Learning to Generate Text-grounded Mask for Open-world Semantic Segmentation from Only Image-Text Pairs

Junbum Cha Jonghwan Mun Byungseok Roh

Kakao Brain

{junbum.cha, jason.mun, peter.roh}@kakaobrain.com

Abstract

*We tackle open-world semantic segmentation, which aims at learning to segment arbitrary visual concepts in images, by using only image-text pairs without dense annotations. Existing open-world segmentation methods have shown impressive advances by employing contrastive learning (CL) to learn diverse visual concepts and transferring the learned image-level understanding to the segmentation task. However, these CL-based methods suffer from a train-test discrepancy, since it only considers image-text alignment during training, whereas segmentation requires region-text alignment during testing. In this paper, we proposed a novel **T**ext-grounded **C**ontrastive **L**earning (TCL) framework that enables a model to directly learn region-text alignment. Our method generates a segmentation mask for a given text, extracts text-grounded image embedding from the masked region, and aligns it with text embedding via TCL. By learning region-text alignment directly, our framework encourages a model to directly improve the quality of generated segmentation masks. In addition, for a rigorous and fair comparison, we present a unified evaluation protocol with widely used 8 semantic segmentation datasets. TCL achieves state-of-the-art zero-shot segmentation performances with large margins in all datasets. Code is available at https://github.com/kakaobrain/tcl.*

1. Introduction

Open-world semantic segmentation aims to identify the arbitrary semantic concepts in the open world[1]. Conventional semantic segmentation aims to learn segmentation capability for the small number of pre-defined target categories, whereas open-world semantic segmentation addresses unrestricted arbitrary categories or free-form texts. Such segmentation capability over unlimited targets drastically extends the application scope of the open-world segmentation models.

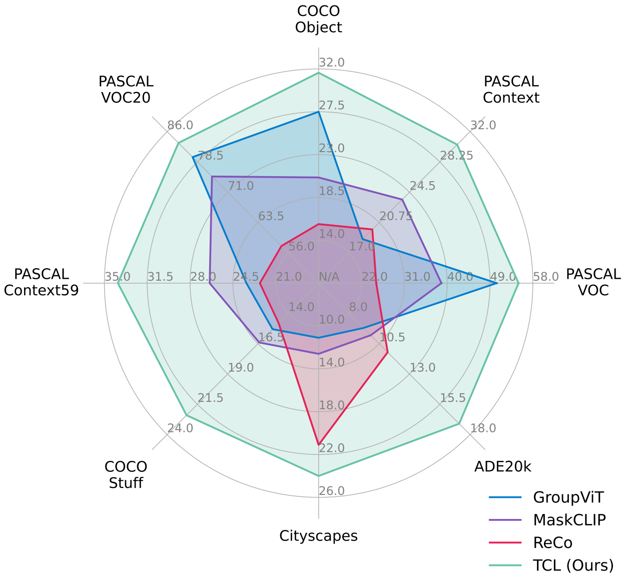


Figure 1. **Open-world segmentation performance comparison.** The proposed method remarkably outperforms existing methods in all 8 segmentation benchmark datasets.

The first challenge for open-world segmentation is how to learn arbitrary concepts, beyond pre-defined categories. Inspired by the success of CLIP [23], previous approaches [11, 17–19, 28, 30, 33] tackle this challenge by exploiting massive web-crawled image-text paired data; since the texts in web-crawled data contain a global semantic description for the paired images, the large-scale image-text pairs can provide rich knowledge for arbitrary semantic categories. However, there still remains another challenge in *how to achieve precise localization of arbitrary concepts without dense annotations*. There are several approaches that simply address this issue using dense annotation (segmentation masks) in addition to image-text pairs [11, 17, 18]. The dense annotation helps to improve segmentation perfor-

[1] This setting is often called both *open-world* and *open-vocabulary*. In this paper, we mainly refer to this setting as *open-world* for clarity.

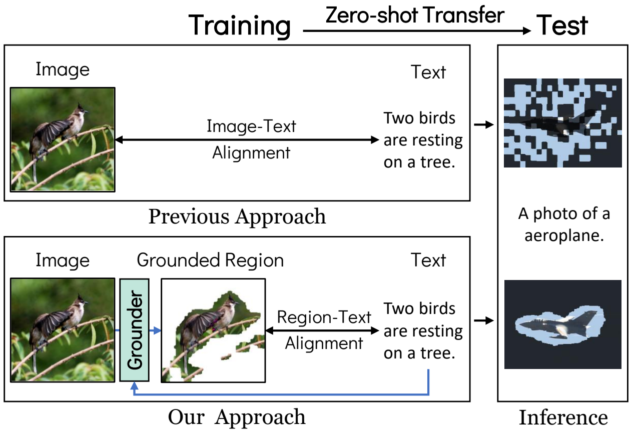


Figure 2. **A conceptual comparison between the previous approach and ours.** Open-world segmentation is typically achieved through region-text alignment, which involves matching region features and text embeddings. However, previous methods learn image-text alignment during training, thus suffering from the alignment-level discrepancy between training and testing. In contrast, our method facilitates end-to-end learning of region-text alignment with only image-text pairs.

mance in a fixed benchmark dataset, but the requirements of expensive dense annotation still limit the applicable domains and scalability of the method.

