Efficiency of Training Costs in Semantic Segmentation

November 2024

Yasufumi Kawano

A thesis for the degree of Ph.D. in Engineering

Efficiency of Training Costs in Semantic Segmentation

November 2024

Graduate School of Science and Technology
Keio University

Yasufumi Kawano

論文要旨

セマンティックセグメンテーションとは，画像中の全画素にクラスラベルを付与するタスクである．セマンティックセグメンテーションは都市画像のような複雑な画像でも高い精度で認識でき，ロボティクスや衛星画像分析など多様な分野で応用されている．しかし，その学習はピクセルレベルのアノテーションと大規模な計算リソースが必要である．また，教師付き学習は事前定義されたカテゴリセットに依存しているため，予測時に珍しいクラスや完全に新しいクラスを検出することは難しいといった課題がある．本研究では，Active Learning や教師無しドメイン適応，オープンボキャブラリセマンティックセグメンテーションといったタスクにおいて新たな手法を提案し，これらの課題に対処する．第 1 章では，セマンティックセグメンテーションの位置付けと課題を提示し，本研究の問題定義と研究目的について述べる．第 2 章では，セマンティックセグメンテーションにおける学習効率化に関する既存手法を概観する．第 3 章では，Active Learning を用いた不確実性を活用する手法を導入し，効率的なデータアノテーションを実現する．モデルの出力から計算できる不確実性指標を用いた Active Learning 手法を提案することによりよりアノテーションコストを低下させ，従来手法よりも精度と時間の面でコスト削減が可能となる．第 4 章では，モデルの出力から計算できる不確実性を loss として最小化する新たな教師無しドメイン適応手法によりアノテーションを不要にしつつ，よりターゲットドメインに頑健なモデルになり最先端の結果を実現する．第 5 章では，学習済み Diffusion モデルの内部特徴を分析し，セマンティックセグメンテーションにおける応用の可能性を探り，追加の学習やアノテーションなしで効果的なセグメンテーションを実現する新しい手法を提案して，従来手法と比較して，様々なドメインの画像に対してオープンボキャブラリセマンティックセグメンテーション性能における優位性を実証する．第 6 章では，外部データベースからクラスタのカテゴリを検索することで，事前に定義されたクラスに依存せず，追加学習，アノテーション，ガイダンス不要なオープンボキャブラリセマンティックセグメンテーションを実現する新しいアプローチを提案し，従来手法と比較して，様々なドメインの画像に対してオープンボキャブラリセマンティックセグメンテーション性能における優位性を実証する．第 7 章では，本研究の成果を

まとめ，今後の課題について議論する．本論文ではセマンティックセグメンテーションにおけるアノテーションや計算リソースなどの学習コストを効率化するための新しい手法を複数のアプローチで開拓する．本研究で提案する方法は，実際のアプリケーションにおいても有効であり，特にアノテーションのコストが高い状況や専門知識が求められる状況において，大きな利益をもたらすことが期待できる．

Abstract

Semantic segmentation is the task of assigning a class label to every pixel in an image. It can accurately recognize complex images, such as urban scenes, and is applied in various fields including robotics and satellite image analysis. However, training for semantic segmentation requires pixel-level annotations and extensive computational resources. Furthermore, supervised learning depends on predefined category sets, making detecting rare or entirely new classes challenging during prediction. This thesis proposes new methods to address tasks such as active learning, unsupervised domain adaptation, and open vocabulary semantic segmentation. Chapter 1 presents the status and issues of semantic segmentation and defines the problem and objectives of this research. Chapter 2 reviews existing methods for improving learning efficiency in semantic segmentation. Chapter 3 introduces a model that can reduce annotation costs more effectively in terms of accuracy and time than traditional methods by active learning using an uncertainty metric calculated from the model's output. Chapter 4 proposes a new unsupervised domain adaptation method to reduce annotation costs and achieves state-of-the-art results by making the model more robust to the target domain by minimizing the uncertainty calculated from the model's output as a loss. Chapter 5 analyzes the internal features of pre-trained diffusion models, explores their applicability in semantic segmentation, and proposes a new method that enables effective segmentation without additional training or annotations, demonstrating the superiority of this method in open vocabulary semantic segmentation performance compared to traditional methods. Chapter 6 proposes a new approach that uses external database searches to identify cluster categories, enabling open vocabulary semantic segmentation without dependence on predefined classes, additional learning, annotations, or guidance, and to demonstrate the superiority of this method in open vocabulary semantic segmentation performance compared to traditional methods. Chapter 7 summarizes the results of this work and discusses future work. This thesis develops several new approaches to optimize the learn-

ing costs associated with semantic segmentation, such as annotations and computational resources. The methods proposed in this study are effective in practical applications and are particularly beneficial in situations where annotation costs are high or where specialized knowledge is required.

Contents

1 Introduction **1**

1.1 Semantic Segmentation 1

1.2 Problem Definition and Research Objectives 4

1.2.1 Efficiency of Annotation through Active Learning . . . 5

1.2.2 Efficiency of Annotation through Unsupervised Domain Adaptation 5

1.2.3 Applying Pre-trained Diffusion Models to Semantic Segmentation 7

1.2.4 Guidance-free Open Vocabulary Semantic Segmentation 8

1.3 Publication List 9

1.4 Structure of This Thesis 10

2 Related Works **11**

2.1 Active Learning 11

2.2 Unsupervised Domain Adaptation 13

2.3 Unsupervised Learning and Open-Vocabulary Semantic Segmentation 14

2.3.1 Generative Models for Segmentation 15

2.4 Zero Guidance Semantic Segmentation 16

2.5 Text Retrieval from CLIP Embedding 16

3 Efficiency of Annotation through Active Learning **18**

3.1 Overview of the Proposed Method 19

3.2 Uncertainty Indicator Generator 21

3.3 Experiment Overview of Active Learning 22

3.4 Experiment Results of Active Learning 24

3.4.1 Analysis of Selected Samples 30
3.4.2 Analysis of Uncertainty Indicator 30
3.4.3 Comparison of Model Computational Complexity . . . 32
3.5 Limitation 33
3.6 Conclusion 34

4 Efficiency of Annotation through Unsupervised Domain Adaptation 35
4.1 Overview of Proposed Unsupervised Domain Adaptation . . . 37
4.2 Uncertainty Loss 38
4.3 Experiment 39
4.4 Conclusion 40

5 Applying Pre-trained Diffusion Models to Semantic Segmentation 42
5.1 Overview of the Proposed Method 43
5.2 Method 45
5.2.1 Diffusion Model Architecture 46
5.2.2 MaskDiffusion 47
5.2.3 Unsupervised MaskDiffusion 50
5.2.4 Main Results 51
5.2.5 Internal Feature 54
5.2.6 Open Vocabulary Segmentation on Web-Crawled Images 56
5.2.7 Ablation Study 58
5.3 Limitation 61
5.4 Conclusion 62

6 Guidance-free Open-Vocabulary Semantic Segmentation 63
6.1 Overview of the Proposed Method 64
6.2 Method 65
6.2.1 Segment Candidates with DINO 66
6.2.2 Representative Segment Embeddings with CLIP 67
6.2.3 Segment Category Retrieval 69
6.3 Experiment 70

6.3.1 Implementation Details 72
6.3.2 Main Results 72
6.3.3 Open Vocabulary Segmentation on Web-Crawled Images 74
6.3.4 Ablation Study 76
6.4 Limitation 80
6.5 Conclusion 81

7 Conclusion **82**

Acknowledgement **84**

References **85**

List of Figures

1.1 **Input image and outputs for each image recognition task.** In general classification tasks, the output is the class name of the object mainly featured in the image. In object detection, the output includes both the location in the image and the class name of the object. In semantic segmentation, each pixel is assigned and outputs an object class. In supervised learning, the class names for these tasks are chosen from a predefined set of multiple candidates, regardless of the task. 2

1.2 **Typical pool-based active learning cycle.** In each iteration, a task model is trained using labeled images. After training, unlabeled images are selected based on the model's inference and then labeled by an oracle. This cycle is repeated until the model's performance meets the user's requirements or the budget is exhausted. 4

1.3 **Overview of unsupervised domain adaptation.** Training is performed using synthetic data that does not require annotation, and evaluation is conducted on real images. 6

1.4 **Overview of applying pre-trained diffusion models.** Capable of segmenting a wide range of classes, including proper nouns, in an open vocabulary without the need for additional training. 7

1.5 **Comparison between segmentation methods that do not require the input of class names (guidance) and conventional methods.** Our approach eliminates the need for additional training and guidance, enabling the segmentation of a wide range of classes, including proper nouns, in an open vocabulary. 8

3.1 **Model of our proposed method.** The red arrows show the path for the unlabeled images, and the blue arrows show the path for the labeled images. The labeled images are used to train the task model. Using the output of the trained task model, the uncertainty indicator generator (UIG) outputs the uncertainty indicator for unlabeled images. Unlabeled images with high uncertainty are labeled first. 19

3.2 **Comparison of methods on Cityscapes dataset [1].** . . . 26

3.3 **Improvements over random sampling by each method on Cityscapes dataset [1].** 26

3.4 **Qualitative evaluation on Cityscapes datset [1].** The left image shows the ground truth labels, the center image shows the results of training with all images, and the right image shows the results of training with 50% of the data sampled using the proposed method. Using the proposed method, nearly the same outcomes are achieved with only half the amount of annotations as when training with all images. . . . 29

3.5 **Visual results of active learning methods.** SRAAL [2], our UIG) on a toy example. The red points represent classes with 0, and blue points with 1. The black faction marks are the samples selected by each method. The oracle decision boundary is shown as a black dashed line. We trained on 500 samples and selected 30 samples from the 500 unlabeled samples. It can be seen that the proposed method is able to select samples near the decision boundary better than SRAAL. 30

3.6 **Relationships between actual loss (X-axis) and uncertainty metric (Y-axis) for SRAAL [2].** 31
3.7 **Relationships between actual loss (X-axis) and uncertainty metric (Y-axis) for entropy.** 32
3.8 **Relationships between actual loss (X-axis) and uncertainty metric (Y-axis) for the proposed method.** 33

4.1 **Overview of unsupervised domain adaptation.** Training is performed on synthetic data that does not require annotation, and evaluation is performed on real images 36
4.2 **Model of the proposed method in unsupervised domain adaptation.** Red arrows indicate the path of target data, while blue arrows represent the path of source data. . . . 37
4.3 **Results of qualitative comparison in unsupervised domain adaptation.** 40

5.1 **Comparison of MaskDiffusion with previous method MaskCLIP [3] on a Cityscapes [1] image.** k-means clustering on the internal features of the Diffusion U-Net (a) shows that each determined cluster roughly partitions the image according to some classes, indicating that the semantic information is well preserved. MaskDiffusion (b) yields well-partitioned segments consistent with the shape of the object and exhibits minimal noise. In comparison, MaskCLIP [3] (c) results in smaller and noisy segments. 43
5.2 **High-level overview of our MaskDiffusion architecture.** MaskDiffusion uses a frozen pre-trained diffusion model. The UNet is given latent images compressed by a VAE as well as text prompts embedded by CLIP. The prompts are the names of all the potential classes to be segmented. The output of each layer of the U-Net is extracted as a concatenated internal feature $\mathbf{f}$ and a cross-attention map, which are subsequently post-processed into a segmentation image. 47

5.3 **Overview of the post processing step.** This post-processing module mathematically transforms the internal features **f** using the cross-attention map. First, a representative **f** is computed for each category through a weighted average of **f** based on the values of the cross-attention map. In the next step, we determine the semantic segmentation result by evaluating the cosine similarity between **f** and the representative **f** for each category and then assign each **f** to the category that has the closest similarity. 48

5.4 **Overview of Unsupervised MaskDiffusion architecture.** we employ spectral clustering [4] to include the spatial relationships of the internal features in the segmentation process. 50

5.5 **Qualitative results.** Images (a) to (c) depict scenes from PascalVOC [5], images (d) and (e) represent scenes from Cityscapes [1], and image (f) shows scene from Potsdam [6]. We compare MaskCLIP [3], GEM [7], and our proposed MaskDiffusion. MaskCLIP [3] and GEM [7] provide fragmented and noisy segmentations, whereas MaskDiffusion exhibits a more cohesive and accurate segmentation for each object. 52

5.6 **Overview result of Unsupervised MaskDiffusion, k-means on internal features and previous method STEGO [8] with COCO-Stuff [9] and Cityscapes [1].** It demonstrates that Unsupervised MaskDiffusion produces qualitatively cleaner results. 55

5.7 **Open-vocabulary segmentation results.** In (a) we test on a general image, (b) and (c) show images generated by Stable Diffusion, and (d) and (e) are images featuring specific proper nouns. The corresponding prompt for each class is displayed below the respective image. The successful segmentation of challenging concepts, rare classes, and proper nouns highlights the effectiveness of MaskDiffusion in handling diverse segmentation tasks. Note that these word candidates function properly even with similar alternative words. 57

5.8 **Open Vocabulary Segmentation with similar words.** This demonstrates that Open Vocabulary Segmentation can accurately segment even when the given class candidates are similar words. 58
5.9 **Ablation study on internal feature size.** Qualitative results of internal features k-means show that earlier layers of the UNet (128, 64) do not provide an appropriate segmentation, whereas the inner layers (32, 16, 8) produce semantic meaningful results. 59

6.1 **High-level overview of our TAG architecture.** Our TAG can partition images into semantic segments and label each segment with open-vocabulary categories. First, TAG identifies segment candidates using per-pixel features obtained from DINOv2 [10]. Then, it acquires representative segment embeddings for segment candidates using per-pixel features from a ViT pre-trained with CLIP [11]. Finally, the categories are assigned to each candidate segment by retrieving the closest matching sentence from an external database. Note that the input is only the image, with no need to input category candidates as guidance. 66
6.2 **Overview of the flow for each segment.** Each segment independently retrieves for category candidates and assigns a category. 67
6.3 **Comparison results with CLIP base open-vocabulary segmentation methods on PascalVOC [5]** Note that MaskCLIP [3] and GEM [7] uses text guidance while our TAG does not use. 70
6.4 **Qualitative results.** We compare GroupViT [12], ZeroSeg [13], and our TAG on images containing general objects from PascalContext [14]. This figure indicates that TAG can segment and label correctly. 71

6.5 **Open-vocabulary segmentation results.** In (a) we test on a general image, (b) and (c) show images generated by Stable Diffusion [15], and (d) and (e) are images featuring specific proper nouns. 75
6.6 **Qualitative results of ablation study on PascalVOC [5].** The database and the number of k-means clusters are shown with the results. 77

List of Tables

1.1 **Relationships among proposed methods.** Classification of proposed methods into five categories: training-free, annotation-free, uncertainty metric, guidance-free, and open-vocabulary. . 3

2.1 **Relationships among related studies.** Classification of related studies into four categories: training-free, annotation-free, guidance-free, and open-vocabulary. 14

3.1 **Comparison of class-wise IoU between the proposed method and random sampling on Cityscapes dataset [1].** 27

3.2 **Average, minimum, and maximum IoU for each class from five trials of 50% random sampling on Cityscapes dataset [1].** 28

4.1 **Experiment results across various methods and dataset combinations.** 41

5.1 **Comparison of MaskCLIP [3], GEM [7], and our MaskDiffusion.** We evaluate on Potsdam [6], Cityscapes [1], PascalVOC [5] and COCO-Stuff [9] and report the mIoU. 51

5.2 **Comparison of Unsupervised MaskDiffusion with state-of-the-art unsupervised segmentation methods.** We use the Cityscapes [1] and COCO-Stuff [9] datasets for comparison and report the mIoU. The clear boost in performance by including the internal features underlines their semantic richness and effectiveness. 54

5.3 **Ablation study on Diffusion time step.** We evaluate MaskDiffusion on the Cityscapes dataset [1] using mIoU. . . . 60
5.4 **Ablation study on output size of the cross-attention map.** We evaluate MaskDiffusion on the Cityscapes dataset [1] using mIoU. 60
5.5 **Additional experiment results of MaskDiffusion.** we conduct an experiment calculating representative $\mathbf{f}$ with ground truth and using only the classes contained in the images as prompt input (dynamic prompts). 61
5.6 **Ablation study on U-Net layers to be adopted as internal features.** We perform k-means on various U-Net layer outputs and evaluate on the Cityscapes dataset [1] using the unsupervised mIoU. 62

6.1 **Comparison of state-of-the-art methods.** We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. The clear boost in performance by retrieving open-vocabulary segment categories underlines their semantic richness and effectiveness. 72
6.2 **Ablation study on database and numbers of cluster used for k-means.** We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. These results reveal that using PMD [17] and CC12M [18] as the database and setting the number of k-means clusters to 15 is the most robust choice, consistently yielding favorable outcomes across multiple datasets. 76
6.3 **Ablation study on filtering operations.** We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. This table shows that utilizing all three filtering operations yields the best results. 79
6.4 **Ablation study on the threshold of frequency filtering.** We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. 79

6.5 **Ablation study on the number of retrieved captions.** We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. 80
6.6 **Ablation study on the threshold of Sentence-BERT similarity.** We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. 80

Chapter 1

Introduction

In this chapter, we first introduce the positioning and challenges of semantic segmentation within the field of computer vision, followed by the problem definition and research objectives of this study. Afterward, we summarize the structure of this thesis.

1.1 Semantic Segmentation

Image recognition is a key task in the field of computer vision. Several methods for recognizing images include image classification, object detection, and semantic segmentation. Image classification involves recognizing objects depicted in an image, while object detection can roughly identify where objects are located within the image. Semantic segmentation goes further by recognizing objects at the pixel level. Figure 1.1 shows examples of these tasks; notably, object detection uses bounding-box-based recognition to roughly identify an airplane in the image, whereas semantic segmentation allows for the recognition of the airplane and its background on a pixel-by-pixel basis. Semantic segmentation is critical in applications where not only the presence of an object is necessary, but also a precise understanding of its shape and boundaries. For example, in medical image analysis, accurate segmentation is essential for diagnosing diseases and formulating treatment plans. Similarly, in autonomous driving, understanding the exact location of obstacles

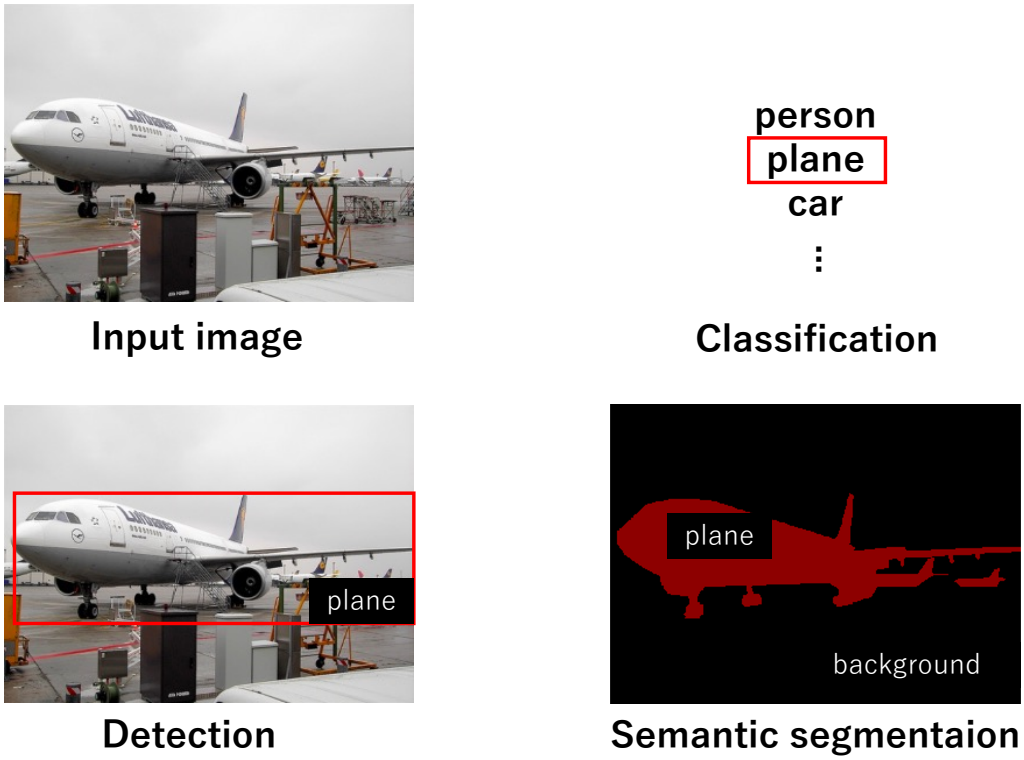

Figure 1.1 Input image and outputs for each image recognition task. In general classification tasks, the output is the class name of the object mainly featured in the image. In object detection, the output includes both the location in the image and the class name of the object. In semantic segmentation, each pixel is assigned and outputs an object class. In supervised learning, the class names for these tasks are chosen from a predefined set of multiple candidates, regardless of the task.

Method	Training Free	Annotation Free	Uncertainty Metric	Guidance Free	Open Vocabulary
Active learning (Sec3)	-	50%	√	√	-
Unsupervised domain adaptation (Sec4)	-	100%	√	√	-
MaskDiffusion (Sec5)	√	100%	-	-	√
Unsupervised MaskDiffusion (Sec5)	√	100%	-	√	√
TAG (Sec6)	√	100%	-	√	√

Table 1.1 Relationships among proposed methods. Classification of proposed methods into five categories: training-free, annotation-free, uncertainty metric, guidance-free, and open-vocabulary.

and other vehicles on the road is crucial for safe driving.

Semantic segmentation implementations widely employ deep learning technologies, particularly Convolutional Neural Networks (CNN) and Vision Transformers (ViT) [19]. These models, based on their robust feature extraction capabilities, have significantly contributed to advancements in this field. CNN and ViT enable end-to-end prediction of pixel-level labels directly from images, yielding results that are visually and intuitively understandable. However, there are several challenges associated with this approach. Achieving high-quality segmentation requires a large amount of pixel-level annotation, which is a time-consuming and costly task. Additionally, the training process involves high computational costs, making the development of efficient model training methods crucial. Furthermore, conventional supervised learning depends on a predefined set of categories, making it difficult to annotate and learn the specific categories of individuals, buildings, proper nouns, and more. This study proposes a new approach to address these challenges associated with training in semantic segmentation.

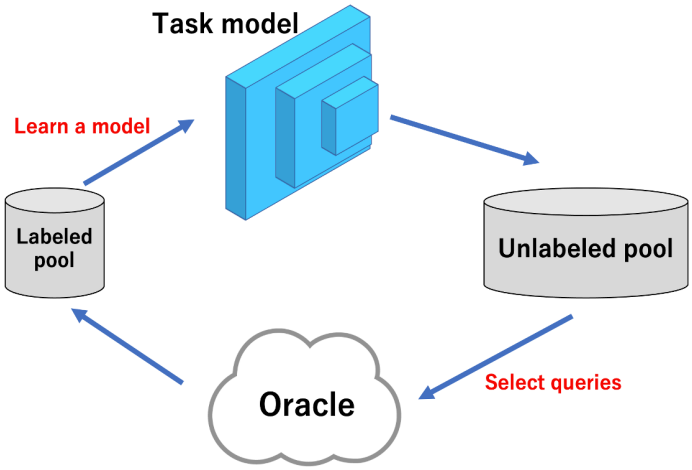

Figure 1.2 Typical pool-based active learning cycle. In each iteration, a task model is trained using labeled images. After training, unlabeled images are selected based on the model's inference and then labeled by an oracle. This cycle is repeated until the model's performance meets the user's requirements or the budget is exhausted.

1.2 Problem Definition and Research Objectives

This thesis addresses the efficiency of training costs associated with semantic segmentation. Training for semantic segmentation requires pixel-level annotations and substantial computational resources. Additionally, supervised learning relies on a predefined set of categories, which makes it challenging to detect rare or entirely new classes during prediction. We shows the relationships among the proposed methods in Table 1.1. Below, we present the objectives for each problem tackled in this thesis.

1.2.1 Efficiency of Annotation through Active Learning

Active learning, as illustrated in Figure 1.2, involves selecting the most informative images from a pool of unlabeled images for model training, having those selected images labeled by an Oracle (annotator), and then adding the labeled images to the labeled pool to update the task model. This process is repeated until the model's performance meets the requirements or the budget is exhausted.

