

Semantic-Centric Alignment for Zero-shot Panoptic Segmentation with Limited Data

Jialei Chen¹, Daisuke Deguchi¹, Dongyue Li¹, Xu Zheng²,
Seigo Ito¹, Hiroshi Murase¹, Qi Fan^{3*}

¹Nagoya University, Furo-cho, Chikusa-ku, Nagoya, Aichi, Japan.

²The Hong Kong University of Science and Technology (Guangzhou),
Guangzhou, Guangdong, China.

³Nanjing University, Nanjing, Jinagsu, China.

*Corresponding author(s). E-mail(s): fanqi@nju.edu.cn;

Contributing authors: jialeichen2021@gmail.com; ddeguchi@nagoya-u.jp;
nv.31n.2702@s.thers.ac.jp; xzheng287@connect.hkust-gz.edu.cn;
iseigo@vislab.is.i.nagoya-u.ac.jp; murase@nagoya-u.jp;

Abstract

Zero-shot segmentation has shown strong performance by generating features from semantic embeddings to adapt models to unseen classes. These generated features are typically aligned with the visual distribution of seen classes to improve generalization on visual features. However, this vision-centric alignment may overfit to seen classes due to the absence of visual data for unseen classes. To address this, we propose a semantic-centric alignment method that aligns generated features with a well-structured semantic distribution spanning all classes. First, we align vision backbone features with CLIP tokens via Vision-to-CLIP alignment. This alignment leverages CLIP’s vision-language matching capabilities to produce semantically aligned backbone features. Next, we generate synthetic features from semantic embeddings for unseen classes. These features are supervised by semantically aligned visual features and CLIP semantic embeddings to enhance visual diversity while preserving semantic consistency. Finally, we finetune the class projector using real features from seen classes and pseudo features from unseen classes to improve generalization and reduce overfitting. Our semantic-centric alignment improves zero-shot generalization by constructing a unified and semantically aligned feature space. With limited data, our method achieves state-of-the-art performance in zero-shot panoptic segmentation and readily extends to zero-shot semantic segmentation and achieving great generalization capability across datasets.

Keywords: Semantic-centric Alignment, Zero-shot Learning, Panoptic Segmentation

1 Introduction

Image segmentation [1, 2] aims to assign each pixel in an image to its corresponding class and has achieved remarkable progress in recent years [3–5]. However, most existing methods require retraining to adapt new categories, limiting their scalability and generalization. To address this, recent studies have explored approaches that can generalize to classes absent during training [6–20]. Among these research directions, zero-shot segmentation [12–14] that trains exclusively on seen classes and aims to segment unseen classes at inference has emerged as a promising approach. Despite their success, these methods often suffer from overfitting to seen classes, which limits their generalization capability to unseen classes.

To tackle this challenge, recent methods [12, 14, 21] exploit the zero-shot capabilities of vision-language models like CLIP [22] for segmentation tasks. However, these approaches typically rely on high-quality pixel-level annotations for the seen classes, which are expensive and labor-intensive to obtain, making them difficult to deploy in scenarios with limited labeled data. To bridge this gap, prior methods [5, 23, 24] typically introduce a pseudo feature generator that synthesizes visual features conditioned on semantic embeddings. The generator is commonly trained via a vision-centric strategy, where the synthesized features are aligned with real visual features extracted from the segmentation model, as illustrated in Fig. 1. The segmentation model is then fine-tuned by both real visual features from seen classes and pseudo features generated for unseen classes, facilitating generalization to novel categories without compromising performance on seen classes.

However, due to the lack of real visual features for unseen classes, the generator often fails to fully leverage the semantics embedded in unseen semantic embeddings [11, 24, 25]. As a result, the synthesized features often appear semantically misaligned and visually entangled, hindering the segmentation model’s ability to generalize to novel categories. To verify this hypothesis, we visualize the t-SNE distributions of unseen visual features generated by vision-centric methods. As shown at the top of Fig. 2, the synthesized features produced by the vision-centric method are highly entangled, leading to poor class separability and making it difficult to distinguish between unseen classes. Although some methods attempt to guide visual feature generation by relationships among semantic embeddings [5, 26], the resulting features typically remain suboptimal due to the inherent modality gap between vision and language. For example, while dog and wolf may be semantically similar, their visual appearances can differ significantly, highlighting the difficulty of aligning visual features based solely on semantic similarity. This challenge becomes even more pronounced in panoptic segmentation, which requires accurate separation of both object instances and background regions [27–29]. Therefore, Zero-shot Panoptic Segmentation (ZPS) remains largely unexplored, particularly under scenarios with limited annotated data.

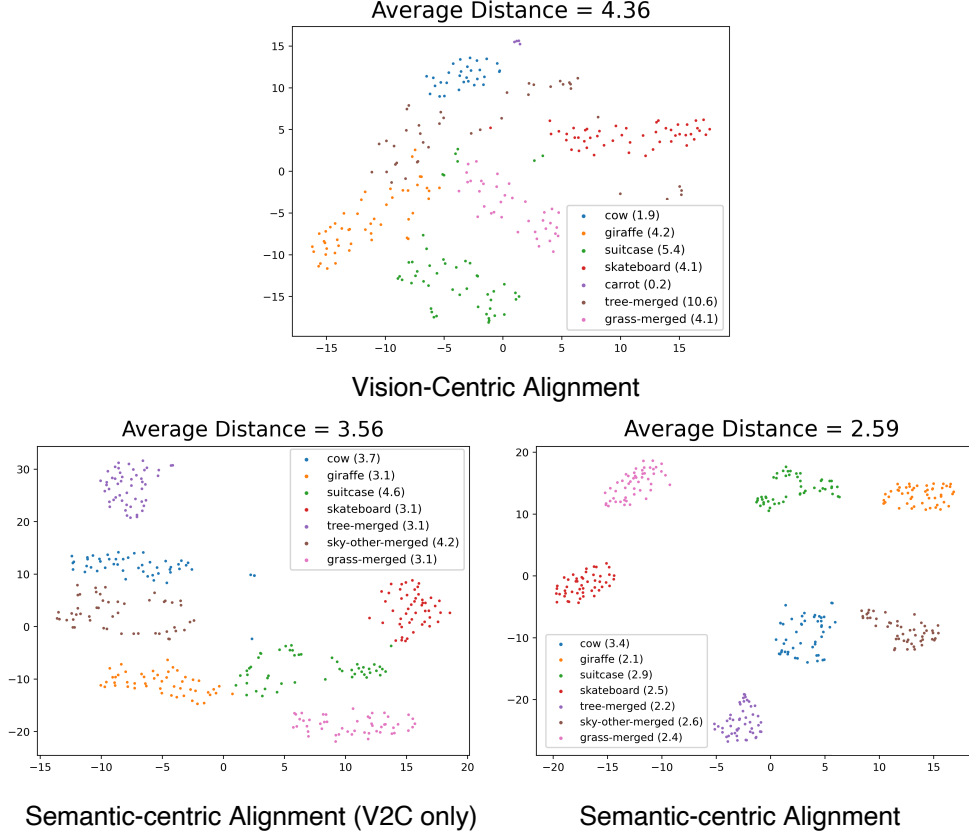


Fig. 2: Comparison of the t-SNE visualizations of features from unseen classes between vision-centric and our semantic-centric methods, where the numbers in the legend denote the pairwise distances between class centers.

space. In contrast, we propose a semantic-centric alignment approach that aligns both real visual features from the segmentation model and pseudo-generated features with CLIP’s well-structured semantic embeddings. This enhances zero-shot generalization by leveraging CLIP’s optimized semantic space and rich visual attributes, effectively bridging the modality gap and mitigating overfitting to seen classes. Unlike open-vocabulary and relaxed zero-shot segmentation methods that rely on the full dataset [12, 14, 21], our method operates under a more challenging setting, using only a subset (*e.g.*, approximately $\frac{1}{3}$ of COCO’s visual data), yet still achieves strong generalization. Additionally, our approach is simple and flexible, integrating seamlessly into existing frameworks such as MaskFormer [32] and Mask2Former [33], while achieving state-of-the-art performance on multiple zero-shot panoptic and semantic segmentation benchmarks. Extensive experiments demonstrate the effectiveness of our semantic-centric alignment. In summary, our main contributions are as follows:

- We propose a novel semantic-centric alignment method to align features with well-structured semantic embeddings for zero-shot panoptic segmentation.
- We achieve semantic-centric alignment through V2C and G2S alignment, which align real visual features from the segmentation model and pseudo-generated visual features with semantic embeddings, respectively.
- Our method achieves state-of-the-art performance on multiple zero-shot panoptic and semantic segmentation benchmarks and can be flexibly integrated into powerful segmentation models.

2 Related Works

Close-set Image Segmentation. Image segmentation is an important task in computer vision that involves assigning a category label to each individual pixel. It is typically categorized into three main types: (1) Semantic segmentation[18–20], which classifies each pixel into a predefined category but does not differentiate between distinct object instances; (2) Instance segmentation[34–36], which identifies and separates individual instances of the same category; (3) Panoptic segmentation [5, 27, 37], which unifies semantic and instance segmentation by simultaneously handling foreground objects (things) and amorphous background regions (stuff). Panoptic segmentation has garnered increasing interest from the research community due to its comprehensive formulation, which jointly addresses the challenges of semantic consistency and instance-level discrimination within a single framework.

Before the introduction of Transformers [38], many methods [27] treated panoptic segmentation as a combination of instance and semantic segmentation, solving the two tasks separately. With the introduction of DETR [37], panoptic segmentation shifted from CNN-based methods to Transformer-based methods. These approaches assign predictions to ground truth using the Hungarian matching algorithm [32, 33, 37], and unify semantic and instance segmentation without separate processing branches. In this paper, we propose a method that can handle both semantic and panoptic segmentation in the more challenging zero-shot setting with limited data.

Zero-shot classification. Zero-shot classification focuses on recognizing categories that are not seen during training [39, 40]. Due to its strong potential for real-world deployment, especially in scenarios with limited labeled data, it has attracted increasing interest from the research community. Zero-shot classification typically relies on attribute-based datasets such as AWA [41], where each class is annotated with semantic attributes (*e.g.*, a seal may be described as furry, large, and capable of swimming) [40, 42]. These attributes serve as high-level semantic cues that guide the training of a generator, which learns to synthesize pseudo visual features from attributes. The generated features are then used to further finetune the classification model, either by fine-tuning a trainable classifier [40, 42] or by directly applying a non-parametric method such as k-NN [43, 44]. While effective, such methods often overlook fine-grained visual cues that are essential for dense prediction tasks like segmentation. Moreover, zero-shot segmentation typically relies only on class names without

detailed attribute descriptions, making direct application of attribute-based classification methods infeasible. Nevertheless, the insights from zero-shot classification remain valuable and inspire the development of segmentation approaches.