In this paper, therefore, we focus on open-world semantic segmentation from only image-text pairs without any dense annotation. For this setting, the existing methods [19, 28, 30, 33] learn an image-text alignment capability during training and heavily rely on the transferability of the image-text alignment to perform region-text alignment at inference. More specifically, MaskCLIP [33] leverages CLIP models pre-trained to learn image-text alignment. To perform region-text alignment using CLIP, MaskCLIP applies a simple heuristic modification to the CLIP image encoder. GroupViT [30] and ViL-Seg [19] propose to cluster region-level visual features into distinct groups and generate segmentation masks by matching the groups and texts. Note that they match the text embeddings and clustered region features in test time, but in training time, the text embeddings are aligned with global image embeddings. While the existing methods have shown impressive results even through the training with image-text alignment, they still suffer from the alignment-level discrepancy between training and testing phases as depicted in Fig. 2.

To address this train-test discrepancy, we propose the **T**ext-grounded **C**ontrastive **L**earning (TCL) framework, which allows a model to learn region-text alignment directly from the image-text pairs without any dense annotations. Our key idea is to incorporate a text grounding procedure within contrastive learning as illustrated in Fig. 2, where TCL generates a segmentation mask indicating text-grounded regions, computes grounded region embeddings using the mask, and applies contrastive learning between text and grounded region. By re-formulating the contrastive loss to be directly affected by the segmentation quality, TCL enables end-to-end training of the grounder and directly improves the quality of region-text level alignment. We also present a unified evaluation protocol using widely used 8 semantic segmentation datasets and compare existing methods in the same setting. As a result, TCL achieves state-of-the-art zero-shot segmentation performance with large margins in all datasets, as shown in Fig. 1.

Our main contributions are summarized as follows:

- We introduce a novel framework for open-world segmentation, named Text-grounded Contrastive Learning (TCL), which enables learning region-text alignment directly without train-test discrepancy, thus learning to generate more precise segmentation masks through only image-text pairs.
- We present a unified evaluation protocol and re-evaluate recent open-world segmentation models for a fair and direct comparison.
- We achieve the new state-of-the-art zero-shot segmentation performance on 8 segmentation datasets with large margins compared to existing methods.

2. Related Works

2.1. Open-world Semantic Segmentation

Open-world scenario aims to recognize arbitrary concepts in the open world. It is also called **open-vocabulary** because the target vocabulary is open rather than closed. Contrastive Language-Image Pre-training (CLIP) [23] ushered in the era of open-world image recognition using large-scale image-text pairs [2, 5, 26, 27]. CLIP learns the alignment between an image and a text in training time, then transfer it to the zero-shot classification by aligning image and texts indicating target classes at inference time. The advent of CLIP enables open-world settings in various fields such as object detection [12, 31], image captioning [13], or semantic segmentation [30, 33].

Open-world semantic segmentation with image-text pairs is addressed in two different settings. The first is a **semi-supervised setting**, which uses dense annotation (*i.e*., segmentation masks) in addition to image-text pairs [11, 17, 18]. Semi-supervised approaches learn segmentation capability using dense annotation and expand the target vocabulary using image-text supervision. LSeg [17] expands target class vocabulary using image-label datasets and CLIP text encoder [23]. OpenSeg [11] and OVSeg [18] first train a mask generator using dense annotation and expand target vocabulary using image-text datasets. The use of dense annotation makes the model learn region-level alignment instead of image-level alignment, leading to high-quality segmentation masks. However, it still relies on costly dense annotation, and applicable domains are limited to the domains where dense annotation is available.

The target of this paper is an **unsupervised setting**, which aims to learn segmentation from only image-text pairs without any dense annotation [19, 28, 30, 33]. Since the massive image-text pairs are easily obtained by web crawling without human annotators, applicable domains of unsupervised methods become almost unlimited. In order to achieve segmentation capability using only image-text pairs, we need to learn region-text alignment instead of image-text alignment and train a text-grounded mask generator. However, the absence of dense annotation makes this approach challenging. Existing open-world semantic segmentation studies have taken a strategy to bypass this issue. Instead of learning region-level alignment directly, they transfer image-level alignment to region-level by heuristic modification [28, 33] or clustering [19, 30]. MaskCLIP [33] proposes to obtain a dense image embedding from CLIP image encoder through heuristic modification of the last attention layer. Even though it has several limitations, such as low output resolution or noisy segmentation results, they show it is a simple yet effective way to obtain an initial segmentation map for refinement. ReCo [28] proposes an advanced refinement method based on MaskCLIP, by retrieval and co-segmentation. Clustering-based methods [19,30] learn representations using CL with image-text pairs. They compute region-level image embedding by clustering sub-region embeddings. These approaches also have shown impressive results but have several limitations: (i) the learning objective is still image-level alignment due to lack of the region annotation, (ii) the number of clusters is pre-defined independent of the given image, and (iii) clustering sub-region image embeddings is independent of the query text. In summary, existing methods indirectly address region-level alignment problems by learning image-level alignment. To tackle this problem, we propose a novel region-level alignment objective, named Text-grounded Contrastive Learning (TCL).

2.2. Region-level Contrastive Learning

Learning region-level alignment instead of image-level alignment is a fundamental target objective in dense tasks, such as segmentation or object detection. There are approaches to learn region-level alignment using dense annotation in the semi-supervised setting. They first train mask or region proposal networks using dense annotation and learn alignment between the proposals and texts [11,15,31]. For example, OpenSeg [11] trains a class-agnostic mask generator using dense annotation. In the object detection field, RegionCLIP [31] employs an off-the-shelf region proposal network and learns region-level alignment. In contrast to the existing region-level methods, the proposed method learns region-level alignment without any dense annotation.