In this study, we propose a method to select the most effective images from the unlabeled pool within the framework of active learning. Our approach uses an uncertainty sampling strategy to determine the most effective images. Uncertainty sampling is a method that involves labeling data with the highest uncertainty during task model inference. Recently, traditional methods using uncertainty sampling have been prevalent, with approaches using Variational Autoencoders (VAE) [2, 20, 21, 22, 23], and deep learning models estimating loss (learning-loss) to select uncertain images in [24].

These prior studies often result in complex and inefficient pipelines that are difficult to operationalize. Specifically, the task model and the uncertainty computation model operate independently, making it challenging for the task model to select truly necessary images. Additionally, there are issues with an increased number of computational nodes and longer computation times, which can deviate from the goal of efficiency in annotation.

The objectives of this study are: (1) to reduce annotation costs using an active learning method that employs an uncertainty indicator computable from model outputs, and (2) to propose a model that can reduce annotation costs more effectively in terms of accuracy and time than traditional methods.

1.2.2 Efficiency of Annotation through Unsupervised Domain Adaptation

While active learning can improve annotation efficiency in semantic segmentation, it still requires annotations. Semantic segmentation, which involves

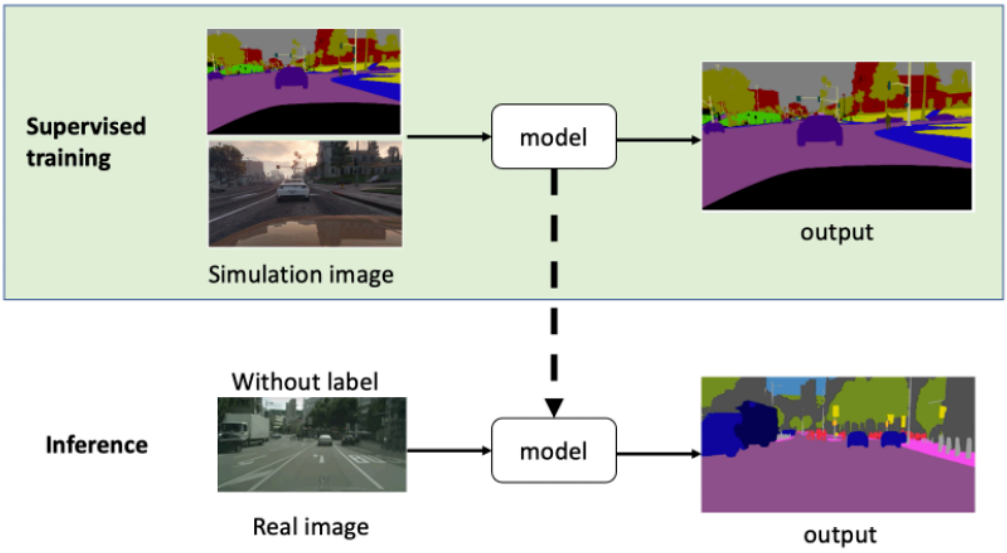

Figure 1.3 Overview of unsupervised domain adaptation. Training is performed using synthetic data that does not require annotation, and evaluation is conducted on real images.

assigning class labels to every pixel in an image, has higher annotation costs than other tasks. Therefore, we propose a method using uncertainty for unsupervised domain adaptation, which does not require any annotations.

Unsupervised domain adaptation is a task aimed at adapting models trained on synthetic data, as illustrated in Figure 1.3, to real-world data without needing costly annotations on real images. The training data consists of images and their correct labels generated by a game engine, eliminating the need for manual annotation. However, training on source data (synthetic data) using conventional supervised learning methods can lead to significant performance degradation due to domain differences with the target data (real image data).

The objectives of this study are: (1) to reduce annotation costs by minimizing the uncertainty calculated from model outputs as a loss using a new unsupervised domain adaptation method, and (2) to enhance existing unsupervised domain adaptation methods by adding this approach, thereby creating a more robust model for the target domain and achieving state-of-

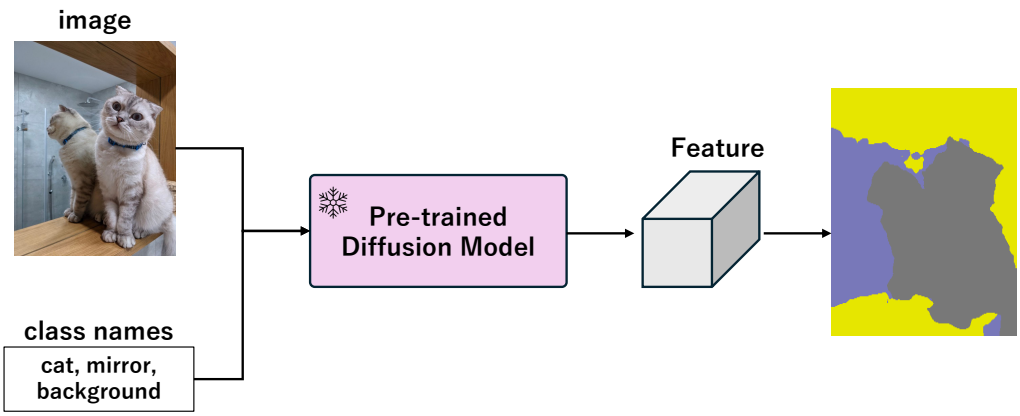

Figure 1.4 Overview of applying pre-trained diffusion models. Capable of segmenting a wide range of classes, including proper nouns, in an open vocabulary without the need for additional training.

the-art results.

1.2.3 Applying Pre-trained Diffusion Models to Semantic Segmentation

While active learning and unsupervised domain adaptation can improve annotation efficiency, semantic segmentation still requires annotations and extensive training. Therefore, we propose a method that applies pre-trained image generation models to semantic segmentation, eliminating the need for data preparation or additional training.

This method, as illustrated in Figure 1.4, involves inputting an image and the class names for segmentation into a pre-trained diffusion model. This allows for the extraction of features from the diffusion model without additional training, enabling the segmentation of a wide range of classes, including proper nouns, in an open vocabulary.

In this study, we (1) analyze the internal features of pre-trained diffusion models to explore their potential applications in semantic segmentation, (2) propose a new method that achieves effective segmentation without additional training or annotation, and (3) demonstrate the superiority of our

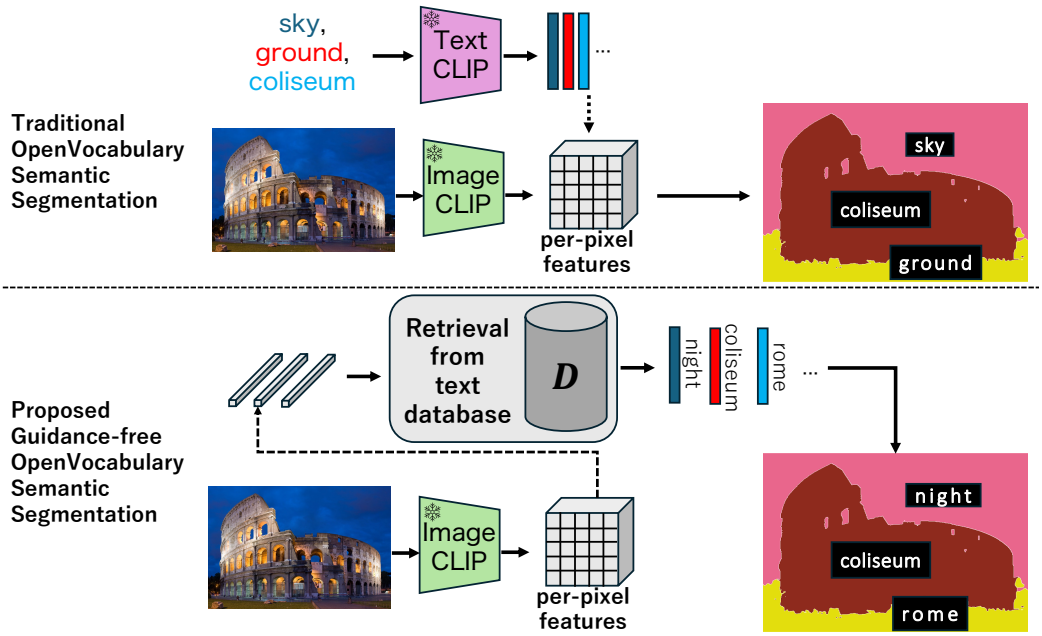

Figure 1.5 Comparison between segmentation methods that do not require the input of class names (guidance) and conventional methods. Our approach eliminates the need for additional training and guidance, enabling the segmentation of a wide range of classes, including proper nouns, in an open vocabulary.

method over conventional techniques in performing open vocabulary semantic segmentation across various image domains.

1.2.4 Guidance-free Open Vocabulary Semantic Segmentation

Traditional open-vocabulary semantic segmentation methods allow for the segmentation of a wide range of classes but require the input of class names (guidance) as candidates. This involves inputting candidates that include the class names of objects appearing in the image, which means it is not entirely annotation-free. To address this, we propose a new approach using a pre-trained CLIP model [11] and retrieving open-vocabulary classes from

an external database.

As shown in Figure 1.5, this method proposes a new open-vocabulary semantic segmentation technique that does not require additional training, annotation, or guidance. It solves the issue with conventional methods where manual input of candidate classes is necessary.

In this study, we (1) propose a new approach that achieves open-vocabulary semantic segmentation without reliance on predefined classes, additional training, annotation, or guidance by searching cluster categories from an external database, and (2) demonstrate the superiority of our method over traditional techniques in performing open vocabulary semantic segmentation across various image domains.

1.3 Publication List

The publication list by the author and the related thesis chapters are as follows:

1. Yasufumi Kawano, Yoshiki Nota, Rinpei Mochizuki, and Yoshimitsu Aoki. Non-deep active learning for deep neural networks. *Sensors*, Vol. 22, No. 14, p. 5244, 2022 (Chapter 3)

2. 川野恭史, 野田祥希, 望月凜平, 青木義満. Active learning における不確実サンプル選択によるアノテーション効率化. 精密工学会誌, Vol. 88, No. 2, pp. 211–216, 2022 (Chapter 3)

3. 川野恭史, 野田祥希, 青木義満. 不確実性ロスによるセマンティックセグメンテーションの教師無しドメイン適応. 精密工学会誌, Vol. 89, No. 12, pp. 921–925, 2023 (Chapter 4)

4. Yasufumi Kawano and Yoshimitsu Aoki. Maskdiffusion: Exploiting pre-trained diffusion models for semantic segmentation. *IEEE Access*, Vol. 12, pp. 127283–127293, 2024 (Chapter 5)

5. Yasufumi Kawano and Yoshimitsu Aoki. Tag: Guidance-free open-vocabulary semantic segmentation. *IEEE Access*, Vol. 12, pp. 88322–

88331, 2024 (Chapter 6)

1.4 Structure of This Thesis

Chapter 1 discusses the background of semantic segmentation, which is the subject of this study, and describes four problem settings addressed. Chapter 2 explains the related research and underlying technologies relevant to these issues. Chapter 3 discusses the efficiency of annotation through active learning. The main content of this chapter is based on findings published in [25, 26]. Chapter 4 discusses the efficiency of annotation through unsupervised domain adaptation. The main content of this chapter is published in [27]. Chapter 5 describes the application of pre-trained diffusion models to semantic segmentation. The main content of this chapter is published in [28]. Chapter 6 discusses guidance-free open-vocabulary semantic segmentation. The main content of this chapter is based on findings published in [29]. Chapter 7 presents the conclusions of this thesis.

Chapter 2

Related Works

2.1 Active Learning

There are three main scenarios for active learning [30]: The first is membership query synthesis, which generates valid data for model training. The second is stream-based selective sampling, which sifts the data to label or discard. The third is pool-based sampling, which selects the most effective images for model learning from an unlabeled pool of data. Of the three, pool-based sampling, shown in Figure 1.2, is the most common [2, 20, 23, 24, 31, 32]. Specifically, a representative image is selected from a large amount of unlabeled data and annotated by an oracle (called the annotator), thus becoming the labeled data. There are multiple approaches to using pool-based sampling.

Some studies focus on how to map the distance of a distribution to the information content of data points [33, 34, 35], while others estimate the diversity of a distribution by observing the gradient [36], future errors [37], or changes in the output of the trained model [38]. However, these methods are computationally inefficient for current deep networks and large datasets.

To solve this problem, active learning methods [2, 20, 21, 22, 23, 24] effective for deep networks and large datasets have appeared. Ref. [24] proposed the learning-loss method. Their model consists of a task module and a loss prediction module that predicts the loss of the task module. These

two modules are learned simultaneously, and the loss prediction module is trained to estimate the target loss for unlabeled samples based on mid-layer feature information as a proxy for model uncertainty. Since the accuracy of loss prediction is affected by the performance of the task module, there is a disadvantage that if the task module is inaccurate, the predicted loss will not reflect how informative the sample is.

The methods that have become mainstream in recent years are those that use adversarial training [2, 20, 21, 22, 23]. In these methods, a variational autoencoder (VAE) reduces an image to a low-dimensional representation, and the discriminator determines whether the obtained representation was derived from a labeled or unlabeled image. By training the VAE and discriminator adversarially, we can select images from the unlabeled data pool that have features that the model has not yet seen. Because these methods sample images based on image features, they not only have the advantage of performing task-independent active learning, such as classification and segmentation but also the disadvantage of taking more time than task learning. SRAAL [2] is an improved method of VAAL [20]. By relabeling the output of the discriminator with an online uncertainty indicator (OUI), SRAAL achieves a better selection of uncertain images than VAAL. We think that the contribution of SRAAL is more in OUI than in VAE. OUI is used to relabel the discriminator, which is the same as distilling the discriminator at the output of OUI. In other words, the contribution that represents uncertainty can be found in the OUI. The verification of this hypothesis is one of the significant aspects of our research.

TA-VAAL [23] is one of the state-of-the-art methods that combine VAAL [20] and learning-loss [24]. By changing the task-learning-based loss prediction to ranking loss prediction and embedding normalized ranking loss information into VAAL using a ranking-conditional generative adversarial network, the data distribution of both labeled and unlabeled pools is taken into account. Some methods use VAE [39, 40] or GAN [41, 42] to generate new images that are more informative for the current model. However, these synthesis methods have disadvantages such as high computational complexity and unstable performance [43].

These studies rely on specialized learning methods such as adversarial learning. This tends to result in complex and inefficient pipelines that are difficult to use in practical applications. Specifically, they have a model for computing uncertainty in addition to the task model, and since these are independent of each other, it is difficult for the task model to select the images it needs, and the number of computation nodes is large, which increases the computation time. The disadvantage of the time required is that it detracts from the objective of improving the efficiency of annotation.

To solve this problem, Refs. [44, 45, 46] have been proposed to compute uncertainty algorithmically, using only task model learning. These methods do not require learning to predict uncertainty, making them easy to use in practice. We propose an uncertainty indicator generator (UIG), which is an algorithmic method similar to these methods.

The uncertainty sampling approach [47, 48] annotates the data with the highest level of uncertainty in the results when inferring the model. Essentially, uncertainty is the degree to which the model cannot recognize the image. In other words, a sample that is closer to the decision boundary of the model can be regarded as more uncertain [49, 50]. Therefore, our study focused on selecting samples around the decision boundary.

2.2 Unsupervised Domain Adaptation

Unsupervised domain adaptation aims to adapt models trained on synthetic data to real-world data without the need for costly real image annotations. To acquire the target domain, unsupervised domain adaptation techniques include adversarial learning methods [51, 52] and self-learning methods [53, 54]. Adversarial learning aims to align the distributions of the source and target domains as seen in GANs. Self-learning methods train the network using pseudo-labels [55] for the target domain. Most unsupervised domain adaptation processes involve pre-computing pseudo-labels offline and then repeating the model training process.

In self-learning methods, pseudo-labels utilize pixels whose predictions

Table 2.1 Relationships among related studies. Classification of related studies into four categories: training-free, annotation-free, guidance-free, and open-vocabulary.

Method	Pretrained Backbone	Training Free	Annotation Free	Guidance Free	Open Vocabulary
Traditional Semantic Segmentation					
DeepLab [57]	-	-	-	-	-
Unsupervised Segmentation					
STEGO [8]	DINO [58]	-	√	√	-
HP [59]	DINOv2 [10]	-	√	√	-
Open Vocabulary Segmentation					
ODISE [60]	StableDiffusion [15]	-	-	-	√
MaskCLIP [3]	CLIP [11]	√	√	-	√
Our method (Sec5)	StableDiffusion [15]	√	√	-	√
Zero-Guidance Segmentation					
ZeroSeg [13]	CLIP[11]& DINO[58]& GPT-2[61]	√	√	√	√
SelfSeg [62]	BLIP[63] & X-Decoder[64]	√	-	√	√
Our method (Sec6)	CLIP[11] & DINOv2[10]	√	√	√	√

are sufficiently reliable. [56] proposed constraining the model to generate high-confidence predictions by minimizing the entropy of the predicted values, instead of using high-confidence pixels. While [56] uses entropy as a measure of reliability, this study employs a metric shown to measure reliability more accurately than entropy in Active Learning, allowing for further improvements in accuracy beyond previous research.

2.3 Unsupervised Learning and Open-Vocabulary Semantic Segmentation

Unsupervised semantic segmentation [8, 59, 65, 66] is the task of performing semantic segmentation without using supervision. STEGO [8] and HP [59] use a two-layer structure comprising a backbone and a head, where the backbone is trained using the unsupervised learning method DINO [58] to extract image features that are then used to optimize the segmentation model's head. However, a challenge with unsupervised semantic segmentation is that although it clusters images by class, it cannot specifically identify each cluster's

class.

Open-vocabulary semantic segmentation [3, 7, 67, 68, 69, 70, 71] is the task of segmenting objects across domains without being limited to predefined categories. MaskCLIP [3], GEM [7] and CLIPSeg [68], based on the CLIP [11] visual language model, have advanced open vocabulary semantic segmentation. In particular, MaskCLIP [3] and GEM [7] segment various categories without additional training. These two approaches, reducing annotation and training costs and overcoming category limitations, have inspired a new direction in our research, motivating us to explore similar problem formulations. Furthermore, ODISE [60] has extended the capabilities of pre-trained diffusion models, incorporating additional training of a segmentation network to achieve open-vocabulary panoptic segmentation. The application of image generation models to segmentation is discussed in detail in Sec 2.3.1.

Table 2.1 classifies related studies based on four criteria: the pre-trained models used, the necessity of additional training, the necessity of text input (guidance) for segmentation, class identifiability across images, and the ability to perform open-vocabulary segmentation.

This study aims to propose methods that do not require annotations and additional training for semantic segmentation, building on the use of pre-trained models in unsupervised semantic segmentation [8, 59] and the training-free segmentation methods in open-vocabulary semantic segmentation [3, 7].

2.3.1 Generative Models for Segmentation

The use of image generative models, including Generative Adversarial Networks (GAN) [41, 42, 72, 73, 74, 75] and diffusion models [60, 76, 77, 78, 79, 80, 81, 82, 83], has been a focal point in various prior studies concerning semantic segmentation. DDPMSeg [76] revealed that a generative model's internal representation correlates with visual semantics, aiding semantic segmentation but limited to a predefined set of labels. Addressing this, ODISE [60] enables panoptic segmentation with open vocabularies. DiffSeg [83], similarly to our method, provides unsupervised segmentation with-

out additional training by using KL divergence in UNet's self-attention maps for segmentation, but it cannot consistently map the same class to the same index across images.

Several methodologies [60, 76, 77, 78, 79, 80] have leveraged diffusion models as a foundational backbone for segmentation tasks, often incorporating additional models and training to facilitate the segmentation process.

In alignment with the existing research landscape, we propose an open-vocabulary semantic segmentation method without additional training.

2.4 Zero Guidance Semantic Segmentation

Open-vocabulary semantic segmentation faces the challenge of requiring guidance (the names of classes to be segmented). This study aims to enable open-vocabulary semantic segmentation without such guidance. Concurrently, ZeroSeg [13] utilizes ZeroCap [84], which combines the CLIP [11] features with the language model GPT-2 [61] to generate captions. This involves tuning a portion of GPT-2 [61] to fit the prompt "Image of a ...", ensuring the text closely matches the image according to CLIP [11]'s understanding.

SelfSeg [62] achieves zero-guidance semantic segmentation by segmenting internal features obtained from the image-to-text model BLIP [63] using X-Decoder [64].

However, SelfSeg [62] requires additional training, and ZeroSeg [13] still has room for performance improvement. This study hypothesizes that the challenges of ZeroSeg [13] are linked to the performance of ZeroCap [84] and proposes a new approach that retrieves candidate class names from a database to estimate class names for each segment.

2.5 Text Retrieval from CLIP Embedding

In natural language processing, retrieving information from external databases has been shown to boost the performance of large language models [85, 86, 87]. This concept is also explored in computer vision, particularly for ad-

dressing class imbalance by using databases to retrieve training samples or image-text pairs. RAC [88] and VIC [89] achieved image classification without relying on predefined classes by utilizing an external database. It has the advantage of low memory consumption because it only uses captions from databases like Public Multimodal Datasets (PMD) [17] collecting image-text pairs from different public datasets.

Chapter 3

Efficiency of Annotation through Active Learning

One way to improve annotation efficiency is active learning. The goal of active learning is to select images from many unlabeled images, where labeling will improve the accuracy of the machine learning model the most. To select the most informative unlabeled images, conventional methods use deep neural networks with a large number of computation nodes and long computation time, but we propose a non-deep neural network method that does not require any additional training for unlabeled image selection. The proposed method trains a task model on labeled images, and then the model predicts unlabeled images. Based on this prediction, an uncertainty indicator is generated for each unlabeled image. Images with a high uncertainty index are considered to have a high information content and are selected for annotation. Our proposed method is based on a very simple and powerful idea: select samples near the decision boundary of the model.

In this study, we reduce annotation costs using an active learning method that employs an uncertainty indicator computable from model outputs, and propose a model that can reduce annotation costs more effectively in terms of accuracy and time than traditional methods.

3.1 Overview of the Proposed Method

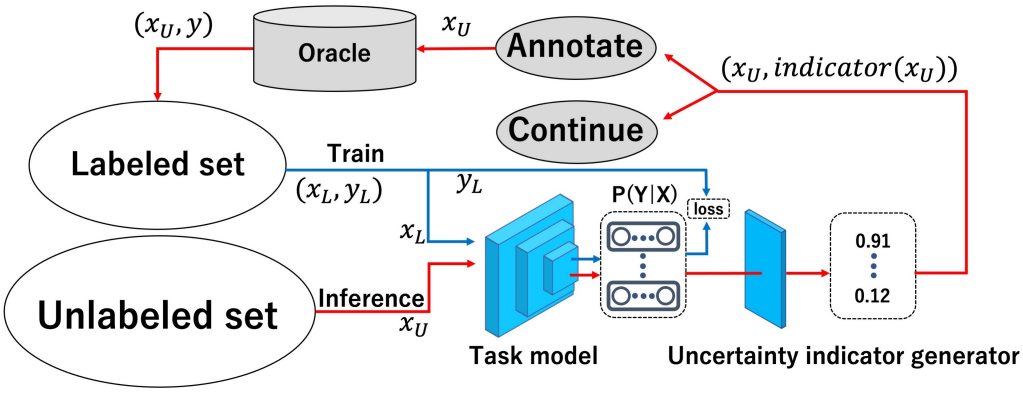

Figure 3.1 Model of our proposed method. The red arrows show the path for the unlabeled images, and the blue arrows show the path for the labeled images. The labeled images are used to train the task model. Using the output of the trained task model, the uncertainty indicator generator (UIG) outputs the uncertainty indicator for unlabeled images. Unlabeled images with high uncertainty are labeled first.

Active learning involves selecting the most informative images from a pool of unlabeled images for model training, having those selected images labeled by an Oracle (annotator), and then adding the labeled images to the labeled pool to update the task model. This process is repeated until the task model's performance meets the requirements or the budget is exhausted.

As shown in Figure 3.1, the proposed method is based on pool-based uncertainty sampling in Active Learning. Typically, tasks include classification, multi-label classification, and semantic segmentation; however, this study specifically targets semantic segmentation.

