YEDITEPE UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

KNOWLEDGE DISTILLATION WITH FOUNDATION MODELS FOR IMAGE SEGMENTATION

A Thesis Submitted

by

Merve Noyan

In Partial Fulfillment

of the Requirements for the Degree of

Master of Science

in

Data Science

Supervisor

Assist. Prof. Dr. İpek Baz

Istanbul - 2024

KNOWLEDGE DISTILLATION WITH FOUNDATION MODELS FOR IMAGE SEGMENTATION

by

Merve Noyan

Approved by:

Assist. Prof. Dr. İpek Baz
(Yeditepe University) ……
(Thesis Supervisor)

Assist. Prof. Dr. Onur Demir
(Yeditepe University) ……

Prof. Dr. Emin Murat Esin
(Maltepe University) ……

DATE OF APPROVAL: …. / …. / 20….

DECLARATION OF ORIGINALITY

I hereby declare that this thesis is my own work and that all information in this thesis has been obtained and presented following academic rules and ethical conduct. I have fully cited and referenced all material and results as required by these rules and conduct, and this thesis study does not contain any plagiarism. The necessary permissions have been obtained if any material used in the thesis requires copyright. No material from this thesis has been used to award another degree.

To the best of my knowledge and belief, it contains no material previously published or written by another person nor material accepted for the award of any other degree except where due acknowledgment has been made in the text.

I accept all kinds of legal liability that may arise in cases contrary to these situations.

Merve Noyan

……

ABSTRACT

KNOWLEDGE DISTILLATION WITH FOUNDATION MODELS FOR IMAGE SEGMENTATION

Open vocabulary image segmentation is the task of predicting segmentation masks for any label. State-of-the-art open vocabulary image segmentation approaches combine zero-shot object detection and mask generation models. Mask generation models use point and box prompts to segment anything, and there have been many approaches in distilling these. This approach includes distilling open vocabulary image segmentation models into small student models to guide the student model training. This dissertation also contributes a new open-vocabulary image segmentation model, named OWLSAM, that can be used for open-vocabulary image segmentation and synthetic data labelling.

ÖZET

GÖRÜNTÜ SEGMENTASYONU IÇIN TEMEL MODELLERIN DAMITILMASI

Açık sözlük görüntü segmentasyonu, herhangi bir sınıf için segmentasyon maskelerini tahmin etme görevidir. Maske oluşturma modelleri, herhangi bir şeyi bölümlere ayırmak için nokta ve sınırlayıcı kutu komutlarını kullanır. Son teknoloji ürünü açık sözlük görüntü segmentasyonu yaklaşımları, sıfır atışlı nesne algılama ve maske oluşturma modellerini birleştirerek tahminde bulunur. Bu yaklaşım, öğrenci modeli eğitimine rehberlik etmek için açık sözlük görüntü segmentasyonu modellerini öğretmen model olarak kullanır. Bu tez, aynı zamanda yeni bir açık sözlük görüntü segmentationu modeli (OWLSAM) sunar.

DEDICATION

I would like to dedicate this thesis to my mother and grandmother...

ACKNOWLEDGEMENTS

It is with immense gratitude that I acknowledge the support and help of my Professor Ms. Ipek Baz. She is immensely knowledgeable, patient and helpful. I was very inspired by her suggestions whenever I was stuck.

I would like to thank Julien Chaumond, Omar Sanseviero and Pedro Cuenca for providing compute support and allowing me to work on my thesis, and Google ML Developers Program for compute credits to run the experiments. I would like to thank Ross Wightman for his suggestions. Lastly I would like to thank my friends Anton Lozhkov, Zachary Mueller, Teven Le Scao for their support throughout my thesis writing journey.

TABLE OF CONTENTS

DECLARATION OF ORIGINALITY iii
ABSTRACT iv
ÖZET v
DEDICATION vi
ACKNOWLEDGEMENTS vii
TABLE OF CONTENTS viii
LIST OF FIGURES x
LIST OF TABLES xi
LIST OF ABBREVIATIONS xii
1. INTRODUCTION 1
1.1. PROBLEM STATEMENT 1
1.2. AIM OF THIS DISSERATION 2
1.3. CONTRIBUTION OF THIS THESIS 2
1.4. OUTLINE 3
2. LITERATURE REVIEW 4
2.1. VISION TRANSFORMERS 4
2.2. CONTRASTIVE PRE-TRAINING AND MULTIMODALITY 5
2.3. OWLV2 MODEL 6
2.4. SEGMENT ANYTHING MODEL 7
2.5. SEMANTIC SEGMENTATION 8
2.6. OPEN-VOCABULARY IMAGE SEGMENTATION 8
2.7. KNOWLEDGE DISTILLATION 10
2.8. MOBILENET AND DEEPLAB 11
2.9. DISTILLATION IN SEGMENTATION 12
2.10. DISTILLATION IN WITH PSEUDOLABELS FROM FOUNDATION MODELS 15
3. METHODOLOGY 17
3.1. PROBLEM DEFINITION 17
3.2. PROPOSED APPROACH 19
3.3. TEACHER MODEL 19

3.4. IMPLEMENTATION DETAILS 21
3.4.1. PASCAL-VOC Dataset 21
3.4.2. Knowledge Distillation Details 22
3.4.3. Other Technical Details 25
3.4.4. Model Fine-tuning Details 29
4. RESULTS 32
4.1. OWLSAM RESULTS 32
4.2. DISTILLATION RESULTS 34
5. DISCUSSION 36
6. CONCLUSIONS 38
6.1. FUTURE WORK 38
REFERENCES 40