3. Methods

3.1. Overview

Open-world semantic segmentation is a task that aims to learn a model capable of zero-shot segmentation for arbitrary visual concepts, not restricted to pre-defined ones. Our main goal is to develop an open-world segmentation algorithm using only image-text pairs. However, achieving this objective is challenging because there is no explicit supervision (*i.e.*, pixel-level dense annotations) for text-described region segmentation. Existing methods learn models parametrized by θ to maximize the mutual information between paired images and texts [22, 23] as follows:

$$\arg\max_{\theta} I_\theta(\mathbf{x}^V;\mathbf{x}^T), \tag{1}$$

where $(\mathbf{x}^V,\mathbf{x}^T)$ is a random image and text pair. This objective encourages the model to learn the alignment between images and texts, however, at test time, the learned model generates segmentation masks for arbitrary concepts by computing region-text alignments. Such alignment-level discrepancy between train and test time can lead the model to a sub-optimal solution as shown in Fig. 2. With this in consideration, to bridge the gap between the objective of conventional contrastive learning (CL) and the requirement of the zero-shot segmentation, we propose Text-grounded Contrastive Learning (TCL) which incorporates a text grounding process within CL to enable learning region-text alignment directly. As a text grounding module, we introduce a grounder to generate segmentation masks for the given texts. In a nutshell, TCL learns a model to maximize mutual information between text-grounded regions and texts as follows:

$$\arg\max_{\theta} I_\theta(\mathbf{m}\cdot\mathbf{x}^V;\mathbf{x}^T), \tag{2}$$

where $\mathbf{m}$ is a text-grounded mask of random variable indicating the text-described region. Compared to contrastive learning that implicitly learns a grounding capability, TCL has a clear advantage of explicitly learning the grounding capability through the end-to-end trainable grounder.

In the rest of this section, we first explain the text-grounded mask generation procedure by the grounder. Then, we describe how we define losses using the generated mask to train our open-world grounder with text-grounded contrastive learning. Lastly, we explain how our model performs zero-shot inference for arbitrary concepts.

3.2. Grounder

Fig. 3 illustrates our overall training pipeline. For an input batch of paired texts $\mathbf{X}^T$ and images $\mathbf{X}^V$, TCL first performs a grounding process to identify text-grounded regions for a text via a grounder. The grounder consists of

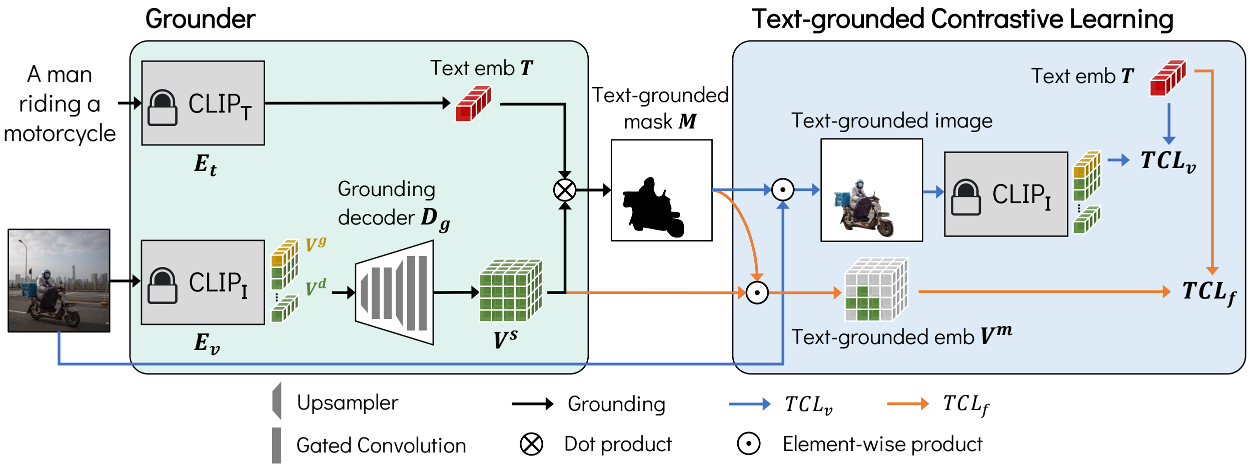


Figure 3. **Overall training pipeline of TCL.** The proposed TCL framework first obtains text-grounded masks using a grounder and then learns the grounder using text-grounded contrastive learning. By incorporating the text grounding process with contrastive learning, our framework can directly learn region-text alignment that is required for precise segmentation. `CLIP`$_\text{T}$ and `CLIP`$_\text{I}$ indicate CLIP text and image encoders, respectively. CLIP encoders are frozen and we train the grounding decoder only. After training, the TCL block is discarded, and only the Grounder block is used to generate the text-grounded segmentation mask for inference.