In Figure 3.1, the red arrows indicate the path of unlabeled images, while the blue arrows indicate the path of labeled images. Initially, the task model is trained using a small set of labeled images. Subsequently, images from the unlabeled pool are input into the Uncertainty Indicator Generator (UIG) using the trained task model, producing an uncertainty score for each image. The top K uncertain images are selected, annotated, and added to the labeled

Algorithm 1 Sampling from Uncertainty Indicator

Input: D_L: labeled data pool, D_U: unlabeled pool, T: random initialized task model, C: number of cycles in active learning, N: number of samplings;

Output: Final learned parameters of T;

```
for i = 1 to C do
    train T with D_L
    for x_i in D_U do
        V = T(x_i)
        indicator_i = UIG(V)
        if (indicator_i is top-N) then
            y_i = ORACLE(x_i)
            D_L = D_L + (x_i, y_i)
            D_U = D_U − x_i
        end if
    end for
end for
```

pool.

By repeating this process, high performance can be achieved with minimal annotation cost. The sampling flow using UIG is also detailed in Algorithm 1.

Our contributions are the following:

1. We propose a model that can reduce annotation costs more effectively in terms of accuracy and time than traditional methods.

2. We demonstrate that deep learning models are not necessary for active learning in semantic segmentation, achieving more efficient active learning without requiring additional training to calculate uncertainty indicators.

3.2 Uncertainty Indicator Generator

We defined UIG based on the hypothesis that the contribution in SRAAL [2] is more in OUI than in VAE. In SRAAL [2], OUI is used to relabel the discriminator, which is the same as distilling the discriminator at the output of OUI. In other words, the contribution that represents uncertainty can be found in the OUI.

UIG can be used for multi-label classification in addition to the usual one-class classification task. The fundamental idea behind both UIG is the same: to compute uncertainty by using least confident and margin sampling. Least confident is a method of selecting images in which the maximum value of V is the smallest, and margin sampling is a method of selecting images in which the difference between the maximum value of V and the second largest value of V is the smallest. It is designed to express the uncertainty of each image in the unlabeled pool with a specific importance using continuous values in the range [0, 1). The UIG calculates the uncertainty indicators based on the prediction vectors of the task model (image classifier and semantic segmentation model). Specifically, in the case of image classification, the prediction is a probability vector for each category. In segmentation, each pixel has a probability vector, and the prediction vector is the average value of each probability vector. The calculation of the uncertainty indicator can be formulated as

$$\text{Indicator}(x_U) = 1 - \frac{\text{MINVar}(V)}{\text{Var}(V)} \times \max(V), \tag{3.1}$$

where x_U is the unlabeled image, and $V = p\left(x_U | D_L\right)$ is the probability vector of x_U based on the trained task model in the current labeled pool D_L. Assuming that the number of classes is C, $\text{MINVar}(V)$ can be formulated as

$$\begin{aligned} \text{MINVar}(V) &= \text{Var}(V') \\ &= \frac{1}{C}\left(\left(\frac{1}{C} - \max(V)\right)^2 + (C-1)\left(\frac{1}{C} - \frac{\max(V)}{C-1}\right)^2\right). \end{aligned} \tag{3.2}$$

MINVar(V) is the variance of the vector V', where the top m elements are the same as V, and the other $C-1$ elements are $1-max(V)/(C-1)$.

V' is a vector where all elements have the same value except for the maximum, which equals the maximum of V. Therefore, MINVar(V)/ Var(V) represents the concentration of a probability distribution and can be viewed as margin sampling. The more the value of $VAR(V)$ deviates from $MINVAR(V)$, the higher the uncertainty is considered. $max(V)$ indicates that the lower the maximum value, the higher the uncertainty, representing the least confident metric. From this, we can see that the uncertainty metric has the following characteristics: it ranges between [0,1), is negatively correlated with the maximum probability value, and is negatively correlated with the concentration of the probability distribution. In essence, it estimates the uncertainty of an image from the perspectives of least confident and margin sampling. An uncertainty metric close to 1 implies higher image uncertainty and a higher likelihood of being selected.

As a concrete example, take the three-class classification of dogs, cats, and tigers. Consider that the output of the task model for a certain image is $[0.6, 0.1, 0.3]$. In this case, $V = [0.6, 0.1, 0.3]$, $V' is [0.6, 0.2, 0.2]$. Since VAR is the variance of V and $MINVAR$ is the variance of V', they are 0.042 and 0.036, respectively. Additionally, $max(V)$ is the maximum value of V, which is 0.6. Applying these values to Equation 3.1, we obtain an uncertainty indicator of 0.49. If this is calculated for an image such as $V = [0.4, 0.4, 0.2]$, where it is not known whether the image is of a dog or a cat, the indicator is obtained as 0.90, which can be regarded as high uncertainty.

3.3 Experiment Overview of Active Learning

In this study, the proposed method was compared against five types of sampling methods: VAAL [20], SRAAL [2], TA-VAAL [23], Entropy [30], and random sampling. Furthermore, to verify that the proposed method can select more uncertain images than conventional methods, a comparison between the uncertainty metric and actual losses is conducted.

Additionally, the execution speeds for each dataset are measured and compared with VAAL [20] and SRAAL [2].

Experiments are also conducted using 2D data created with the makemoons library from scikit-learn [90] to examine whether samples near the decision boundary can be effectively selected.

The dataset was initially randomly sampled such that 10% of the data was labeled and the remaining 90% was unlabeled, marking the beginning of the experiment. Subsequently, sampling was conducted in 5% increments until the labeled data constituted 40% of the total dataset. A comparison of accuracy between each sampling method was performed at each sampling increment.

The experiments were conducted five times each, and the average of the results was taken as the final value. The evaluation metric used was the mean Intersection over Union (mIOU).

The Cityscapes dataset [1] was used in the experiments. For training the model, the image size was resized from $1024 \times 2048 \times 3$ to $688 \times 688 \times 3$, with 2975 images used for training and 500 images for testing. Cityscapes is annotated with 34 classes, including ambiguous classes, which were reduced to 19 classes for clarity.

To verify the selection of samples near decision boundaries, a binary classification 2D dataset was generated using the makemoons library from scikit-learn [90], as shown in Figure 3.5. The noise option was set to 0.2, and the sizes of the labeled and unlabeled datasets were each set to 500 samples. From 500 unlabeled samples, 30 samples were selected.

For training in semantic segmentation, a Dilated Residual Network (DRN) [91] was used, and the model was trained over 100 epochs. The optimizer used was Adam, with a learning rate of 0.0003, and the loss function specified was Cross-entropy loss. Data augmentation was performed using only RandomHorizontalFlip.

For the experiments in Section 3.4.1, a simplified dataset was used; therefore, a simplified model was also employed. The task model consisted of a three-layer perceptron with a learning rate of 0.1 and used SGD. The SRAAL's VAE employed two-layer MLPs with ReLU for both the encoder

and decoder, and a two-layer MLP for the discriminator. Adam was used for both the VAE and the discriminator. All experiments were set to 100 epochs. No hyperparameters were needed for the generation of the uncertainty metric.

3.4 Experiment Results of Active Learning

Experimental results are shown in Figure 3.2. When training on the entire Cityscapes dataset [1] using the DRN [91], the mIoU reaches 58.14%. With the proposed method using the same task model on 70% of the data, a mIoU of 56.72% is achieved. A random sampling of 70% of the data results in a mIoU of 54.98%, which is almost equal to the mIoU obtained with 30% sampling using the proposed method. Moreover, the conventional method VAAL achieves a mIoU of 56.10% with 70% of the data, which is close to the 50% value of the proposed method. From this, it can be inferred that the proposed method demonstrates superior performance in semantic segmentation compared to traditional methods.

The results for Cityscapes shown in Figure 3.3 seem to indicate a decrease in mIoU as sampling progresses, compared to other datasets. This is likely because sampling up to 70% in Cityscapes results in outcomes closer to random sampling than in other datasets.

After conducting 5 rounds of 50% sampling, a comparison of the average IoU for each class obtained through random sampling and the proposed method is presented in Table 3.1. Also, the average, minimum, and maximum IoUs for each class in random sampling are shown in Table 3.2.

Results in Table 3.1 reveal that the proposed method achieved higher IoUs in 17 out of 19 classes compared to random sampling (the exceptions being the "Road" and "Fence" classes). Moreover, while the average of the maximum values for each class was 53.07, the average for the proposed method was 52.90, indicating that the proposed method is conducting near-ideal sampling.

Furthermore, a qualitative comparison between the results of training

with all images in Cityscapes and training with 50% of the data sampled using the proposed method is presented in Figure 3.4. Using the proposed method allows for nearly the same outcomes with only half the data compared to using all the data.

These quantitative and qualitative comparisons thus demonstrate the superiority of the proposed method.

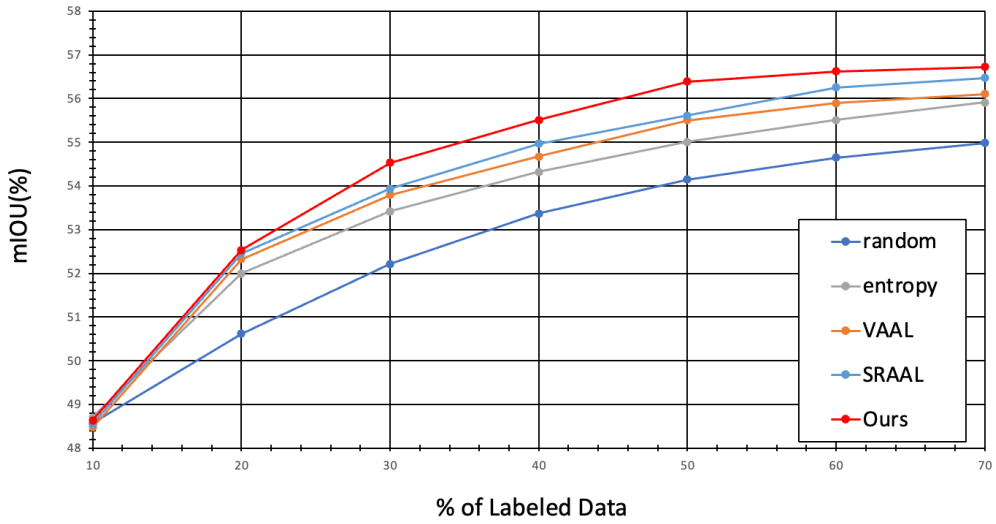

Figure 3.2 Comparison of methods on Cityscapes dataset [1].

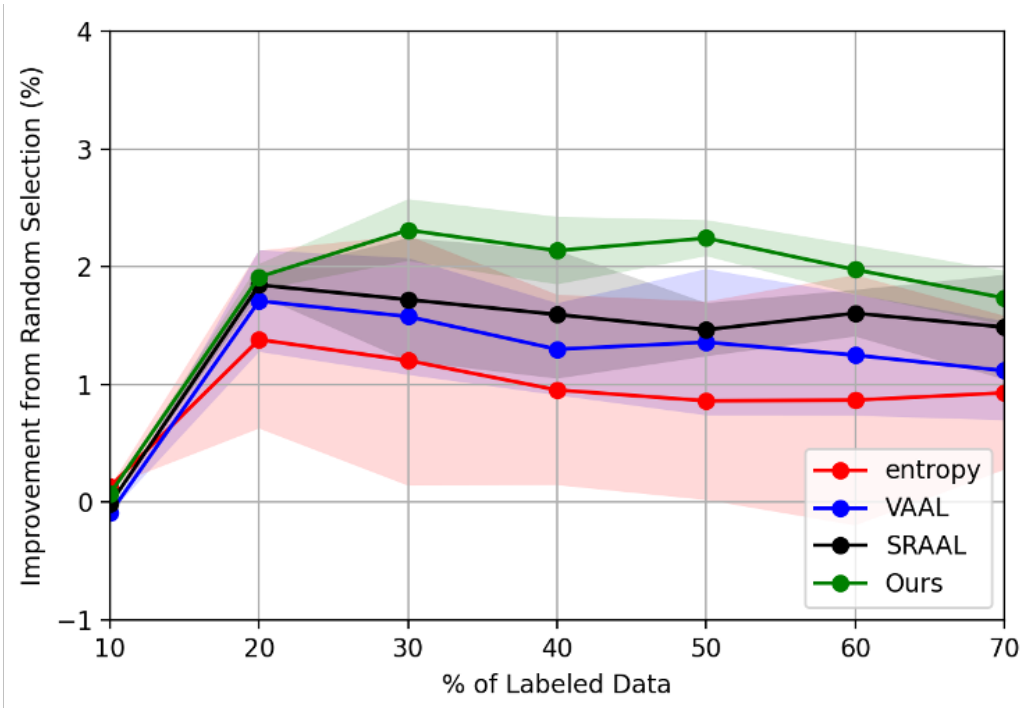

Figure 3.3 Improvements over random sampling by each method on Cityscapes dataset [1].

Table 3.1 Comparison of class-wise IoU between the proposed method and random sampling on Cityscapes dataset [1].

Class	Random50%	Proposed50%	Occurrence
Road	**93.75**	92.04	96
Sidewalk	65.69	**67.90**	92
Building	79.06	**83.82**	98
Wall	22.83	**27.68**	39
Fence	**27.52**	27.26	37
Poll	27.77	**32.14**	99
Traffic light	32.33	**46.37**	60
Traffic sign	45.89	**52.87**	96
Vegetation	84.46	**88.15**	96
Terrain	30.00	**32.89**	47
Sky	85.26	**89.28**	91
Person	40.18	**55.92**	82
Rider	21.25	**28.84**	48
Car	80.70	**83.53**	97
Truck	19.67	**19.96**	21
Bus	29.21	**46.36**	17
Train	42.01	**46.61**	2
Motorcycle	20.90	**30.76**	11
Bicycle	42.82	**52.77**	65
AVERAGE	46.91	**52.90**	-

Table 3.2 Average, minimum, and maximum IoU for each class from five trials of 50% random sampling on Cityscapes dataset [1].

Class	Average	Min	Max
Road	93.75	91.36	94.66
Sidewalk	65.69	64.56	66.60
Building	79.06	77.53	83.58
Wall	22.83	19.91	28.11
Fence	27.52	26.32	29.16
Poll	27.77	25.86	30.83
Traffic light	32.33	28.64	42.76
Traffic sign	45.89	43.39	50.16
Vegetation	84.46	83.10	88.00
Terrain	30.00	27.87	35.84
Sky	85.26	82.92	90.44
Person	40.18	35.29	54.52
Rider	21.25	14.25	25.59
Car	80.70	79.06	84.42
Truck	19.67	11.13	27.78
Bus	29.21	25.29	34.78
Train	42.01	31.97	62.79
Motorcycle	20.90	16.39	25.23
Bicycle	42.82	38.56	53.07
AVERAGE	46.91	43.34	53.07

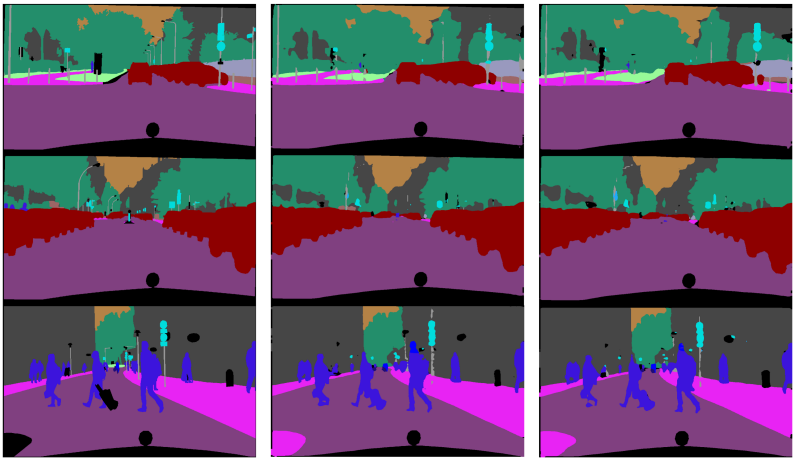

Figure 3.4 Qualitative evaluation on Cityscapes datset [1]. The left image shows the ground truth labels, the center image shows the results of training with all images, and the right image shows the results of training with 50% of the data sampled using the proposed method. Using the proposed method, nearly the same outcomes are achieved with only half the amount of annotations as when training with all images.

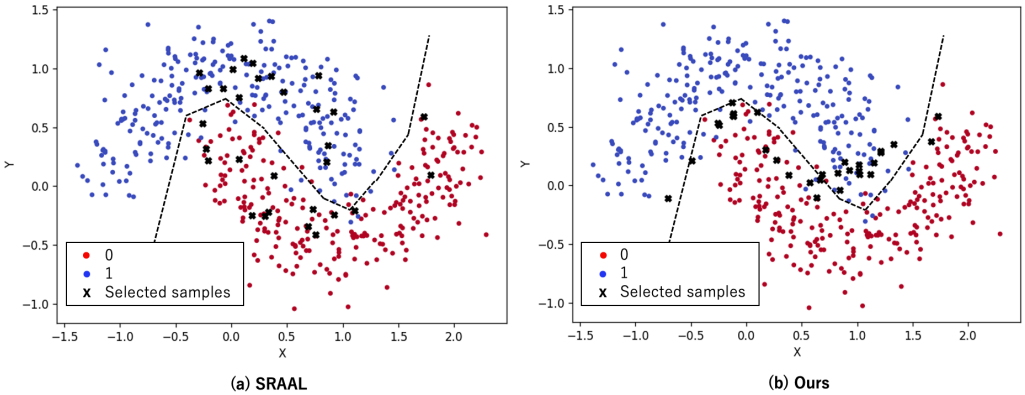

Figure 3.5 Visual results of active learning methods. SRAAL [2], our UIG) on a toy example. The red points represent classes with 0, and blue points with 1. The black faction marks are the samples selected by each method. The oracle decision boundary is shown as a black dashed line. We trained on 500 samples and selected 30 samples from the 500 unlabeled samples. It can be seen that the proposed method is able to select samples near the decision boundary better than SRAAL.

3.4.1 Analysis of Selected Samples

Figure 3.5 shows which samples were selected in the two-dimensional dataset for SRAAL and the proposed method. The samples selected for SRAAL are spread over the whole area, whereas the proposed method selects samples close to the decision boundary line. Our method aims at efficient annotation by selecting samples near the decision boundary, and Figure 3.5 shows that our method selects samples closer to the decision boundary than conventional methods.

3.4.2 Analysis of Uncertainty Indicator

Using the CIFAR-10 dataset [92], graphs with actual loss on the X-axis and the uncertainty metric of each method on the Y-axis are presented in Figure 3.6 for SRAAL, Figure 3.7 for Entropy, and Figure 3.8 for the proposed

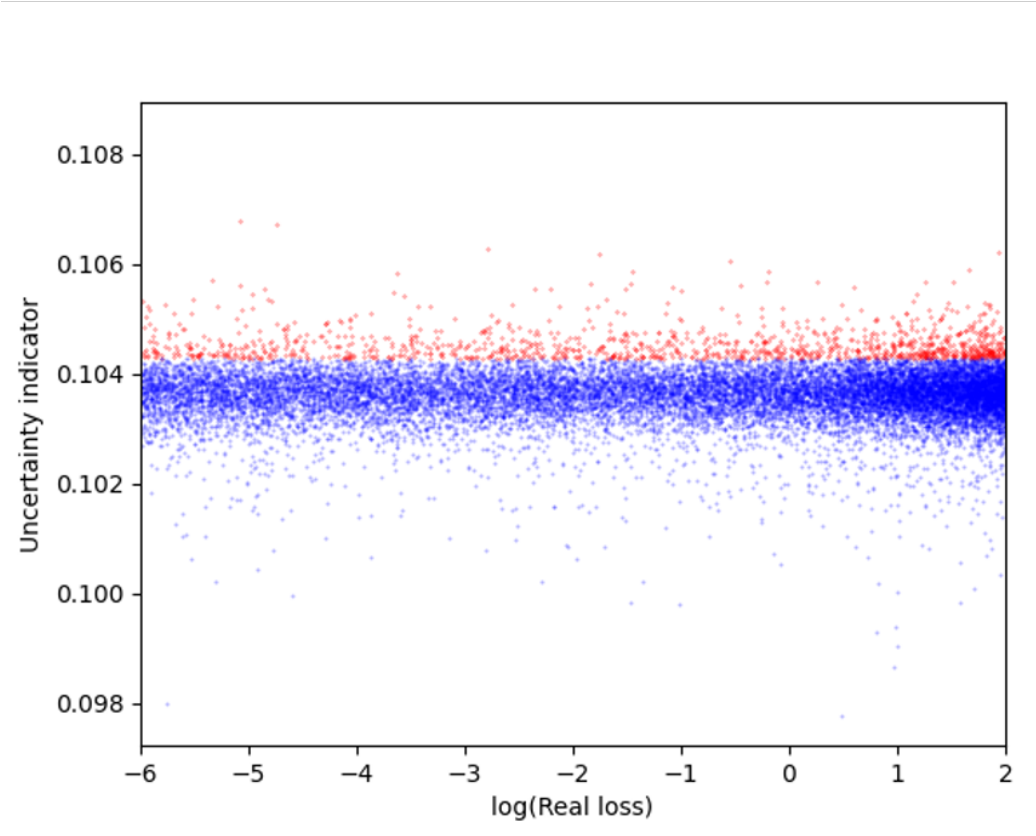

Figure 3.6 Relationships between actual loss (X-axis) and uncertainty metric (Y-axis) for SRAAL [2].

method. Points selected during a 5% sampling after training with a randomly chosen 10% of the outputs are marked in red. In Active Learning, the goal is to sample data with high actual loss from unlabeled data, and it is desirable to have an uncertainty metric that correlates with actual loss. In these figures, ideally, points on the right side (those with high actual loss) should be sampled. SRAAL, as shown in Figure 3.6, selects images from a broad range of losses, from low to high, whereas the proposed method, as seen in Figure 3.8, is able to select images with high loss. Furthermore, the correlation coefficients between loss and uncertainty metrics are 0.01 for SRAAL, 0.07 for Entropy, and 0.43 for the proposed method, indicating that the proposed method has the highest correlation. Therefore, it can be concluded that the proposed method provides better performance in measuring

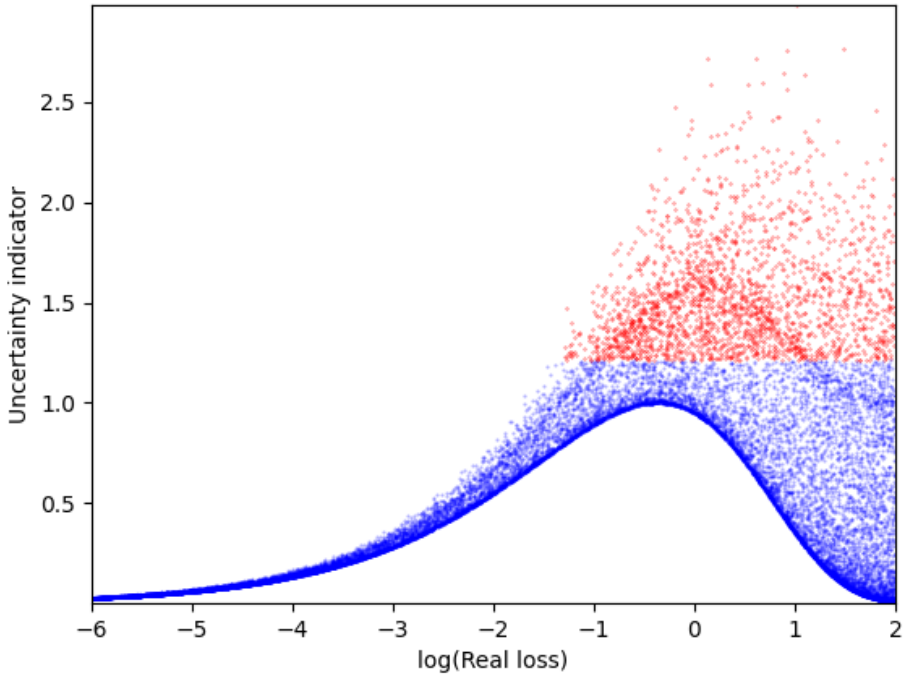

Figure 3.7 Relationships between actual loss (X-axis) and uncertainty metric (Y-axis) for entropy.

uncertainty compared to traditional methods.