Zero-shot Segmentation. Most existing zero-shot segmentation approaches can be broadly classified into two categories: projection-based methods [12, 14] and generation-based methods [5, 23, 25]. Projection-based methods [21, 45] align visual and semantic features within a shared embedding space, where labels are assigned based on similarity scores. However, since these models are often trained without any visual examples from unseen classes, they may struggle to generalize to novel categories. To address this issue, generation-based methods [11, 24] train a generator to synthesize visual features for unseen classes, and then learn a classifier using both real features from seen classes and generated features from unseen ones, effectively preserving the performance for seen classes while improving model’s generalization. However, these generation-based methods typically follow a vision-centric paradigm, where the training process focuses on matching the generated visual features to the visual distribution of seen classes, leading to the underutilization of unseen semantic embeddings. This often results in unrealistic and less discriminative features for unseen categories, reducing the model’s generalization capability. In this paper, we propose a novel semantic-centric alignment method for zero-shot segmentation, following the generation-based paradigm. Unlike existing vision-centric approaches [5, 25] that adapt generated features to the raw visual distribution of seen classes, our method learns to generate features aligned with a well-structured semantic distribution. To enhance visual diversity while preserving semantic consistency, we explicitly align the generated features with this semantically-informed visual distribution, and additionally project them back into the semantic space to ensure alignment with the original class embeddings. Through this bidirectional alignment, our method produces more realistic and discriminative features for unseen categories, leading to improved generalization in the zero-shot setting.

Compared to open-vocabulary semantic segmentation methods [15, 17, 21, 46], which typically rely on fully annotated datasets during training, our approach operates under a more constrained setting by utilizing a small subset only containing seen classes. Furthermore, unlike prior distillation-based object detection methods [47, 48] that rely on region proposals to transfer potential unseen knowledge, our method learns solely from seen class supervision without requiring any proposal generation mechanism. In addition, unlike weakly [19, 20, 49] and semi-supervised methods [7, 18], which may utilize images containing unseen categories during training, our method is trained strictly on images containing only seen classes. Finally, unlike open-vocabulary instance segmentation [16], which only segments the foreground object, our approach tackles the more challenging task of zero-shot panoptic segmentation, which requires jointly segmenting both foreground objects and background regions.

3 Method

Task Formulation. Before introducing our method, we first define the zero-shot segmentation setting. Given a dataset $\mathcal{D} = \{\mathbf{X}^i, \mathbf{Y}^i\}_{i=1}^M$ and the semantic embeddings

Table 1: Comparison of existing zero-shot settings.

	Inductive	Transductive
Soft	Images containing $\mathbf{A}_u$ can be used $\mathbf{A}_u$ can not be used	Images containing $\mathbf{A}_u$ can be used $\mathbf{A}_u$ can be used
Strict	Images containing $\mathbf{A}_u$ can not be used $\mathbf{A}_u$ can not be used	Images containing $\mathbf{A}_u$ can not be used $\mathbf{A}_u$ can be used

$\mathbf{A} \in \mathbb{R}^{N \times C}$ from a text encoder, where $\mathbf{X} \in \mathbb{R}^{H \times W \times 3}$ represents an image of height H and width W , and $\mathbf{Y}$ denotes the corresponding segmentation label. The dataset consists of M images, and $\mathbf{A}$ is divided into two disjoint subsets: seen classes $\mathbf{A}_s \in \mathbb{R}^{N_s \times C}$ and unseen classes $\mathbf{A}_u \in \mathbb{R}^{N_u \times C}$, where $\mathbf{A}_s \cap \mathbf{A}_u = \emptyset$ and $N_s + N_u = N$. Zero-shot segmentation methods can be categorized based on whether unseen semantic embeddings are used during training, resulting in **inductive** and **transductive** settings as shown in the column of Table 1. In the inductive setting, the model is trained solely with $\mathbf{A}_s$, while in the transductive setting, both $\mathbf{A}_s$ and $\mathbf{A}_u$ can be used for training. Likewise, zero-shot methods can also be categorized based on whether training images containing unseen classes can be used for training, leading to **soft** and **strict** settings as shown in the row of Table 1. In the soft setting, images containing unseen classes can be used for training, whereas in the strict setting, only images containing seen classes are permitted, with images containing $\mathbf{A}_u$ remaining restricted. Our ZPS method adopts the **strict image setting** and supports both inductive and transductive settings for semantic embeddings, unlike most existing methods that use the soft setting, thereby posing a more challenging zero-shot segmentation task.

Basic Idea and Method Overview. Training a segmentation model with strong generalization to unseen classes remains challenging due to the limited availability of visual data. A key solution is to generate realistic visual features for unseen classes that contain rich visual cues necessary for accurate segmentation. To address this, our approach adopts a structured three-step framework. **First**, we construct a segmentation model that is inherently aligned with semantic embeddings, providing a strong foundation for zero-shot generalization. **Second**, we introduce a feature generator that aligns with both the segmentation model and semantic embeddings, ensuring that the generated visual features are semantically consistent and compatible with the model. **Finally**, we fine-tune the segmentation model using a combination of real features from seen classes and synthetic features for unseen classes, further enhancing its ability to generalize beyond the training distribution.

To implement our approach, we first decompose the segmentation model into two components: a visual backbone B , which is responsible for extracting spatially dense and semantically consistent visual features from input images, and a semantic projector P , which transforms these visual features into the semantic space where classification or alignment with textual embeddings is performed. Our training pipeline consists of three sequential stages as shown in Fig. 3 : **1. Vision-to-CLIP (V2C) Alignment:** We first train the backbone to extract visual features and the semantic projector to map them into the semantic space by aligning with both CLIP semantic embeddings and visual CLS tokens. This step transfers CLIP’s vision-language alignment capabilities to the segmentation model, enabling it to extract rich visual

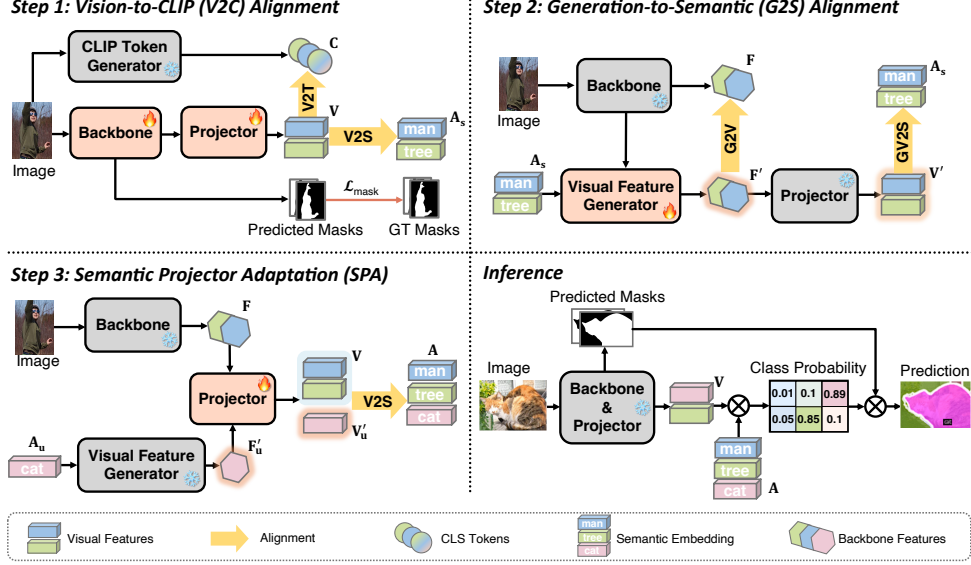


Fig. 3: The training process consists of three stages: Stage 1: We train the visual backbone and semantic projector via Vision-to-CLIP (V2C) alignment, which includes Vision-to-Token (V2T) and Vision-to-Semantic (V2S) components, aligning visual features with CLIP encoders. Stage 2: With the backbone and projector frozen, we train a generator using Generation-to-Semantic (G2S) alignment, including Generator-to-Vision (G2V) and Generated-Vision-to-Semantic (GV2S), to synthesize features that are visually realistic and semantically aligned. Stage 3: We freeze the backbone and generator, and adapt the semantic projector using both real features from seen classes and generated features from unseen classes. At inference, our model segments novel classes directly with new semantic embeddings, without requiring classifier retraining.

attributes while maintaining semantic consistency. **2. Generation-to-Semantic (G2S) Alignment:** Next, we freeze both the backbone and semantic projector and train a feature generator G . In this stage, the generator is supervised to produce synthetic features that align with the frozen backbone features and preserve semantic consistency with the input embeddings, while promoting visual diversity. **3. Semantic Projector Adaptation:** Finally, we freeze both the backbone and generator, and use them to generate real features for seen classes and synthetic features for unseen classes. These features are then used to adapt the semantic projector, enhancing its ability to generalize to unseen classes. Note that Steps 2 and 3 rely on semantic embeddings from unseen classes and are therefore applicable only in transductive settings. In contrast, the V2C alignment module supports both inductive and transductive scenarios. Given the multi-stage nature of our training pipeline, we include the pseudocode in Algorithm 1 to provide a clear overview of the procedure. Compared to previous vision-centric alignment methods, which often struggle to separate unseen class features, our semantic-centric approach generates more distinguishable pseudo-unseen features and improves generalization to unseen classes.

Algorithm 1: Training pipeline of the proposed method

Input: Training images $\mathbf{X}$ with segmentation labels $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^O$; semantic embeddings $\mathbf{A} = \{\mathbf{A}_s, \mathbf{A}_u\}$ for seen and unseen classes; frozen CLIP encoder; generator G , backbone B .

Stage 1: Vision-to-CLIP (V2C) alignment

1. Freeze G and CLIP encoders; train B and P .
2. For each batch:
 - 2.1) Extract visual features $\mathbf{F} = B(\mathbf{X})$; obtain mask predictions $\mathbf{M}$ and map $\mathbf{F}$ to the semantic space via $\mathbf{V} = P(\mathbf{F})$, forming predictions $\mathbf{Z}$.
 - 2.2) Obtain global token banks and local CLS tokens $\{\mathbf{C}_b = CLIP(\mathbf{X}), \mathbf{C}_m = CLIP(\mathbf{X} \odot \mathbf{Y})\}$.
 - 2.3) Align $\mathbf{C}_b$ and $\mathbf{C}_m$ with $\mathbf{V}$, and supervises $\mathbf{V}$ using ground-truth labels $\mathbf{Y}$ to update B and P with $\mathcal{L}_{v2c}$.