three components: (i) image encoder E_v is in charge of providing a single (L2-normalized) global feature as well as dense patch-level features, (ii) text encoder E_t provides a (L2-normalized) text embedding feature, and (iii) grounding decoder D_g converts dense features from image encoder into finer pixel-level embeddings for alignment with text. In practice, we adopt a pre-trained CLIP model [23] to initialize two encoders and freeze them to preserve and exploit the rich knowledge of CLIP learned during large-scale pre-training. The text-grounded masks are computed by the position-wise dot product between text embedding and dense pixel-level embedding. The overall process of grounder is summarized as follows:

$$\mathbf{T} = E_t\left(\mathbf{X}^T\right) \quad \text{and} \quad \mathbf{V}^g, \mathbf{V}^d = E_v\left(\mathbf{X}^V\right), \tag{3}$$

$$\mathbf{V}^s = D_g(\mathbf{V}^d), \tag{4}$$

$$\mathbf{M}_{i,j} = \sigma\left(w \cdot \mathbf{t}_j^\top \mathbf{V}_i^s + b\right), \tag{5}$$

where $\mathbf{T} \in \mathbb{R}^{B\times C}$, $\mathbf{V}^g \in \mathbb{R}^{B\times C}$, and $\mathbf{V}^d \in \mathbb{R}^{B\times L\times C}$ are normalized text embeddings, normalized global image embeddings, and dense image features from CLIP encoders, $\mathbf{V}^s \in \mathbb{R}^{B\times C\times H\times W}$ is normalized pixel-level dense embeddings by the grounding decoder, and $\mathbf{M} \in \mathbb{R}^{B\times B\times H\times W}$ is text-grounded masks between images and texts in the batch. B, C, and L indicate a batch size, the embedding dimension size, and the number of patches, respectively. σ is a sigmoid function, and w, b are learnable scalar projection.

The generated text-grounded masks are used to extract text-grounded image embedding. By replacing the global image embedding with text-grounded image embedding in the contrastive learning framework, TCL enables the model to learn region-text alignment in an end-to-end manner. In the following section, we describe how the generated mask $\mathbf{M}$ is used for text-grounded contrastive learning.

3.3. Text-grounded Contrastive Learning

Recall that the main idea of TCL is to use text-grounded images instead of whole images, unlike conventional CL. For this purpose, we define TCL losses in three different levels—image-level, feature-level, and area-level—using the generated masks $\mathbf{M}$ for all pairs of images and texts in a batch; the detailed pseudo code to compute TCL losses is given in Appendix A. We also employ smooth regularization to further improve the quality of generated masks.

Image-level TCL loss. One intuitive way to compute the text-grounded image embedding is to encode only the regions of the image that contains the semantics of the paired text, using the image encoder. To make the whole process end-to-end trainable, we compute a differentiable masked image by multiplying the given image $\mathbf{X}_i^V$ and a binarized mask $\mathbf{M}_{i,i}^b$ obtained from the generated mask $\mathbf{M}_{i,i}$ using Gumbel-Max [14]. The masked image is then fed into the image encoder, $\tilde{\mathbf{v}}_i^g, \tilde{\mathbf{v}}_i^d = E_v\left(\mathbf{M}_{i,i}^b \cdot \mathbf{X}_i^V\right)$, to obtain text-grounded image embedding $\tilde{\mathbf{v}}_i^g$. We compute the cosine similarity matrix between text-grounded image embeddings and text embeddings in a batch by $S_{i,j}^m = {\tilde{\mathbf{v}}_i^g}^\top \mathbf{t}_j$. Finally, we use the symmetric version of InfoNCE [22, 23] to define the image-level TCL loss $\mathcal{L}_{\text{TCL}_v}$ to make the representations of positive image-text pairs similar to each other while the representations of negative pairs dissimilar based on the similarity matrix:

$$\mathcal{L}_{\text{TCL}_v} = \text{InfoNCE}\left(\mathbf{S}^m\right), \tag{6}$$

$$\text{InfoNCE}\left(\mathbf{S}\right) = -\frac{1}{2B}\sum_{i}^{B}\log\frac{\exp(S_{i,i}/\tau)}{\sum_{j}^{B}\exp(S_{i,j}/\tau)} - \frac{1}{2B}\sum_{i}^{B}\log\frac{\exp(S_{i,i}/\tau)}{\sum_{j}^{B}\exp(S_{j,i}/\tau)}, \tag{7}$$

where τ is a learnable temperature.

Feature-level TCL loss. The image-level TCL loss drives a model to generate segmentation masks for the paired texts (*i.e.*, texts of positive pairs). However, we observe that this loss alone is insufficient to prevent the model from generating masks for regions not described in the text, particularly for salient regions. This raises the need to suppress negative masks obtained from unrelated texts (*i.e.*, texts of negative pairs), but computing the image-level TCL loss for negative masks is infeasible due to the high computational cost of encoding text-grounded images. To overcome this challenge, we introduce the feature-level TCL loss, which enables the effective computation of features of the negative masks. Specifically, for pixel-level dense embeddings $\mathbf{V}_i^s \in \mathbb{R}^{C\times H\times W}$ from grounding decoder and a text embedding vector $\mathbf{t}_j$, we compute feature-level text-grounded image embedding $\mathbf{v}_{i,j}^f \in \mathbb{R}^C$ by:

$$\mathbf{v}_{i,j}^f = \frac{\sum_{h,w} M_{i,j,h,w}\cdot\mathbf{v}_{i,:,h,w}^s}{\sum_{h,w} M_{i,j,h,w}}. \tag{8}$$

Note that this feature-level embedding is computed using negative masks $\mathbf{M}_{i,j\ (i\neq j)}$, different from the image-level TCL loss. We then compute the cosine similarity $S_{i,j}^f = {\mathbf{v}_{i,j}^f}^\top \mathbf{t}_j$ between all pairs of text embeddings and feature-level text-grounded image embeddings in the batch. The feature-level TCL loss is defined as follows:

$$\mathcal{L}_{\text{TCL}_f} = \text{InfoNCE}\left(\mathbf{S}^f\right). \tag{9}$$