3.4.3 Comparison of Model Computational Complexity

A comparison of execution times for the Active Learning pipeline has been conducted using previous studies as benchmarks. The experiments were performed on an NVIDIA GeForce RTX 2080Ti. VAAL [20] required 3.0×10^5 seconds, and SRAAL [2] took 7.0×10^5 seconds, whereas the proposed method completed the pipeline in just 1.3×10^5 seconds. Additionally, the memory required for sampling on the CIFAR-10 dataset with a batch size of 128 was 4114 MiB for the proposed method, compared to 5586 MiB for both VAAL and SRAAL.

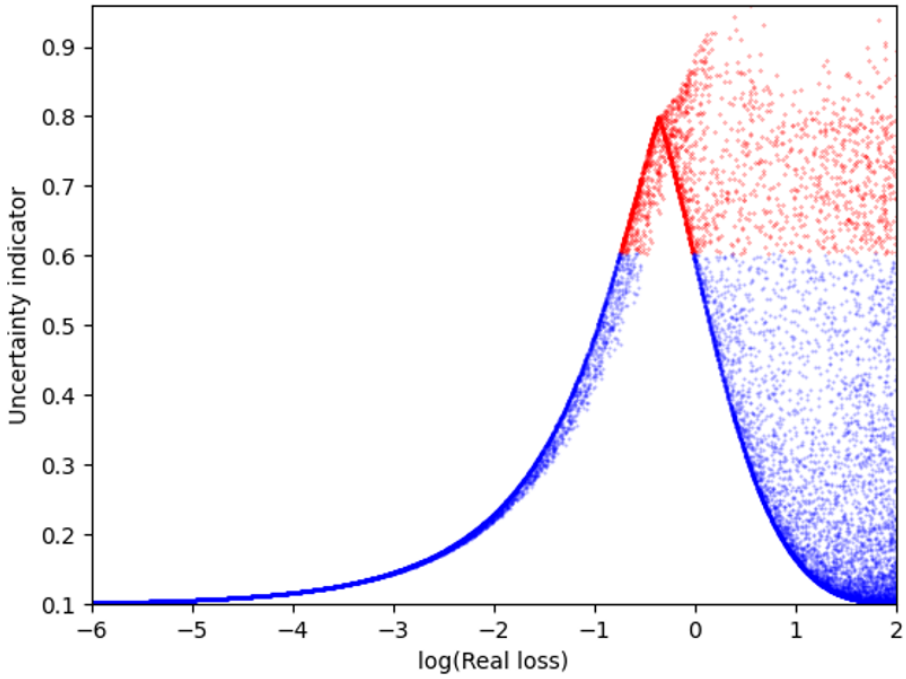

Figure 3.8 Relationships between actual loss (X-axis) and uncertainty metric (Y-axis) for the proposed method.

3.5 Limitation

In recent years, active learning has required considerable time to train models, making it often faster to simply perform annotations directly. We proposed a method that does not require training beyond the task model itself, solving this issue. We consider our approach to be optimal within the context of active learning. However, semi-supervised and unsupervised learning methods have been producing good results recently, and the potential improvements in accuracy through active learning are modest in comparison, with our proposed method being no exception. While active learning selects effective images from unlabeled data, these methods utilize all images for learning. Active learning is particularly suited to physically costly cases like semantic segmentation, whereas semi-supervised and unsupervised learning often fit cases requiring specialized knowledge, such as medical image diag-

nosis. Moving forward, it is necessary to explore integrating Active Learning with these methods to develop approaches that are practical for real-world applications.

3.6 Conclusion

In order to select the most informative sample from an unlabeled pool, we proposed a model to derive the most informative unlabeled sample from the output of the task model, a UIG. Experimental results show that the proposed method can select images with higher loss than the conventional method, and the annotation cost can be reduced by minimizing the amount of labeled data that is used. The proposed method outperforms the current SoTA method by 1% accuracy in CIFAR-10. We also succeeded in reducing the execution time of the models by about 90% for CIFAR-10 and CIFAR-100, about 50% for Cityscapes, and about 80% for CelebA, as compared to the conventional methods.

In this study, we have shown the superiority of our method for images, but we believe that it can also be applied to more advanced tasks, such as video. We will also consider useful methods for combining active learning with semi-supervised and unsupervised learning for real-world applications.

Chapter 4

Efficiency of Annotation through Unsupervised Domain Adaptation

While active learning can enhance annotation efficiency in semantic segmentation, it still requires annotations. Semantic segmentation, which involves assigning class labels to every pixel in an image, is notably costlier in terms of annotation compared to other tasks. Therefore, we propose a method using uncertainty for unsupervised domain adaptation, which requires no annotations at all. Unsupervised domain adaptation aims to adapt models trained on synthetic data to real-world data without the need for costly real image annotations, as shown in Figure 4.1. This approach can achieve accuracies comparable to supervised learning while reducing annotation costs [53, 54, 56]. The training data consists of images and labels generated by a game engine, eliminating the need for manual annotation. However, when traditional supervised learning is applied directly to the source data (synthetic data), the performance significantly degrades due to the domain discrepancy with the target data (real images).

Recent approaches in unsupervised domain adaptation utilize pseudo-labels of the target domain [55] to train the network, thereby acquiring the target domain characteristics [53][54]. Pseudo-labels are generated from the

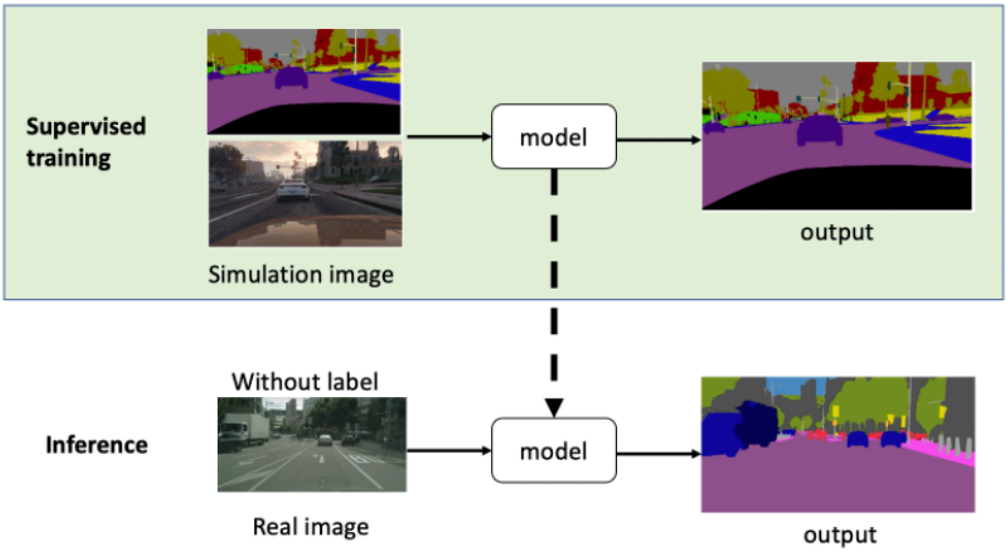

Figure 4.1 Overview of unsupervised domain adaptation. Training is performed on synthetic data that does not require annotation, and evaluation is performed on real images

model's predictions, assuming as correct those pixels where the predicted values exceed a threshold. The proposed method introduces an uncertainty loss that represents uncertainty at the pixel level for the target data, optimizing even for pixels with low confidence. Unlike pseudo-labels that constrain high-confidence pixels to maintain their confidence, uncertainty loss imposes constraints to increase the confidence of low-certainty pixels. Therefore, uncertainty loss can enhance robustness to noise not present in the source domain and improve the classification of similar classes.

While entropy has traditionally been used to measure uncertainty [56], it does not consider the probabilities of other classes, focusing only on individual class values. This thesis employs variance to consider the relative magnitudes between all class values. By accounting for the relationships between class values, the method can express uncertainty in more scenarios than methods that do not. For instance, where the probability vector for all classes is below $1/e$, entropy encourages learning to reduce all class values, whereas the proposed method encourages increasing the certainty of the

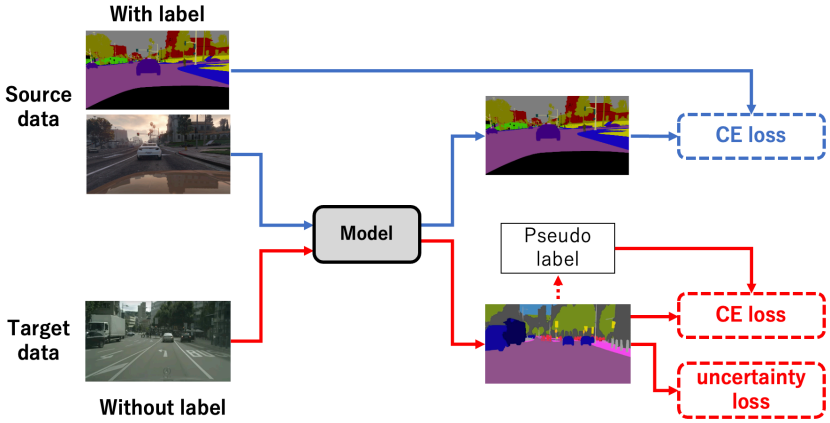

Figure 4.2 Model of the proposed method in unsupervised domain adaptation. Red arrows indicate the path of target data, while blue arrows represent the path of source data.

more reliable classes within the vector.

In this study, we reduce annotation costs by minimizing the uncertainty calculated from model outputs as a loss using a new unsupervised domain adaptation method, and enhance existing unsupervised domain adaptation methods by adding this approach, thereby creating a more robust model for the target domain and achieving state-of-the-art results.

4.1 Overview of Proposed Unsupervised Domain Adaptation

Unsupervised domain adaptation aims to adapt models trained on synthetic data to real-world data without the need for costly annotations on actual images. The training data comprises images and correct labels generated by a game engine, thereby eliminating the need for manual annotation. As shown in Figure 4.2, the red arrows indicate the path of the target data, while the blue arrows represent the path of the source data. In unsupervised domain adaptation, the labeled source data is trained in a manner similar to supervised learning. Simultaneously, for the unlabeled target data, pseudo-

labels are created based on the model's outputs, and these pseudo-labels are used as teachers for learning. In this study, for the target data, in addition to the loss from pseudo-labels, an uncertainty loss proposed in this study is added. The uncertainty loss constrains the model to generate high-confidence predictions by minimizing the uncertainty of the predictions, complementing the pseudo-label loss, which uses high-confidence pixels.

When creating pseudo-labels, only pixels whose maximum prediction values exceed a threshold are used; however, by introducing uncertainty loss, high-confidence predictions are encouraged across all pixels. While pseudo-labels aim to increase the certainty of already high-confidence pixels, uncertainty loss aims to increase the certainty of less confident pixels. Therefore, by employing uncertainty loss, the model is expected to become more robust to noise and classification of similar classes that might not be present in the source domain.

Our contributions are the following:

1. We propose a novel uncertainty-based loss for unsupervised domain adaptation that does not require costly annotation of real images.

2. We design the proposed method to complement the loss from pseudo-labels, achieving more accurate and robust semantic segmentation than conventional entropy-based loss methods.

4.2 Uncertainty Loss

Define the uncertainty loss applied to unsupervised domain adaptation in semantic segmentation. Let the output image of the model be denoted as x, and let $V_{i,j}$ be the probability vector at the pixel located at row i and column j of x. The uncertainty loss for this pixel can be formulated as follows.

$$\mathcal{L}_{uncertainty}(x) = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} \left(1 - \frac{\mathrm{MINVar}(V_{i,j})}{\mathrm{Var}(V_{i,j})} \times \max(V_{i,j}) \right) \tag{4.1}$$

Additionally, MINVar($V_{i,j}$) can be formulated as follows.

$$\mathrm{MINVar}(V_{i,j}) = \frac{1}{C}\left(\left(\frac{1}{C} - \max\left(V_{i,j}\right)\right)^2 + (C-1)\left(\frac{1}{C} - \frac{1-\max\left(V_{i,j}\right)}{C-1}\right)^2\right) \tag{4.2}$$

MINVar($V_{i,j}$) is the variance of a vector of length C, where the maximum value is the same as $V_{i,j}$, and the remaining $C-1$ elements are $1 - max(V)/(C-1)$.

This proposed uncertainty loss has the following two characteristics:

1. **Pixel-wise self-learning:** By calculating the uncertainty loss for all pixels, it is possible to optimize even for pixels that are more likely to be uncertain, such as near the boundaries between objects.

2. **Class-wise uncertainty:** By proposing an alternative to entropy for uncertainty, it becomes possible to consider the relative magnitudes of predictions across all classes, allowing for the representation of uncertainty in a greater variety of patterns.

4.3 Experiment

Comparisons are conducted by adding uncertainty loss to previous studies, HRDA [54] and DAFormer [53]. The source data used are GTA-5 [93] and Synthia [94], with Cityscapes [1] utilized as the target data. The evaluation metric employed is mIOU. Experiments are also conducted using entropy as a loss component, with a coefficient of 0.01 for uncertainty loss and 0.001 for entropy loss. The results presented are the averages of three experimental runs.

The results of experiments conducted using multiple methods and dataset combinations are presented in Table 4.1. For the GTA5 [93] to Cityscapes [1] scenario, applying uncertainty loss to DAFormer [53] increased 1.2 mIoU, and applying it to HRDA [54] increased 0.4 mIoU. For the Synthia [94] to

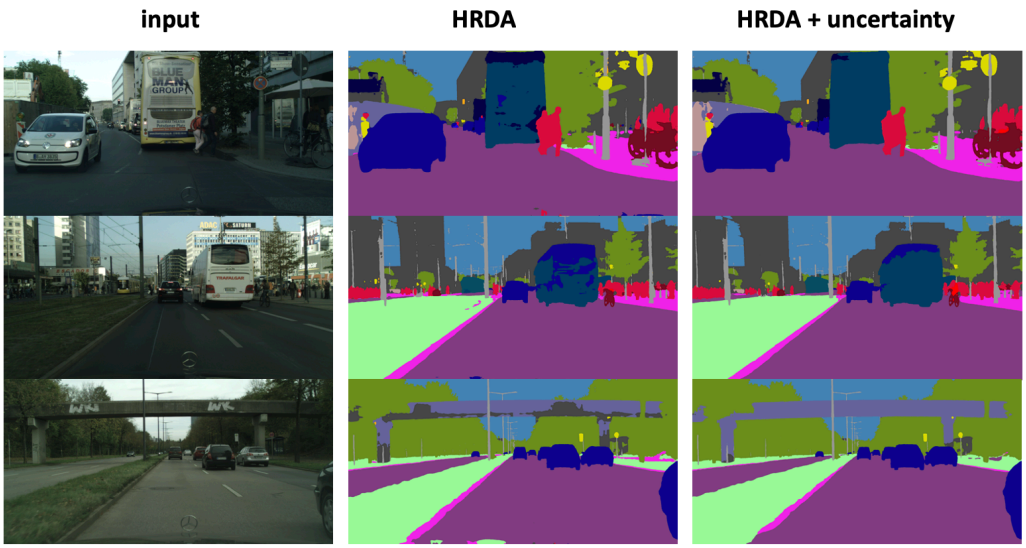

Figure 4.3 Results of qualitative comparison in unsupervised domain adaptation.

Cityscapes [1] scenario, applying uncertainty loss to DAFormer [53] increased 0.9 mIoU, and applying it to HRDA [54] increased 0.4 mIoU. Additionally, comparing the results of applying entropy to HRDA with those of the proposed uncertainty loss, entropy showed no change in mIoU, while the proposed method resulted in an improvement of 0.4.

Qualitative results are shown in Figure 4.3. It can be observed that the model has become more robust to challenging classes such as buses and graffiti on walls.

4.4 Conclusion

By using the uncertainty introduced in active learning as a loss in unsupervised domain adaptation, the model generates higher-confidence predictions than when trained solely with pseudo-labels, achieving more robust and accurate semantic segmentation results compared to traditional methods. Future work is expected to explore different uses of uncertainty, such as incorporating it into the generation of pseudo-labels.

Table 4.1 Experiment results across various methods and dataset combinations.

	Road	S.walk	Build.	Wall	Fence	Pole	Tr.Light	Sign	Veget.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	M.bike	Bike	mIoU
GTA–5 → Cityscapes																				
CBST [95]	91.8	53.5	80.5	32.7	21.0	34.0	28.9	20.4	83.9	34.2	80.9	53.1	24.0	82.7	30.3	35.9	16.0	25.9	42.8	45.9
DACS [96]	89.9	39.7	87.9	30.7	39.5	38.5	46.4	52.8	88.0	44.0	88.8	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.1
CorDA [97]	94.7	63.1	87.6	30.7	40.6	40.2	47.8	51.6	87.6	47.0	89.7	66.7	35.9	90.2	48.9	57.5	0.0	39.8	56.0	56.6
BAPA [98]	94.4	61.0	88.0	26.8	39.9	38.3	46.1	55.3	87.8	46.1	89.4	68.8	40.0	90.2	60.4	59.0	0.0	45.1	54.2	57.4
ProDA [99]	87.8	56.0	79.7	46.3	44.8	45.6	53.5	53.5	88.6	45.2	82.1	70.7	39.2	88.8	45.5	59.4	1.0	48.9	56.4	57.5
DAFormer [53]	95.7	70.2	89.4	53.5	48.1	49.6	55.8	59.4	89.9	47.9	92.5	72.2	44.7	92.3	74.5	78.2	65.1	55.9	61.8	68.3
DAFormer+UL	96.3	73.1	89.7	56.4	47.3	49.8	56.5	62.1	90.0	48.0	92.0	71.8	44.4	92.5	77.0	81.9	71.0	57.3	63.9	69.5
HRDA [54]	**96.4**	**74.4**	91.0	**61.6**	51.5	57.1	**63.9**	69.3	91.3	48.4	94.2	79.0	52.9	93.9	84.1	**85.7**	75.9	**63.9**	67.5	73.8
HRDA+Entropy	**96.4**	74.1	**91.3**	60.9	**53.3**	**59.3**	62.6	**71.1**	91.5	**49.0**	94.0	78.5	48.7	**94.2**	**84.7**	84.7	75.2	**64.6**	68.5	73.8
HRDA+UL	**96.4**	74.1	**91.3**	60.6	53.2	58.1	63.0	70.3	**91.6**	**49.0**	**94.4**	**79.3**	**55.1**	94.1	84.3	85.6	**77.2**	63.6	**68.7**	**74.2**
Synthia → Cityscapes																				
CBST [95]	68.0	29.9	76.3	10.8	1.4	33.9	22.8	29.5	77.6	–	78.3	60.6	28.3	81.6	–	23.5	–	18.8	39.8	42.6
DACS [96]	80.6	25.1	81.9	21.5	2.9	37.2	22.7	24.0	83.7	–	90.8	67.6	38.3	82.9	–	38.9	–	28.5	47.6	48.3
BAPA [98]	91.7	53.8	83.9	22.4	0.8	34.9	30.5	42.8	86.6	–	88.2	66.0	34.1	86.6	–	51.3	–	29.4	50.5	53.3
CorDA [97]	**93.3**	**61.6**	85.3	19.6	5.1	37.8	36.6	42.8	84.9	–	90.4	69.7	41.8	85.6	–	38.4	–	32.6	53.9	55.0
ProDA [99]	87.8	45.7	84.6	37.1	0.6	44.0	54.6	37.0	88.1	–	84.4	74.2	24.3	88.2	–	51.1	–	40.5	45.6	55.5
DAFormer [53]	84.5	40.7	88.4	41.5	6.5	50.0	55.0	54.6	86.0	–	89.8	73.2	48.2	87.2	–	53.2	–	53.9	61.7	60.9
DAFormer+UL	85.2	43.1	88.2	42.1	**7.5**	49.7	55.5	55.8	87.4	–	88.7	73.0	48.3	87.8	–	59.3	–	54.8	62.0	61.8
HRDA [54]	85.2	47.7	**88.8**	**49.5**	4.8	**57.2**	**65.7**	**60.9**	85.3	–	92.9	79.4	52.8	89.0	–	64.7	–	**63.9**	**64.9**	65.8
HRDA+UL	87.3	49.5	87.6	44.8	6.8	55.7	64.4	58.6	**88.4**	–	**94.2**	**80.0**	**56.1**	**90.3**	–	**68.1**	–	63.6	64.6	**66.2**

Chapter 5

Applying Pre-trained Diffusion Models to Semantic Segmentation

While active learning and unsupervised domain adaptation can improve annotation efficiency, semantic segmentation still requires extensive annotations and training. In addition, since supervised learning relies on a pre-defined set of categories, detecting extremely rare or even completely new classes during prediction becomes virtually impossible.

In this study, we address these limitations by combining two related tasks, namely unsupervised and open-vocabulary semantic segmentation. Unsupervised semantic segmentation [8, 59, 66] avoids costly annotation by leveraging representations obtained through a model [10, 58] which has been trained on a different task. On the other hand, open-vocabulary semantic segmentation [3, 60, 67, 100, 101] allows recognizing a wide array of categories through natural language and is not bound to a pre-defined set of categories. To integrate these two approaches, we leverage the internal feature space as well as the attention maps of a pre-trained Stable Diffusion model [15]. We propose new training, annotation-free open-vocabulary semantic segmentation methods, MaskDiffusion and Unsupervised MaskDiffusion by using a pre-trained Stable Diffusion model [15].

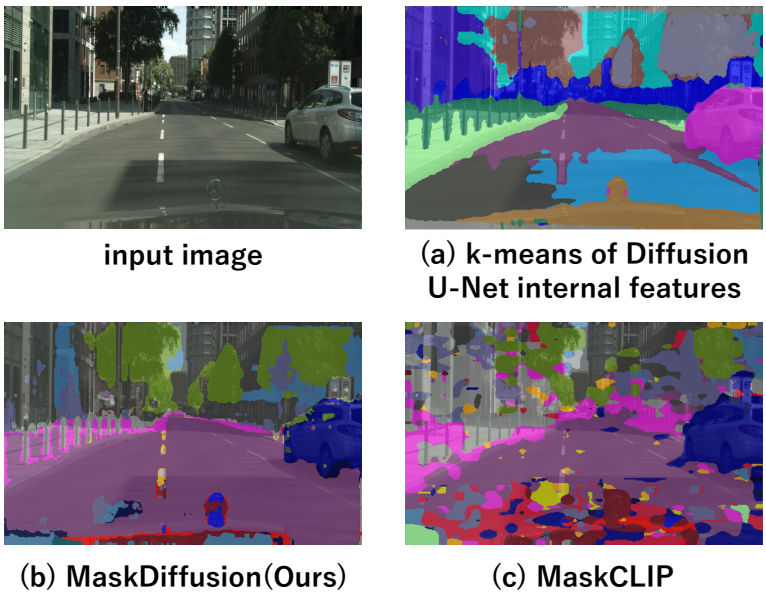

Figure 5.1 Comparison of MaskDiffusion with previous method MaskCLIP [3] on a Cityscapes [1] image. k-means clustering on the internal features of the Diffusion U-Net (a) shows that each determined cluster roughly partitions the image according to some classes, indicating that the semantic information is well preserved. MaskDiffusion (b) yields well-partitioned segments consistent with the shape of the object and exhibits minimal noise. In comparison, MaskCLIP [3] (c) results in smaller and noisy segments.

In this study, we analyze the internal features of pre-trained diffusion models to explore their potential applications in semantic segmentation, propose a new method that achieves effective segmentation without additional training or annotation, and demonstrate the superiority of our method over conventional techniques in performing open vocabulary semantic segmentation across various image domains.

5.1 Overview of the Proposed Method

Diffusion models, trained on large datasets and influenced by text embeddings from extensive pre-trained visual-language models like CLIP [11], have

achieved significant success in the field of image generation. These diffusion models have acquired a broad understanding of open-vocabulary concepts and are believed to be particularly useful for dense prediction tasks such as semantic segmentation. In ODISE [60], the internal features of a frozen diffusion model are utilized for open-vocabulary panoptic segmentation with additional training. Based on these observations, this research further investigates the use of diffusion model internals in semantic segmentation and proposes **MaskDiffusion**, which achieves effective segmentation without additional training.