$\Downarrow$

Stage 2: Generation-to-Semantic (G2S) alignment

1. Freeze B , P , and CLIP encoders; train generator G .
2. For each batch:
 - 2.1) Split $\mathbf{F}$ into those assigned with targets $\mathbf{F}_s$ and without targets $\mathbf{F}^\varnothing$.
 - 2.2) Generate pseudo visual features $\mathbf{F}'_s$ by feeding semantic embeddings $\mathbf{A}_s$ into G .
 - 2.3) Project $\mathbf{F}'_s$ into the semantic space using the frozen P , obtaining $\mathbf{V}'_s$.
 - 2.4) Match $\mathbf{F}'_s$ with both $\mathbf{F}_s$ and $\mathbf{F}^\varnothing$, and align $\mathbf{V}'_s$ with $\mathbf{A}_s$ to update G using $\mathcal{L}_{g2s}$.

$\Downarrow$

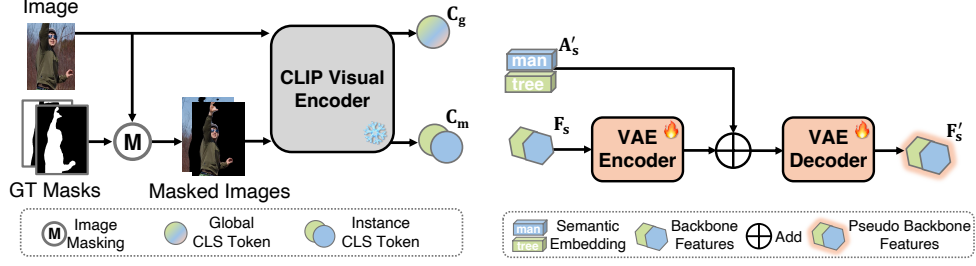
Stage 3: Semantic Projector Adaptation (SPA)

1. Freeze B , G and CLIP encoders; train projector P .
 2. For each batch:
 - 2.1) Extract features $\mathbf{F} = B(\mathbf{X}_s)$ for input images.
 - 2.2) Generate pseudo unseen features $\mathbf{F}'_u = G(\mathbf{A}_u)$ for unseen classes.
 - 2.3) Project both $\mathbf{F}$ and $\mathbf{F}'_u$ into semantic space via P for $\mathbf{V}$ and $\mathbf{V}'_u$.
 - 2.4) Align $\mathbf{V}$ with $\mathbf{A}_s$ and match $\mathbf{V}'_u$ to $\mathbf{A}$ via $\mathcal{L}_f$ to update P .
-

3.1 Vision-to-CLIP Alignment for Backbone Adaption

V2C alignment aims to align the visual features of the segmentation model with the CLIP model from two perspectives: (1) semantic embeddings from the text encoder and (2) visual features from the visual encoder. These are achieved through two components: **Vision-to-Semantic (V2S)** alignment and **Vision-to-Token (V2T)** alignment, respectively, as illustrated in the top-left of Fig. 3. Notably, V2C alignment can be applied in both **inductive** and **transductive** settings.

The goal of V2S alignment is to match visual features with their corresponding semantic embeddings, ensuring that the visual representations from the backbone are semantically consistent. Given an input image $\mathbf{X}$ and its label set $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^O$, where each $\mathbf{y}_i$ represents a non-overlapping segment and O denotes the number of unique objects, we first feed $\mathbf{X}$ into the backbone B and a transformer decoder to extract



(a) The CLIP visual encoder extracts a global token from the full image and an instance token from the masked image based on the ground-truth mask. (b) A VAE takes semantic embeddings as conditions and backbone features as input to produce pseudo visual features aligned with both visual and semantic spaces.

Fig. 4: The process of (a) CLS token and (b) synthetic visual feature generation.

object-level visual features $\mathbf{F} \in \mathbb{R}^{Q \times C}$ and predicted masks $\mathbf{M}$. The extracted features $\mathbf{F}$ are then passed through the semantic projector P to obtain projected visual features $\mathbf{V} \in \mathbb{R}^{Q \times C}$. These features, together with the masks $\mathbf{M}$, form the prediction set $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^Q$, where each $\mathbf{z}_i = \{\mathbf{v}_i, \mathbf{m}_i\}$ consists of projected features $\mathbf{v}_i \in \mathbb{R}^{1 \times C}$ and mask predictions $\mathbf{m}_i \in \mathbb{R}^{H \times W}$. Here, Q denotes the number of object features. Since the features are not spatially aligned, we adopt Hungarian matching [5, 37] to find the optimal assignment $\hat{\sigma}$ that minimizes the matching loss $\mathcal{L}_{match}$ between $\mathbf{Z}$ and $\mathbf{Y}$:

$$\hat{\sigma} = \arg \min_{\sigma} \sum_{i=1}^Q \mathcal{L}_{match}(\mathbf{Z}^{\sigma(i)}, \mathbf{Y}). \quad (1)$$

The matching loss $\mathcal{L}_{match}$ is formulated as a linear combination of focal loss [50] and DICE loss [33], where the class similarity in the focal loss is computed as $\text{sigmoid}(\mathbf{V}^\top \mathbf{A}_s)$. Finally, using the target assignment $\hat{\sigma}$, we apply $\mathcal{L}_{match}$ to align the visual features with the semantic embeddings, thereby transferring visual-semantic alignment and promoting class-agnostic mask generation. The reason to replace the widely used softmax cross-entropy loss [32, 33] with focal loss [50] is motivated by the characteristics of the Zero-shot Panoptic Segmentation (ZPS) setting. In ZPS, the model is trained solely on seen classes, while inference involves both seen and unseen classes. The softmax function enforces mutual exclusivity among classes, which biases the model toward seen categories and suppresses the activation of unseen ones during prediction. In contrast, focal loss treats each class independently, effectively relaxing this constraint. This allows the model to reduce the dominance of seen classes and better generalize to unseen categories.

While V2S ensures alignment between visual features and semantic embeddings, it may not fully exploit the generalization capabilities of CLIP. Therefore, we introduce **Vision-to-Token (V2T)** alignment as a complementary strategy to enhance V2S by aligning backbone with CLIP visual encoder. To apply V2T alignment, we generate both global and instance-level CLS tokens, as illustrated in Fig. 4a. To extract the

instance-level tokens, we first apply ground-truth masks to remove irrelevant regions. Specifically, the masked image is computed as $\mathbf{X}_m = \mathbf{Y} \odot \mathbf{X}$, where $\odot$ denotes element-wise multiplication (i.e., masking the image). We then feed both the masked image $\mathbf{X}_m$ and the original image $\mathbf{X}$ into the frozen CLIP visual encoder to obtain the instance CLS tokens $\mathbf{C}_m \in \mathbb{R}^{O \times C}$ and the global CLS token $\mathbf{C}_g \in \mathbb{R}^{1 \times C}$, respectively. To leverage these CLS tokens, we propose **target attention** and **instance alignment** for the global and instance-level tokens, respectively. In target attention, we compute the similarity $\mathbf{W} \in [0, 1]^{1 \times Q}$ between $\mathbf{C}_g$ and the visual features $\mathbf{V}$, with higher weights assigned to $\mathbf{V}_s$, which correspond to objects under $\hat{\sigma}$:

$$\mathbf{W} = \text{softmax} \left(\frac{\mathbf{C}_g^\top \mathbf{V}}{\|\mathbf{V}\|_2 \|\mathbf{C}_g\|_2} (\gamma \cdot \mathbb{1}(\mathbf{v} \in \mathbf{V}_s) + 1) \right), \quad (2)$$

where the softmax is applied along the query dimension, *i.e.*, the number of queries, and γ is a hyperparameter that controls the weight for target-relevant features. Using the feature-token similarity $\mathbf{W}$, we aggregate $\mathbf{V}$ to form an weighted combination of features: $\mathbf{V}_a = \mathbf{W}^\top \mathbf{V}$. Finally, we align $\mathbf{V}_a$ with $\mathbf{C}_g$, treating CLS tokens from other images in the same mini-batch as negative samples.

$$\mathcal{L}_g(\mathbf{V}_a, \mathbf{C}_g, \mathbf{C}_b) = -\log \frac{\exp(\mathbf{V}_a^\top \mathbf{C}_g / \tau)}{\sum_{i=1}^{B_s} \exp(\mathbf{V}_a^\top \mathbf{c}_i / \tau)}, \quad (3)$$

where $\mathbf{c}_i \in \mathbf{C}_b$, and $\mathbf{C}_b \in \mathbb{R}^{B_s \times C}$ denotes the batch of CLS tokens, with B_s representing the mini-batch size, and τ is a temperature hyperparameter controlling the loss scale. However, relying solely on global CLS tokens may limit the model’s ability to differentiate individual object instances. To address this, we introduce **instance alignment**, which directly supervises the object-level features $\mathbf{V}_s$ assigned to labeled segments under $\hat{\sigma}$, as follows:

$$\mathcal{L}_l(\mathbf{V}_s, \mathbf{C}_m) = \sum_{i=1}^O -\log \frac{\exp(\mathbf{v}_i^\top \mathbf{c}_i / \tau)}{\sum_{j \neq i}^O \exp(\mathbf{v}_i^\top \mathbf{c}_j / \tau) + \exp(\mathbf{v}_i^\top \mathbf{c}_i / \tau)}, \quad (4)$$

where $\mathbf{v} \in \mathbf{V}_s$ and $\mathbf{c} \in \mathbf{C}_m$. To recap, Vision-to-CLIP alignment consists of:

$$\mathcal{L}_{v2c} = \mathcal{L}_{match}(\mathbf{Z}^{\hat{\sigma}}, \mathbf{Y}) + \mathcal{L}_g(\mathbf{V}_a, \mathbf{C}_g, \mathbf{C}_b) + \mathcal{L}_l(\mathbf{V}_s, \mathbf{C}_m), \quad (5)$$

V2C alignment transfers CLIP’s vision-language matching capabilities to the segmentation model, ensuring that its visual features are aligned with the well-structured semantic space. This alignment provides the generator with a semantically consistent visual distribution, thereby improving generalization to unseen classes. Compared to previous distillation-based object detection methods that rely on an object proposer to identify potential regions for knowledge transfer [47, 48], our approach requires supervision from only seen classes. Moreover, unlike some weakly supervised object detection methods [49], our method does not require any additional training data.