Area TCL loss. The image-level and feature-level TCL losses focus on generating a mask to capture the text-described region in the image. However, the model can collapse into a trivial solution with only these losses—generating a mask for the entire image instead of the desired region. To prevent this collapse, we introduce an additional objective to our TCL framework, named area TCL loss, which incorporates priors on the mask area to ensure capturing only the text-described region. To be specific, for the positive masks (masks from positive pairs) $\mathbf{M}^+$ and the negative masks (masks from negative pair) $\mathbf{M}^-$, we denote the area of positive and negative masks by $\overline{\mathbf{M}^+}$ and $\overline{\mathbf{M}^-}$, respectively. The area TCL loss is defined by L1-distance between the area priors and the expected area of each mask:

$$\mathcal{L}_{\text{area}} = \left\| p^+ - \mathbb{E}\left[\overline{\mathbf{M}^+}\right]\right\|_1 + \left\| p^- - \mathbb{E}\left[\overline{\mathbf{M}^-}\right]\right\|_1, \tag{10}$$

where p^+ and p^- are positive and negative area priors. For the negative area prior p^-, intuitively, we can expect the area of the negative masks to be 0.0. We set the positive area prior p^+ to 0.4, which is the average text-described region area measured by MaskCLIP [33] in the CC3M dataset [27].

Smooth regularization. In the image-text dataset, a text usually describes the salient object or concept in the paired image. We observe that the regions described by the text are generally smooth rather than noisy. We employ total variation (TV) regularization loss [24] to incorporate this smoothness observation in the objective. The TV loss is applied to both mask and pixel-level dense embedding:

$$\mathcal{L}_{\text{tv}} = \|\mathbf{M}\|_{\text{TV}} + \|\mathbf{V}^s\|_{\text{TV}}, \tag{11}$$

where $\|\cdot\|_{\text{TV}}$ is the anisotropic TV norm.

Final loss. Our final loss function is defined by:

$$\mathcal{L} = \underbrace{\lambda_{\text{TCL}}\mathcal{L}_{\text{TCL}} + \lambda_{\text{area}}\mathcal{L}_{\text{area}}}_{\text{TCL losses}} + \underbrace{\lambda_{\text{tv}}\mathcal{L}_{\text{tv}}}_{\text{regularization}}, \tag{12}$$

where $\mathcal{L}_{\text{TCL}} = \mathcal{L}_{\text{TCL}_v} + \mathcal{L}_{\text{TCL}_f}$, and λ_{TCL}, λ_{area}, λ_{tv} are hyperparameters to balance three losses.

3.4. Inference Pipeline

The zero-shot inference pipeline is similar to CLIP [23], except for performing pixel-level classification instead of image-level classification. Specifically, for text embeddings $\mathbf{T} \in \mathbb{R}^{N\times C}$ and a pixel-level dense embedding $\mathbf{v}^s \in \mathbb{R}^{C\times H\times W}$, text-grounded mask $\mathbf{M} \in \mathbb{R}^{N\times H\times W}$ is computed by Eq. (5), where N is the number of target classes. The final segmentation map $\mathcal{M}$ is computed by:

$$\mathcal{M}_{h,w} = \arg\max_n M_{n,h,w}. \tag{13}$$

Prompt templates such as "`a photo of a {label}.`" are used to generate text embeddings as in CLIP [23].

4. Experiments

4.1. Experiment Settings

Unified evaluation protocol. In open-world semantic segmentation, a standard evaluation protocol is not yet established. Previous studies conduct an evaluation using their own protocols such as different data processing strategies on different datasets [19, 28, 30, 33]; surprisingly, even for the same dataset, the target classes are sometimes different across studies. For a fair comparison, we present a unified evaluation protocol following the open-world scenario where prior access to the target data before evaluation is not allowed. Under this scenario, the proposed protocol prohibits dataset-specific hyperparameters or tricks, *e.g.*, class

	with background class			without background class					
Methods	VOC	Context	Object	VOC20	Context59	Stuff	City	ADE	Avg.
GroupViT (YFCC)	49.5	19.0	24.3	74.1	20.8	12.6	6.9	8.7	27.0
GroupViT (RedCaps)	50.4	18.7	27.5	79.7	23.4	15.3	11.1	9.2	29.4
MaskCLIP†	29.3	21.1	15.5	53.7	23.3	14.7	21.6	10.8	23.7
MaskCLIP	38.8	23.6	20.6	74.9	26.4	16.4	12.6	9.8	27.9
ReCo	25.1	19.9	15.7	57.7	22.3	14.8	21.1	11.2	23.5
TCL (Ours)	**55.0** **(+4.6)**	**30.4** **(+6.8)**	**31.6** **(+4.1)**	**83.2** **(+3.5)**	**33.9** **(+7.5)**	**22.4** **(+6.0)**	**24.0** **(+2.4)**	**17.1** **(+5.9)**	**37.2** **(+7.8)**

Table 1. **Zero-shot segmentation performance comparison on 8 semantic segmentation datasets.** mIoU metric is used in every experiment. We highlight **the best** and second-best results. MaskCLIP† indicates their baseline method without additional refinement techniques. The YFCC and RedCaps of GroupViT indicate their training datasets in addition to CC12M. Each dataset abbreviation stands for VOC: PASCAL VOC, Context: PASCAL Context, Object: COCO-Object, Stuff: COCO-Stuff, City: Cityscapes, ADE: ADE20K.