The strengths of our MaskDiffusion are manifold. First, it eliminates the need for pixel-by-pixel annotations typically required by popular semantic segmentation techniques. Second, it can segment any class of objects, distinguishing it from traditional diffusion-based methods [60, 76, 82].

In MaskDiffusion, compared to traditional CLIP-based methods [3, 7], applying a diffusion model trained on 5B larger datasets allows for better prediction of less frequent classes and new classes while CLIP [11] is trained on 2B dataset. Additionally, CLIP [11] is trained to match images with corresponding text and does not inherently train at the pixel level. As demonstrated in methods like MaskCLIP [3] and GEM [7], segmentation is achieved by obtaining patch-level features without using the final attention pooling layer in CLIP [11]. In contrast, diffusion models generate images and calculate loss at the pixel level, making them more suitable for segmentation tasks. Especially for minor classes, diffusion models are particularly effective due to their ability to model diverse and detailed aspects of images at the pixel level. This capability stems from the iterative training process to denoise images, enabling the models to capture nuances and variations that might be overlooked by models not focused on pixel-level fidelity, such as CLIP [11]. [102] also mentions that diffusion models are particularly effective in attribute binding tasks, which are challenging for CLIP [11], demonstrating that diffusion models can integrate multiple pieces of information and understand more complex scenarios and abstract concepts.

As shown in Figure 5.1, applying k-means clustering on the internal features of the U-Net in Stable Difusion already provides a rough yet consistent

segmentation, whereas MaskCLIP [3] fails to uniformly segment the image. We are able to further improve segmentation results leveraging internal features with our proposed MaskDiffusion.

In addition, based on the observation that internal features are useful for segmentation, we propose an unsupervised segmentation method, called **Unsupervised MaskDiffusion**. Unsupervised MaskDiffusion, unlike MaskDiffusion, operates solely on image inputs without requiring class candidate prompts. This is why it is "Unsupervised". Unsupervised MaskDiffusion can segment objects of the same class by spectral clustering [4], computing a Laplacian matrix from the similarity between pixels of internal features.

Our contributions are the following:

1. We analyze the internal features of diffusion models and show that they are useful for semantic segmentation under the concept of no training. The results of k-means classification of internal features are comparable to those of unsupervised mIoU, a conventional unsupervised segmentation method.

2. We introduce MaskDiffusion and Unsupervised MaskDiffusion achieving compelling segmentation results for all categories in the wild without any additional training.

3. MaskDiffusion model outperforms GEM [7] by 10.5 mIoU on the Potsdam dataset [6], demonstrating the superior segmentation performance of our proposed approach. Moreover, the Unsupervised MaskDiffusion model surpasses the performance of DiffSeg [83] by 14.8 mIoU on the COCO-Stuff dataset [9], as measured by unsupervised mIoU.

5.2 Method

Our research attempts to explore the applicability of diffusion models to semantic segmentation. We begin with a brief introduction to the architecture of diffusion models in Sec 5.2.1, followed by a detailed description of

the proposed MaskDiffusion in Sec 5.2.2 and Unsupervised MaskDiffusion in Sec 5.2.3.

5.2.1 Diffusion Model Architecture

Stable Diffusion [15] follows the two processes in diffusion models, namely the diffusion process and the inverse diffusion process. The former involves the introduction of Gaussian noise to a clean image, while the latter employs a single UNet architecture to eliminate the noise and restore the original image. The diffusion and inverse diffusion processes each consist of 1000 steps, and a single UNet is responsible for all time steps. During training, we estimate the noise as

$$L_{LDM} := \mathbb{E}_{\varepsilon(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))\|_2^2\right], \tag{5.1}$$

where x is the input image, z and $\varepsilon(x)$ are the latent images, ϵ is the noise sampled from a normal distribution $\mathcal{N}(0,1)$, t is the time step, y is the text prompt, τ_θ is a pre-trained text encoder and $\epsilon_\theta(z_t, t, \tau_\theta(y))$ is the predicted noise, which is the output of the UNet.

A significant aspect of Stable Diffusion [15] lies in the utilization of a trained CLIP [11] as the text encoder, facilitating the conversion of textual inputs into corresponding vectors. Embedding these vectors into the UNet architecture enables the integration of text-based conditioning during the denoising process, achieved through cross-attention mechanisms. To preprocess y, the text prompt, τ_θ is introduced as a text encoder that projects y to an intermediate representation $\tau_\theta(y) \in \mathbb{R}^{M\times d_\tau}$, which is then mapped to the intermediate layers of the UNet via a cross-attention layer. Here M represents the number of tokens in the text prompt, and d_τ denotes the dimensionality of the text features. Attention consists of three vectors: query (Q), key (K), and value (V), and is defined as

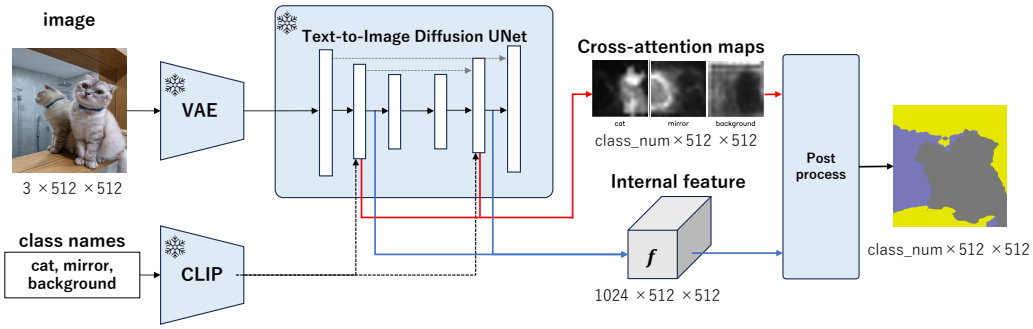

Figure 5.2 High-level overview of our MaskDiffusion architecture. MaskDiffusion uses a frozen pre-trained diffusion model. The UNet is given latent images compressed by a VAE as well as text prompts embedded by CLIP. The prompts are the names of all the potential classes to be segmented. The output of each layer of the U-Net is extracted as a concatenated internal feature $\mathbf{f}$ and a cross-attention map, which are subsequently post-processed into a segmentation image.

$$Q = W_Q^{(i)} \cdot \phi_i(z_t), K = W_K^{(i)} \cdot \tau_\theta(y), V = W_V^{(i)} \cdot \tau_\theta(y),$$
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V, \tag{5.2}$$

where d is the scaling factor, N is the number of tokens in the input sequence, $\phi_i(z_t) \in \mathbb{R}^{N \times d_i}$ is a (flattened) intermediate representation of the UNet implementing ϵ_θ and $W_V^{(i)} \in \mathbb{R}^{d \times d_i}$, $W_Q^{(i)} \in \mathbb{R}^{d \times d_\tau}$, $W_K^{(i)} \in \mathbb{R}^{d \times d_\tau}$ are projection matrices. The query is a vector created from the input image, while the key and value are created from the text vector.

5.2.2 MaskDiffusion

Figure 5.2 shows our proposed method which we call MaskDiffusion, a novel approach that leverages a pre-trained diffusion model as a semantic segmentation model, eliminating the need for additional training. MaskDiffusion leverages the observation that the intrinsic features extracted from the internal layer of the UNet in Stable Diffusion [15] inherently contain semantic

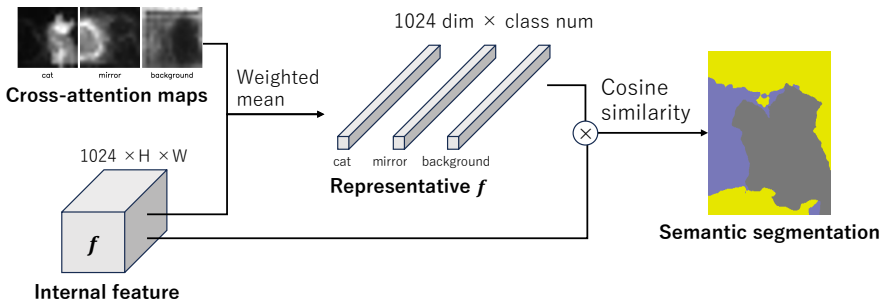

Figure 5.3 Overview of the post processing step. This post-processing module mathematically transforms the internal features $\mathbf{f}$ using the cross-attention map. First, a representative $\mathbf{f}$ is computed for each category through a weighted average of $\mathbf{f}$ based on the values of the cross-attention map. In the next step, we determine the semantic segmentation result by evaluating the cosine similarity between $\mathbf{f}$ and the representative $\mathbf{f}$ for each category and then assign each $\mathbf{f}$ to the category that has the closest similarity.

information, as demonstrated in the experimental results outlined in Section 5.2.5. For each pixel of the input image we obtain an internal feature, denoted as $\mathbf{f} \in \mathbb{R}^{1024}$, by upsampling and combining the output of the internal layers as follows,

$$\begin{aligned} \mathbf{f}_i &= \text{UNetLayer}_i(x_i, \tau_\theta(s)), \\ \mathbf{f} &= \text{Concatenate}(\mathbf{f}_1, ..., \mathbf{f}_n), \end{aligned} \tag{5.3}$$

where x_i denotes the input to the internal UNet layer, s represents the prompt and n is the number of internal UNet layers. Note that x_0 is the input image and x_1 and beyond are the output of previous layers.

Due to its high dimensionality, determining the class of each pixel solely based on $\mathbf{f}$ proves to be a challenging task. Therefore, we propose to incorporate the information contained in the cross-attention maps in the UNet architecture into $\mathbf{f}$ in order to assign a class to each pixel.

$$\text{Cross-attention map}(Q, K) = \frac{QK^T}{\sqrt{d}} \tag{5.4}$$

We define the cross-attention map as shown in Equation 5.4 based on the definition of the cross-attention in Equation 5.2. Note that we do not use the softmax function in Equation 5.4 as it introduces undesirable normalization across classes, which disrupts the intrinsic relationship between pixels and consequently leads to poorer performance.

The cross-attention map is calculated from Q and K. As shown in Equation 5.2, Q is generated from image features, and K is generated from text inputs (category candidates), allowing class labels to be assigned to the cross-attention.

Cross-attention establishes a relationship between the textual input prompts, which include the category names for segmentation, and the pixel-wise features of the image. In essence, the cross-attention map can be viewed as a preliminary semantic segmentation map, highlighting areas of potential object localization. However, using the cross-attention map as a standalone solution does not lead to accurate results. Instead, we leverage the cross-attention map in combination with the extracted high-dimensional internal features, $\mathbf{f}$, to assign each pixel a corresponding class.

Next, we explain how to incorporate the information of the cross-attention map into the internal features $\mathbf{f} \in \mathbb{R}^{1024 \times H \times W}$ as illustrated in Figure 5.3. First, we compute a representative internal feature $\bar{\mathbf{f}}_c \in \mathbb{R}^{1024}$ for each category c through a weighted average based on the values $a_{cnm} \in \mathbb{R}$ of the cross-attention map for each pixel position (n, m), as follows:

$$\bar{\mathbf{f}}_c = \frac{1}{A_c} \sum_{n,m} a_{cnm} \cdot \mathbf{f}_{nm}, \quad A_c = \sum_{n,m} a_{cnm} \tag{5.5}$$

In the subsequent step, the semantic map $\mathbf{s} \in \mathbb{R}^{H \times W}$ is obtained by calculating the cosine similarity for each pixel position (n, m) between the internal feature vector and the representative feature $\bar{\mathbf{f}}_c$ for each class and subsequently assigning the class with the highest similarity. Mathematically, this process is described as

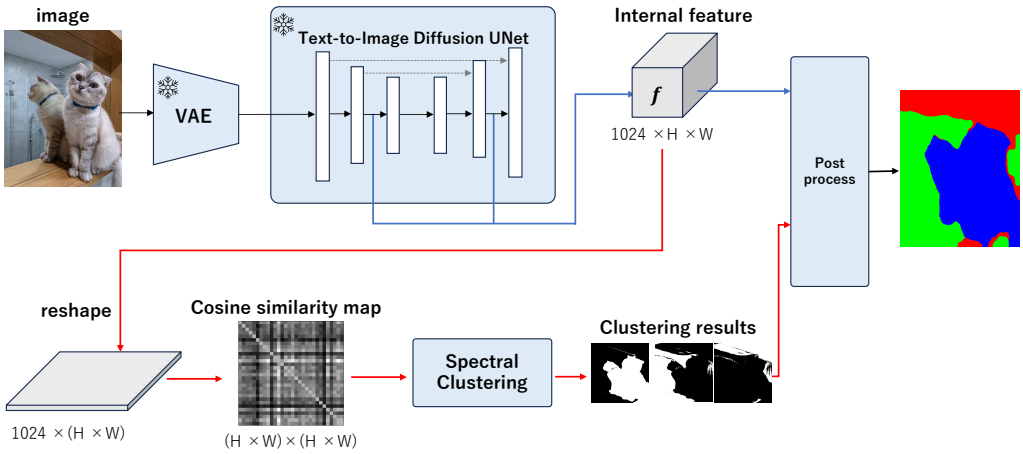

Figure 5.4 Overview of Unsupervised MaskDiffusion architecture. we employ spectral clustering [4] to include the spatial relationships of the internal features in the segmentation process.

$$s_{nm} = \underset{c}{\operatorname{argmax}} \frac{\mathbf{f}_{nm}^T \cdot \bar{\mathbf{f}}_c}{\|\mathbf{f}_{nm}\| \cdot \|\bar{\mathbf{f}}_c\|}, \tag{5.6}$$

and can be interpreted as assigning the category, represented by $\bar{\mathbf{f}}_c$, which exhibits the closest resemblance with the internal feature $\mathbf{f}_{nm}$ of the corresponding pixel.

5.2.3 Unsupervised MaskDiffusion

In our experimental results with MaskDiffusion, as detailed in Section 5.2.4, we found that the internal features seem to be quite powerful, while the cross-attention map demonstrates some limitations. As a consequence, we also explore an alternative approach that only uses internal features in the context of unsupervised segmentation. As such, this method does not require any prompts and only relies on images as input, which corresponds to the same setting as DiffSeg [83].

Unsupervised MaskDiffusion overview is shown in Figure 5.4. In order to cluster the obtained internal features in the text-free setting, we explore two different clustering methods, namely k-means and spectral clustering [4].

Table 5.1 Comparison of MaskCLIP [3], GEM [7], and our MaskDiffusion. We evaluate on Potsdam [6], Cityscapes [1], PascalVOC [5] and COCO-Stuff [9] and report the mIoU.

Method	Potsdam [6]	Cityscapes [1]	PascalVOC [5]	COCO-Stuff [9]
	6classes	19class	20class	171class
MaskCLIP [3]	10.4	15.5	28.6	1.6
GEM [7]	10.7	16.3	26.5	0.8
MaskDiffusion (Ours)	**21.2**	**17.1**	**29.9**	**5.5**

In particular, we employ spectral clustering [4] to include the spatial relationships of the internal features in the segmentation process. Based on the observation that internal features representing the same class should be close concerning the distance in pixel space, we compute the similarity map $\mathbf{h} \in \mathbb{R}^{HW \times HW}$ between the reshaped internal features $\hat{\mathbf{f}} \in \mathbb{R}^{HW \times 1024}$ as

$$\mathbf{h}_{ij} = \frac{\hat{\mathbf{f}}_\mathbf{i}^T \cdot \hat{\mathbf{f}}_\mathbf{j}}{\|\hat{\mathbf{f}}_\mathbf{i}\| \cdot \|\hat{\mathbf{f}}_\mathbf{j}\|}, \tag{5.7}$$

where i and j denote the pixel location. A large value of $\mathbf{h}_{ij}$ in this map indicates that the pixels at i and j very likely belong to the same class. Note that we set H and W to 100 and accordingly resize $\mathbf{f}$ to 100×100 to reduce computational complexity.

Next, we construct a Laplacian matrix from the similarity map and classify clusters based on their smallest eigenvectors. The clustering results are then post-processed using $\mathbf{f}$, where the clustering results replace the cross-attention map in Equation 5.5 to produce the final segmentation image. Note that the values of the spectral clustering [4] are either 0 or 1, whereas the cross-attention map consists of values between 0 and 1. One of our novelty lies in the adoption of spectral clustering with appropriate motivation.

5.2.4 Main Results

To validate the performance of MaskDiffusion, we conducted comprehensive comparative experiments with its closest counterpart, MaskCLIP [3] and

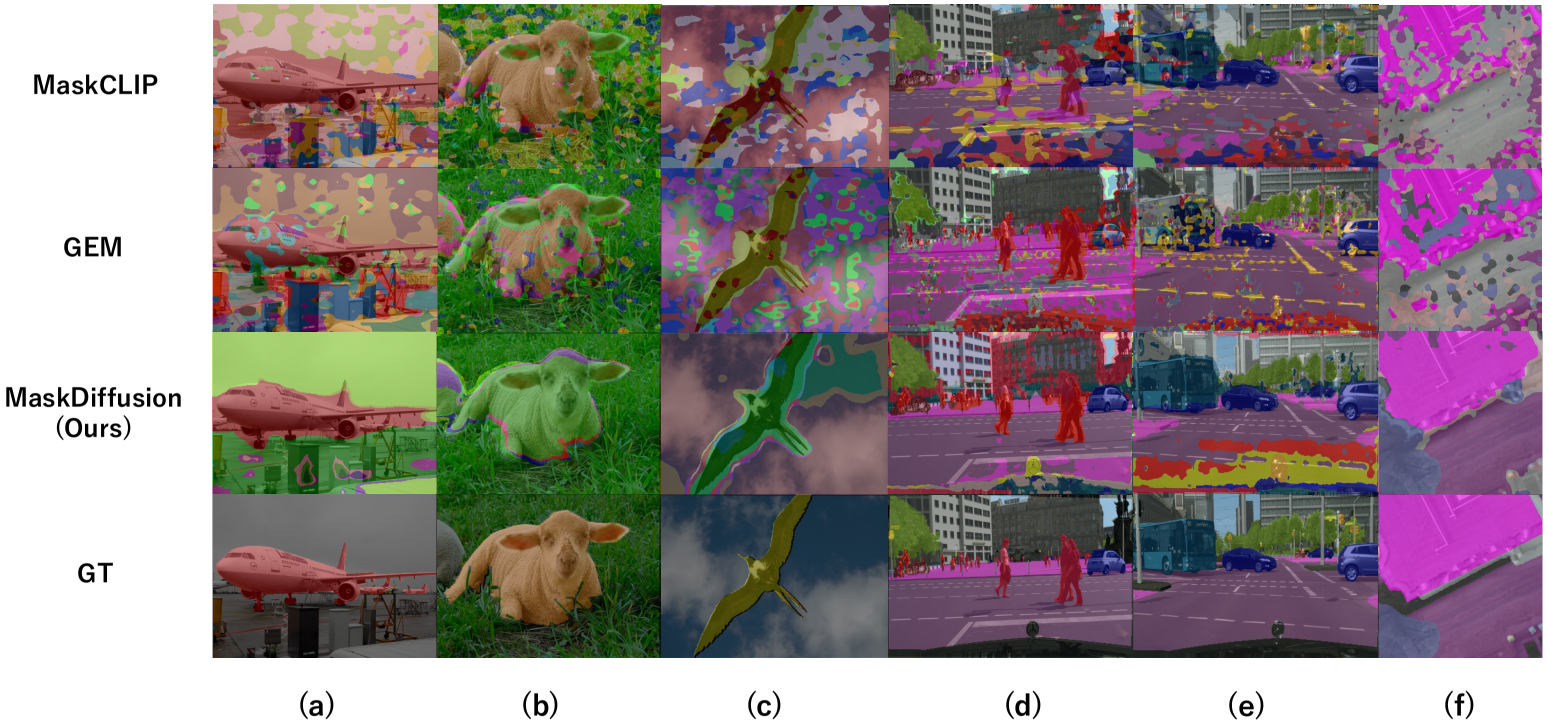

Figure 5.5 **Qualitative results.** Images (a) to (c) depict scenes from PascalVOC [5], images (d) and (e) represent scenes from Cityscapes [1], and image (f) shows scene from Potsdam [6]. We compare MaskCLIP [3], GEM [7], and our proposed MaskDiffusion. MaskCLIP [3] and GEM [7] provide fragmented and noisy segmentations, whereas MaskDiffusion exhibits a more cohesive and accurate segmentation for each object.

GEM [7]. For the evaluation, we used the mean Intersection over Union (mIoU) as the primary metric. We perform our experiments on the Potsdam [6] dataset comprising of 6 classes, Cityscapes [1] with 19 classes, and PascalVOC [5] with 20 classes, as well as COCO-Stuff [9] consisting of 171 classes. Note that we input the names of all the classes included in the dataset as prompts.

The quantitative results are presented in detail in Table 5.1. Our method outperforms previous research across all datasets, particularly achieving a +10.5 mIoU on the Potsdam [6] dataset, a land cover task dataset, demonstrating the robustness of the internal features for class estimation over CLIP-based approaches.

The qualitative results are shown in Figure 5.5. Images (a) to (c) from PascalVOC [5], (d) and (e) from Cityscapes [1], and (f) from Potsdam [6] compare MaskCLIP [3], GEM [7], and our MaskDiffusion. MaskCLIP [3] and GEM [7] result in a noisy and fragmented segmentation, whereas MaskDiffusion achieves more consistent segments that better correspond with the shape of the object and exhibit minimal noise.

To validate its performance, we conducted experiments comparing our Unsupervised MaskDiffusion to its closest counterpart, DiffSeg [83] under the same conditions of image unit segmentation as specified in their paper [83].

In unsupervised segmentation, the categories are not explicitly labeled. Therefore, it is necessary to align the unlabeled clusters with the ground truth for evaluation purposes. Following other unsupervised segmentation methods [8, 59, 66, 83], we quantify the performance of each method by employing the unsupervised mIoU, which matches unlabeled clusters with ground-truth labels using a Hungarian matching algorithm. The Hungarian matching algorithm is used to find the optimal matching in a bipartite graph. It calculates the cost between each cluster and each label, aiming to minimize the overall cost. Based on the obtained matching, the IoU between each cluster and its corresponding ground truth label is calculated, and the average of these values is taken to obtain the unsupervised mIoU.

We test on Cityscapes [1] with 19 and 27 classes and COCO-Stuff [9] with 27 classes. For a fair comparison, we input an empty string as a prompt.

Table 5.2 Comparison of Unsupervised MaskDiffusion with state-of-the-art unsupervised segmentation methods. We use the Cityscapes [1] and COCO-Stuff [9] datasets for comparison and report the mIoU. The clear boost in performance by including the internal features underlines their semantic richness and effectiveness.

Method	Backbone pretraining	Training free	Cityscapes [1]		COCO-Stuff [9]
			19class	27class	27class
IIC [65]	-	-	-	6.4	6.7
PiCIE [66]	-	-	-	12.3	13.8
STEGO [8]	DINO [58]	-	-	21.0	28.2
HP [59]	DINO [58]	-	-	18.4	24.6
k-means	DINOv2 [10]	√	20.5	19.3	33.8
k-means	SD [15] Internal feature	√	23.3	20.1	35.0
DiffSeg [83]	SD [15] self-attention	√	-	21.2	43.6
MaskDiffusion (U)	SD [15] Internal feature	√	**28.5**	**25.3**	**58.4**

The experimental results are shown in Table 5.2, which confirms that our proposed Unsupervised MaskDiffusion outperforms conventional segmentation methods using diffusion models and further underscores the effectiveness of internal features and their similarity map. Additionally, the results of k-means and SD internal features indicate that spectral clustering is more suitable for Unsupervised MaskDiffusion.

The qualitative results of STEGO [8] and Unsupervised MaskDiffusion are shown in Figure 5.6. While STEGO [8] produces noisy results, Unsupervised MaskDiffusion yields well-partitioned segments consistent with the shape of the object, especially on the COCO-Stuff [9] dataset. For the Cityscapes [1] dataset, it can also be observed that Unsupervised MaskDiffusion can distinguish between similar categories, such as buses and cars.