LIST OF FIGURES

Figure 2.1. SAM Architecture, from the paper [1] 7

Figure 2.2. OWL Architecture from paper [2] 8

Figure 2.3. An example input output from PASCAL VOC 2012 semantic segmentation dataset 9

Figure 2.4. Knowledge Distillation 10

Figure 3.1. An example input output from PASCAL VOC 2012 semantic segmentation dataset 18

Figure 3.2. Overview of the OWLSAM Model 19

Figure 3.3. Open-vocabulary Example for OWLSAM Text Detection 20

Figure 3.4. Open-vocabulary Example for OWLSAM 20

Figure 3.5. Distillation Process 21

Figure 3.6. Example Images and their masks from PASCAL-VOC 2012 Dataset 23

Figure 3.7. Preprocessing Pipeline to Obtain Soft Targets 28

Figure 4.1. Visualized Intersection-over-Union from [3] 32

Figure 4.2. OWLSAM Outputs on SegInW 33

Figure 4.3. Intersection-over-Union distributions per class 34

Figure 5.1. Illustrative Example for Tuned OWL Confidence Thresholds 36

Figure 5.2. Illustrative Example for Tuned OWLSAM Outputs 37

LIST OF TABLES

Table 3.1. Hyperparameters for Distillation 30

Table 3.2. Hyperparameters for Training 31

Table 4.1. Mean Average Precision for SegInW 33

Table 4.2. Mean IoU IoU on PASCAL-VOC 2012 Validation Set 34

Table 4.3. Comparison of Distilled and Fine-tuned Model Performance in Intersection-over-Union for Each Class 35

LIST OF ABBREVIATIONS

ViT	Vision Transformer
VLM	Vision Language Model
SAM	Segment Anything Model
KL-divergence	Kullback-Leibler Divergence
IoU	Intersection-over-Union
CNN	Convolutional Neural Network
mAP	Mean Average Precision
mIoU	Mean Intersection-over-Union
SginW	Segmentation in the Wild Benchmark
PASCAL-VOC	Pascal Visual Object Classes Benchmark
T	Temperature term in loss

1. INTRODUCTION

This section gives a brief overview of the problem, aim and contribution of this dissertation and finally, an outline.

1.1. PROBLEM STATEMENT

Image segmentation is an important task in computer vision that aims to divide an image into semantically meaningful regions. These regions encompass objects or things. Image segmentation models often classify a limited set of labels, which the model is trained on. Open vocabulary image segmentation overcomes this by adding a natural language interface to the models which can input any text query to segment the object described by the query.

Recent paradigm shift in AI came with the emergence of foundation models that leverage not only massive scale of compute but large datasets that are created with data engines, self-training and pseudo-labelling using the models that are trained. This has unlocked universal generalist models. The initial foundation models, such as CLIP [4] and contrastive pre-training paved the way for open-vocabulary and multimodal models. CLIP is a multimodal model that consist of an image and text encoder that are jointly pre-trained to yield multimodal embeddings. Vision foundation models have gained significant attention due to their ability to perform zero-shot learning on various computer vision tasks. These models are pre-trained on large-scale datasets and can be used directly for inference without the need for fine-tuning on specific downstream tasks. Examples of vision foundation models include ALIGN [5], Florence [6], and CoCa [7]. While vision foundation models excel at zero-shot learning, fine-tuning them on specific tasks can further improve their performance. Popular model architectures for fine-tuning include Vision Transformers (ViT) [8] and ConvNeXt [9]. These architectures have demonstrated superior performance when fine-tuned on various computer vision benchmarks. Following CLIP, OWL-ViT [2] model have repurposed contrastive pre-training for open-vocabulary object detection. OWLv2 [10] has scaled OWL-ViT to increase zero-shot performance using large scale compute and dataset. Another notable vision foundation model is the Segment Anything Model (SAM) [1], a large mask generation model. SAM leverages a combination of supervised and self-supervised

learning techniques to generate high-quality object masks from input prompts, enabling interactive segmentation applications. SAM only accepts point and box prompt, and due to this it is not an open-vocabulary segmentation model. Combining SAM with different open-vocabulary detection models through prompting text to open-vocabulary detection models, getting bounding boxes and passing boxes to SAM model composes state-of-the-art pipelines for generalist open-vocabulary image segmentation task. However, there hasn't been an attempt to combine OWLv2 with SAM to create this pipeline. As the field of foundation models continues to evolve, researchers are exploring ways to improve their efficiency, interpretability, and adaptability to various tasks. Techniques such as knowledge distillation [11], and few-shot learning are being investigated to make foundation models more accessible and applicable to real-world scenarios.

1.2. AIM OF THIS DISSERATION

This thesis aims to investigates two topics: can we get state-of-the-art results by combining OWLv2 model with SAM model for open-vocabulary image-segmentation, and ways to transfer the information from this model to a smaller architecture. The latter unlocks two enhancements, firstly, less dependency on labelling services, as it proves we can use pseudolabels from larger models instead of labelling services. To do so, we distill the model. Thus, secondly, distillation also unlocks better training performance with the presence of soft targets (foundation model logits).

1.3. CONTRIBUTION OF THIS THESIS

Open vocabulary image segmentation is an important task in computer vision with many practical applications. However, state-of-the-art models that combine zero-shot object detection and mask generation tend to be large and computationally expensive. While there have been significant efforts to distill models in other machine learning domains, relatively less work has explored distillation techniques for vision models, especially in the emerging area of open vocabulary segmentation.

