{R}^{M\times D}, \tag{8}$$

where W_q^{dec}, W_k^{dec} and W_v^{dec} are linear transformations, and $\tilde{\mathcal{T}}^{\text{m-cap}}$ denotes the updated vector sequence, with the masked entities being completed by querying group tokens. For training, we extract the [EOT] token features z^{M} and z^{E} from $\tilde{\mathcal{T}}^{\text{m-cap}}$ and $\mathcal{T}^{\text{entity}}$, and construct a contrastive loss:

$$\mathcal{L}_{\text{entity}} = \frac{1}{2}\left(\frac{\exp(z_i^{\text{M}}\cdot z_i^{\text{E}})}{\sum_l \exp(z_i^{\text{M}}\cdot z_l^{\text{E}})} + \frac{\exp(z_i^{\text{M}}\cdot z_i^{\text{E}})}{\sum_l \exp(z_l^{\text{M}}\cdot z_i^{\text{E}})}\right). \tag{9}$$

Intuitively, the entity completion task enables better alignment between the groups and entities.

Cross-image Mask Consistency. To encourage visual invariance, we enforce consistent mask predictions between images that contain shared entities. Specifically, for each entity of interest, we can easily source multiple image-caption pairs from the dataset by text searching. Given one sampled image-caption-entity triplet, (I_1, T_1, E_1), we can easily search for another sample (I_2, T_2, E_2), with both triplet sharing one entity, *i.e.*, $e \in E_1 \cap E_2$. Given encoded groups $\mathcal{G}_1$, $\mathcal{G}_2$, two sets of subgroups $\mathcal{G}_1^{\text{sub}}, \mathcal{G}_2^{\text{sub}} \in \mathbb{R}^{K'\times D}$ that represent the common entity e (*e.g.*, cat in Fig. 2) are obtained by choosing groups with higher similarity to the entity embedding, where $K' = rK$ and r is the selection ratio. The masks of co-attentive entity in image I_1 can be grounded by both entity-specific subgroups $\mathcal{G}_1^{\text{sub}}, \mathcal{G}_2^{\text{sub}}$ as:

$$\mathcal{M}_1 = \sigma(\mathcal{I}_1^\top \mathcal{G}_1^{\text{sub}}),\ \hat{\mathcal{M}}_1 = \sigma(\mathcal{I}_1^\top \mathcal{G}_2^{\text{sub}}) \in [0,1]^{L\times K'}, \tag{10}$$

where σ is the sigmoid activation; both $\mathcal{M}_1 = \{m_k^1\}_{k=1}^{K'}$ and $\hat{\mathcal{M}}_1 = \{\hat{m}_k^1\}_{k=1}^{K'}$ consist of K' unordered masks.

To align $\hat{\mathcal{M}}_1$ with $\mathcal{M}_1$, we first adopt the bipartite matching to find the optimal permutation p_1^* over K' subgroups with the lowest matching cost as:

$$p_1^* = \underset{p\in\mathcal{P}}{\arg\min} \sum_k -\cos\left(m_k^1, \hat{m}_{p(k)}^1\right), \tag{11}$$

where $\mathcal{P}$ is the full permutation and $\cos(\cdot)$ denotes the cosine similarity. Eq. 11 is solved via the efficient Hungarian algorithm [25]. In this way, the symmetric cross-image mask consistency loss $\mathcal{L}_{mask}$ is defined as:

$$\mathcal{L}_{mask} = \frac{1}{2}\left(\sum_k \text{D}\left(\text{sg}(\mathbf{m}_k^1), \hat{m}^1_{p_1^*(k)}\right) + \text{D}\left(\text{sg}(\mathbf{m}_k^2), \hat{m}^2_{p_2^*(k)}\right)\right), \tag{12}$$

where $\text{sg}(\cdot)$ denotes the stop gradient operation; the target mask $\mathbf{m}_k$ is achieved by binarizing m_k with a threshold δ; $\text{D}(\mathbf{m}, \hat{m}) = 1 - 2|\mathbf{m} \bigcap \hat{m}|/(|\mathbf{m}| + |\hat{m}|)$ stands for the standard Dice loss. To guarantee the quality of pseudo mask targets, the masks are generated by an extra momentum model, which is updated by the exponential-moving-average (EMA) of the online model.