name expansion or rephrasing, leading to performance overestimation. For example, we observe that TCL can get significant performance gains by expanding the target class of "person" to its sub-concepts (*e.g*., man, woman, worker, rider, etc.), but the kinds of class name-based tricks are not allowed in our unified evaluation protocol because the expansion depends on the target class names. With this consideration, we evaluate models using unified class names from the default version of MMSegmentation [8] without class name-based tricks. Dense CRF [16] is not used identically due to its expensive computational cost. All other evaluation settings follow GroupViT [30], where the input image is resized to have a shorter side of 448. We employ mean intersection-over-union (mIoU) as a performance metric, which is a standard metric in semantic segmentation. While we aim to provide a fair comparison, defining fair conditions can be subjective. Thus, we provide further results and discussion on this topic in Appendix C, especially regarding dataset scale and refinement methods.

Benchmark datasets and comparison methods. We provide an extensive evaluation on widely used 8 benchmarks, categorized into two groups: (*i*) with background class (PASCAL VOC [10], PASCAL Context [21], and COCO-Object [3]), and (*ii*) without background class (PASCAL VOC20 [10], PASCAL Context59 [21], COCO-Stuff [3], Cityscapes [9], and ADE20K [32]). Note that open-world segmentation methods rely on the textual description of class names, which may require additional considerations for the background class, such as probability thresholding instead of using the "background" description as is. The datasets with background class evaluate this aspect. We compare TCL with all existing open-sourced methods, including GroupViT [30], MaskCLIP [33], and ReCo [28] under the unified protocol. We also include their variants in comparison baselines for an extensive comparison. Additional details and comparisons are given in Appendix E.

Implementation details. For the grounder, we use the CLIP ViT-B/16 model where the size of input images is 224×224 and the patch size is 16×16. Following MaskCLIP [33], we modify the last attention layer of the CLIP image encoder to acquire the dense embedding representing local semantics. The grounding decoder consists of four gated convolution blocks with two upsampling interpolations, and we use pixel-adaptive mask refinement (PAMR) [1] for mask refinement. Further details on the model architecture are provided in Appendix B. We use CC 3M and 12M datasets [5, 27] for training. The loss weights of $\lambda_{\text{TCL}} = 0.1, \lambda_{\text{area}} = 0.4, \lambda_{\text{tv}} = 1.0$ are used. We train the model with a batch size of 1024 and a learning rate of 7.5×10^{-5} for total $50,000$ iterations with $15,000$ warmup steps and cosine schedule. AdamW optimizer [20] is used with a weight decay of 0.05.

4.2. Zero-shot Transfer to Semantic Segmentation

Comparison of existing methods. We extensively compare existing open-world semantic segmentation methods in Table 1 using the proposed unified protocol, including two checkpoints of GroupViT [30] and two variants of MaskCLIP [33]. Between the existing methods, GroupViT achieves the best average performance, particularly on object-oriented datasets such as VOC, VOC20, and COCO-Object. However, its performance tends to decrease when the target dataset is dominated by stuff classes. On the other hand, MaskCLIP performs the best on stuff-oriented datasets such as Context, Context59, and COCO-Stuff, benefiting from the large-scale pre-trained CLIP model. We conjecture it benefits from leveraging a large-scale pre-trained CLIP model. The refinement techniques proposed in MaskCLIP [33] improve the average performance but significantly degrade it on Cityscapes (21.6 → 12.6), suggesting the limitation of the heuristic refinement methods. The significant performance degradation of MaskCLIP and ReCo between VOC20 and VOC may imply the need for consideration for background class.

TCL remarkably outperforms existing methods. Although the performances of existing methods vary depend-

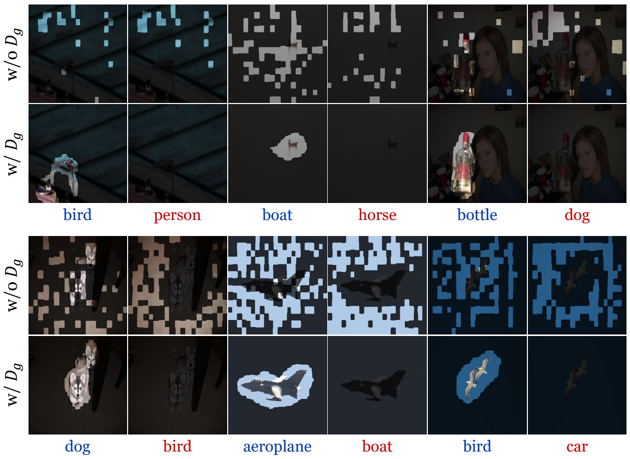


Figure 4. **Visualization of the generated text-grounded masks.** "w/o D_g" rows show the generated text-grounded masks without the grounding decoder (D_g), *i.e.*, CLIP dense features $\mathbf{V}^d$ are used instead of pixel-level dense embeddings $\mathbf{V}^s$. "w/ D_g" rows show the results with the grounding decoder. Each image is compared using both positive (blue) and negative (red) prompts. The results show that the grounder accurately and finely captures the text-described region with less noise via the grounding decoder.

ing on the characteristics of the evaluation datasets, TCL outperforms all the other methods by large margins across all datasets as shown in Table 1. These results demonstrate that our TCL framework successfully addresses the alignment-level train-test discrepancy that exists in the previous methods by learning the region-level alignment. In addition, region-level alignment learning of TCL allows our model to learn the capability to distinguish the background region in a data-driven manner, thus, our method can address the background class without any heuristic post-processing that the previous methods typically rely on.