3.2 Generation-to-Semantic Alignment for Generator Adaptation

Another challenge in zero-shot segmentation lies in the generator’s limited ability to effectively leverage semantic information. To generate unseen visual features that (1) differentiate between instances and (2) incorporate semantic information from unseen classes, we propose **G2S alignment**, as illustrated in the top-right of Fig. 3. G2S alignment consists of two components: **Generation-to-Vision (G2V)** alignment, which bridges the gap between generated and real visual features to address (1), and **Generated-Vision-to-Semantic (GV2S)** alignment, which preserves semantic consistency to address (2). Since G2S alignment requires access to semantic embeddings of unseen classes, it is applicable only in the transductive setting.

To apply G2V alignment, we first pass the input $\mathbf{X}$ through the frozen backbone to obtain real visual features $\mathbf{F}$. We then apply the optimal assignment $\hat{\sigma}$ to split $\mathbf{F}$ into $\mathbf{F}_s \in \mathbb{R}^{O \times C}$ (features assigned to labels) and $\mathbf{F}^\varnothing \in \mathbb{R}^{(Q-O) \times C}$ (features without label assignments). Next, we generate pseudo visual features $\mathbf{F}'_s$ by G , as illustrated in Fig. 4b. Formally, $\mathbf{F}_s$ is fed into a VAE encoder [51] to produce a reparameterized latent representation. The corresponding semantic embeddings $\mathbf{A}'_s$ are then added to the latent vector as conditional input. The conditioned vector is subsequently passed through a VAE decoder to generate the pseudo visual features $\mathbf{F}'_s$. Once $\mathbf{F}'_s$ is obtained, we first align it with the real visual features $\mathbf{F}_s$. Following prior work [5, 11, 23], we define $\mathcal{L}_{mmd}$ to align $\mathbf{F}'_s$ with $\mathbf{F}_s$.

$$\mathcal{L}_{mmd}(\mathbf{F}'_s, \mathbf{F}_s) = \sum_{\mathbf{f}'_s \in \mathbf{F}'_s} k(\mathbf{f}'_s, \mathbf{F}_s) + \sum_{\mathbf{f}_s \in \mathbf{F}_s} k(\mathbf{f}_s, \mathbf{F}'_s) - 2 \sum_{\mathbf{f}'_s \in \mathbf{F}'_s} \sum_{\mathbf{f}_s \in \mathbf{F}_s} k(\mathbf{f}'_s, \mathbf{f}_s), \quad (6)$$

where $k(a, b) = \exp(-\frac{1}{2\lambda^2}\|a - b\|^2)$ is a Gaussian kernel function with bandwidth λ . Due to the absence of visual data from unseen classes, the backbone captures only a partial view of the true visual distribution, leading to limited feature diversity. To mitigate this limitation, we introduce contrastive alignment to supervise $\mathbf{F}'_s$ using $\mathbf{F}_s$ and $\mathbf{F}^\varnothing$, thereby promoting the generation of more diverse visual features.

$$\mathcal{L}_{div}(\mathbf{F}'_s, \mathbf{F}_s, \mathbf{F}^\varnothing) = \sum_{i=1}^O \frac{\exp(\mathbf{f}'_i{}^\top \mathbf{f}_i / \tau)}{\sum_{j=1}^O \exp(\mathbf{f}'_i{}^\top \mathbf{f}_j / \tau) + \sum_{k=1}^{Q-O} \exp(\mathbf{f}'_i{}^\top \mathbf{f}_k^\varnothing / \tau)}, \quad (7)$$

where $\mathbf{f} \in \mathbf{F}$, $\mathbf{f}' \in \mathbf{F}'$, $\mathbf{f}^\varnothing \in \mathbf{F}^\varnothing$. By encouraging separation between different elements in $\mathbf{F}_s$, the generator learns to differentiate object instances. Simultaneously, pushing away from $\mathbf{F}^\varnothing$ promotes the generation of features that are more representative of object-containing regions. The overall loss for G2V alignment is defined as:

$$\mathcal{L}_{g2v} = \mathcal{L}_{mmd}(\mathbf{F}'_s, \mathbf{F}_s) + \mathcal{L}_{div}(\mathbf{F}'_s, \mathbf{F}_s, \mathbf{F}^\varnothing) + \mathcal{L}_{kld}. \quad (8)$$

where $\mathcal{L}_{kld}$ indicates the Kullback-Leibler divergence (KLD) loss for VAE [51]. Although G2V alignment aligns pseudo-visual features with real, semantic-aligned

visual features, the generated features may still deviate from their original semantic embeddings. To address this issue, we introduce **GV2S alignment**. Specifically, after aligning $\mathbf{F}'_s$ with $\mathbf{F}$ via G2V alignment, we pass $\mathbf{F}'_s$ through the frozen semantic projector P to obtain $\mathbf{V}'_s$. We then align $\mathbf{V}'_s$ with all seen semantic embeddings $\mathbf{A}_s$ through GV2S alignment. $\mathcal{L}_{gv2s} = \mathcal{L}_{focal}(\mathbf{V}'_s, \mathbf{A}_s)$, where $\mathcal{L}_{focal}$ denotes the same focal loss [50] used in Vision-to-CLIP alignment. It enforces semantic consistency between the generator and the backbone, ensuring that the generated visual features retain semantic information. In summary, the total loss for G2S alignment is defined as:

$$\mathcal{L}_{g2s} = \mathcal{L}_{g2v} + \mathcal{L}_{gv2s}. \quad (9)$$

In G2S alignment, the combination of G2V and GV2S enables the generator to produce visual features that are both diverse and semantically consistent, thereby facilitating more realistic generalization to unseen classes.

3.3 Semantic Projector Adaptation

Unlike most generation-based methods, which finetune a trainable classifier [5, 23] and therefore tend to suffer from limited generalization, we instead finetune only the semantic projector, which is responsible for mapping visual features into the semantic space, as shown in the bottom-left of Fig. 3. Specifically, we freeze both the backbone B and the generator G . We then randomly sample noise features $\mathbf{U} \in \mathbb{R}^{T \times C}$ from a standard Gaussian distribution and feed them into the VAE decoder [51]. In parallel, we randomly sample T unseen semantic embeddings from $\mathbf{A}_u$ to serve as conditional input for the frozen VAE decoder [51], which generates pseudo visual features for unseen classes. Finally, the generated features are passed through the trainable semantic projector to obtain the pseudo unseen semantic features $\mathbf{V}'_u$. The semantic projector P is finetuned using the following objective:

$$\mathcal{L}_f = \mathcal{L}_{focal}(\mathbf{V}, \mathbf{A}_s) + \mathcal{L}_{focal}(\mathbf{V}'_u, \mathbf{A}). \quad (10)$$

3.4 Training Objectives and Inference

Our method consists of three training stages. First, we train the backbone and semantic projector using supervision from seen classes via V2C alignment (Eq. 5). Second, we freeze both the backbone and semantic projector and train the generator with G2S alignment (Eq. 9). Finally, we use real visual features from seen classes (extracted by the backbone) and pseudo visual features from unseen classes (generated by the generator) to finetune the semantic projector (Eq. 10).

During inference, as shown in the bottom-right of Fig. 3, the input image is passed through the backbone to extract visual features and generate corresponding masks. These features are then passed through the fine-tuned semantic projector. The probability of each visual feature is computed via the inner product between visual and semantic features. Finally, the segmentation result is obtained by combining the probability scores with the predicted masks via an inner product.

Table 2: Unseen classes for ZPS and ZSS.

ZPS		ZSS
Thing	Stuff	
cow, giraffe, suitcase, frisbee, skateboard, carrot, scissors	cardboard, sky, grass, playingfield, river, road, tree	cow, giraffe, suitcase, frisbee, skateboard, carrot, scissors, cardboard, clouds, grass, playingfield, river, road, tree, concrete

4 Experiments

4.1 Experiments Setup

Dataset. We conduct experiments on the COCO dataset [52]. For ZPS, we follow the setup of PAding [5], selecting 73 out of 80 thing classes and 46 out of 53 stuff classes as seen. For ZSS, we adopt the setting of DeOP [53], where 156 out of 171 classes are considered seen. The unseen class names are listed in Table 2. Due to the large number of total classes, we only present the unseen categories in the main paper for clarity. The complete class set, including both seen and unseen classes, follows the official definitions provided by the original datasets.

Implementation Details. All experiments are conducted on 8 NVIDIA V100 GPUs. Semantic embeddings are extracted using the CLIP text encoder [22] with a ViT-B/16 backbone [38]. We follow the same text templates as prior works [5, 12, 13]. Our base model, Mask2Former [33], uses ResNet-101 [54] as the backbone and is trained for 48 epochs following the original training settings. The semantic projector is implemented as a simple MLP. The generator follows a VAE architecture with four layers of MLP-BatchNorm1d-LeakyReLU for both the encoder and decoder. For G2S alignment, the generator is trained for 12 epochs. In the inductive setting, a bias term $\mu = 2.0$ is added to the logits of unseen classes during inference. V2C alignment is applied in both inductive and transductive settings, while G2S is only used in transductive setting. For all the experiment, we set the number of pseudo unseen features T as 150.

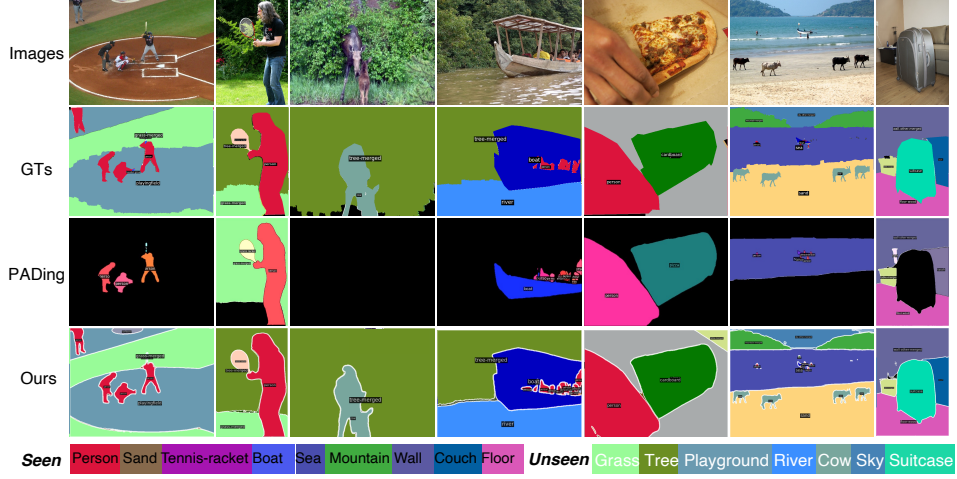
Evaluation Metric. We follow the generalized zero-shot learning (GZSL) setting [5, 12, 13], where the model is required to segment both seen and unseen classes correctly. To comprehensively evaluate performance on both seen and unseen classes, we adopt harmonic panoptic quality (hPQ) for panoptic segmentation and harmonic Intersection over Union (hIoU) for semantic segmentation, defined as: $hPQ = \frac{2 \cdot sPQ \cdot uPQ}{sPQ + uPQ}$ and $hIoU = \frac{2 \cdot sIoU \cdot uIoU}{sIoU + uIoU}$, where sPQ and uPQ denote the panoptic quality [5] for seen and unseen classes, respectively, and $sIoU$ and $uIoU$ denote the mIoU for seen and unseen classes. Inference speed is measured in frames per second (FPS) using a single NVIDIA V100.