This thesis makes several key contributions to fill this gap. We present a novel approach

to distill an open-vocabulary segmentation model by leveraging both ground truth masks and pseudo-labels generated by the model itself. To our knowledge, this is the first work to explore distillation for the specific case of open vocabulary segmentation models that integrate zero-shot detection and segmentation. Secondly, propose a new open vocabulary segmentation model architecture called OWLSAM, which combines OWLv2 zero-shot segmentation model and Segment Anything model that serves as the teacher model, enabling high-quality open vocabulary segmentation. Through extensive experiments, we demonstrate that our distillation approach using OWLSAM can produce efficient student models that maintain high segmentation accuracy while being significantly smaller and faster than the teacher. This shows the potential for distillation to make open vocabulary segmentation more practical for resource-constrained settings. Furthermore, we show how OWLSAM can be used to automatically generate high-quality synthetic training data by segmenting objects in unlabeled images. This can help to address the data bottleneck often faced when training segmentation models for new domains. Our work enables more efficient and flexible segmentation models that can be applied to a wider range of real-world computer vision tasks. We distill OWLSAM model to a smaller architecture and compare the same architecture fine-tuned on the same dataset. We show that it is possible to transfer the knowledge in open-vocabulary segmentation models to make use of it for edge device deployment.

1.4. OUTLINE

The thesis is following six key chapters. First chapter is introduction to clarify the aim and point of this dissertation. It is followed by the literature review, which provides a comprehensive survey of relevant research and theoretical frameworks that inform, and presents the research and key advancements that follows to this day. In the methodology there is the details of the preprocessing pipeline and the experiments. Results are then presented, showcasing the findings from the experiments. The discussion interprets the implications of these findings in relation to the research questions and practical applications, addressing limitations and potential future directions. Finally, the conclusion encapsulates the primary contributions of the study, emphasizing its overall impact and summarizing key insights.

2. LITERATURE REVIEW

This chapter elaborates on the research and the key advancements made that led up to this thesis. It includes the main architectures and models that scaled well, enabling the foundation models in computer vision. The chapter also includes a technique called knowledge distillation as well which this thesis heavily relies on. Last section contains the researches that are closest to this thesis.

2.1. VISION TRANSFORMERS

Vision Transformers (ViTs) [12] represent a recent breakthrough in computer vision, departing from the traditional convolutional neural networks (CNNs) [13] that have long dominated the field. ViTs aim to leverage the success of transformers in natural language processing for image understanding tasks.

CNNs have been the cornerstone of various computer vision tasks, characterized by their ability to capture spatial hierarchies through convolutional layers. However, as image datasets grow larger and tasks become more complex, CNNs face challenges in capturing global dependencies across the image. Transformers, initially designed for sequential data in NLP, offer a promising alternative by allowing for global self-attention mechanisms, enabling the model to learn relationships between all image pixels directly.

ViTs replace the convolutional layers of CNNs with a stack of transformer blocks [14]. Each block consists of self-attention layers and feed-forward neural networks. This architecture enables ViTs to process images as sequences of patches, thereby facilitating better long-range interactions and context aggregation compared to CNNs.

Empirical studies have demonstrated that ViTs achieve competitive performance against CNNs [15] on various image classification benchmarks such as ImageNet [16] and COCO [17]. Moreover, ViTs excel in tasks requiring fine-grained understanding and context-aware feature extraction, making them suitable for applications beyond classification, including object detection, segmentation, and even generative tasks like image synthesis.

Despite their success, ViTs pose challenges such as computational efficiency, especially for high-resolution images, and sensitivity to data augmentation strategies. Ongoing research focuses on addressing these limitations through hybrid architectures combining CNNs and transformers, as well as quantization and knowledge distillation to smaller and computationally efficient architectures.

2.2. CONTRASTIVE PRE-TRAINING AND MULTIMODALITY

Contrastive Language–Image Pretraining (CLIP) [4] model represents a significant advancement in the field of artificial intelligence, specifically in multimodal understanding. Developed to bridge the gap between natural language processing and computer vision, CLIP can understand and generate text from images and vice versa, making it a powerful model to unlock multimodal applications. CLIP is built on contrastive learning, a technique where the model learns to associate images and text that are semantically related while distinguishing them from unrelated pairs. The model architecture comprises two primary components: an image encoder and a text encoder. The image encoder in CLIP is typically a convolutional neural network (CNN) or a Vision Transformer (ViT). This component processes the input images to produce a fixed-dimensional feature vector. The encoder is trained to capture high-level visual features that are pertinent to the semantic content of the images. The text encoder is a transformer-based model to encode inputs into embeddings, similar to BERT model [18]. The text encoder processes the input text (such as captions, descriptions, or labels) and converts it into a corresponding feature vector. The text encoder is designed to understand and represent the semantic nuances of natural language. During training, CLIP maximizes the cosine similarity between the feature vectors of matched image-text pairs (image and text encoder outputs) while minimizing the similarity between unmatched pairs. This process is known as contrastive learning and is crucial for aligning the visual and textual representations in a shared latent space. One of the most remarkable features of CLIP is its ability to perform zero-shot learning. This means that the model can generalize to open-ended labels thanks to text encoder without explicit task-specific training on limited set of labels. Inherently, CLIP can accomplish zero-shot image classification with open-vocabulary labels through assessing similarity between text and image queries in shared latent space. This

behavior enabled training of open-vocabulary zero-shot object detection models like OWL.