4.3. Qualitative Results

Visualization of the generated text-grounded masks. Fig. 4 illustrates the impact of the learned grounding decoder. Since we follow MaskCLIP [33] modification, the results in "w/o D_g" rows can be regarded as the initial results of MaskCLIP before refinement. Despite the vast pretraining scale and remarkable zero-shot classification performance of CLIP [23], its grounding capability is limited because the learning objective targets image-level alignment (See "w/o D_g" rows). In contrast, the grounding decoder (D_g) learns the region-level alignment by TCL, resulting in more precise, finer, and less noisy generated masks (See "w/ D_g" rows).

Qualitative comparison. We qualitatively compare the proposed method in Fig. 5. On the PASCAL VOC dataset (Fig. 5a), we observe various types of errors in each comparison method. The grouping procedure of GroupViT [30] makes the segmentation results less noisy, but it also causes an incorrect segmentation of a large group. ReCo [28] struggles with the segmentation of background regions due to the lack of consideration about the background class. MaskCLIP [33] does not take this into account as well, but its refinement methods make the results less noisy. In addition, we present examples in the wild to show open-world segmentation capability in Fig. 5b. We collect test samples containing visual concepts not included in conventional segmentation datasets (*e.g.*, moon, sunset) or free-form texts (*e.g.*, "two women and one man with a smiling snowman"). GroupViT tends to focus on the main object of the image and regard the other objects as background, which is consistent with its good performance in object-oriented datasets. Interestingly, in this qualitative comparison in the wild, we observe ReCo consistently outperforms MaskCLIP contrary to the quantitative results. We conjecture that this is because the refinement approach of ReCo is data-driven, while the refinement approach of MaskCLIP depends on heuristic post-processing, which may not guarantee general improvement. Compared to the baselines, TCL consistently generates more precise segmentation masks. These results demonstrate that our proposed method, which learns region-level alignment, improves the segmentation quality both in the evaluation dataset and in web images in the wild. Additional qualitative results are provided in Appendix H.

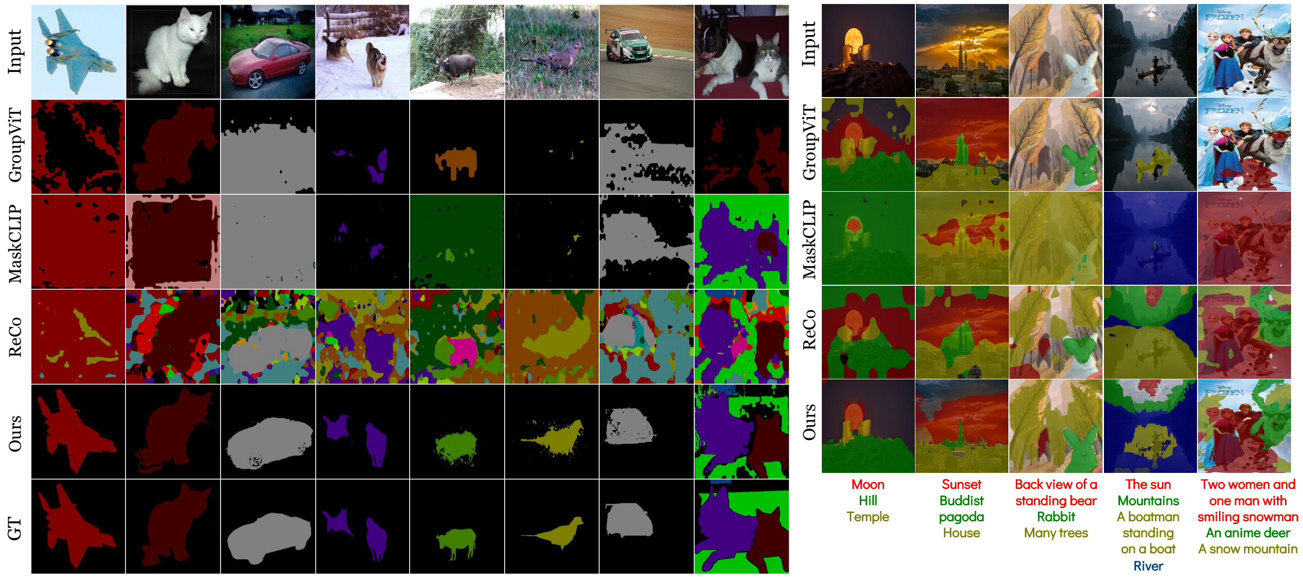
Additional analysis on failure cases and model behavior are provided in Appendices F and G, respectively.

4.4. Ablation Studies

We investigate the impact of individual components of the proposed framework by ablation studies on the training split of the PASCAL VOC20 dataset. We use a short learning schedule with a batch size of 512 for total $40,000$ iterations including $10,000$ warmup steps.

Baseline to TCL. Table 2a presents cumulative ablation studies on the grounding decoder and the TCL losses. Our initial model before training based on MaskCLIP [33] is referred to as the baseline (A), which modifies the last attention layer of the CLIP image encoder. When we add only the grounding decoder to the baseline without TCL loss (B), there is no improvement in performance. This suggests that training the decoder with the same CL loss as the pre-training (CLIP) does not enhance the localization capabilities. As shown in (C), the proposed framework becomes complete with TCL loss.