4.2 Comparison with other methods

Comparison on Zero-Shot Panoptic Segmentation (ZPS). We first compare our method with the state-of-the-art zero-shot panoptic segmentation (ZPS) method [5] under both inductive and transductive settings. To the best of our knowledge, PAding is the only existing method specifically designed for zero-shot panoptic

Table 3: Comparison on Zero-Shot Panoptic Segmentation.

Model	Inductive			Transductive		
	hPQ	sPQ	uPQ	hPQ	sPQ	uPQ
PADing [5]	0.0	43.3	0.0	22.3	41.5	15.3
Ours	20.8	40.5	14.0	27.5	39.2	21.2

**Fig. 5:** Visualization of our proposed method on panoptic segmentation. The first row displays the input images, followed by rows showing the labels, the results from PADing [5], and the results from our method. Compared to PADing [5], our method yields more complete and coherent segmentations, particularly on unseen categories.

segmentation (ZPS). In contrast, prior works focus solely on zero-shot semantic segmentation [14, 21] or zero-shot instance segmentation [35, 36], and therefore lack a comprehensive understanding of the entire scene. The results are shown in Table 3. In the inductive setting, where no finetuning is performed on synthetic unseen features, PADing obtains 0 hPQ, indicating a complete failure to segment unseen categories holistically. Although PADing shows a slightly higher sPQ, this is largely due to overfitting on seen classes without generalization. In contrast, our method achieves a +14.0% improvement in uPQ, demonstrating significantly stronger generalization to unseen categories. Under the transductive setting, where unseen synthetic features are available during training, our method consistently outperforms PADing across most metrics, with +5.2% hPQ and +5.9% uPQ, at only a small cost in sPQ. These results confirm that our approach not only preserves segmentation quality on seen classes but also improves completeness and coherence in unseen regions.

To qualitatively demonstrate the effectiveness of the proposed method, we visualize its predictions in Fig. 5. As shown in the figure, our method can accurately segment both seen classes (*e.g.*, *boat*) and unseen classes that are often missed or misclassified by other methods (*e.g.*, *cow* in the third image). Compared to the baseline, our approach produces more coherent and complete segmentation masks.

Table 4: Comparison on Zero-shot Semantic Segmentation. The best results are noted in **bold** and the second best is noted as underline.

Method	Backbone	Embed	hIoU	sIoU	uIoU	FPS
SPNet [45]	ResNet-101 [54]	Word2vec	14.0	35.2	8.7	-
ZS3 [11]		Word2vec	15.0	34.7	9.5	-
CaGNet [24]		Word2vec	18.2	33.5	12.2	-
SIGN [23]		Word2vec	20.9	32.2	15.5	-
Zzseg [55]		CLIP	8.7	38.7	4.9	1.11
ZegFormer [13]		CLIP	27.2	37.4	21.4	6.69
ZegCLIP [12]	ViT-B [38]	CLIP	40.8	40.2	41.1	-
CLIP-RC [14]		CLIP	41.1	40.9	<u>41.6</u>	-
OTSeg+ [21]		CLIP	<u>41.5</u>	41.3	41.8	-
PADing [5]	ResNet-50 [54]	CLIP	30.7	40.4	24.8	-
Ours			36.1	40.1	32.9	24.0
DeOP [53]	ResNet-101 [54]	CLIP	38.2	38.0	38.4	4.37
Ours			42.0	42.5	<u>41.6</u>	<u>9.24</u>

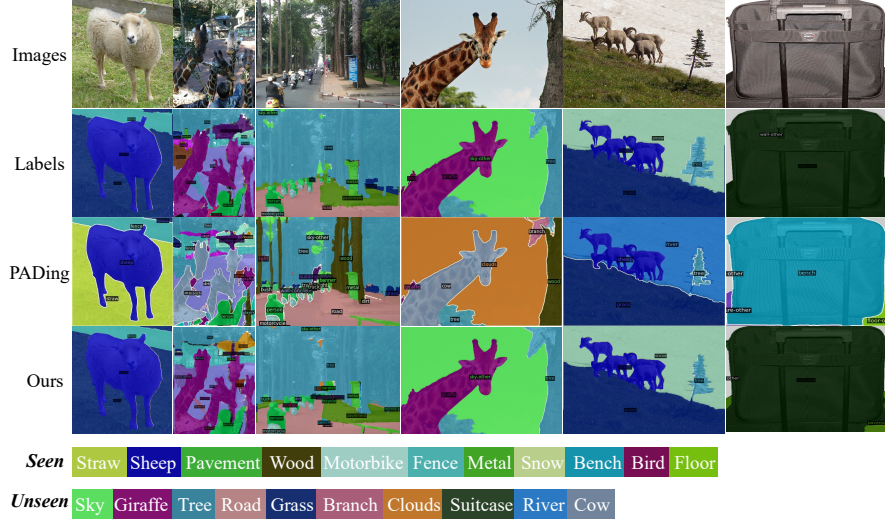


Fig. 6: Qualitative comparison of zero-shot semantic segmentation. From top to bottom: input images, ground-truth annotations, predictions by PADing [5], and predictions by our method. Compared to PADing [5], our approach yields more accurate and semantically consistent segmentation, particularly for unseen categories.

Comparison on Zero-shot Semantic Segmentation (ZSS). Table 4 presents a comparison of state-of-the-art methods for zero-shot semantic segmentation (ZSS). Focusing on two representative baselines, PADing [5] and DeOP [53], PADing achieves 30.7% hIoU, while our method improves it to 36.1%. Similarly, compared to DeOP [53], our method achieves consistent improvements across all metrics, reaching the highest hIoU and sIoU, while maintaining competitive uIoU. In terms of efficiency, FPS values are reported for selected methods. Our method achieves an inference speed of 9.24 FPS, nearly twice as fast as DeOP [53], while maintaining strong segmentation performance. Moreover, as shown in Fig. 6, we qualitatively compare our method with PADing on zero-shot semantic segmentation. Compared to PADing [5], our model

Table 5: Comparison on cross-dataset generalization. VOC indicates the PASCAL VOC [57], PC indicates the PASCAL Context [58], and A indicates the ADE20K dataset [59]. The number after the dataset indicates the class numbers in this dataset.

Method	Backbone	Training Dataset	VOC-20	PC-59	A-150	FPS
ZS3 [11]	ResNet-101 [60]	PASCAL VOC [57]	38.3	19.4	-	-
LSeg [61]		PASCAL VOC [57]	47.4	-	-	-
OpenSeg [62]		COCO-Stuff [52]	60.0	36.9	15.3	-
OpenSeg [62]		COCO-Stuff [52] + Loc. Narr. [63]	63.8	40.1	17.5	-
PixelCLIP [56]	ConvNext-B [64]	SA-1B [65]	62.9	32.7	19.4	-
GKC [26]	ResNet-101 [60]	COCO-Panoptic [52]	83.2	45.2	18.8	-
Zzseg [55]	ResNet-101 [60]	COCO-Stuff-156 [53]	88.4	47.7	20.5	1.11
DeOP [53]			91.7	48.8	22.9	4.37
Ours			93.7	49.7	20.9	9.24

Table 6: Ablations on our modules where “V2C” indicates the Vision-to-CLIP alignment, “G2S” indicates the Generation-to-Semantic alignment, and “SPA” indicates the semantic projector adaptation. Without SPA, G2S cannot contribute effectively to performance as the synthetic features for unseen classes cannot be utilized.

Method	Inductive			Transductive		
	hPQ	sPQ	uPQ	hPQ	sPQ	uPQ
Baseline	10.9	31.0	6.6	9.4	26.0	5.7
Baseline + G2S & SPA				18.1	28.7	13.2
Baseline + V2C	15.8	31.2	10.5	12.3	19.7	9.0
Baseline + V2C + G2S & SPA				22.6	31.0	17.7

produces more accurate segmentation masks, particularly for unseen categories such as giraffe and suitcase, indicating better semantic alignment and visual consistency.

Comparison on cross-dataset. Table 5 compares the cross-dataset generalization performance of various zero-shot segmentation methods on VOC-20, PC-59, and A-150, with frames per second (FPS) reported to indicate computational efficiency. Methods trained on PASCAL VOC exhibit limited generalization, while those trained on the full COCO-Stuff dataset, *i.e.*, with annotations for all classes, demonstrate better transferability. PixelCLIP [56] and GKC [26], trained on larger datasets, further improve performance. ZSseg [55] and DeOP [53] achieve strong generalization but suffer from low FPS. In contrast, our method outperforms all previous approaches, achieving 93.7% on VOC-20 [57], 49.7% on PC-59 [58], and 20.9% on A-150 [59], while maintaining the highest FPS. Notably, our model uses only partial annotations and operates, demonstrating superior accuracy and computational efficiency.

4.3 Ablation Study

For the ablation study, we employ MaskFormer [32] with a ResNet-50 [54] backbone as the baseline. All training stages are conducted for 12 epochs, focusing solely on panoptic segmentation due to its higher complexity compared to other tasks.

Ablations on the proposed methods. Table 6 presents an ablation study of the proposed modules, evaluating V2C alignment in the inductive setting and analyzing the effects of G2S alignment and SPA in the transductive setting. In the inductive setting, adding V2C significantly improves performance, increasing hPQ and uPQ,

Table 7: Ablations on target attention and instance alignment in V2T alignment.

Target Attention	Instance Alignment	Inductive			Transductive		
		hPQ	sPQ	uPQ	hPQ	sPQ	uPQ
-	-	15.4	31.2	10.2	20.3	30.1	15.3
-	✓	14.0	31.0	9.0	20.9	30.5	15.8
✓	-	14.5	31.0	9.4	22.0	31.0	17.0
✓	✓	15.8	31.2	10.5	22.6	31.0	17.7

Table 8: Ablations on G2S alignment.