Training Objective. We adopt a combination of three different loss functions:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contrast}} + \mathcal{L}_{\text{entity}} + \lambda\mathcal{L}_{\text{mask}}, \tag{13}$$

where λ is the weight for balancing the mask consistency.

3.3. Discussion

One work that is closely related to ours is GroupViT [53]. They extracted multiple nouns and prompted each noun to a sentence to serve as extra matched captions for the image. The model is thus supervised by a multi-label contrastive loss. In contrast, our paper differs from GroupViT in three critical aspects: (1) Entities vs nouns. Rather than using all nouns, we leverage the entities that match to visual objects, enabling high-quality image-caption correspondence. (2) Network architecture. We only adopt single-stage grouping (i.e. one binding module) as we observe no further performance gain of applying multi-stage grouping in GroupViT. The hard-assignment trick [53] is also adopted. Besides, we devise a (very) lightweight decoder to model the fine-grained, token-wise group-word correlation. (3) Proxy tasks for training. We propose two different proxy tasks, *i.e.*, masked entity completion and cross-image mask consistency to improve the entity-specific group semantics and further encourage visual invariance.

4. Experiments

4.1. Experimental Setups

Pre-training Dataset. Following [32, 53], we use Conceptual Captions 12M [7] for training, which is originally constructed with over 12M image-text pairs collected from the Internet. However, due to some links have been expired, we have downloaded about 10M image-text pairs. The constructed entity set in Sec. 3.1.2 includes a total number of 100 frequently appeared entities while abstract nouns (*e.g.*, art, view) are discarded. After filtering CC12M, we obtain 4.3 million image-text pairs for pre-training, which is termed as CC4M. Examples of entities include *people*, *car*, *cup*, *chair*, *T-shirt*, *house*, *bed*, *cat*, *ball*, *pizza*, *etc.* Please refer to the supplementary material for the full entity set.

Downstream Evaluation Datasets. We evaluate our model on four benchmarks, namely, PASCAL VOC 2012 [18], PASCAL Context [38], COCO Object [31], and ADE20K [59] with 20, 59, 80 and 150 foreground classes,

Method	Backbone	Pretrain dataset	Supervision	Zero-shot transfer	Downstream datasets			
					PASCAL VOC	PASCAL Context	COCO Object	ADE20K
DeiT [46]	ViT-S	IN-1K	class label	✗	53.0	35.9	-	-
MoCo [10]	ViT-S	IN-1K	self	✗	34.3	21.3	-	-
DINO [6]	ViT-S	IN-1K	self	✗	39.1	20.4	-	-
MoCo [10]	ViT-S	CC12M+YFCC15M	self	✗	36.1	23.0	-	-
DINO [6]	ViT-S	CC12M+YFCC15M	self	✗	37.6	22.8	-	-
ViL-Seg [32]	ViT-B	CC12M	self+text	✓	33.6	15.9	-	-
GroupViT [53]	ViT-S	CC12M	text	✓	41.1	-	-	-
GroupViT [53]	ViT-S	CC12M+YFCC15M	text	✓	51.2	22.3	20.9	-
ViewCo [43]	ViT-S	CC12M+YFCC15M	self+text	✓	52.4	**23.0**	23.5	-
CLIPpy [42]	ViT-B	HQITP-134M	text	✓	52.2	-	25.5†	13.5
GroupViT* [53]	ViT-S	CC4M	text	✓	19.8	8.8	9.1	3.4
GroupViT* [53]	ViT-B	CC4M	text	✓	25.8	11.3	10.7	3.6
GroupViT* [53]	ViT-S	CC12M	text	✓	40.2	18.7	17.7	**6.2**
OVSegmentor (ours)	ViT-S	CC4M	text	✓	44.5	18.3	19.0	4.3
OVSegmentor (ours)	ViT-B	CC4M	text	✓	**53.8**	20.4	**25.1**	5.6