2.3. OWLV2 MODEL

OWL-ViT is an open vocabulary object detector capable of detecting objects with open-ended labels. The model architecture comprises two main components: a text encoder and an image encoder, resembling a CLIP-like architecture. The training process of OWL-ViT involves two stages. First, image-text contrastive pre-training is performed using a large-scale dataset to learn a shared representation space for both visual and textual data. This stage leverages a contrastive loss that maximizes the similarity between corresponding image-text pairs while minimizing the similarity between non-corresponding pairs.

For the image encoder, OWL-ViT employs a Vision Transformer (ViT) architecture. During the transfer to open-vocabulary object detection, the final token pooling layer and projection layer are removed. Instead, a classification and object detection head is attached to the output tokens of the image encoder. Each output token is linearly projected to obtain per-object image embeddings, which are then used for classification and bounding box prediction via a small multi-layer perceptron (MLP). The text embeddings, generated by passing category names or textual descriptions through the text encoder, serve as queries for open-vocabulary classification. These queries enable the model to predict the presence and location of objects that may not have been explicitly present in the training data.

The OWLv2 model extends OWL-ViT by scaling through self-training [12]. This involves using pseudo-annotations from the largest OWL-ViT model to weakly supervise the training process, thereby increasing data coverage. Additionally, OWLv2 employs advanced techniques such as multi-head attention pooling for aggregating image representations and logit scaling for improved prediction confidence.

The OWL-ViT model architecture does not fuse image and text encoders early in the process, allowing the model to process thousands of text queries independently for each image. This design choice enhances the efficiency and flexibility of the model during inference, as query embeddings can be precomputed and reused across different images.

Overall, OWL-ViT and its successor OWLv2 represent significant advancements in

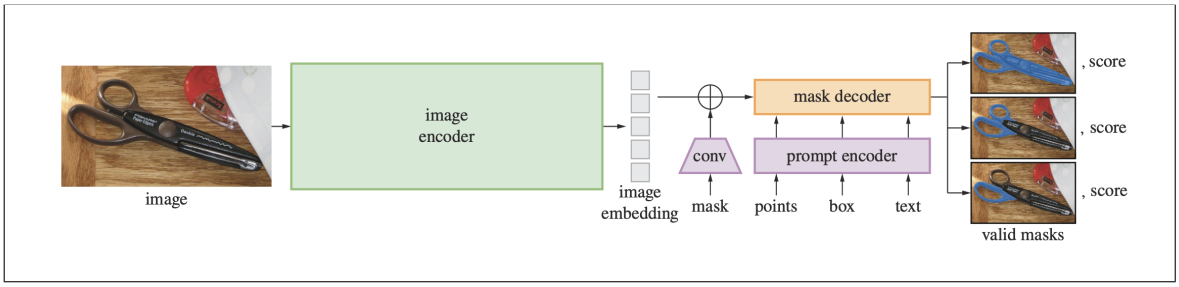


Figure 2.1. SAM Architecture, from the paper [1]

open-vocabulary object detection, combining the strengths of Vision Transformers with large-scale contrastive pre-training and innovative transfer learning techniques to achieve robust and scalable object detection capabilities. Architecture of OWL can be seen in Figure 2.2

2.4. SEGMENT ANYTHING MODEL

Segment Anything Model [1] can accomplish two tasks introduced in the Segment Anything paper, called Segment Anything and Segment Everything. The SAM model is essentially built for promptable segmentation, where prompts consist of bounding boxes and points around and within objects or regions to be segmented, and the model doesn't accept text prompts. The Segment Anything task refers to the capability of the SAM model to segment objects or regions of interest based on various types of prompts. In contrast, the Segment Everything task involves fully automatic segmentation of all regions and objects in an entire image without any specific prompts, achieved by creating a grid of points and feeding them as point prompts. In this task, SAM aims to identify and segment all objects and regions within the image independently.

SAM consists of three main components: an image encoder, a prompt encoder, and a mask decoder. The mask decoder takes in the outputs of the image encoder and prompt encoder and produces masks. SAM's success can be largely attributed to its training on the extensive SA-1B dataset, which includes 11 million images and one billion masks generated through pseudo-labelling using the model itself. Architecture of SAM can be seen in Figure 2.1

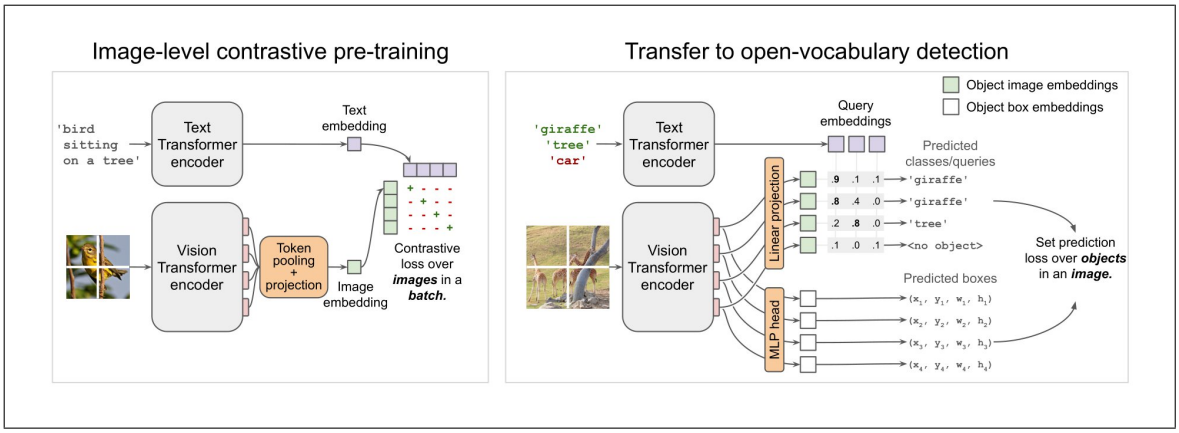


Figure 2.2. OWL Architecture from paper [2]