$\mathcal{L}_{g2v}$	$\mathcal{L}_{gv2s}$	Transductive		
		hPQ	sPQ	uPQ
-	-	10.6	12.6	9.1
-	✓	21.9	31.1	16.9
✓	-	20.8	30.6	15.8
✓	✓	22.6	31.0	17.7

demonstrating its effectiveness in enhancing generalization to unseen classes. For the transductive setting, we evaluate the impact of G2S and SPA, which leverage synthetic features to enhance adaptation. Without V2C, adding G2S and SPA increases hPQ and uPQ, showing their ability to bridge the domain gap. When V2C, G2S, and SPA are combined, the model achieves the best results. We do not report an ablation of G2S in isolation, as it is tightly coupled with SPA. G2S is responsible for generating semantically consistent and visually diverse features for unseen classes, which are then utilized by SPA to improve model generalization. Without SPA, the model cannot leverage the features generated by G2S, leading to the same performance as the inductive setting. Conversely, applying SPA without G2S forces the model to adapt to noise-like features, resulting in potential model collapse.

Ablations on the components in V2C. We set the model without target attention and instance alignment as the baseline. Table 7 presents the ablation results. First, we add instance alignment to the baseline and observe a performance drop in the inductive setting, primarily due to reduced accuracy on unseen classes. However, in the transductive setting, instance alignment outperforms the baseline, showing improvements in both hPQ and uPQ. Next, we evaluate the effect of target attention by replacing instance alignment. While its inductive performance remains lower than the baseline, it achieves better results than instance alignment alone. In the transductive setting, target attention yields a notable improvement, achieving 1.7% higher hPQ and uPQ compared to the baseline. Finally, combining both instance alignment and target attention leads to performance gains in both settings, surpassing the baseline and further boosting hPQ and uPQ in the transductive setting by 2.3% and 2.4%.

Ablations on the components in G2S. Table 8 presents the ablation study on the proposed G2S alignment, which includes two key components: the vision-guided loss $\mathcal{L}_{g2v}$ and the semantic-guided loss $\mathcal{L}_{gv2s}$. We evaluate each component individually and in combination under the transductive setting. Without either loss, the model performs poorly across all metrics, indicating the necessity of semantic-visual alignment. Introducing $\mathcal{L}_{gv2s}$ alone yields significant gains, especially in uPQ and sPQ, demonstrating the effectiveness of leveraging visual signals to enhance semantic representation learning. On the other hand, using only $\mathcal{L}_{g2v}$ also improves all metrics, though slightly less than $\mathcal{L}_{gv2s}$ alone. When both losses are applied, the model achieves the best overall performance, validating that the bidirectional alignment between generated semantics and visual features is essential for improving generalization.

Ablation studies on the $\mathcal{L}_{div}$ in G2S. We presents an ablation study on the effectiveness of $\mathcal{L}_{div}$ in the G2S alignment and the results are shown in Table 9. Adding $\mathcal{L}_{div}$ leads to improvements in both hPQ and uPQ, while maintaining comparable

Table 9: Ablation study on $\mathcal{L}_{div}$ in G2V alignment.

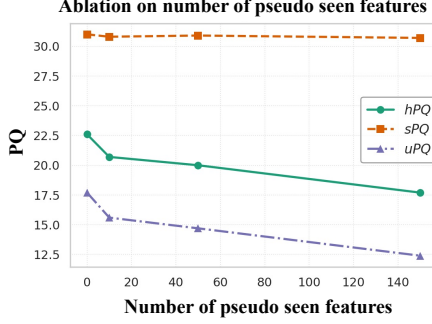
$\mathcal{L}_{div}$	Transductive		
	hPQ	sPQ	uPQ
w/o	21.9	31.1	17.0
w	22.6	31.0	17.7

Table 10: Ablation study on $\mathbf{F}^\varnothing$ in G2V alignment.

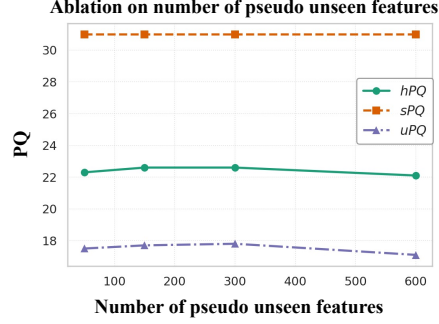
$\mathbf{F}^\varnothing$	Transductive		
	hPQ	sPQ	uPQ
w/o	22.1	31.0	17.2
w	22.6	31.0	17.7

Table 11: Ablation study on τ in G2V alignment.

τ	Transductive		
	hPQ	sPQ	uPQ
0.07	22.6	31.0	17.7
0.15	22.0	31.1	17.0
0.30	22.1	31.0	17.2



(a) Number of pseudo seen embeddings.



(b) Number of pseudo unseen embeddings.

Fig. 7: Ablations on (a) number of pseudo seen, and (b) pseudo unseen embeddings

performance in sPQ. This indicates that the diversity loss enhances the generator’s ability to produce more discriminative and diverse features, benefiting unseen class.

Ablation studies on the $\mathbf{F}^\varnothing$ in G2S. We conduct experiments to evaluate the effectiveness of $\mathbf{F}^\varnothing$, as shown in Table 10. Under the transductive setting, our approach achieves higher hPQ compared to its counterpart without $\mathbf{F}^\varnothing$, primarily due to improved uPQ performance. This improvement in uPQ is attributed to the more diverse and distributed visual features enabled by the G2V alignment, which benefits from the inclusion of background and non-target representations through $\mathbf{F}^\varnothing$.

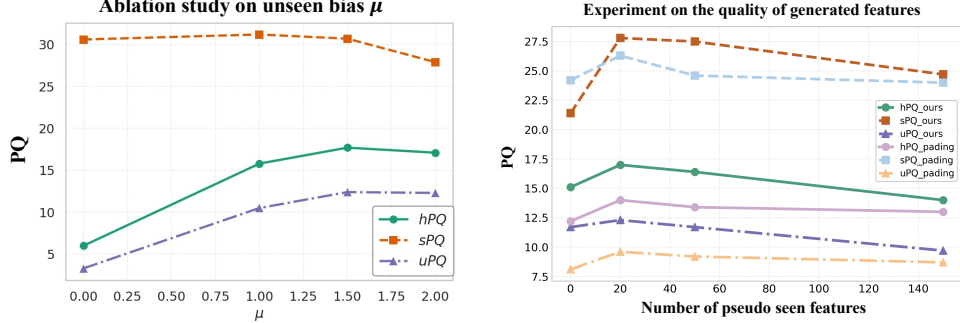
Ablation studies on the τ in G2S. Table 11 presents an ablation study on the temperature parameter τ , evaluating its impact on segmentation performance in both inductive and transductive settings. The reported metrics include hPQ, sPQ, and uPQ for each setting. In the inductive setting, the best sPQ is achieved at $\tau = 0.15$, although uPQ drops to 10.5%. In contrast, in the transductive setting, $\tau = 0.07$ yields the highest hPQ and uPQ, while $\tau = 0.15$ still provides the best sPQ. These results suggest that the choice of τ affects segmentation differently across settings.

4.4 Further Analysis

Experiments on the number of pseudo seen embeddings in SPA. As shown in Fig. 7a, we conduct an additional experiment to evaluate the impact on the pseudo seen features. In the semantic projector adaptation stage, we do not generate pseudo *seen* visual features by default. As the number of pseudo seen embeddings increases, the hPQ consistently decreases, primarily due to a drop in uPQ. Meanwhile, sPQ also shows a slight decline. These results suggest that a significant gap still exists between real and pseudo embeddings and that the generated pseudo visual features for seen and unseen classes are not sufficiently separable.

Table 12: Comparison of distance between generated and real visual features for unseen classes.

Method	Distance ↓
PADing [5]	4.05
Ours	3.10



(a) Ablation on different unseen bias μ in inference. (b) Experiments of replacing real seen features with pseudo seen features.

Fig. 8: Ablation studies on (a) different μ values during inference, and (b) experiments on generated visual features only.

Experiments on the number of pseudo unseen embeddings in SPA. We conduct experiments to investigate the effect of the noise sampling number in the semantic projector adaptation stage, as shown in Fig. 7b. Specifically, we increase the number of samples from 50 to 300 to generate more pseudo-unseen visual features. As a result, both hPQ and uPQ improve and reach their peak values of 22.6% and 17.8%, respectively. However, when the sampling number is increased beyond 300, we observe a performance degradation, with both hPQ and uPQ dropping to their lowest.

Experiments on unseen bias μ for inductive settings. We conduct experiments to investigate how increasing the logit values of unseen classes influences segmentation performance, as illustrated in Fig. 8a. This analysis is motivated by the observation that models tend to exhibit a bias toward seen classes, often leading to under-segmentation of unseen categories. As shown in the figure, when no bias term is added to the logits of unseen classes, *i.e.*, $\mu = 0$, the overall hPQ remains low, primarily due to the poor performance on uPQ. This suggests that the model struggles to activate unseen class under the default logit scale, highlighting the necessity of logit calibration to balance the confidence between seen and unseen classes.

Experiments on the quality of generated features. Table 12 presents a quantitative comparison of the feature similarity between generated and real unseen features. To perform this evaluation, we measure the similarity between real and generated features for unseen classes. Specifically, we first extract all real unseen visual features from the test set using ground-truth labels. Based on the labels of these real unseen features, we then generate corresponding pseudo unseen features using both PADing [5] and our method. Feature similarity is measured as the scaled average Euclidean distance to real unseen features, with smaller values indicating better alignment with

Table 13: Ablation study on the hyper-parameter γ in V2C alignment.

γ	Inductive			Transductive		
	hPQ	sPQ	uPQ	hPQ	sPQ	uPQ
1.5	13.3	31.4	8.4	20.2	31.2	14.9
2.0	15.8	31.2	10.5	22.6	31.0	17.7
3.0	16.0	30.5	10.8	22.4	30.1	17.9

Table 14: Ablation study on enhancement strategies in V2C alignment.

Enhancement	Inductive			Transductive		
	hPQ	sPQ	uPQ	hPQ	sPQ	uPQ
multiply	15.8	31.2	10.5	22.6	31.0	17.7
plus	14.5	31.4	9.4	22.3	31.1	17.4
minus	14.1	31.4	9.1	21.0	30.8	15.9

Table 15: Comparison between softmax and focal loss in V2C alignment.