Impact of individual TCL losses. The influence of each component of the proposed TCL loss and its effect on the segmentation performance compared to the conventional CL loss are shown in Table 2b. Smooth regularization is used for all experiments in this table. The CL loss (D) is computed by applying attention pooling [23] to the dense



(a) **Examples in PASCAL VOC.**

(b) **Examples in the wild.**

Figure 5. (a) The comparison shows the error types of each method in the VOC dataset. GroupViT tends to make an error on a large group rather than noisy results. ReCo suffers from segmentation of the background region. MaskCLIP tends to fail at capturing the target area precisely. (b) shows results on the wild web images and free-form texts. Texts used as target classes are shown at the bottom of the images.

image embedding $\mathbf{V}^s$. When comparing (D) and (C), the proposed TCL loss remarkably improves the segmentation performance (61.1 → 77.4). Image-level or feature-level TCL loss (E, F) solely improves the performance significantly, and using both losses together provides further performance gain. Using CL in addition to TCL (G) does not improve performance, and it is essential to use area TCL loss in TCL framework to prevent model collapse (H), as described in Sec. 3.3. The difference between (B) and (D) is the use of smooth regularization.

Hyperparameters. Tables 2c to 2e shows the performance changes according to the variation of the loss weight hyperparameters (HPs). The first rows show the importance of each loss ($\lambda = 0.0$ cases). The absence of area TCL loss causes a significant performance drop (Table 2d), as mentioned above. Smooth regularization also significantly contributes to the final performance (Table 2e), supporting our assumption that the text-described region is smooth rather than noisy. Note that the sensitivity on HPs is about loss balancing, not about the target dataset. As an open-world segmentation method, *TCL does not require any tuning with the target dataset, including model fine-tuning and inference HPs tuning*. Once a TCL model is trained, we evaluate the model for every benchmark without any fine-tuning.

	Method	VOC20
A	Baseline	53.2
B	+ Decoder	52.3
C	+ TCL	**77.4**

(a) **Baseline to TCL.**

	TCL_v	TCL_f	$\mathcal{L}_{\text{area}}$	CL	VOC20
D				✓	61.1
E	✓		✓		74.6
F		✓	✓		76.0
C	✓	✓	✓		**77.4**
G	✓	✓	✓	✓	75.6
H	✓	✓			67.1

(b) **TCL losses.**

λ_{TCL}	VOC20
0.0	-
0.01	76.8
0.1	**77.4**
1.0	68.2

(c) $\mathcal{L}_{\text{TCL}}$

λ_{area}	VOC20
0.0	67.1
0.04	69.5
0.4	**77.4**
4.0	76.7

(d) $\mathcal{L}_{\text{area}}$

λ_{tv}	VOC20
0.0	73.8
0.1	75.2
1.0	**77.4**
10.0	70.7

(e) $\mathcal{L}_{\text{tv}}$

Table 2. **Ablation studies on TCL losses and hyperparameters.** Refinement techniques are not applied to reveal the effect of each loss function clearly. Default settings are marked in gray.

5. Conclusion

We propose a novel framework for open-world semantic segmentation with only image-text pairs, addressing the alignment-level discrepancy between training (image-text) and testing (region-text) in existing methods. In the proposed framework, we incorporate the grounding process within contrastive learning, thus allowing explicitly learning alignment between text and text-grounded regions (*i.e.*, segmentation mask). We also present a unified evaluation protocol for a fair comparison of existing methods, where TCL achieves state-of-the-art zero-shot segmentation performance on all 8 benchmarks, remarkably surpassing previous methods. We hope that this study encourages a new research direction of explicitly learning region-text alignment for open-world semantic segmentation.

References

[1] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 6, 12

[2] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset, 2022. 2, 13

[3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018. 6, 14

[4] Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. *European Conference on Computer Vision (ECCV)*, 2022. 11

[5] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2, 6

[6] 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. 15

[7] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 13

[8] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 6

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 6

[10] 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(2):303–338, 2010. 6

[11] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision*, pages 540–557. Springer, 2022. 1, 2, 3

[12] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *ICLR*, 2022. 2

[13] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *EMNLP*, 2021. 2

[14] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017. 4

[15] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790, 2021. 3

[16] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24, 2011. 6

[17] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2021. 1, 2

[18] 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. *arXiv preprint arXiv:2210.04150*, 2022. 1, 2

[19] Quande Liu, Youpeng Wen, Jianhua Han, Chunjing Xu, Hang Xu, and Xiaodan Liang. Open-world semantic segmentation via contrasting and clustering vision-language embedding. *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 5, 14

[20] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 6

[21] 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. *CVPR*, 2014. 6

[22] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3, 4

[23] 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, 5, 7, 11, 12, 13

[24] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992. 5

[25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 13

[26] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 2

[27] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018. 2, 5, 6

[28] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. *Advances in*

Neural Information Processing Systems, 2022. 1, 2, 3, 5, 6, 7, 13, 15, 16

[29] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34, 2021. 15

[30] 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*, 2022. 1, 2, 3, 5, 6, 7, 14, 15, 16

[31] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, and Jianfeng Gao. Regionclip: Region-based language-image pretraining. *CVPR*, 2022. 2, 3

[32] 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(3):302–321, 2019. 6, 14

[33] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 5, 6, 7, 11, 14, 15, 16