2.5. SEMANTIC SEGMENTATION

Semantic segmentation is a critical task in computer vision that involves classifying each pixel in an image into one of several predefined categories, effectively partitioning the image into semantically meaningful regions or objects of interest. Unlike image classification, which assigns a single label to an entire image, semantic segmentation provides a detailed understanding of the scene by labeling each pixel, making it essential for applications like autonomous driving, medical imaging, and scene understanding. Early approaches leveraged traditional machine learning techniques, but the field has seen significant advancements with the advent of deep learning. The state-of-the approaches include models based on Fully Convolutional Network (FCN) [19], and ViT based approaches. An example input output for a semantic segmentation can be seen in Figure 2.3.

2.6. OPEN-VOCABULARY IMAGE SEGMENTATION

Open-vocabulary image segmentation is an generalization task in computer vision designed to identify and delineate objects in images based on textual descriptions, without being constrained to a predefined set of classes. This approach leverages the capabilities of image encoder and text decoder models to recognize and segment objects. As of now, there are three methods to achieve open-vocabulary image segmentation. First method is combining a backbone, a feature enhancer and a mask head with a text encoder. OFA [20] integrates this

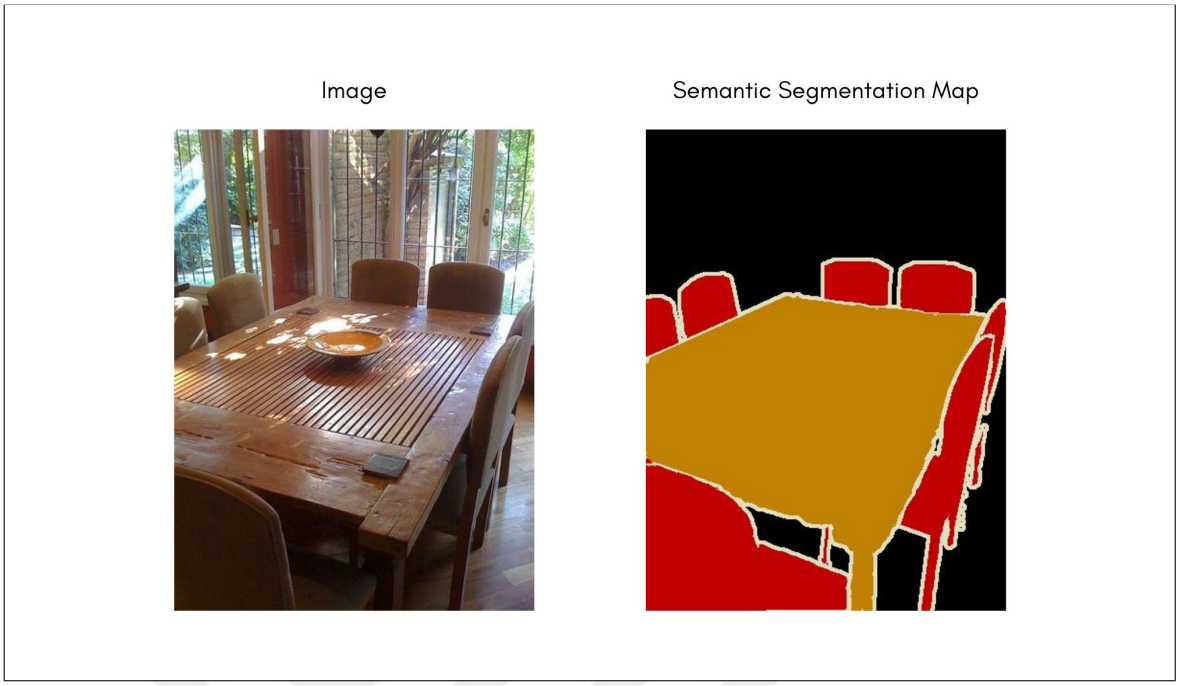


Figure 2.3. An example input output from PASCAL VOC 2012 semantic segmentation dataset

Mask R-CNN architecture with a text encoder, which allows it to interpret textual descriptions and segment objects that fit that description. The text encoder converts the textual input into a format that the visual model can process, enabling the segmentation of objects based on textual queries. This integration allows the model to perform instance segmentation tasks by leveraging both visual and textual information, enhancing its flexibility and capability to handle diverse and unseen objects. Another recent approach is using vision language models for image segmentation. Vision language models (VLMs) are used to do a variety of tasks that take in text and image and output text, including visual question answering or image captioning. These models, when pre-trained with image inputs and bounding box coordinate and segmentation token outputs, are capable of doing zero-shot object detection and image segmentation. PaliGemma [21] is the first and as of now only model to accomplish such task. It outputs segmentation tokens which are later decoded to a segmentation mask by another network trained separately. Last approach is to combine a zero-shot object detector with mask generation model. GroundedSAM [22] combines GroundingDINO [23], a zero-shot object detector, with the state-of-the-art mask generation model, Segment Anything Model (SAM). This paper combines OWLv2 with SAM for open vocabulary image segmentation, and explores the use of such model as a teacher model for knowledge distillation.