Activation	Inductive			Transductive		
	hPQ	sPQ	uPQ	hPQ	sPQ	uPQ
Softmax	5.8	27.1	3.2	13.5	27.6	8.9
Ours	15.8	31.2	10.5	22.6	31.0	17.7

Table 16: Experiment of replacing G2S with DDPM [66] for generator training.

Method	Transductive		
	hPQ	sPQ	uPQ
DDPM [66]	18.8	30.2	13.6
Ours	22.6	31.0	17.7

the true distribution. As shown in the table, Our method results in a lower distance than PADing [5], indicating our effectiveness in generating realistic features.

To further assess the quality of the generated features, we replace the real seen features with varying numbers of generated ones while keeping the number of pseudo unseen features fixed. As shown in Fig. 8b, when no seen features are used, the overall performance is suboptimal. As the number of generated features increases, performance improves and reaches a peak, indicating that our method effectively captures the underlying visual distribution and produces realistic features. Moreover, generating an excessive number of features causes a steady performance decline due to a widened gap from the real feature distribution. Notably, our method generally outperforms PADing [5] across hPQ and uPQ, regardless of the number of generated seen features. These results demonstrate that our G2S alignment can effectively help to produce visually diverse and semantically consistent features for both seen and unseen classes, enabling strong generalization even without relying on real seen features.

Experiments on the γ in V2C alignment. We conduct experiments to investigate the impact of enhancing visual features assigned with labels, as shown in Table 13. In the inductive setting, increasing the enhancement level (controlled by γ) leads to a higher hPQ, but also causes a more significant drop in sPQ than the corresponding gain in uPQ. A similar trend is observed in the transductive setting, where the effect becomes even more pronounced. Specifically, the hPQ at $\gamma = 2.0$ is higher than that at $\gamma = 3.0$, primarily due to the substantial decrease in sPQ.

Experiments on how is γ used in Eq. 2. We investigate the effect of different operations involving γ in target attention, as shown in Table 14. Specifically, we compare multiplication ($\times(\gamma \cdot \mathbb{1}(\mathbf{v} \in \mathbf{V}_s) + 1)$), addition ($+(\gamma \cdot \mathbb{1}(\mathbf{v} \in \mathbf{V}_s) + 1)$), and subtraction ($-(\gamma \cdot \mathbb{1}(\mathbf{v} \in \mathbf{V}_s) + 1)$) applied in Eq. 2. In the inductive setting, the multiplication strategy achieves the highest hPQ. While both addition and subtraction result in slightly higher sPQ, they yield lower hPQ. In the transductive setting, multiplication and addition achieve comparable hIoU scores, both outperforming subtraction.

Comparison between softmax [32, 33] and focal loss [50]. Table 15 compares the performance of softmax [32, 33] and focal loss [50] under both inductive and transductive settings. Across all evaluation metrics, focal loss consistently outperforms

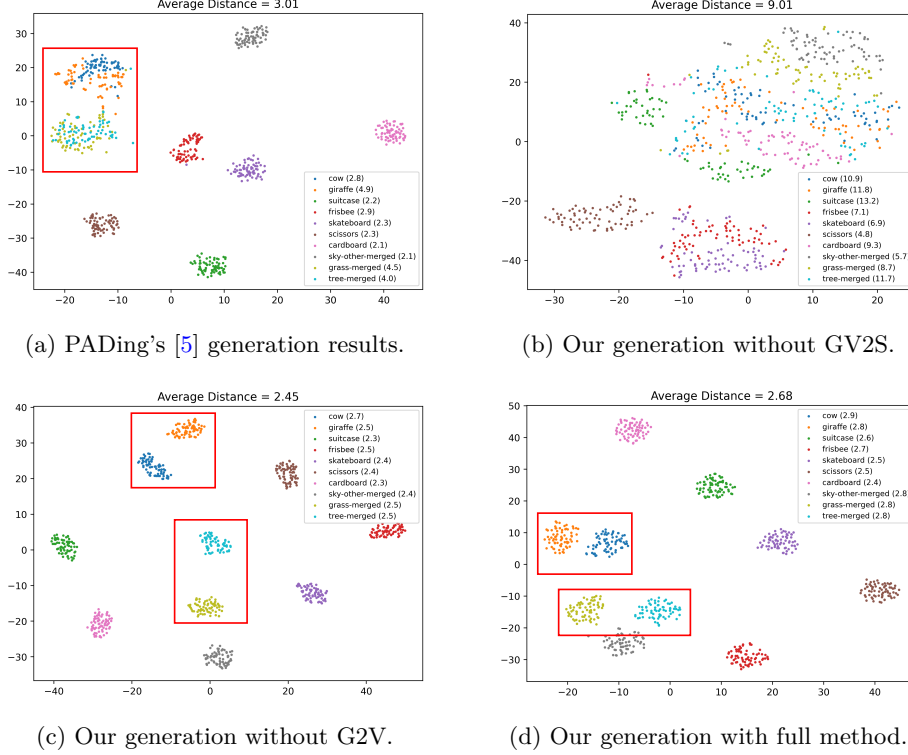
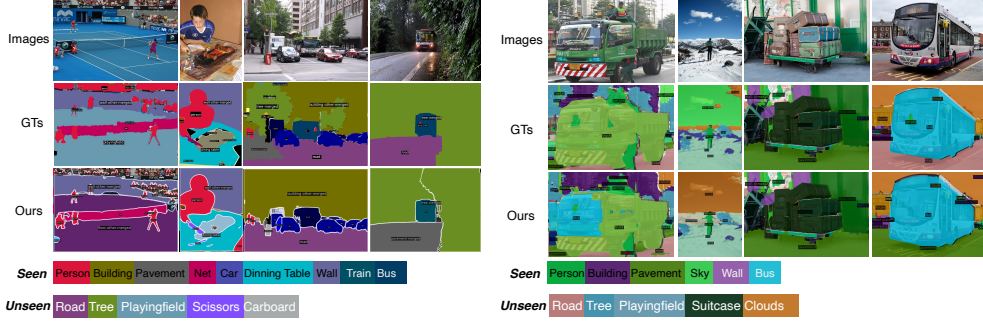


Fig. 9: T-SNE visualization of the generated visual features for unseen classes: (a) PADing generations, (b) our method without GV2S, (c) our method without G2V, and (d) our full method. The number in the title denotes the average pairwise distance among all class centers, while the numbers in the legend indicate distances between specific class pairs. Red boxes highlight the most discriminative regions. Compared to the baselines, our method better exploits semantic information of unseen classes and produces more discriminative and well-separated feature distributions.

softmax. In the transductive setting, for instance, focal loss achieves a uPQ of 23.5% compared to 18.1% with softmax. Similar improvements are observed in hPQ and sPQ, confirming the effectiveness of focal loss across different evaluation perspectives in ZPS. We attribute the performance degradation to the limitations of softmax in zero-shot settings. Softmax enforces mutual exclusivity by normalizing class probabilities, which biases the model to seen classes and suppresses any activation for unseen classes. In contrast, focal loss scores each class independently, allowing the model to assign non-zero confidence to multiple plausible categories. This reduces overfitting to seen classes and enhances the model’s generalization to unseen classes.

Comparison between DDPM [66] and our VAE-based [51] methods. Table 16 presents a comparison between our method and a representative diffusion-based model, *i.e.*, DDPM [66], under the transductive setting. In this experiment, we replace



(a) Failure cases of zero-shot panoptic segmentation. (b) Failure cases of zero-shot semantic segmentation.

Fig. 10: Failure cases on (a) zero-shot panoptic segmentation, and (b) zero-shot semantic segmentation.

the G2S alignment module in Stage 2 with a DDPM-based generator, while keeping Stages 1 and 3 unchanged. Specifically, our method achieve 3.8% hPQ and 4.1% uPQ performance gains over DDPM [66]. Although the diffusion model achieves moderate hPQ and uPQ, its generative process focuses on predicting noise rather than directly optimizing feature quality, making it difficult to enforce visual diversity and semantic consistency. In contrast, our method explicitly generates pseudo visual features and enforces both semantic consistency and visual diversity by aligning them with semantic embeddings and semantically aligned visual representations. As a result, our approach consistently outperforms the diffusion baseline across all metrics, demonstrating superior generalization to unseen classes.

Visualization of the generated class features. Fig. 9 presents t-SNE visualizations of the generated visual features for unseen classes, comparing PADing generation, our method without GV2S, our method without G2V, and our full method. The average distance between class centers is reported in the title of each plot, and the legend displays distances for individual classes. In (a) PADing generation, semantically similar classes such as *tree* and *grass* appear entangled, indicating limited class separability. (b) Our method without GV2S produces highly entangled features, resulting in an average class center distance of 9.01, significantly higher than other settings, suggesting poor semantic preservation. (c) Our method without G2V reduces the distance between features and their class centers, improving separability. However, this comes at the expense of visual diversity, limiting the model’s adaptability to unseen classes. (d) Our full method achieves the best trade-off, generating features with both strong class separation and preserved diversity. It yields a lower average distance (2.68), outperforming PADing (3.01), and demonstrates that our approach produces more structured and generalizable visual representations.

Visualization of failure cases. Fig. 10 presents representative failure cases from both zero-shot (a) panoptic segmentation and (b) semantic segmentation. While our method generally performs well, these examples reveal several remaining challenges. Specifically, the model may misclassify semantically similar object categories, such as confusing bus with train. It may also fail to capture complete object boundaries, resulting in

fragmented predictions, *e.g.*, predicting parts of a truck as bus, which highlights the need for further improvements in both semantic discrimination and spatial consistency.

5 Conclusion

In this paper, we propose a novel **semantic-centric alignment** approach for zero-shot segmentation, addressing the overfitting issues commonly observed in vision-centric methods. By aligning generated features with well-structured semantic embeddings, our method significantly enhances generalization to unseen classes. Our framework consists of two key components: **Vision-to-CLIP (V2C) alignment**, which leverages CLIP’s vision-language matching capabilities to improve semantic consistency, and **Generation-to-Semantic (G2S) alignment**, which ensures that the pseudo-unseen features maintain both visual realism and semantic coherence. By integrating real and synthetic features, our model constructs a unified feature space that supports robust zero-shot generalization. Extensive experiments demonstrate that our method achieves **state-of-the-art** performance in zero-shot panoptic segmentation and is highly adaptable to zero-shot semantic segmentation. Future work may explore more advanced visual feature generation and extend semantic-centric alignment to broader vision-language understanding tasks.

6 Acknowledgment

Support for this work was given by the Toyota Motor Corporation (TMC) and JSPS KAKENHI Grant Number 23K28164 and JST CREST Grant Number JPMJCR22D1. However, note that this article solely reflects the opinions and conclusions of its authors and not TMC or any other Toyota entity. Computations are done on “Flow” at the Information Technology Center, Nagoya University. The author Jielei Chen is sponsored by the China Scholarship Council.