Table 1. **Comparison with existing methods.** Models in the first five rows are finetuned on target datasets while the rest perform zero-shot transfer. GroupViT (with *) refers to the results by training officially released code. Results of DeiT, MoCo, and DINO are copied from [53]. Bold fonts refer to the best results among the models that enable zero-shot transfer. CLIPpy uses 134M in-house data while we use an order of magnitude less data. † indicates evaluation on 133 COCO classes claimed by CLIPpy, which differs from ours (80 classes).

respectively. An extra background class is considered in VOC, Context, and COCO. We ignore their training sets and directly evaluate our method on the validation sets without any finetuning, including 1449, 5105, 5000, and 2000 images, respectively. In general, we report the mean Intersection-over-Union (mIoU) on all the classes.

Implementation Details. In our model, the self-attention layers in the visual encoder are initialised with DINO [6] pre-trained on ImageNet. The text encoder is initialised with BERT [16] model pre-trained on BookCorpus and English Wikipedia. Our decoder with one randomly initialised Transformer Decoder layer performs reasonably well. The input image is randomly cropped to 224×224 at training time, and the batch size is set to 2048 with an initial learning rate 3.2×10^{-4}. We train our model for 40 epochs using the Adam optimizer with weight decay set to 0.5. The coefficient for updating the momentum model is 0.99. As the generated masks are unreliable in early epochs, we set the mask consistency coefficient λ=0 for the first 30 epochs and λ=0.1 for the remaining epochs. The group selection ratio r is 0.5. As for the threshold in mask consistency loss, we use δ=0.65. At inference time, the image is resized with a shorter length of 448. We follow [53] to set a threshold for the background class, which is 0.9, 0.35, 0.9 and 0.95 on VOC, Context, COCO, and ADE20K, respectively.

4.2. Comparison with Existing Methods

In Table 1, we compare our model with the existing models that have been trained with (1) fully-supervised finetuning transfer and (2) zero-shot transfer. In Table 2, zero-shot segmentation (ZSS) approaches are listed for comparison.

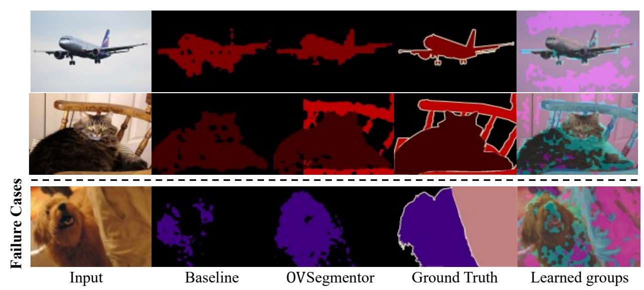

Figure 3. **Qualitative results on PASCAL VOC.** We show the learned groups in the last column with each color representing a group. In the last row, the model fails to segment the person's leg.

Comparison with Finetuning Transfer. We compare our method with DeiT [46], MoCo [10] and DINO [6], which are pre-trained on ImageNet [15] or CC12M+YFCC15M datasets with class labels [46] or self-supervision [6, 10], and finetuned on the training set from downstream benchmarks, with a randomly initialised convolution head appended on the backbone network. As shown in Table 1, our model achieves competitive performance on PASCAL Context, and outperforms the self-supervised methods by over 10% on PASCAL VOC, with zero-shot transfer.