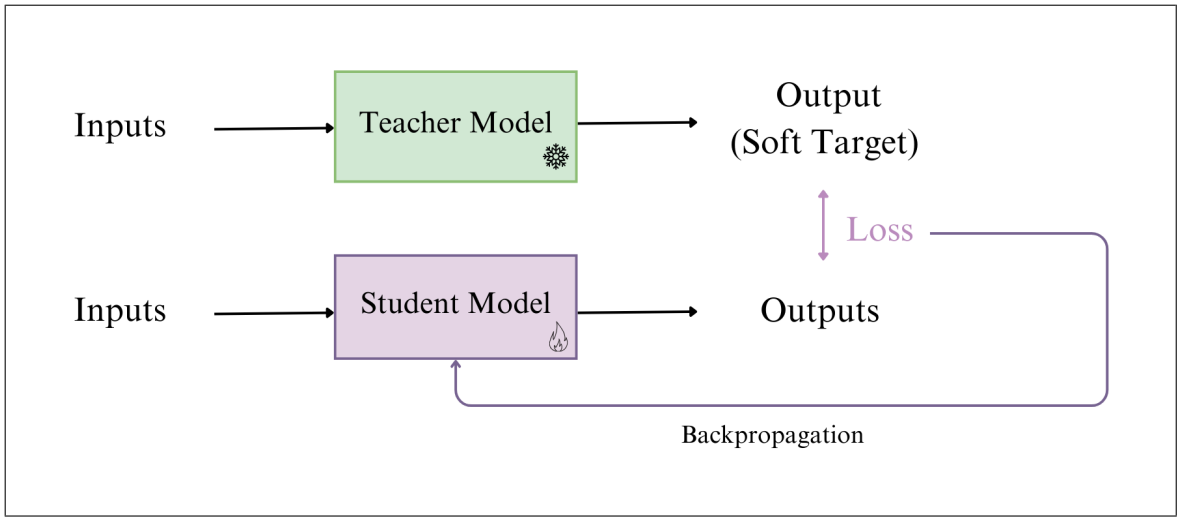


Figure 2.4. Knowledge Distillation

2.7. KNOWLEDGE DISTILLATION

Knowledge distillation (KD) is a widely used technique to enhance the performance of deep learning models without modifying their architectures. The concept of distilling the knowledge from a larger, pre-trained teacher model to a smaller student model was firstly introduced by Hinton et. al [11]. This process involves supervising the student model using both hard labels (the ground truth) and soft labels (the probability distribution over classes output by the teacher model). This dual supervision helps the student model learn not just the correct output, but also the nuances of the teacher's learned knowledge.

Building on this foundational work, numerous studies have sought to enhance the utilization of soft labels to transfer more comprehensive knowledge. Tian et al. [19] refined the distillation process by decoupling representation learning from classification, enabling the student model to first focus on learning improved feature representations before tackling the classification task.

Another major advancement in knowledge distillation involves the transfer of intermediate features from the teacher model. Another novel method involves a self-supervised teaching assistant to guide training of a ViT [12] based student model [24]. FitNet is another work, where the semantic information from the teacher's intermediate layers is directly distilled into the student model. This approach takes advantage of the rich hierarchical features learned by

the teacher, providing the student with more detailed guidance [25].

2.8. MOBILENET AND DEEPLAB

MobileNet [26] is a family of neural network architectures designed for efficient computation and high performance on mobile and embedded devices. Developed by Google, MobileNet models are optimized for applications requiring low latency and limited computational resources, such as real-time object detection and image classification on smartphones and IoT devices. The key innovation of MobileNet is the use of depthwise separable convolutions, which significantly reduce the number of parameters and computational cost compared to standard convolutions, without sacrificing accuracy.

MobileNetV1 [26] introduced depthwise separable convolutions, splitting the standard convolution into a depthwise convolution and a pointwise convolution. This decomposition reduces the number of parameters and multiplications, making the network lighter and faster. MobileNetV2 [27] built upon this by introducing inverted residuals and linear bottlenecks, further enhancing performance while maintaining efficiency. These architectural improvements allow MobileNetV2 to achieve higher accuracy with fewer parameters and less computational load than its predecessor.

MobileNet employing very lightweight architecture significantly reduces the number of parameters and computational load without substantially sacrificing accuracy, which makes it a great candidate for edge device applications. Additionally, MobileNet is designed for efficient inference, enabling quick data processing with low latency, essential for real-time applications on edge devices. Its scalability, facilitated by width and resolution multipliers, allows the model to be adjusted based on the available computational resources, making it versatile for a wide range of edge devices. These attributes collectively render MobileNet an ideal choice for edge computing scenarios where resource constraints demand both efficient and effective computation.

DeepLab is a family of neural network models designed for semantic image segmentation [28]. DeepLab introduced atrous (or dilated) convolutions, which allows controlling the resolution at which feature responses are computed within Deep Convolutional Neural Networks. This

enables a wider field of view without losing resolution. DeepLabV2 introduced Atrous Spatial Pyramid Pooling (ASPP), which is an enhanced atrous convolution approach that uses multiple atrous convolutions with different rates in parallel, capturing multi-scale contextual information [28]. DeepLabV3 builds on DeepLabV2 by adding batch normalization to ASPP module and adding image-level features [29]. This model also removed conditional random field modules. Lastly, DeepLabV3+ improves DeepLabV3 by using DeepLabV3 as an encoder and adding a decoder on top of it, which refines the segmentation results [30].