5.2.5 Internal Feature

In this section, we experimentally investigate the role of the internal features, which are represented by 1024-dimensional vectors for each pixel. To prove

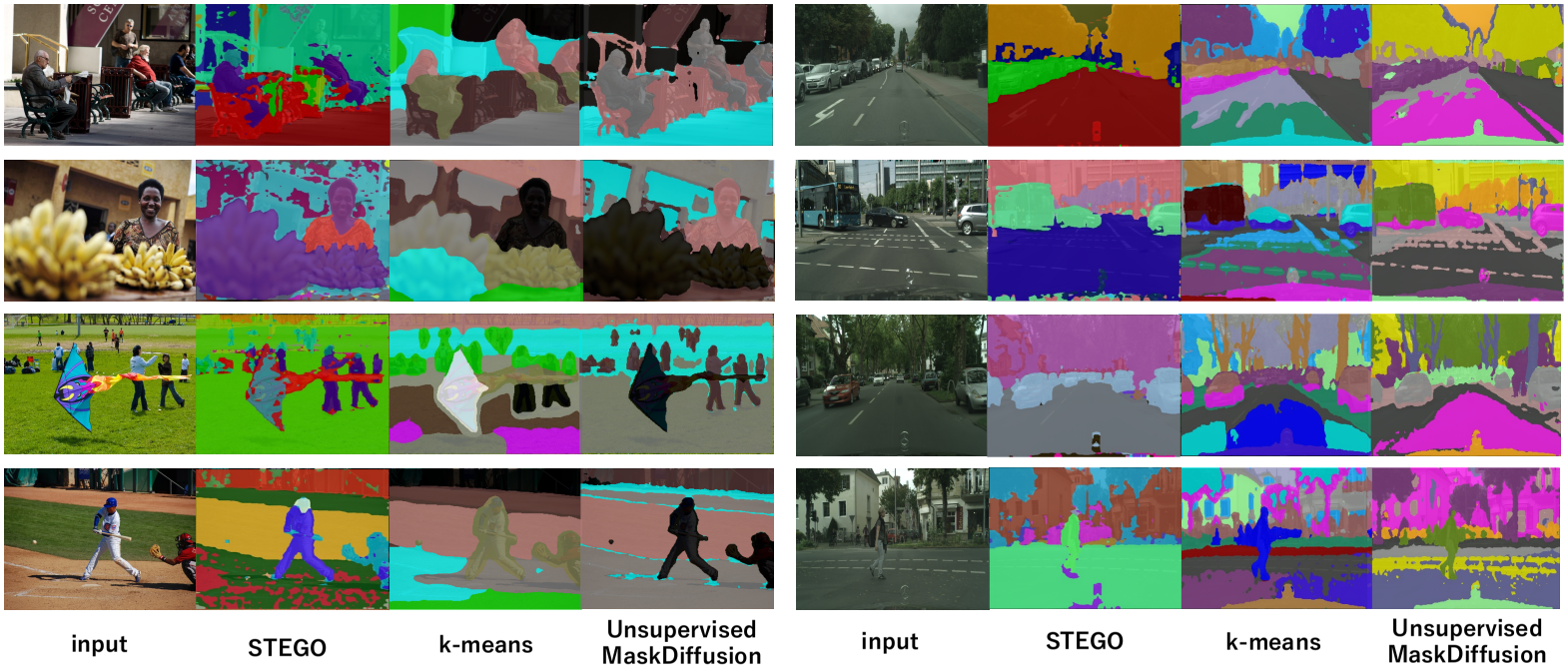

Figure 5.6 Overview result of Unsupervised MaskDiffusion, k-means on internal features and previous method STEGO [8] with COCO-Stuff [9] and Cityscapes [1]. It demonstrates that Unsupervised MaskDiffusion produces qualitatively cleaner results.

that these internal features hold sufficient semantic information, we conduct k-means segmentation on a pixel-wise basis. To that end, we choose a specific number of k-means clusters for each dataset.

For comparison, we use common unsupervised segmentation methods [8, 59, 65, 66] as well as k-means on DINOv2 [10]. The experimental setup is the same as in the previous section.

The results of our experiments are presented in Table 5.2, highlighting the strong performance of the internal features across diverse datasets. Particularly, the results demonstrate that applying k-means on the internal features significantly outperforms traditional unsupervised segmentation, for example by 6.8 mIoU for the COCO-Stuff dataset. Moreover, the performance exceeds that of DINOv2 k-means across all datasets, showcasing the robust capabilities of the internal features without the need for specific training. Collectively, our findings establish the semantic nature of the internal features.

Figure 5.6 shows the clustering results of Unsupervised MaskDiffusion and internal features via k-means. It demonstrates that Unsupervised MaskDiffusion can more cleanly classify identical classes into the same segment and different classes into separate segments, compared to k-means of internal feature.

5.2.6 Open Vocabulary Segmentation on Web-Crawled Images

In this section, we thoroughly assess the performance of MaskDiffusion using open vocabulary segmentation experiments on web-crawled images, where we test the model's ability to accurately segment various unseen classes, including specific and detailed categories such as 'joker' and 'porsche'.

The qualitative outcomes of the experiments are visually depicted in Figure 5.7. In this figure, (a) represents a general image, while (b) and (c) showcase images created by Stable Diffusion. Furthermore, (d) and (e) exhibit images containing proper nouns, with the respective class entered as a prompt displayed below each image. Our results highlight the successful

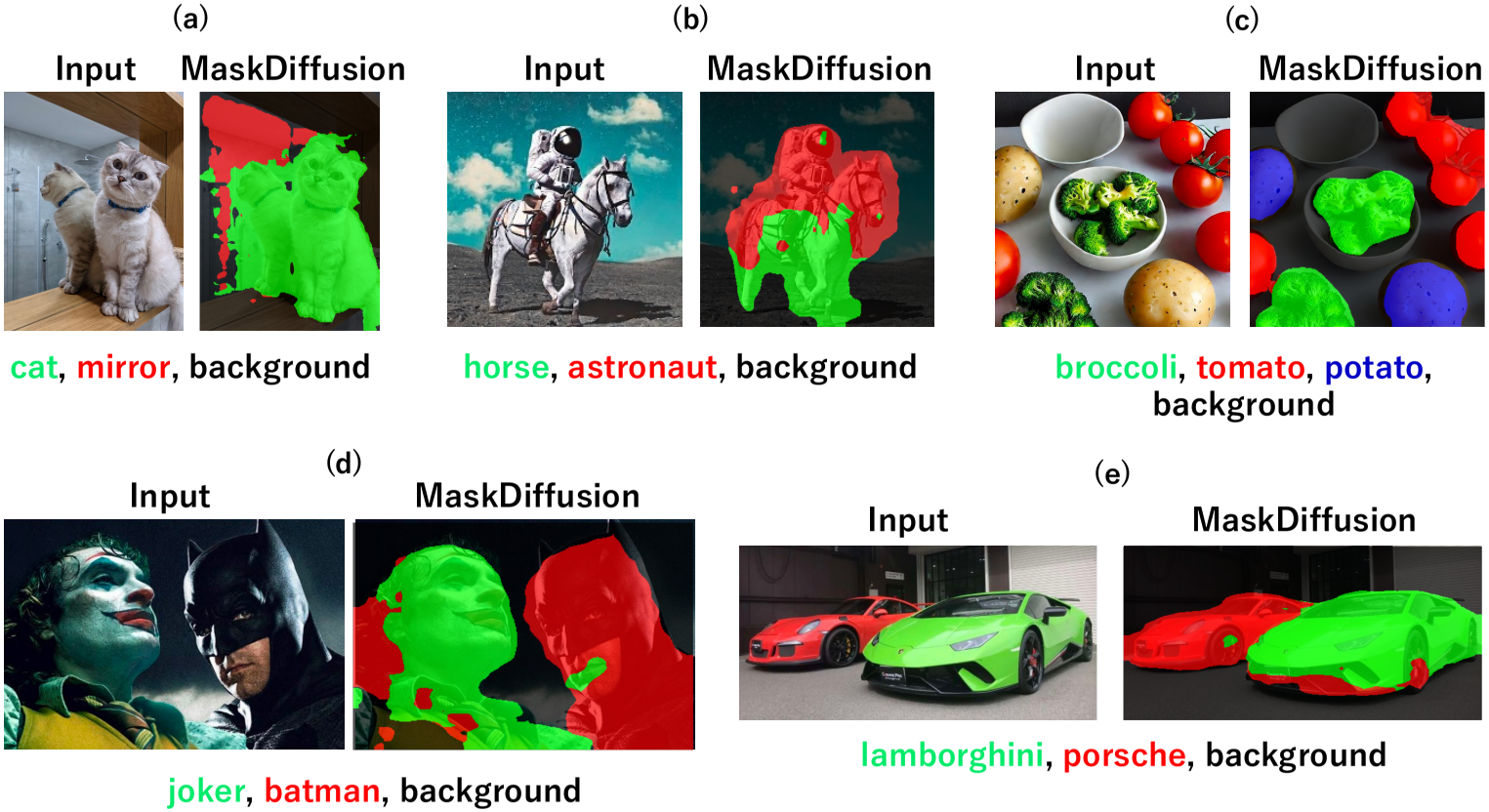

Figure 5.7 **Open-vocabulary segmentation results.** In (a) we test on a general image, (b) and (c) show images generated by Stable Diffusion, and (d) and (e) are images featuring specific proper nouns. The corresponding prompt for each class is displayed below the respective image. The successful segmentation of challenging concepts, rare classes, and proper nouns highlights the effectiveness of MaskDiffusion in handling diverse segmentation tasks. Note that these word candidates function properly even with similar alternative words.

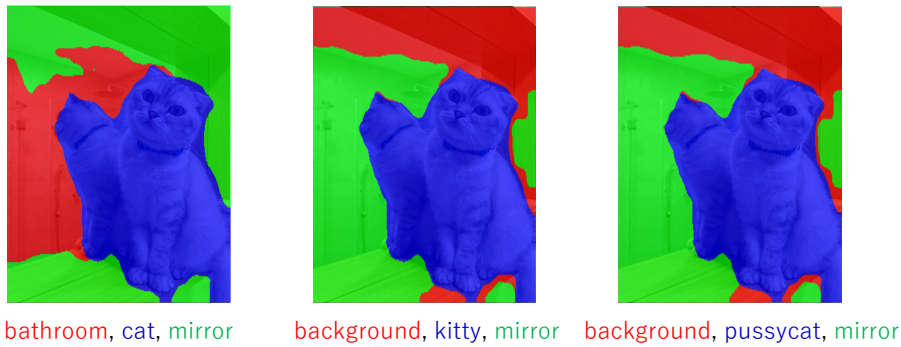

Figure 5.8 Open Vocabulary Segmentation with similar words. This demonstrates that Open Vocabulary Segmentation can accurately segment even when the given class candidates are similar words.

segmentation of challenging concepts such as 'mirror' in (a) and rare segmentation tasks such as 'astronaut' in (b). Additionally, the model demonstrates the capability to identify general classes, as depicted in (c), indicating that its segmentation performance improves with more general classes. Impressively, the segmentation of proper nouns is also achievable, as evidenced in results (d) and (e). These findings serve as compelling evidence that MaskDiffusion exhibits a robust capability for accurately segmenting open vocabularies, including complex and specific categories, thus underscoring its versatility and effectiveness in handling diverse and intricate segmentation tasks.

Furthermore, Figure 5.8 demonstrates that MaskDiffusion can accurately segment even when the given class candidates are similar words, as shown with the image from Figure 5.7 (a). When the candidates are 'mirror' and 'background', the mirror area is segmented as 'mirror', When the candidates are 'bathroom' and 'mirror', the area showing the bathroom is prioritized for segmentation, while the other areas are segmented as 'mirror'.

5.2.7 Ablation Study

In this section, we perform ablation studies on MaskDiffusion, examining how various Stable Diffusion [15] time steps and UNet outputs for internal features affect performance, using the 19class Cityscapes [1] dataset.

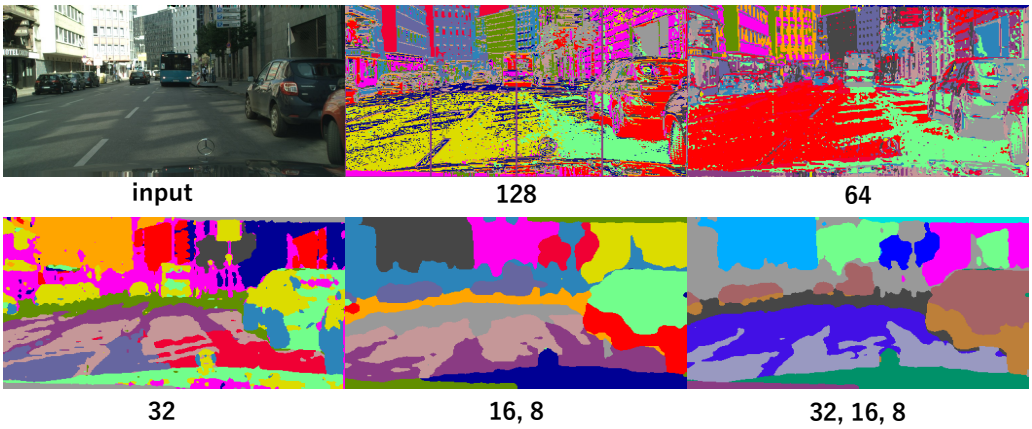

Figure 5.9 Ablation study on internal feature size. Qualitative results of internal features k-means show that earlier layers of the UNet (128, 64) do not provide an appropriate segmentation, whereas the inner layers (32, 16, 8) produce semantic meaningful results.

Table 5.3 presents the results of the ablation experiments comparing the effect of using MaskDiffusion with different time steps on the mIoU. The results reveal that employing a time step of $t = 1$ yields the most favorable outcome. Typically, in Stable Diffusion [15], larger values of t are employed when restoring images closer to Gaussian noise, requiring an attention map that broadly identifies the location of the object. Conversely, smaller values of t are utilized to restore images closer to the original image, thereby producing an attention map that closely resembles the shape of the object in the real image.

Furthermore, we conduct ablation experiments on the cross-attention map and internal features, which involve examining the concatenated outputs of different layers. Considering the structure of UNet, the outer intermediate outputs are of larger size compared to the smaller inner intermediate outputs. Therefore, we compare various combinations at different positions, which are detailed in Table 5.4. Overall, combining the cross-attention maps with a resolution of 8×8 and 16×16 yields the best results.

Similarly, we examine the effect of combining different internal feature sizes by applying k-means and evaluating the resulting segmentation with the unsupervised mIoU. The quantitative results are summarized in Table 5.6,

Table 5.3 Ablation study on Diffusion time step. We evaluate MaskDiffusion on the Cityscapes dataset [1] using mIoU.

t	1	10	100	500
mIoU	**17.1**	5.5	2.4	0.9

Table 5.4 Ablation study on output size of the cross-attention map. We evaluate MaskDiffusion on the Cityscapes dataset [1] using mIoU.

Cross-attention map size			mIoU
64	32	16,8	
√	√	√	12.5
	√	√	14.5
		√	**17.1**

indicating that combining feature maps of size 8, 16, and 32 clearly outperforms other size combinations. Figure 5.9 presents the qualitative results of the ablation study. Since earlier layers of the U-Net typically generate low-level features, the output is less suitable for segmentation, whereas inner outputs provide more semantic information.

These findings highlight the critical role that both, the time step and the internal features, play in the segmentation performance of MaskDiffusion.

Next, to explore the limitations of the internal features $\mathbf{f}$, we created representative $\mathbf{f}$ using images and ground truth on the training dataset. In this experiment, $\mathbf{f}$ is first created from images in the training dataset, and then the $\mathbf{f}$ and ground truth are used to create a representative $\mathbf{f}$. Representative $\mathbf{f}$ is the mean of $\mathbf{f}$ for each class in ground truth. In the same context, we also conduct an experiment using only the classes contained in the images as prompt input. The evaluation results are shown in Table 5.5.

Our method achieves 30-60% of the performance compared to when calculating representative $\mathbf{f}$ with ground truth, without training from the ground truth. Furthermore, it is observed that using dynamic prompts leads to

Table 5.5 Additional experiment results of MaskDiffusion. we conduct an experiment calculating representative $\mathbf{f}$ with ground truth and using only the classes contained in the images as prompt input (dynamic prompts).

Method	Cityscapes [1]	PascalVOC [5]	COCO-Stuff [9]
	19class	20class	27class
MaskDiffusion	17.1	29.9	13.0
MaskDiffusion w/ ground truth	35.0	53.7	39.8
MaskDiffusion w/ dynamic prompts	21.6	87.2	40.5

improved performance in proportion to the number of classes per image. Cityscapes [1], COCO-Stuff [9] contain about 10 objects per image, whereas VOC [5] contains a few objects. mIoU increase is largely influenced by the number of objects in one image, especially with dynamic prompts.

5.3 Limitation

While MaskDiffusion achieves remarkable results, especially compared to related approaches, our proposed method still comes with certain limitations. First, the cross-attention map has a low level of assigning internal features to each class. This can be seen in Figure 5.5(b), where the segmentation is clean but the class assignment is incorrect. Second, our setting assumes that potential candidates for the classes appearing in the images are known beforehand. When category candidates are insufficient, it becomes challenging to perform accurate segmentation. While we consider this to be outside of the scope of this study, it would be possible to use MaskDiffusion in conjunction with models that are able to detect the presence of objects in images, e.g. CLIP [11], to solve this limitation.

Table 5.6 Ablation study on U-Net layers to be adopted as internal features. We perform k-means on various U-Net layer outputs and evaluate on the Cityscapes dataset [1] using the unsupervised mIoU.

Internal feature size				Unsupervised
128	64	32	16,8	mIoU
√				13.8
	√			14.8
		√		17.2
			√	20.3
√	√	√	√	14.8
	√	√	√	14.8
		√	√	**23.3**

5.4 Conclusion

In this study, we proposed MaskDiffusion, a novel approach for semantic segmentation that leverages Stable Diffusion and internal features to achieve superior segmentation results. Through a series of comprehensive experiments and analyses, we have demonstrated the effectiveness and versatility of MaskDiffusion across various datasets and challenging segmentation tasks. Our results indicate that MaskDiffusion exhibits robust performance in handling diverse categories, including general classes and fine-grained, proper noun-based segments.

Overall, our findings highlight the potential of MaskDiffusion as a powerful and effective tool in the field of semantic segmentation. By leveraging the strengths of Stable Diffusion and internal features, we have successfully demonstrated the model's capability to handle diverse datasets and open vocabularies, paving the way for future advancements and applications in this critical area of computer vision.

Chapter 6

Guidance-free Open-Vocabulary Semantic Segmentation

Unsupervised semantic segmentation [8, 59, 66] avoids the expensive annotation process by using representations obtained through a backbone model [10, 58] trained on a different task. Open-vocabulary semantic segmentation [3, 60, 67, 100, 101] enables the identification of a wide array of categories through natural language and is not bound to a pre-defined set of categories.

However, there are still challenges to be solved with these methods. Unsupervised semantic segmentation clusters images by class but cannot identify the class of each cluster, while open-vocabulary segmentation assumes that text queries describing objects in the image are provided by the user. To address these challenges, zero-guidance segmentation emerged in [13], enabling open-vocabulary segmentation without the need for inputting class candidates (guidance), yet there is still room for improvement in terms of performance.

Based on these backgrounds, we further improved this approach by introducing a novel method named TAG, which offers higher performance and flexibility. As its primary strength, TAG achieves **T**raining, **A**nnotation, and **G**uidance-free open-vocabulary semantic segmentation. This method

employs a novel approach by extracting semantic features from each pixel in an image using CLIP [11], and then retrieving the open-vocabulary classes based on these features from an external database [17, 18, 103, 104]. TAG operates using pre-trained frozen models CLIP [11] and DINOv2 [10], eliminating the need for an additional training process. Our TAG can segment an image into meaningful segments as shown in Figure 6.3 without any text guidance.

In this study, we propose a new approach that achieves open-vocabulary semantic segmentation without reliance on predefined classes, additional training, annotation, or guidance by searching cluster categories from an external database, and demonstrate the superiority of our method over traditional techniques in performing open vocabulary semantic segmentation across various image domains.

6.1 Overview of the Proposed Method

TAG, **T**raining, **A**nnotation, and **G**uidance-free open-vocabulary semantic segmentation, employs a novel approach by extracting semantic features from each pixel in an image using CLIP [11], and then retrieving the open-vocabulary classes based on these features from an external database [17, 18, 103, 104]. CLIP [11] can identify diverse objects and scenes while its segmentation results are often coarse and noisy, necessitating refinement. We use DINOv2 [10] to excel in capturing fine details and global context, enabling precise segmentation. Combining these models leverages CLIP [11]'s generalization and DINO [10]'s detailed feature extraction for more accurate segmentation. These models do not utilize the dense and costly annotations traditionally required for semantic segmentation.

Furthermore, through the extensibility of its database, this method also incorporates flexibility, making it easy to adapt to new classes or scenarios. A major distinction between previous methods [13, 62], and our TAG is that it provides more flexibility as it can be extended to include new concepts by adding them to the database while previous methods require re-training. It

is important to note that while the database used in TAG is finite, language models like BLIP [63] or GPT [61] are also constructed from similarly finite datasets. In [89], it is even reported the retrieval-based methods provided superior results over BLIP [63] in the context of image classification.

Our TAG can segment an image into meaningful segments as shown in Figure 6.3 without any text guidance. In particular, TAG is able to accurately segment structures with their proper nouns, such as the *Leaning Tower of Pisa* and the *Coliseum*. In addition, TAG shows significant improvements in contrast to other comparable segmentation methods, i.e. on the PascalVOC [5] dataset (+15.3 mIoU).

Our contributions are the following:

1. We propose a novel approach, namely TAG, to achieve open-vocabulary semantic segmentation that does not require pre-defined categories by retrieving segment categories from an external database.

2. TAG achieves compelling segmentation results for all categories in the wild without any additional training, high-cost dense annotation, or text query guidance.

3. TAG outperforms the previous state-of-the-art methods by 15.3 mIoU on the PascalVOC [5] dataset, demonstrating the superior segmentation performance of our proposed approach.

6.2 Method

Figure 6.1 shows an overview of our proposed method which we call TAG, a novel approach. Our TAG attempts to partition input images into semantic segments and label each segment with open-vocabulary categories. To this end, we propose to identify segment candidates using per-pixel features obtained from DINOv2 [10] (Sec. 6.2.1), acquire representative segment embeddings for segment candidates using per-pixel features from a ViT pre-trained with CLIP [11] (Sec. 6.2.2), and assign categories to each candidate segment by retrieving the closest matching sentence from an external database

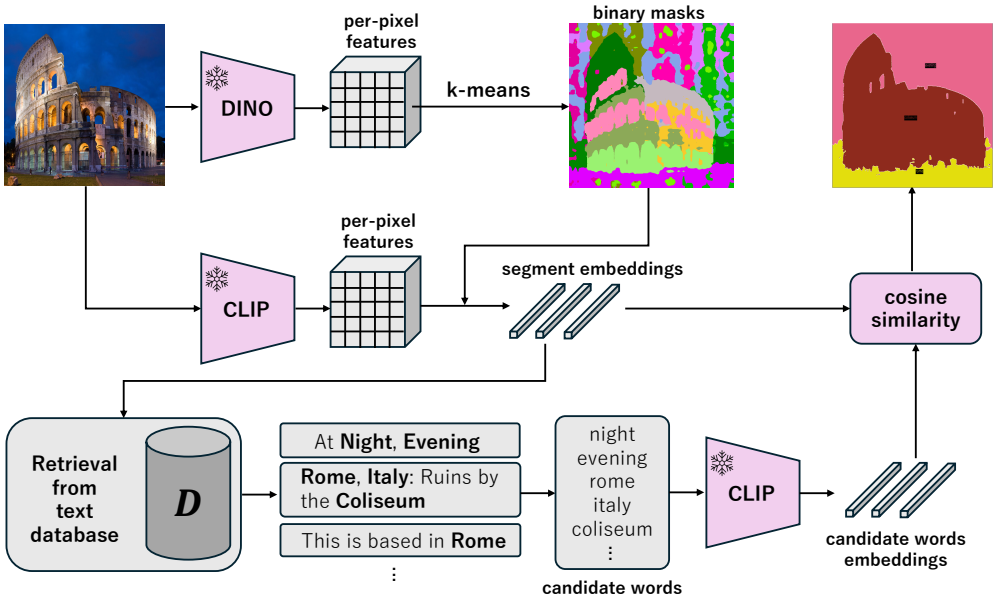

Figure 6.1 **High-level overview of our TAG architecture.** Our TAG can partition images into semantic segments and label each segment with open-vocabulary categories. First, TAG identifies segment candidates using per-pixel features obtained from DINOv2 [10]. Then, it acquires representative segment embeddings for segment candidates using per-pixel features from a ViT pre-trained with CLIP [11]. Finally, the categories are assigned to each candidate segment by retrieving the closest matching sentence from an external database. Note that the input is only the image, with no need to input category candidates as guidance.