Comparison with Zero-shot Transfer. Here, we compare with existing works under the zero-shot transfer scenario, including GroupViT [53], ViL-Seg [32] and CLIPpy [42], with the pre-training data ranging from CC12M [7] to 134M in-house dataset HQITP-134M [42]. For fair comparison, we re-train GroupViT [53] with their official codebase on our CC4M and CC12M datasets, with the same pre-trained weights as ours being adopted on CC4M. However, we observe no further performance gain of either applying pre-

Method	Pretrain dataset	Seen	Unseen	VOC	Context
SPNet [51]	-	✔	✗	15.6	4.0
ZS3 [5]	-	✔	✔	17.7	7.7
GaGNet [22]	-	✔	✔	29.9	15.0
SIGN [14]	-	✔	✔	28.9	14.9
Joint [2]	-	✔	✗	32.5	-
ZegFormer [17]	CLIP400M	✔	✗	63.6	
MaskCLIP+ [60]	CLIP400M	✔	✗	86.1	66.7†
ViL-Seg [32]	CC12M	✗	✗	37.3	18.9
GroupViT [53]	CC12M+YFCC15M	✗	✗	43.7	51.3
GroupViT* [53]	CC4M	✗	✗	22.4	24.5
GroupViT* [53]	CC12M	✗	✗	33.1	45.3
OVSegmentor	CC4M	✗	✗	**46.6**	**54.5**

Table 2. **Comparison with zero-shot segmentation methods on unseen classes.** Seen/Unseen denotes whether the model is trained on seen/unseen classes. Our method outperforms most zero-shot segmentation models even without training on seen classes. † indicates a different set of unseen classes.

trained weights to GroupViT or using ViT-B as the backbone on CC12M. The results on CC12M match the reported ones (40.2 vs 41.1). Under the same pre-trained ViT-B backbone and CC4M dataset, our method surpasses the original GroupViT by 35.7%. Additionally, by only using 4M pre-training data, our model yields the best segmentation performance on PASCAL VOC, even outperforming CLIPpy, which is a concurrent work to ours, and pre-trained on 134M data, indicating the effectiveness and training efficiency of our proposed model.

Comparison with Zero-shot Segmentation Methods. In this line of research [17, 22, 51], the idea is to train the model with full mask labels (obtained either from manual groundtruth or pseudo-labelling [60]) on the seen classes and transfer the model to unseen classes, and the task is thus dubbed as zero-shot semantic segmentation (ZSS). To be specific, 5 classes (potted plant, sheep, sofa, train and tv-monitor) in PASCAL VOC and 4 classes (cow, motorbike, sofa and cat) in PASCAL Context are considered unseen while the remaining classes belong to seen. For comparison, we also report the zero-shot transfer performance of our model on these unseen classes, however, **note that**, we do not use any manual mask annotations for training. As observed in Table 2, our model surpasses majority of the models trained under the ZSS scenario, except for [17, 60] that adopted CLIP model pre-trained on 400M data. In terms of transfer efficiency, our model excels at zero-shot transfer ability without the need of training on seen classes.

4.3. Ablation Study

In this section, we conduct thorough ablation studies to validate the necessity of each proposed component.

Ablation Study on Proxy Tasks. Here, we aim to understand the effects of our proposed proxy tasks, *i.e.*, masked entity completion and cross-image mask consistency. As

Baseline	$\mathcal{L}_{\text{entity}}$	$\mathcal{L}_{\text{mask}}$	PASCAL VOC	PASCAL Context
✔			40.5	15.1
✔	✔		48.9	19.9
✔		✔	44.8	17.0
✔	✔	✔	53.8	20.4

Table 3. **Ablation study** on the masked entity completion loss ($\mathcal{L}_{\text{entity}}$) and cross-image mask consistency loss ($\mathcal{L}_{\text{mask}}$). The baseline only uses the image-caption contrastive loss ($\mathcal{L}_{\text{contrast}}$).