2.9. DISTILLATION IN SEGMENTATION

The practical application of foundation models is often hindered by their considerable size and complexity, posing significant challenges in terms of efficiency for compute. Transferring the universal information in these models to smaller architectures became a necessity, which is addressed by applying knowledge distillation through using foundation models as teacher models. The application of knowledge distillation to foundation models is particularly promising. Given their extensive pre-training and high-level feature representations, these models serve as ideal candidates for the role of teacher models. The distilled student models, derived from these robust foundations, can then be deployed in scenarios where computational resources are constrained, such as in edge devices or real-time applications.

There are numerous attempts to distill Segment Anything Model, and most attempts are preserving the universal information rather than accomplishing task-specific knowledge distillation. Segment Anything Model consists of three parts, a heavy image encoder, a prompt encoder and a decoder to decode the encodings to segmentation masks. Most approaches have attempted to replace the image encoder of SAM with a more lightweight architecture.

MobileSAM is a distilled SAM model where heavyweight image encoder is distilled with a lightweight image encoder, making the model 60 times smaller and 5 times faster [31]. The researchers explored three distillation methods to reduce the SAM model's size while maintaining its effectiveness. They first attempted a straightforward approach by directly distilling the decoder outputs. This involved using a small, randomly initialized Vision

Transformer (ViT) and mask decoder. However, this method proved ineffective when both the ViT and decoder were poorly initialized. The researchers subsequently investigated two additional distillation approaches. The first, termed "semi-coupled" distillation, involved randomly initializing only the Vision Transformer image encoder while retaining the original mask decoder. This method is considered semi-coupled as the image encoder distillation process remains dependent on the mask decoder. The second approach, dubbed "decoupled" distillation, represents a more radical departure. In this method, the researchers completely isolated the image encoder distillation process, freezing the mask decoder and eschewing distillation based on generated masks. This strategy aligns with the understanding that the encoder typically constitutes the primary computational bottleneck, and distillation often yields optimal results when applied to encoding processes. Notably, the researchers' findings indicate that the decoupled distillation method outperforms coupled distillation in terms of mean Intersection over Union (IoU), while simultaneously requiring significantly reduced computational resources. This discovery has important implications for model optimization in computer vision tasks.

EfficientSAM [32], a lightweight variant of the Segment Anything Model (SAM) designed to address computational efficiency concerns while maintaining competitive performance. Their approach, called SAM-leveraged Masked Image Pretraining (SAMI), utilizes a masked autoencoder framework to distill knowledge from SAM's large ViT-H image encoder into smaller, more efficient encoders like ViT-Tiny and ViT-Small. SAMI pretrains these lightweight encoders to reconstruct features from the SAM encoder using a mean squared error loss, effectively transferring SAM's rich visual representations. The authors demonstrate that EfficientSAM models, built with SAMI-pretrained encoders and SAM's original mask decoder, achieve comparable performance to the full SAM on zero-shot instance segmentation tasks while significantly reducing computational costs. For instance, EfficientSAM-S reduces inference time by approximately 20x and parameter count by 20x compared to SAM, with only a small performance drop (44.4 AP vs 46.5 AP on COCO). The study also shows SAMI's effectiveness in improving performance on various computer vision tasks beyond segmentation, including image classification and object detection.

ZeroSeg [33] is a novel approach for open-vocabulary semantic segmentation that eliminates the need for human-annotated labels. ZeroSeg leverages knowledge from pretrained

vision-language models, specifically CLIP, to train a segmentation model without relying on text supervision or pixel-level annotations. The method employs a carefully designed architecture that incorporates a masked autoencoder framework for efficiency and a segmentation head with learnable segment tokens. ZeroSeg uses two key loss functions: a multi-scale feature distillation loss and a segment matching loss. These losses enable the model to capture object-localized semantic information at various scales and align segment tokens with corresponding image regions. The model is trained on 1.3M ImageNet images, and it achieves comparable or superior performance to models trained on much larger image-text pair datasets or with segmentation labels. The authors demonstrate ZeroSeg's effectiveness through quantitative evaluations on standard benchmarks (PASCAL VOC, PASCAL Context, and COCO), as well as qualitative assessments and human studies for open-vocabulary segmentation tasks.

Another approach [34] distills the Segment Anything Model (SAM) for planetary geological mapping, specifically for on the detection of Martian skylights. The authors' method employs knowledge distillation from SAM to a domain-specific task with minimal manual annotation. The authors freeze SAM's image encoder and train a lightweight domain-specific decoder, consisting of only three upsampling layers. This approach allows for efficient learning of task-specific semantics while preserving the robust feature extraction capabilities of SAM. The study demonstrates that with as few as five annotated samples, their model achieves performance comparable to or better than a Mask R-CNN baseline trained on 405 manually annotated images. The authors also evaluate different SAM prompt modes for annotation, finding the bounding box mode most effective for their task. This work showcases the potential of foundation models like SAM in accelerating specialized segmentation tasks in planetary science, significantly reducing the manual annotation burden on domain experts.