(Sec. 6.2.3). Note that, unlike traditional open-vocabulary semantic segmentation, the input is only the image, with no need to input category candidates as guidance.

6.2.1 Segment Candidates with DINO

It has been observed that segmentation results obtained from CLIP-based segmentation methods [3, 7] are fragmented and noisy as shown in Figure 6.3. Therefore, the first step in our TAG pipeline is calculating segmentation candidates to achieve more accurate segmentation results. To obtain more pre-

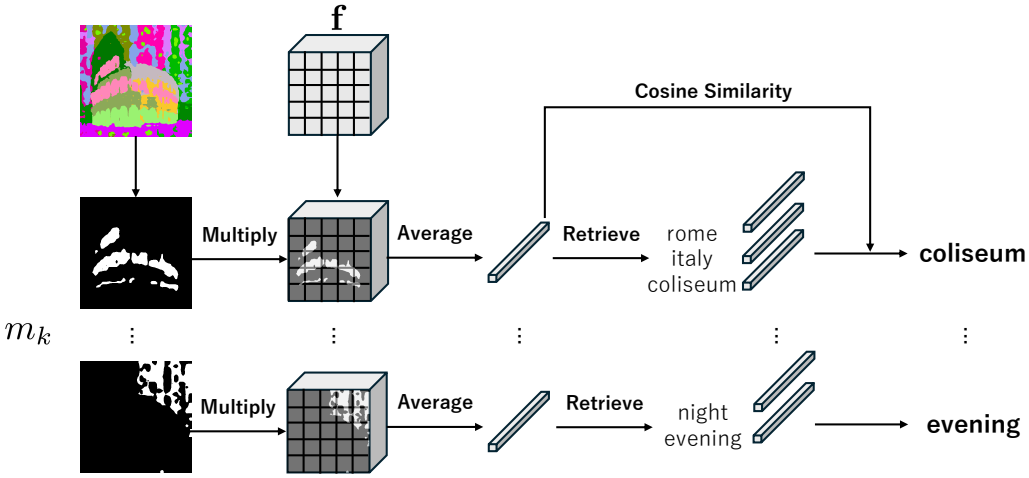

Figure 6.2 Overview of the flow for each segment. Each segment independently retrieves for category candidates and assigns a category.

cise segmentation outcomes than CLIP-based methods without using dense annotations, we reference unsupervised segmentation methods [8, 59] and employ a ViT pre-trained with DINOv2 [10].

The output of DINOv2 [10] is a feature map $\in \mathbb{R}^{D \times \frac{H}{P} \times \frac{W}{P}}$, where D is the dimension of the feature, P is the patch size of the transformer, and H and W are image sizes. This feature map will be upsampled to per-pixel features $\in \mathbb{R}^{D \times H \times W}$. Once the per-pixel feature is obtained, to assign categories for each segmentation candidate, we use k-means to divide the per-pixel features into segmentation candidates, resulting in oversegmentation.

6.2.2 Representative Segment Embeddings with CLIP

CLIP [11] is a ViT model that can embed images and text into the same latent space. To assign natural language categories to each segment, we use CLIP [11] to embed the image at the pixel level.

Instead of directly acquiring pixel-level features from CLIP [11], we extract dense patch-level features from the image encoder of CLIP [11] following CLIP-based segmentation methods [3, 7]. The image encoder of CLIP [11] uses a multi-head attention layer, where the globally average-pooled feature

works as the query, and the feature at each patch generates a key-value pair. Then, this layer outputs a spatial weighted sum of the incoming feature map followed by a linear layer $F(\cdot)$:

$$\begin{aligned}\mathrm{AttnPool}(\overline{q}, k, v) &= F\left(\sum_i \mathrm{softmax}\left(\frac{\overline{q}k_i^T}{C}\right) v_i\right) \\ &= \sum_i \mathrm{softmax}\left(\frac{\overline{q}k_i^T}{C}\right) F(v_i),\end{aligned} \tag{6.1}$$

$$\overline{q} = \mathrm{Emb}_q(\overline{x}), k_i = \mathrm{Emb}_k(x_i), v_i = \mathrm{Emb}_v(x_i), \tag{6.2}$$

where C denotes a constant scaling factor and $\mathrm{Emb}(\cdot)$ represents a linear embedding layer. x_i is the input feature at patch i and $\overline{x}$ is the average of all x_i. The Transformer layer in CLIP [11] outputs a detailed image representation, made possible because $F(v_i)$, computed at each spatial location, captures a rich response of local semantics.

Based on this observation, we utilize the features from the last attention layer of CLIP [11] image encoder by adopting the GEM [7] mechanism.

CLIP model in TAG outputs value features $\in \mathbb{R}^{D \times \frac{H}{P} \times \frac{W}{P}}$, where D is the dimension of the feature, P is the patch size of the transformer. These features contain dense representations of the image, capturing patch-level information, which we upsample to per-pixel features $\in \mathbb{R}^{D \times H \times W}$, corresponding to the same size as the features obtained from DINO [10].

Next, to assign categories to segment candidates, we calculate embedding features representing the segments from CLIP [11] per-pixel features $\mathbf{f} \in \mathbb{R}^{D \times H \times W}$ as shown in Figure 6.2. For each segment k, this representative segment embedding $\bar{\mathbf{f}}_k \in \mathbb{R}^D$ is computed by averaging based on the values $m_{khw} \in \{0, 1\}$, which results from applying k-means to the output of the DINO [10] with k classes, as follows:

$$\bar{\mathbf{f}}_k = \frac{1}{M_k} \sum_{h,w} m_{khw} \cdot \mathbf{f}_{hw}, \quad M_k = \sum_{h,w} m_{khw} \tag{6.3}$$

6.2.3 Segment Category Retrieval

CLIP [11] can embed images and text in the same latent space, but the model itself cannot generate images or text from the embedded features. To address this challenge, our proposed method TAG finds the closest category using multi-modal data from large databases.

First, we retrieve a few of the most probable candidate classes from the large classification space.

Let D be the database of image captions. Given representative segment embedding $\bar{\mathbf{f}}_k$, retrieve the set $D_{\bar{\mathbf{f}}_k} \subset D$ of the n closest captions to each segment embedding by

$$D_{\bar{\mathbf{f}}_k} = \underset{\mathbf{d} \in D}{\text{top-}n} \frac{\bar{\mathbf{f}}_k^T \cdot \mathbf{f}_d}{\|\bar{\mathbf{f}}_k\| \cdot \|\mathbf{f}_d\|}, \quad \mathbf{f}_d = \text{CLIP}_t(d), \tag{6.4}$$

where CLIP_t is the text encoder of the CLIP [11].

Next, to extract candidate words $C_{\bar{\mathbf{f}}_k}$ from the set $D_{\bar{\mathbf{f}}_k}$, we create a set of all words that are contained in the captions. We sequentially apply three operations: (i) remove noisy candidates, (ii) standardize their format, and (iii) filter them.

In the first operation, we remove all the irrelevant words, such as URLs, or file extensions.

Secondly, we align words referring to the same semantic class in a standardized format. Specifically, Converting upper case to lower case and plural words to singular format.

In the final operation, we filter out two types of words: rare and noisy categories based on the frequency of word occurrences, as well as entire categories of words determined by Part-Of-Speech (POS) [105] tagging. Frequency filtering involves retaining only those words that appear more than two times in the input text. If the threshold is set too high and no words meet the criterion, it is lowered to include at least the most frequently occurring words. The POS [105] tagging classifies words into groups like adjectives, articles, nouns, or verbs, allowing us to exclude any terms that do not hold semantic significance as segmentation categories.

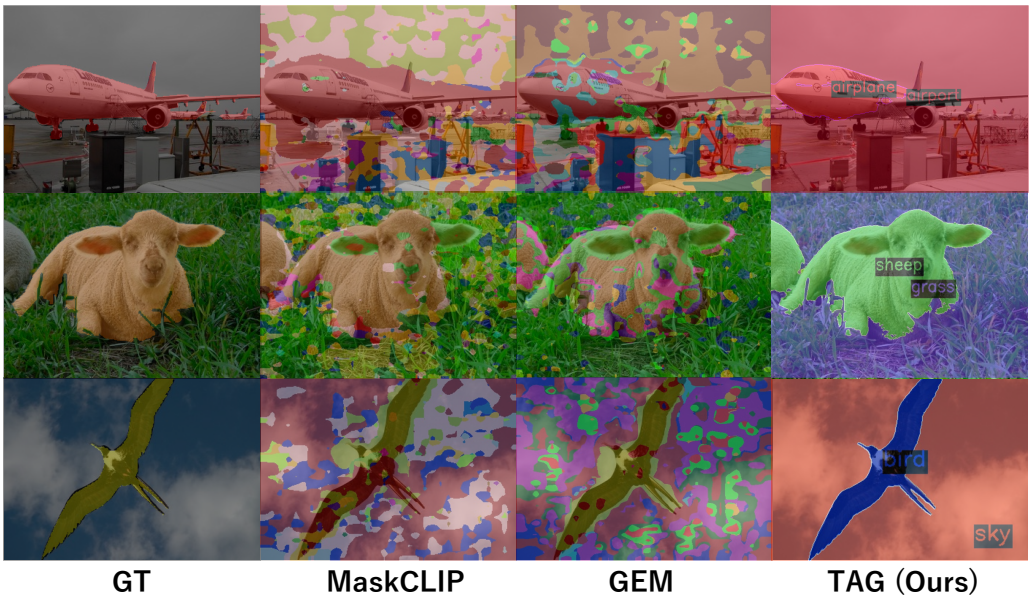

Figure 6.3 Comparison results with CLIP base open-vocabulary segmentation methods on PascalVOC [5] Note that MaskCLIP [3] and GEM [7] uses text guidance while our TAG does not use.

Given candidate words $C_{\bar{\mathbf{f}}_k}$, we assign words to representative segment embedding $\bar{\mathbf{f}}_k$ as

$$W = \underset{\mathbf{c} \in C_{\bar{\mathbf{f}}_k^T}}{\operatorname{argmax}} \frac{\bar{\mathbf{f}}_k^T \cdot \mathbf{f}_c}{\|\bar{\mathbf{f}}_k\| \cdot \|\mathbf{f}_c\|}, \quad \mathbf{f}_c = \mathrm{CLIP}_t(c), \tag{6.5}$$

where W is assigned a category word to segment. Through the above process, we can obtain segmentation results by assigning categories to each segment candidate.

6.3 Experiment

First, we present the implementation details in Section 6.3.1. Next, we compare our results to previous methods in Section 6.3.2 and evaluate the open vocabulary aspect in Section 6.3.3. Finally, we justify the construction of TAG through an ablation study in Section 6.3.4.

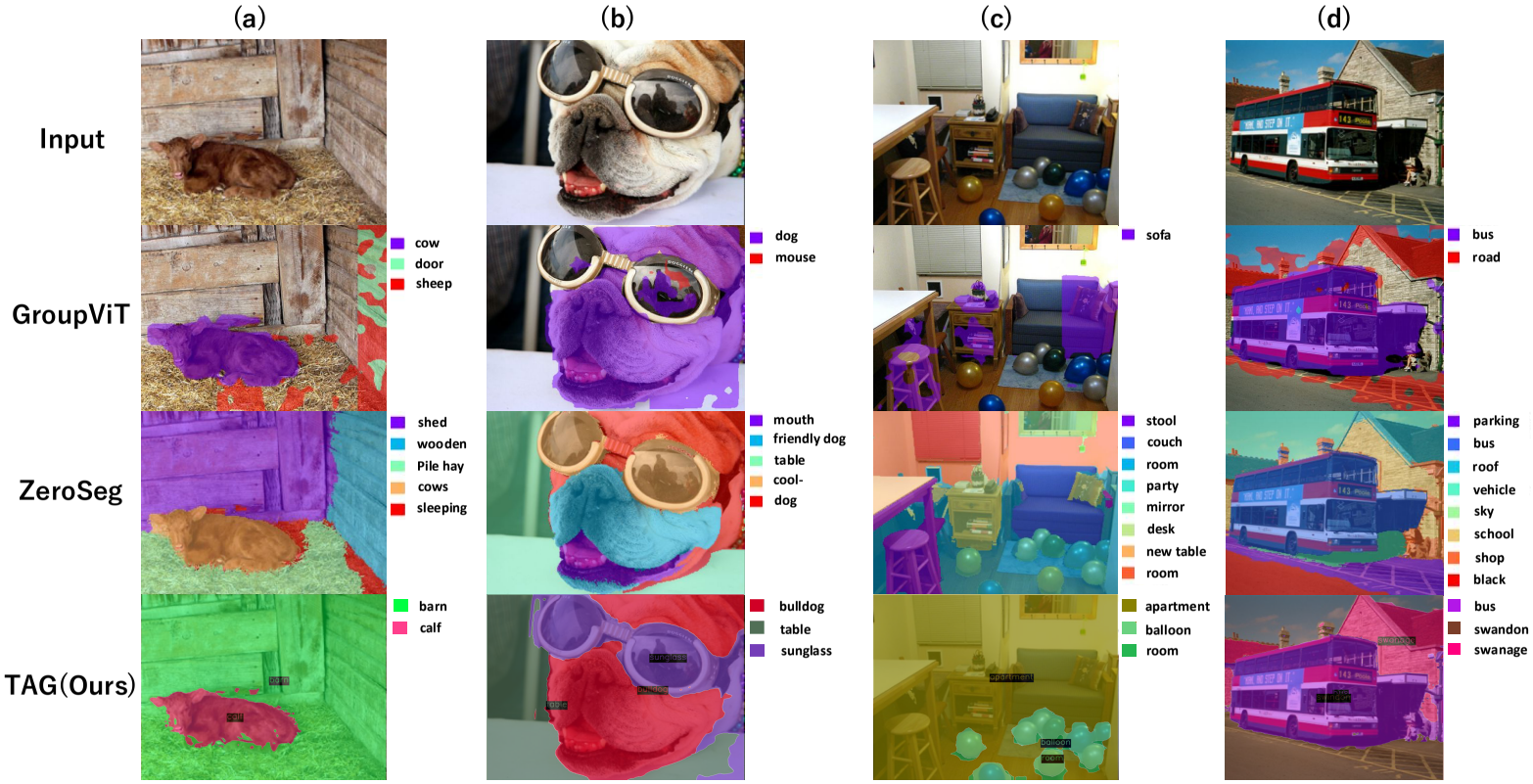

Figure 6.4 **Qualitative results.** We compare GroupViT [12], ZeroSeg [13], and our TAG on images containing general objects from PascalContext [14]. This figure indicates that TAG can segment and label correctly.

Table 6.1 **Comparison of state-of-the-art methods.** We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. The clear boost in performance by retrieving open-vocabulary segment categories underlines their semantic richness and effectiveness.

Method	Training Free	PascalVOC [5] 20class	PascalContext [14] 59class	ADE20K [16] 150class
Open-Vocabulary Segmentation				
X-Decoder [64]	-	96.2	69.2	6.4
ODISE [60]	-	82.7	55.3	11.0
OpenSeg [100]	-	72.2	44.8	8.8
GroupViT [12]	-	50.7	25.9	-
LSeg [106]	-	47.4	-	-
GEM [7]	√	26.5	11.8	4.4
MaskCLIP [3]	√	28.6	25.5	-
Zero-Guidance Segmentation				
ZeroSeg [13]	√	20.1	19.6	-
SelfSeg [62]	√	41.6	-	6.4
TAG (Ours)	√	**56.9**	**20.2**	**6.6**

6.3.1 Implementation Details

For our implementation of TAG, we employed a frozen pre-trained CLIP [11] and DINO [10] with ViT-L/14 architecture and input 448 × 448 images to them. As database, we use PMD [17], CC12M [18], WordNet [104] and EnglishWords [103]. In addition, we use a fast indexing technique, FAISS [107]. Our model works with a GPU memory of 15 GB.

6.3.2 Main Results

To validate the performance of TAG, we conducted comprehensive comparative experiments with its closest counterpart, ZeroSeg [13]. For settings,

TAG uses a PMD database [17]. We set the number of k-means clusters as 15 and the frequency filtering threshold 2. For the evaluation, we used the mean Intersection over Union (mIoU) as the primary metric. The predicted text T_i needs to be assigned to one of the ground truth classes T^{gt}. T_i is assigned to the ground-truth label that is closest in the Sentence-BERT [108] embedding space, following the same approach as ZeroSeg [13]. Formally, the new label T_i^* is computed by

$$T_i^* = \underset{t \in T^{gt}}{\operatorname{argmax}} \, [\text{cossim}^{\text{SBERT}}(T_i, t)]. \tag{6.6}$$

We perform our experiments on the PascalVOC [5] dataset comprising 20 classes, PascalContext [14] with 59 classes, as well as ADE20K [16] consisting of 150 classes.

The qualitative results are shown in Figure 6.3 and Figure 6.4. In Figure 6.3, we compared TAG with CLIP [11] base open-vocabulary method, MaskCLIP [3], and GEM [7]. Using MaskCLIP [3] and GEM [7] results in a noisy and fragmented segmentation, whereas TAG achieves more consistent segments that better correspond with the shape of the object and segment categories. In Figure 6.4, we compare GroupViT [12], ZeroSeg [13], and our TAG on images containing general objects from PascalContext [14]. In the image (a), TAG is the only method that accurately recognizes a cow as a calf. In addition, TAG assigns the precise and relevant class 'barn' to the surroundings of the calf, unlike ZeroSeg which incorrectly includes the class 'sleeping'. In image (b), TAG is the only method to correctly identify 'sunglasses', and it also accurately classifies the dog as a 'bulldog'. However, in image (c), TAG does not distinguish between a desk and a chair but rather assigns the rough class 'room' to the entire space. Occasionally, as shown in (d), TAG assigns proper nouns such as 'swindon' and 'swanage' which are names of cities in South England. While TAG correctly identifies the background as the city 'swanage', the ground is incorrectly assigned to the city of 'swindon'. We hypothesize this is caused by both segments being close in the CLIP [11] embedding space.

The quantitative results are shown in Table 6.1. TAG shows an improvement of +15.3 mIoU on PascalVOC [5], +0.6 mIoU on PascalContext [14], and +0.2 mIoU on ADE20K [16] compared to previous zero-guidance segmentation state-of-the-art results. In particular, TAG shows a dramatic performance improvement on PascalVOC [5], which was identified as a limitation in ZeroSeg [13]. Additionally, our TAG method has made significant improvements compared to untrained open-vocabulary segmentation methods, demonstrating an impressive improvement of +28.3 mIoU on PascalVOC [5], even without text-based guidance.

6.3.3 Open Vocabulary Segmentation on Web-Crawled Images

In this section, we thoroughly assess the performance of TAG using open vocabulary segmentation experiments on web-crawled images, where we test the model's ability to accurately segment various unseen classes, including specific and detailed categories such as 'joker' and 'porsche'.

The qualitative outcomes of the experiments are visually depicted in Figure 6.5. In this figure, (a) represents a general image, while (b) and (c) showcase images created by Stable Diffusion [15]. Furthermore, (d) and (e) show images containing proper nouns.

In image (a), although the complex concept of a 'mirror' is not captured, the segmentation successfully identifies both 'cat' and 'bathroom', resulting in accurate outcomes. In image (b), while failing to recognize the 'astronaut', the model aptly estimates the ground as the 'moon', leading to a logical result. Given TAG's ability to identify the ground as the moon, it is evident that it can understand the whole image while generating segment embeddings. Image (c) showcases the precise segmentation of various foods. Image (d) impressively segments and identifies proper nouns such as 'joker' and 'batman', demonstrating remarkable results. Lastly, image (e), despite containing the specific proper noun 'porsche', is correctly recognized as a supercar, affirming the accuracy of the segmentation.

These findings serve as compelling evidence that TAG exhibits robust

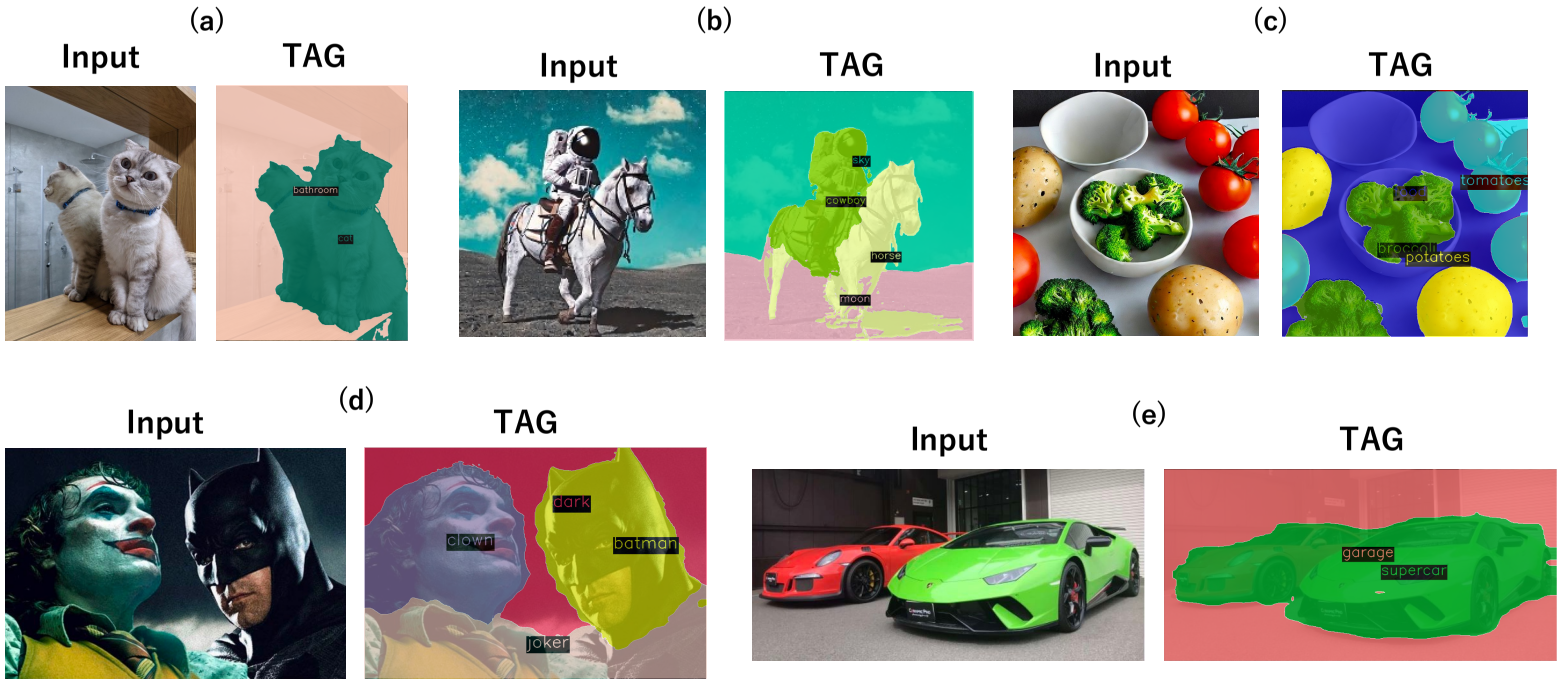

Figure 6.5 **Open-vocabulary segmentation results.** In (a) we test on a general image, (b) and (c) show images generated by Stable Diffusion [15], and (d) and (e) are images featuring specific proper nouns.