7 Data availability statements

The datasets used in this study are all publicly available. Specifically: The COCO-Stuff dataset can be accessed at <https://github.com/nightrone/cocostuff>. The ADE20K dataset is available at <http://groups.csail.mit.edu/vision/datasets/ADE20K>. The PASCAL VOC 2012 dataset is available at <http://host.robots.ox.ac.uk/pascal/VOC/voc2012>. The PASCAL Context dataset can be obtained from <https://cs.stanford.edu/~roozbeh/pascal-context/>.

References

- [1] Chen, J., Deguchi, D., Zhang, C., Zheng, X., Murase, H.: Frozen is better than learning: A new design of prototype-based classifier for semantic segmentation. PR (2024)

- [2] Zheng, X., Luo, Y., Zhou, P., Wang, L.: Distilling efficient vision transformers from cnns for semantic segmentation. PR (2025)
- [3] Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. NeurIPS (2021)
- [4] Guo, M.-H., Lu, C.-Z., Hou, Q., Liu, Z., Cheng, M.-M., Hu, S.-M.: Segnext: Rethinking convolutional attention design for semantic segmentation. NeurIPS (2022)
- [5] He, S., Ding, H., Jiang, W.: Primitive generation and semantic-related alignment for universal zero-shot segmentation. In: CVPR (2023)
- [6] Bi, H., Feng, Y., Mao, Y., Pei, J., Diao, W., Wang, H., Sun, X.: Agmtr: Agent mining transformer for few-shot segmentation in remote sensing. IJCV (2024)
- [7] Xie, H., Wang, C., Zhao, J., Liu, Y., Dan, J., Fu, C., Sun, B.: PrcI: Probabilistic representation contrastive learning for semi-supervised semantic segmentation. IJCV (2024)
- [8] Zhu, L., Wang, X., Feng, J., Cheng, T., Li, Y., Jiang, B., Zhang, D., Han, J.: Weakclip: Adapting clip for weakly-supervised semantic segmentation. IJCV (2024)
- [9] Zeng, Q., Lu, Z., Xie, Y., Xia, Y.: Pick: Predict and mask for semi-supervised medical image segmentation. IJCV (2025)
- [10] Siam, M.: Temporal transductive inference for few-shot video object segmentation. IJCV (2025)
- [11] Bucher, M., Vu, T.-H., Cord, M., Pérez, P.: Zero-shot semantic segmentation. NeurIPS (2019)
- [12] Zhou, Z., Lei, Y., Zhang, B., Liu, L., Liu, Y.: Zegclip: Towards adapting clip for zero-shot semantic segmentation. In: CVPR (2023)
- [13] Ding, J., Xue, N., Xia, G.-S., Dai, D.: Decoupling zero-shot semantic segmentation. In: CVPR (2022)
- [14] Zhang, Y., Guo, M.-H., Wang, M., Hu, S.-M.: Exploring regional clues in clip for zero-shot semantic segmentation. In: CVPR, pp. 3270–3280 (2024)
- [15] Yuan, H., Li, X., Zhou, C., Li, Y., Chen, K., Loy, C.C.: Open-vocabulary sam: Segment and recognize twenty-thousand classes interactively. In: ECCV (2024)
- [16] Li, F., Zhang, H., Sun, P., Zou, X., Liu, S., Li, C., Yang, J., Zhang, L., Gao, J.: Segment and recognize anything at any granularity. In: ECCV (2024)

- [17] Li, X., Yuan, H., Li, W., Ding, H., Wu, S., Zhang, W., Li, Y., Chen, K., Loy, C.C.: Omg-seg: Is one model good enough for all segmentation? In: CVPR (2024)
- [18] Tang, F., Xu, Z., Hu, M., Li, W., Xia, P., Zhong, Y., Wu, H., Su, J., Ge, Z.: Neighbor does matter: Density-aware contrastive learning for medical semi-supervised segmentation. In: AAAI (2025)
- [19] Tang, F., Xu, Z., Qu, Z., Feng, W., Jiang, X., Ge, Z.: Hunting attributes: Context prototype-aware learning for weakly supervised semantic segmentation. In: CVPR (2024)
- [20] Xu, Z., Tang, F., Chen, Z., Su, Y., Zhao, Z., Zhang, G., Su, J., Ge, Z.: Toward modality gap: Vision prototype learning for weakly-supervised semantic segmentation with clip. In: AAAI (2025)
- [21] Kim, K., Oh, Y., Ye, J.C.: Otseg: Multi-prompt sinkhorn attention for zero-shot semantic segmentation. In: ECCV (2024)
- [22] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., *et al.*: Learning transferable visual models from natural language supervision. In: IMCL (2021)
- [23] Cheng, J., Nandi, S., Natarajan, P., Abd-Almageed, W.: Sign: Spatial-information incorporated generative network for generalized zero-shot semantic segmentation. In: ICCV (2021)
- [24] Gu, Z., Zhou, S., Niu, L., Zhao, Z., Zhang, L.: Context-aware feature generation for zero-shot semantic segmentation. In: MM (2020)
- [25] Baek, D., Oh, Y., Ham, B.: Exploiting a joint embedding space for generalized zero-shot semantic segmentation. In: ICCV (2021)
- [26] Han, K., Liu, Y., Liew, J.H., Ding, H., Liu, J., Wang, Y., Tang, Y., Yang, Y., Feng, J., Zhao, Y., *et al.*: Global knowledge calibration for fast open-vocabulary segmentation. In: ICCV, pp. 797–807 (2023)
- [27] Li, Y., Zhao, H., Qi, X., Wang, L., Li, Z., Sun, J., Jia, J.: Fully convolutional networks for panoptic segmentation. In: CVPR, pp. 214–223 (2021)
- [28] Ding, Z., Wang, J., Tu, Z.: Open-vocabulary panoptic segmentation with maskclip. arXiv preprint arXiv:2208.08984 (2022)
- [29] Xu, X., Xiong, T., Ding, Z., Tu, Z.: Masqclip for open-vocabulary universal image segmentation. In: ICCV (2023)
- [30] He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020)

- [31] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: ICCV (2021)
- [32] Cheng, B., Schwing, A., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. NeurIPS (2021)
- [33] Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: CVPR (2022)
- [34] Tian, Z., Shen, C., Chen, H.: Conditional convolutions for instance segmentation. In: ECCV (2020)
- [35] Zheng, Y., Wu, J., Qin, Y., Zhang, F., Cui, L.: Zero-shot instance segmentation. In: CVPR (2021)
- [36] He, S., Ding, H., Jiang, W.: Semantic-promoted debiasing and background disambiguation for zero-shot instance segmentation. In: CVPR (2023)
- [37] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision, pp. 213–229 (2020). Springer
- [38] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
- [39] Quan, Z., Chen, J., Deguchi, D., Sun, J., Zhang, C., Li, Y., Murase, H.: Semantic matters: A constrained approach for zero-shot video action recognition. Pattern Recognition **162** (2025)
- [40] Ge, J., Xie, H., Li, P., Xie, L., Min, S., Zhang, Y.: Towards discriminative feature generation for generalized zero-shot learning. TMM (2024)
- [41] Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR (2009). IEEE
- [42] Hou, W., Chen, S., Chen, S., Hong, Z., Wang, Y., Feng, X., Khan, S., Khan, F.S., You, X.: Visual-augmented dynamic semantic prototype for generative zero-shot learning. In: CVPR (2024)
- [43] Han, Z., Fu, Z., Yang, J.: Learning the redundancy-free features for generalized zero-shot object recognition. In: CVPR (2020)
- [44] Chen, Z., Li, J., Luo, Y., Huang, Z., Yang, Y.: Canzsl: Cycle-consistent adversarial networks for zero-shot learning from natural language. In: WACV (2020)
- [45] Xian, Y., Choudhury, S., He, Y., Schiele, B., Akata, Z.: Semantic projection

- network for zero-and few-label semantic segmentation. In: CVPR (2019)
- [46] Xu, M., Zhang, Z., Wei, F., Hu, H., Bai, X.: Side adapter network for open-vocabulary semantic segmentation. In: CVPR (2023)
 - [47] Gu, X., Lin, T.-Y., Kuo, W., Cui, Y.: Open-vocabulary object detection via vision and language knowledge distillation. In: ICLR (2022)
 - [48] Du, Y., Wei, F., Zhang, Z., Shi, M., Gao, Y., Li, G.: Learning to prompt for open-vocabulary object detection with vision-language model. In: CVPR (2022)
 - [49] Zhou, X., Girdhar, R., Joulin, A., Krähenbühl, P., Misra, I.: Detecting twenty-thousand classes using image-level supervision. In: ECCV (2022)
 - [50] Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV (2017)
 - [51] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
 - [52] Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: CVPR (2018)
 - [53] Han, C., Zhong, Y., Li, D., Han, K., Ma, L.: Open-vocabulary semantic segmentation with decoupled one-pass network. In: ICCV (2023)
 - [54] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
 - [55] Xu, M., Zhang, Z., Wei, F., Lin, Y., Cao, Y., Hu, H., Bai, X.: A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In: ECCV (2022)
 - [56] Shin, H., Kim, C., Hong, S., Cho, S., Arnab, A., Seo, P.H., Kim, S.: Towards open-vocabulary semantic segmentation without semantic labels. NeurIPS (2024)
 - [57] Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. IJCV (2015)
 - [58] Mottaghi, R., Chen, X., Liu, X., Cho, N.-G., Lee, S.-W., Fidler, S., Urtasun, R., Yuille, A.: The role of context for object detection and semantic segmentation in the wild. In: CVPR (2014)
 - [59] Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: CVPR (2017)
 - [60] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder

- with atrous separable convolution for semantic image segmentation. In: ECCV (2018)
- [61] Li, B., Weinberger, K.Q., Belongie, S., Koltun, V., Ranftl, R.: Language-driven semantic segmentation. In: International Conference on Learning Representations (2022)
- [62] Ghiasi, G., Gu, X., Cui, Y., Lin, T.-Y.: Scaling open-vocabulary image segmentation with image-level labels. In: ECCV (2022)
- [63] Pont-Tuset, J., Uijlings, J., Changpinyo, S., Soricut, R., Ferrari, V.: Connecting vision and language with localized narratives. In: ECCV (2020)
- [64] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: CVPR (2022)
- [65] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., *et al.*: Segment anything. In: ICCV (2023)
- [66] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. NeurIPS (2020)