Masking objective	PASCAL VOC	PASCAL Context
All entities (ours)	**48.9**	**19.9**
Single entity	47.0	18.5
All nouns	45.4	17.4
Multi-label contrastive	44.5	17.0
MLM (w/ groups)	42.8	16.3
MLM (w/o groups)	36.3	14.6

Table 4. **Comparison on masking objectives.** MLM (w/ groups) refers to masked language modeling [16] given the group tokens as keys and values in the decoder while MLM (w/o groups) directly predicts masked words from the output of text encoder.

shown in Table 3, the baseline model uses the image-text contrastive loss $\mathcal{L}_{\text{contrast}}$ only, while adding the entity completion task, $\mathcal{L}_{\text{entity}}$, we observe a significant improvement by 8.4% and 4.8% on PASCAL VOC and PASCAL Context, respectively. The performance gain is due to the ability of better aligning the pixel groups with the visual entities. Additionally, the mask consistency also brings improvements, and combing both leads to the best performance. Qualitative results can be seen in Fig. 3. We refer the readers for more visualisations in the supplementary material.

On the Choice of Masking Objectives. We compare our masked entity completion with a series of variants as shown in Table 4. Our proposed objective of masking *all entities* is listed in the first row. **(1) All entities vs one entity**: the masked language modeling (MLM) in prior works [16, 35] normally choose 15% of the token positions in the sentence for prediction, which results in one entity in most of our cases. We observe that masking *all* entities is 2.9% mIoU better than single entity masking, as it forces the network to infer all possible object categories in the image, that is potentially beneficial for the group to category alignment. **(2) Entities vs nouns**: masking and predicting noun phrases in the sentence [20] is one feasible option to learn fine-grained vision-text matching. However, noun masking leads to 3.5% lower mIoU than our entity masking strategy. This is because not all of the nouns in the sentence are visually corresponding to the objects in the image (*e.g.*, illustration, night, etc), thus the group tokens are not expected to align with these nouns. Our method avoids this issue by only masking the visual entities. **(3) Masked entity completion vs multi-label contrastive loss:** comparing with the multi-label contrastive loss used in GroupViT [53], our proposed

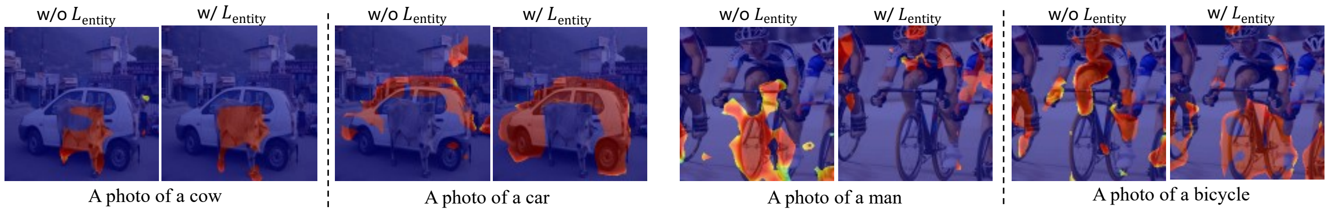

Figure 4. **Qualitative results of the masked entity completion loss.** We show different groups that match the given class. With the boost of group semantics, our model improves visual grouping (left) and solves false matching between group and text (right) of the baseline.

K/τ	0.25	0.5	0.75	1.0
8	49.0	53.6	51.8	50.4
16	42.3	43.2	43.1	42.9

Table 5. Comparison on group numbers K and selection ratios τ on VOC.

Objective	Loss	mIoU
Mask Consistency	Dice	53.8
Group Consistency	NCE	47.9

Table 6. Comparison on different objectives for cross-image consistency on VOC.

Method	VOC		COCO		Context		ADE20K	
	S_{20}	U_0	S_{65}	U_{16}	S_{30}	U_{29}	S_{31}	U_{119}
Baseline	38.5	-	17.2	15.4	20.5	8.8	7.1	3.7
Ours(CC4M)	**52.4**	-	**24.8**	**25.9**	**27.9**	10.9	8.6	4.8
Ours(CC12M)	44.8	-	18.6	21.5	25.4	**12.0**	**10.1**	**6.3**

Table 7. Performance on seen(S)/unseen(U) entities, where indices refer to entity numbers. Background class is not considered.

strategy shows superior performance. **(4) Masked entity completion vs masked language modeling**: MLM [16] originally predicts the masked token over the entire vocabulary via a cross-entropy loss. Here, we restrict the vocabulary to our constructed entity set for fair comparison. Our masking strategy surpasses both MLM variants by a clear margin, which we conjecture is because: (1) MLM classifies each masked token *individually*, and the model can easily refer to the context words without relating to groups. (2) MLM focuses on word-level representations, which is not in accordance with the sentence-level representation of the class embeddings we used during inference. Fig. 4 shows the effect of the masked entity completion in improving visual grouping (left) and group-text alignment (right).