Table 6.2 Ablation study on database and numbers of cluster used for k-means. We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. These results reveal that using PMD [17] and CC12M [18] as the database and setting the number of k-means clusters to 15 is the most robust choice, consistently yielding favorable outcomes across multiple datasets.

Database	Cluster Numbers	PascalVOC [5] 20class	PascalContext [14] 59class	ADE20K [16] 150class
PMD [17]	5	57.5	19.2	5.0
PMD [17]	10	56.9	19.1	6.2
PMD [17]	15	56.9	**20.2**	**6.6**
PMD [17]	20	56.4	18.9	5.2
PMD [17]	25	51.8	19.5	5.2
CC12M [18]	5	**58.5**	19.2	4.9
CC12M [18]	10	58.4	19.2	6.1
CC12M [18]	15	58.1	19.6	6.1
CC12M [18]	20	57.6	19.0	5.0
CC12M [18]	25	52.9	19.1	5.1
WordNet [104]	15	54.3	17.5	4.5
EnglishWords [103]	15	43.7	14.5	3.0

capabilities to accurately segmenting open vocabularies, including complex and specific categories, thus underscoring its versatility and effectiveness in handling diverse and intricate segmentation tasks.

sidewaysfigure

6.3.4 Ablation Study

In this section, we perform ablation studies on TAG, examining how various databases, cluster numbers of k-means, and label reassignment of evaluation affect performance.

Table 6.2 presents the results of the ablation experiments comparing the

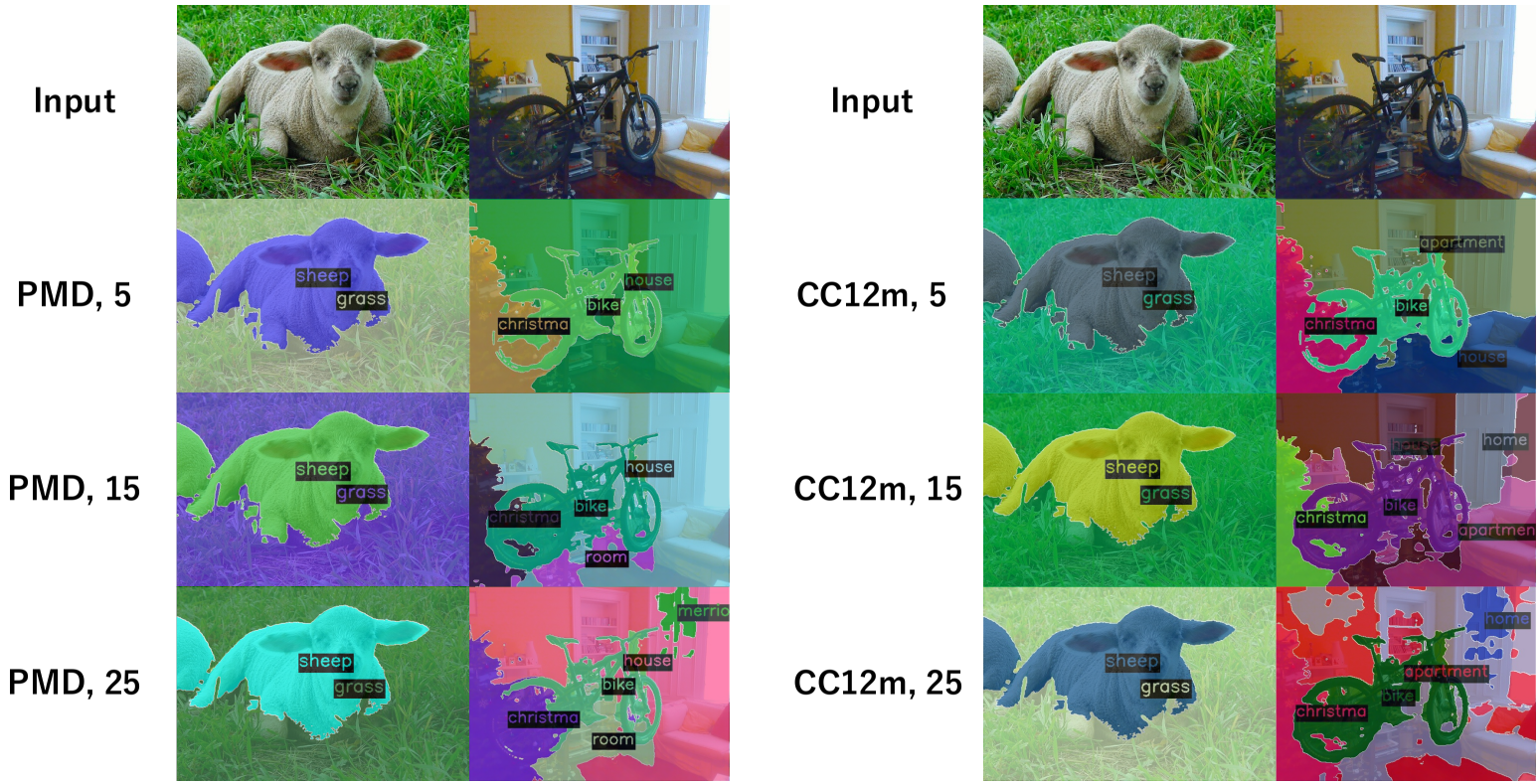

Figure 6.6 **Qualitative results of ablation study on PascalVOC [5].** The database and the number of k-means clusters are shown with the results.

effect of databases and cluster numbers of k-means on the mIoU. The results indicate that PMD [17] and CC12M [18] are preferable datasets for our database. These results also reveal that using PMD [17] as the database and setting the number of k-means clusters to 5 is the most robust choice, consistently yielding favorable outcomes across multiple datasets. Figure 6.6 shows the ablation qualitative results. Left image remains unchanged regardless of variations in the database or the number of clusters, while increasing the number of clusters has been observed to cause segments with the same semantic meaning, such as 'apartment' to be divided into different segments like 'home' or 'house' in right image.

Furthermore, we conduct ablation experiments on filtering operations and the number of captions to select used for segment category retrieval. Table 6.3 shows that utilizing all three filtering operations yields the best results. Similarly, we examine the effect of the threshold on the frequency filtering. The results are shown in Table 6.4 and indicate that using our default threshold of 2 is justified. In addition, Table 6.5 shows the number of captions to select and indicates that our default setting threshold to 10 is appropriate..

For evaluation, the predicted text T_i is assigned to the ground-truth label that is closest in the Sentence-BERT [108] embedding space. We conduct additional experiments on this assignment metrics. By calculating the IoU only for segments with a cosine similarity above a certain threshold, we enable the evaluation of TAG for different values of similarity. The experimental results are shown in Table 6.6. The mIoU for segments with thresholds of 0.5 or higher demonstrates performance comparable to supervised, guided open-vocabulary segmentation methods as shown in Table 6.6. The experimental results also demonstrate that the inferred categories have at least a similarity score of zero or higher within the ground truth label, indicating that the predictions are not entirely off the mark.

Table 6.3 Ablation study on filtering operations. We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU. This table shows that utilizing all three filtering operations yields the best results.

Remove	Standardize	Filter	PascalVOC [5]	PascalContext [14]	ADE20K [16]
			20class	59class	150class
			54.5	18.4	5.0
√			55.2	18.5	5.1
√	√		56.0	18.6	5.8
√	√	√	**56.9**	**20.2**	**6.6**

Table 6.4 Ablation study on the threshold of frequency filtering. We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU.

threshold	PascalVOC [5]	PascalContext [14]	ADE20K [16]
	20class	59class	150class
1	54.3	18.6	5.8
2	**56.9**	**20.2**	6.6
5	54.1	18.6	**6.8**
10	51.7	17.8	6.3
20	51.4	17.8	5.8

Table 6.5 Ablation study on the number of retrieved captions. We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU.

Captions	PascalVOC [5]	PascalContext [14]	ADE20K [16]
Number	20class	59class	150class
5	55.9	19.5	6.3
10	56.9	**20.2**	**6.6**
20	**57.3**	19.9	6.3
40	56.4	19.2	6.0
60	55.6	19.1	5.9

Table 6.6 Ablation study on the threshold of Sentence-BERT similarity. We evaluate on PascalVOC [5], PascalContext [14], and ADE20K [16] and report the mIoU.

threshold	PascalVOC [5]	PascalContext [14]	ADE20K [16]
	20class	59class	150class
-1	56.9	20.2	6.6
0	56.9	20.2	6.6
0.5	79.5	24.4	8.9
0.8	84.8	32.8	15.0

6.4 Limitation

While TAG achieves remarkable results, our proposed method still comes with certain limitations. First, as shown in Table 6.2, TAG depends on the choice of the database, making it challenging to select the optimal database for unknown domains without information on test labels. On the other hand, TAG can flexibly address this limitation by adding new concepts into the

database without retraining, unlike language-based methods [13, 62]. Second, TAG does not distinguish between different levels of class granularity. As shown in Figure 6.5 (e), TAG predicted both 'Porsche' and 'Lamborghini' as 'supercar'. While the predicted categories in the qualitative results are consistently correct, they may not always align with the optimal category desired by the user. Future works might address this issue by considering the frequency of words within the database.

6.5 Conclusion

In this study, we proposed TAG, Training, Annotation, and Guidance-free open-vocabulary semantic segmentation. TAG employs a novel approach by extracting semantic features from each pixel in an image using CLIP [11], and then retrieving the open-vocabulary categories based on these features from an external database. Through a series of comprehensive experiments and analyses, we have demonstrated the effectiveness and versatility of TAG across various datasets and challenging segmentation tasks. Our results indicate that TAG exhibits robust performance in handling diverse categories, including general classes and fine-grained, proper noun-based segments.

Overall, our findings highlight the potential of TAG as a powerful and effective tool in the field of semantic segmentation. By retrieving the open-vocabulary categories, we have successfully demonstrated the model's capability to handle diverse datasets and open vocabularies without text guidance, paving the way for future advancements and applications in this critical area of computer vision.

Chapter 7

Conclusion

In this thesis, we addressed the challenges and inefficiencies associated with training in the field of semantic segmentation. We began by proposing a novel active learning method that leverages uncertainty to select the most informative data points for annotation (Chapter 3). This method circumvents the extensive training times typically associated with active learning and optimizes the use of computational resources, proving especially beneficial in scenarios where physical costs of annotations are high, such as in semantic segmentation.

Building on the foundations of active learning, we introduced an innovative approach to unsupervised domain adaptation in Chapter 4. By employing uncertainty as a loss function, our method enhances the model's ability to generate high-confidence predictions, outperforming traditional techniques that rely solely on pseudo-labels. This advancement not only improves the robustness and accuracy of semantic segmentation but also broadens the model's applicability across diverse domains.

In Chapter 5, we explored the potential of diffusion models, specifically through our MaskDiffusion technique, which integrates Stable Diffusion and internal features to produce superior segmentation results. Our extensive experiments across various datasets underscored ability of MaskDiffusion and Unsupervised MaskDiffusion to adeptly handle a wide range of categories, from general classes to more specific, proper noun-based segments. This

demonstrates the versatility and robustness of using diffusion models for semantic segmentation without the need for direct annotations.

Further innovating in the space of open-vocabulary segmentation, Chapter 6 introduced TAG (Training, Annotation, and Guidance-free open-vocabulary semantic segmentation). TAG utilizes semantic features extracted from pixels combined with external database queries to categorize images in an open-vocabulary manner. This approach eliminates the need for text-based guidance, offering a significant advancement over traditional methods and showcasing the practicality of integrating semantic features with real-world applications.

Overall, our findings highlight a series of methodological innovations that collectively enhance the efficiency and efficacy of semantic segmentation models. These developments not only improve current practices but also pave the way for future research in integrating these techniques with other learning paradigms to tackle the evolving challenges in computer vision. This thesis sets a foundation for further explorations into cost-effective, robust, and versatile semantic segmentation methods, aiming to bridge the gap between theoretical advancements and practical applications in the field.

Acknowledgement

本研究は，著者が慶應義塾大学大学院理工学研究科後期博士課程に在学中，青木義満教授のご指導のもと行われました．本研究の執筆にあたり，多くの方にお世話になりました．この場を借りて感謝の意を述べさせていただきます．

まず，本論文の主査であり，指導教員である慶應義塾大学理工学部青木義満教授に深く感謝いたします．青木義満教授には本研究の立ち上げから本論文の執筆に至るまで，多くの指導をしていただきました．また，研究室の同期，後輩，先輩の皆様にも深く感謝いたします．学部 4 年生から始まり，修士課程，博士課程と青木研究室で得られた経験は今後の筆者の人生において計り知れない価値を持つものと確信しております．

本論文の審査にあたり，副査を快くお引き受けくださった慶應義塾大学理工学部の田中敏幸教授，五十川麻理子准教授，村田真悟准教授にも深く感謝申し上げます．

最後に，常に支えてくださった家族に深く感謝いたします．

References

[1] 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*, pp. 3213–3223, 2016.

[2] Beichen Zhang, Liang Li, Shijie Yang, Shuhui Wang, Zheng-Jun Zha, and Qingming Huang. State-relabeling adversarial active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8756–8765, 2020.

[3] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European Conference on Computer Vision*, pp. 696–712. Springer, 2022.

[4] Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, Vol. 14, , 2001.

[5] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, Vol. 111, pp. 98–136, 2015.

[6] Isprs. isprs 2d semantic labeling contest. http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html.

[7] Vittorio Ferrari Hilde Kuehne Walid Bousselham, Felix Petersen. Grounding everything: Emerging localization properties in vision-language transformers. *arXiv preprint arXiv:2312.00878*, 2023.

[8] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman. Unsupervised semantic segmentation by distilling feature correspondences. In *International Conference on Learning Representations*, 2022.

[9] 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*, pp. 1209–1218, 2018.

[10] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[11] 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*, pp. 8748–8763. PMLR, 2021.

[12] 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*, pp. 18134–18144, 2022.

[13] Pitchaporn Rewatbowornwong, Nattanat Chatthee, Ekapol Chuangsuwanich, and Supasorn Suwajanakorn. Zero-guidance segmentation using zero segment labels. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[14] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[15] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

[16] 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*, Vol. 127, No. 3, pp. 302–321, 2019.

[17] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15638–15650, 2022.

[18] 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*, pp. 3558–3568, 2021.

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

[20] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5972–5981, 2019.

[21] Jifeng Guo, Zhiqi Pang, Wenbo Sun, Shi Li, and Yu Chen. Redundancy removal adversarial active learning based on norm online uncertainty indicator. *Computational Intelligence and Neuroscience*, Vol. 2021, No. 1, p. 4752568, 2021.

[22] Shuo Wang, Yuexiang Li, Kai Ma, Ruhui Ma, Haibing Guan, and Yefeng Zheng. Dual adversarial network for deep active learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pp. 680–696. Springer, 2020.

[23] Kwanyoung Kim, Dongwon Park, Kwang In Kim, and Se Young Chun. Task-aware variational adversarial active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8166–8175, 2021.

[24] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 93–102, 2019.

[25] Yasufumi Kawano, Yoshiki Nota, Rinpei Mochizuki, and Yoshimitsu Aoki. Non-deep active learning for deep neural networks. *Sensors*, Vol. 22, No. 14, p. 5244, 2022.

[26] 川野恭史, 野田祥希, 望月凜平, 青木義満. Active learning における不確実サンプル選択によるアノテーション効率化. 精密工学会誌, Vol. 88, No. 2, pp. 211–216, 2022.

[27] 川野恭史, 野田祥希, 青木義満. 不確実性ロスによるセマンティックセグメンテーションの教師無しドメイン適応. 精密工学会誌, Vol. 89, No. 12, pp. 921–925, 2023.

[28] Yasufumi Kawano and Yoshimitsu Aoki. Maskdiffusion: Exploiting pre-trained diffusion models for semantic segmentation. *IEEE Access*, Vol. 12, pp. 127283–127293, 2024.

[29] Yasufumi Kawano and Yoshimitsu Aoki. Tag: Guidance-free open-vocabulary semantic segmentation. *IEEE Access*, Vol. 12, pp. 88322–88331, 2024.

[30] Burr Settles. Active learning literature survey. 2009.

[31] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

[32] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9368–9377, 2018.

[33] Mustafa Bilgic and Lise Getoor. Link-based active learning. In *NIPS workshop on analyzing networks and learning with graphs*, Vol. 4, p. 9, 2009.

[34] Mahmudul Hasan and Amit K. Roy-Chowdhury. Context aware active learning of activity recognition models. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4543–4551, 2015.

[35] Oisin Mac Aodha, Neill DF Campbell, Jan Kautz, and Gabriel J Brostow. Hierarchical subquery evaluation for active learning on a graph. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 564–571, 2014.

[36] Burr Settles, Mark Craven, and Soumya Ray. Multiple-instance active learning. *Advances in neural information processing systems*, Vol. 20, , 2007.

[37] Nicholas Roy and Andrew McCallum. Toward optimal active learning through monte carlo estimation of error reduction. *Icml, williamstown*, Vol. 2, No. 441-448, p. 4, 2001.

[38] Christoph Käding, Erik Rodner, Alexander Freytag, and Joachim Denzler. Active and continuous exploration with deep neural networks and expected model output changes, 2016.

[39] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[40] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, Vol. 28, , 2015.

[41] Nontawat Tritrong, Pitchaporn Rewatbowornwong, and Supasorn Suwajanakorn. Repurposing gans for one-shot semantic part segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4475–4485, 2021.

[42] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

[43] Yezheng Liu, Zhe Li, Chong Zhou, Yuanchun Jiang, Jianshan Sun, Meng Wang, and Xiangnan He. Generative adversarial active learning for unsupervised outlier detection. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 32, No. 8, pp. 1517–1528, 2019.

[44] Xingjian Li, Pengkun Yang, Tianyang Wang, Min Xu, Dejing Dou, and Cheng zhong Xu. Deep active learning with noise stability, 2022.

[45] Yichen Xie, Masayoshi Tomizuka, and Wei Zhan. Towards general and efficient active learning, 2022.

[46] Mengying Fu Xiangyang Ji Qingming Huang Fang Wan, Tianning Yuan and Qixiang Ye. Nearest neighbor classifier embedded network for active learning. pp. 10041–10048, 2021.

[47] Marc Gorriz, Axel Carlier, Emmanuel Faure, and Xavier Giro-i Nieto. Cost-effective active learning for melanoma segmentation. *arXiv preprint arXiv:1711.09168*, 2017.

[48] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 27, No. 12, pp. 2591–2600, 2017.

[49] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, Vol. 2, No. Nov, pp. 45–66, 2001.

[50] Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pp. 59–66, 2003.

[51] Rui Gong, Wen Li, Yuhua Chen, Dengxin Dai, and Luc Van Gool. Dlow: Domain flow and applications. *International Journal of Computer Vision*, Vol. 129, No. 10, pp. 2865–2888, 2021.

[52] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7472–7481, 2018.

[53] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9924–9935, 2022.

[54] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Hrda: Context-aware high-resolution domain-adaptive semantic segmentation. In *European conference on computer vision*, pp. 372–391. Springer, 2022.

[55] Dong-Hyun Lee, et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, Vol. 3, p. 896. Atlanta, 2013.

[56] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2517–2526, 2019.

[57] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[58] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision*, 2021.

[59] Hyun Seok Seong, WonJun Moon, SuBeen Lee, and Jae-Pil Heo. Leveraging hidden positives for unsupervised semantic segmentation. June 2023.

[60] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2955–2966, 2023.

[61] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, Vol. 1, No. 8, p. 9, 2019.

[62] Osman Ülger, Maksymilian Kulicki, Yuki Asano, and Martin R Oswald.

Self-guided open-vocabulary semantic segmentation. *arXiv preprint arXiv:2312.04539*, 2023.

[63] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900. PMLR, 2022.

[64] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. Generalized decoding for pixel, image, and language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15116–15127, 2023.

[65] Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9865–9874, 2019.

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

[67] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

[68] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7086–7096, 2022.

[69] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. *Advances in Neural Information Processing Systems*, Vol. 35, pp. 33754–33767, 2022.

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

[71] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *arXiv preprint arXiv:2304.06718*, 2023.

[72] Qianli Feng, Raghudeep Gadde, Wentong Liao, Eduard Ramon, and Aleix Martinez. Network-free, unsupervised semantic segmentation with synthetic images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23602–23610, 2023.

[73] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

[74] ZengShun Zhaoa, Yulong Wang, Ke Liu, Haoran Yang, Qian Sun, and Heng Qiao. Semantic segmentation by improved generative adversarial networks. *arXiv preprint arXiv:2104.09917*, 2021.

[75] Daiqing Li, Junlin Yang, Karsten Kreis, Antonio Torralba, and Sanja Fidler. Semantic segmentation with generative models: Semi-supervised learning and strong out-of-domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8300–8311, 2021.

[76] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021.

[77] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. *arXiv preprint arXiv:2305.15347*, 2023.

[78] Weimin Tan, Siyuan Chen, and Bo Yan. Diffss: Diffusion model for few-shot semantic segmentation. *arXiv preprint arXiv:2307.00773*, 2023.

[79] Julia Wolleb, Robin Sandkühler, Florentin Bieder, Philippe Valmaggia, and Philippe C Cattin. Diffusion models for implicit image segmentation ensembles. In *International Conference on Medical Imaging with Deep Learning*, pp. 1336–1348. PMLR, 2022.

[80] Peng Jiang, Fanglin Gu, Yunhai Wang, Changhe Tu, and Baoquan Chen. Difnet: Semantic segmentation by diffusion networks. *Advances in Neural Information Processing Systems*, Vol. 31, , 2018.

[81] Weijia Wu, Yuzhong Zhao, Mike Zheng Shou, Hong Zhou, and Chunhua Shen. Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. *arXiv preprint arXiv:2303.11681*, 2023.

[82] Tomer Amit, Tal Shaharbany, Eliya Nachmani, and Lior Wolf. Segdiff: Image segmentation with diffusion probabilistic models. *arXiv preprint arXiv:2112.00390*, 2021.

[83] Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar Gonzalez-Franco. Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion. *arXiv preprint arXiv:2308.12469*, 2023.

[84] Yoad Tewel, Yoav Shalev, Idan Schwartz, and Lior Wolf. Zerocap: Zero-shot image-to-text generation for visual-semantic arithmetic. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17918–17928, 2022.

[85] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pp. 2206–2240. PMLR, 2022.

[86] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International conference on machine learning*, pp. 3929–3938. PMLR, 2020.

[87] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, Vol. 33, pp. 9459–9474, 2020.

[88] Alexander Long, Wei Yin, Thalaiyasingam Ajanthan, Vu Nguyen, Pulak Purkait, Ravi Garg, Alan Blair, Chunhua Shen, and Anton van den Hengel. Retrieval augmented classification for long-tail visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6959–6969, 2022.

[89] Alessandro Conti, Enrico Fini, Massimiliano Mancini, Paolo Rota, Yiming Wang, and Elisa Ricci. Vocabulary-free image classification, 2023.

[90] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, Vol. 12, pp. 2825–2830, 2011.

[91] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 472–480, 2017.

[92] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[93] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam,*

The Netherlands, October 11-14, 2016, Proceedings, Part II 14, pp. 102–118. Springer, 2016.

[94] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3234–3243, 2016.

[95] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 289–305, 2018.

[96] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 1379–1389, 2021.

[97] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. Domain adaptive semantic segmentation with self-supervised depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8515–8525, 2021.

[98] Yahao Liu, Jinhong Deng, Xinchen Gao, Wen Li, and Lixin Duan. Bapa-net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8801–8811, 2021.

[99] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12414–12424, 2021.

[100] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7061–7070, 2023.

[101] Zheng Ding, Jieke Wang, and Zhuowen Tu. Open-vocabulary universal image segmentation with maskclip. 2023.

[102] Kevin Clark and Priyank Jaini. Text-to-image diffusion models are zero shot classifiers. *Advances in Neural Information Processing Systems*, Vol. 36, , 2024.

[103] Freebsd. web2 dictionary (revision 326913). Accessed February 20, 2024.

[104] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, Vol. 38, No. 11, pp. 39–41, 1995.

[105] Nlp library flair. Accessed February 20, 2024.

[106] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2022.

[107] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, Vol. 7, No. 3, pp. 535–547, 2019.

[108] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.