On Cross-image Mask Consistency. Here, we analyze two key factors in cross-image consistency: (1) group numbers and selection ratios. As shown in Table 5, 8 groups perform comparably well for all selection ratios, and we pick 0.5 as the default. Smaller ratios miss the entities encoded in remaining groups while larger ratios introduce entity-irrelevant information (*e.g.*, background). However, while increasing the group number to 16, it brings over-segmentation for large objects, deteriorating the performance. (2) the objective for cross-image consistency. We study another variant of cross-image consistency by directly aligning two sets of group tokens encoded with shared entity. We adopt the contrastive loss (NCE in Table 6) to pull two sets of group tokens closer, while other group tokens in each mini-batch are pushed farther. Table 6 reveals the superiority of our proposed mask consistency over group consistency, which we believe is because mask consistency involves the image content to realize visual invariance.

Performance on Unseen Entities. We measure the mean IoU for objects within entity set (*e.g.*, person, bus) and those out of entity set on each dataset. As shown in Table 7, our model still achieves a decent improvement over the baseline on unseen entities, indicating it retains strong open-vocabulary segmentation ability. The text encoder pre-trained on large text corpus remains its ability to encode the semantic concept of objects out of entity set.

Scalability. We scale up the datasets by increasing the number of entities, i.e., training the model on the entire CC12M. Ours (CC12M) outperforms ours (CC4M) on the challenging ADE20K and PASCAL Context, reflecting the effect of the scalability of our model. The performance drop on VOC is because the extremely large entity set in CC12M poses extra challenge for our model to learn the entities in VOC.

5. Conclusion

In this paper, we present OVSegmentor, a transformer-based model for open-vocabulary semantic segmentation. The model exploits web-collected image-caption pairs for pre-training without any mask annotations, and transfers to target benchmark segmentation datasets in a zero-shot manner. The model clusters the image pixels into learnable group tokens, which are then aligned with the corresponding caption embeddings. We further devise two proxy tasks, namely masked entity completion and cross-image mask consistency, to learn entity-specific, fine-grained and visually invariant group semantics. OVSegmentor outperforms the state-of-the-art method on PASCAL VOC by using only 3% (4M vs 134M) for pre-training, indicating the effectiveness and training efficiency of our model.

Acknowledgement. This work is supported by Natural Science Foundation of China (No. 61976057; No. 62172101), the Science and Technology Commission of Shanghai Municipality (No. 21511101000; No. 22511106003), the Science and Technology Major Project of Commission of Science and Technology of Shanghai (No. 2021SHZDZX0103). This work is also partially supported by the National Key R&D Program of China (No. 2022ZD0160100), and in part by Shanghai Committee of Science and Technology (No. 21DZ1100100). This work was done during Jilan's parttime internship at Shanghai Artificial Intelligence Laboratory.

References

[1] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018. 2

[2] Donghyeon Baek, Youngmin Oh, and Bumsub Ham. Exploiting a joint embedding space for generalized zero-shot semantic segmentation. In *ICCV*, 2021. 7

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 2

[4] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009. 4

[5] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. *NeurIPS*, 2019. 7

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 6

[7] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 2, 5, 6

[8] 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. *PAMI*, 2017. 2

[9] Qi Chen, Lingxiao Yang, Jian-Huang Lai, and Xiaohua Xie. Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation. In *CVPR*, 2022. 2

[10] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021. 6

[11] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, 2021. 2

[12] 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 *ECCV*, 2020. 2

[13] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *NeurIPS*, 2021. 2

[14] Jiaxin Cheng, Soumyaroop Nandi, Prem Natarajan, and Wael Abd-Almageed. Sign: Spatial-information incorporated generative network for generalized zero-shot semantic segmentation. In *ICCV*, 2021. 7

[15] 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

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 4, 6, 7, 8

[17] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *CVPR*, 2022. 2, 7

[18] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 2, 5

[19] Geoffrey French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham D. Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. In *BMVC*, 2020. 2

[20] Yuying Ge, Yixiao Ge, Xihui Liu, Dian Li, Ying Shan, Xiaohu Qie, and Ping Luo. Bridging video-text retrieval with multiple choice questions. In *CVPR*, 2022. 2, 7

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

[22] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. Context-aware feature generation for zero-shot semantic segmentation. In *ACM MM*, 2020. 2, 7

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

[24] Eunji Kim, Siwon Kim, Jungbeom Lee, Hyunwoo Kim, and Sungroh Yoon. Bridging the gap between classification and localization for weakly supervised object localization. In *CVPR*, 2022. 2

[25] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 5

[26] Boyi Li, Kilian Q. Weinberger, Serge J. Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 2

[27] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *CVPR*, 2022. 2

[28] Juncheng Li, Xin He, Longhui Wei, Long Qian, Linchao Zhu, Lingxi Xie, Yueting Zhuang, Qi Tian, and Siliang Tang. Fine-grained semantically aligned vision-language pre-training. *NeurIPS*, 2022. 2

[29] 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. *NeurIPS*, 2021. 2

[30] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *CVPR*, 2022. 2

[31] 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*. Springer, 2014. 2, 5

[32] 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. 2, 5, 6, 7

[33] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *NeurIPS*, 2020. 2, 4

[34] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2

[35] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *NeurIPS*, 2019. 7

[36] Wenfeng Luo and Meng Yang. Semi-supervised semantic segmentation via strong-weak dual-branch network. In *ECCV*, 2020. 2

[37] Chaofan Ma, Yuhuan Yang, Yanfeng Wang, Ya Zhang, and Weidi Xie. Open-vocabulary semantic segmentation with frozen vision-language models. *BMVC*, 2022. 2

[38] 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

[39] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*, 2020. 2

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

[41] 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, 4

[42] 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. 2, 6

[43] 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. 6

[44] Gyungin Shin, Weidi Xie, and Samuel Albanie. Namedmask: Distilling segmenters from complementary foundation models. *arXiv preprint arXiv:2209.11228*, 2022. 2

[45] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. *NeurIPS*, 2022. 2

[46] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 6

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 4

[48] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *CVPR*, 2021. 2

[49] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. Semi-supervised semantic segmentation using unreliable pseudo-labels. In *CVPR*, 2022. 2

[50] Pingyu Wu, Wei Zhai, and Yang Cao. Background activation suppression for weakly supervised object localization. In *CVPR*, 2022. 2

[51] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *CVPR*, 2019. 2, 7

[52] 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 *CVPR*, 2022. 2

[53] 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, 5, 6, 7

[54] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Rui-Wei Zhao, Tao Zhang, Xuequan Lu, and Shang Gao. Cream: Weakly supervised object localization via class re-activation mapping. In *CVPR*, 2022. 2

[55] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. FILIP: fine-grained interactive language-image pre-training. In *ICLR*, 2022. 2

[56] Yan Zeng, Xinsong Zhang, and Hang Li. Multi-grained vision language pre-training: Aligning texts with visual concepts. In *ICML*, 2022. 2

[57] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. Glipv2: Unifying localization and vision-language understanding. *NeurIPS*, 2022. 2

[58] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 2

[59] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019. 2, 5

[60] Chong Zhou, Chen Change Loy, and Bo Dai. Denseclip: Extract free dense labels from clip. *ECCV*, 2022. 2, 7