SlimSAM [35] presents a novel approach to distilling the Segment Anything Model while maintaining high performance with minimal training data. The authors propose an alternate slimming framework that decomposes the model into two sub-structures: embedding and bottleneck. The process alternates between pruning and distilling these decoupled structures, which helps minimize divergence from the original model and enables more effective knowledge transfer. To address the misalignment between pruning objectives and training targets, the authors propose a novel importance estimation criterion called disturbed Taylor

importance. This approach aligns the pruning criteria with the optimization objectives of subsequent distillation, resulting in improved performance recovery. SlimSAM leverages knowledge distillation from both intermediate layers and final output embeddings. This approach significantly improves the training results, with MIoU improvements of 1.22 percent and 0.57 percent for different distillation steps. One of the most notable aspects of SlimSAM is its ability to achieve high performance with extremely limited training data. The method uses only 0.1 percent (10,000 images) of the SA-1B dataset, which is 10 times less data than comparable methods like EdgeSAM and MobileSAM. The distillation process is carried out in a progressive manner, with two main steps: embedding pruning followed by bottleneck pruning. After each pruning step, knowledge distillation with intermediate layer aligning is employed to recover performance. Global vs. Local Pruning: The authors explore both local and global pruning strategies for bottleneck features, finding that global pruning with appropriate normalization can yield better results, especially with increased training iterations. The authors report that their SlimSAM-50 and SlimSAM-77 models achieve parameter reductions to 4.0 percent (26M) and 1.4 percent (9.1M) of the original count, respectively, while maintaining performance levels comparable to the original SAM-H. This approach to distilling SAM represents a significant advancement in model compression techniques, particularly in scenarios with limited training data availability. The combination of structural pruning, knowledge distillation, and careful consideration of model architecture allows SlimSAM to achieve remarkable efficiency gains while preserving the robust segmentation capabilities of the original SAM.

2.10. DISTILLATION IN WITH PSEUDOLABELS FROM FOUNDATION MODELS

Readily available foundation models can be used for distillation through labels they generate. These labels are called "pseudolabels". Distillation with pseudolabels leverages the knowledge and capabilities of large, pre-trained models to improve smaller, more efficient models. This process involves using a large foundation model (often called the "teacher") to generate high-quality labeled data, which is then used to train a smaller model (the "student"). The key idea is that the foundation model, having been trained on vast amounts of data, can generate accurate and diverse labels for a wide range of tasks. These labels are called

"pseudolabels" - they aren't created by human annotators, but are assumed to be of high quality due to the foundation model's broad knowledge. This reduces the need for expensive and time-consuming human annotation, can generate large amounts of diverse training data quickly and it allows for the transfer of complex knowledge from large models to smaller, more deployable ones. It can be applied to various domains and tasks where the foundation model has relevant knowledge.

Zephyr paper [36] follows aforementioned approach for distilling knowledge from large language models (LLMs) to create smaller, aligned models without relying on human annotations. This process leverages pseudolabels generated by foundation models to improve the capabilities of more compact models. The authors apply distilled supervised fine-tuning from a dataset of AI-generated instructions and responses, called UltraChat. The larger teacher model generates multi-turn dialogues based on seed prompts. These synthetic dialogues serve as training data for the student model, allowing it to learn conversational patterns and knowledge from the teacher.

Gandhi et al. [37] proposed Distil-Whisper, a method for distilling the Whisper speech recognition model into a smaller, faster variant while maintaining its robustness across different acoustic conditions. Their approach uses large-scale pseudo-labeling to create a diverse training dataset spanning 10 domains. They employ a simple word error rate heuristic to filter and select only high-quality pseudo-labels for training. The distillation process involves freezing Whisper's encoder, initializing a lightweight decoder from select layers of Whisper's [38] decoder, and training on a combination of pseudo-label and KL-divergence losses. Notably, Distil-Whisper achieves performance within 1 percent word error rate of the original Whisper model on out-of-distribution test data, while being 5.8 times faster and using 51 percent fewer parameters. The authors also demonstrate that Distil-Whisper can be paired with Whisper for speculative decoding, doubling inference speed while ensuring identical outputs to the original model. This work showcases the potential of using foundation models to generate high-quality pseudo-labels for training efficient, task-specific models without compromising performance or generalization ability.

3. METHODOLOGY

This chapter includes the details about our experiment and the pre-processing setup for the model, the dataset we experiment on and the problem definition.

3.1. PROBLEM DEFINITION

Drawing on recent advancements in foundation models and knowledge distillation techniques within computer vision, this thesis aims to enhance the training of smaller models for semantic segmentation. It proposes leveraging weak supervision from an open-vocabulary image segmentation model by using the output logits as soft targets in task-specific knowledge distillation. This process involves training a compact student model to emulate the behavior of a larger, more powerful teacher model by aligning its predictions to the softened probabilities produced by the teacher. By doing so, the student model not only learns the explicit labels available in the training dataset but also benefits from the rich, implicit knowledge embedded in the teacher's logits, encompassing nuances and variations not explicitly labeled. This method is particularly effective in scenarios where labeled data is scarce or expensive, as it allows the student model to approximate the generalization capabilities of the teacher with significantly reduced computational resources.

Semantic segmentation is a pivotal task in computer vision that involves assigning a specific label to each pixel of an image, thereby enabling machines to understand and interpret detailed visual information with high granularity. This process is crucial in various applications such as autonomous driving, medical image analysis, and scene understanding, where precise pixel-level predictions are necessary. An example input output for semantic segmentation can be seen in Figure 3.1.

The performance of semantic segmentation models is typically evaluated using several key metrics. First is pixel accuracy, which measures the percentage of pixels in the image that are correctly classified. While straightforward, this metric is not sensitive to class imbalances where some classes are more prevalent than others. Another one is Intersection-over-Union (IoU) (also known as the Jaccard Index), this metric evaluates segmentation accuracy by

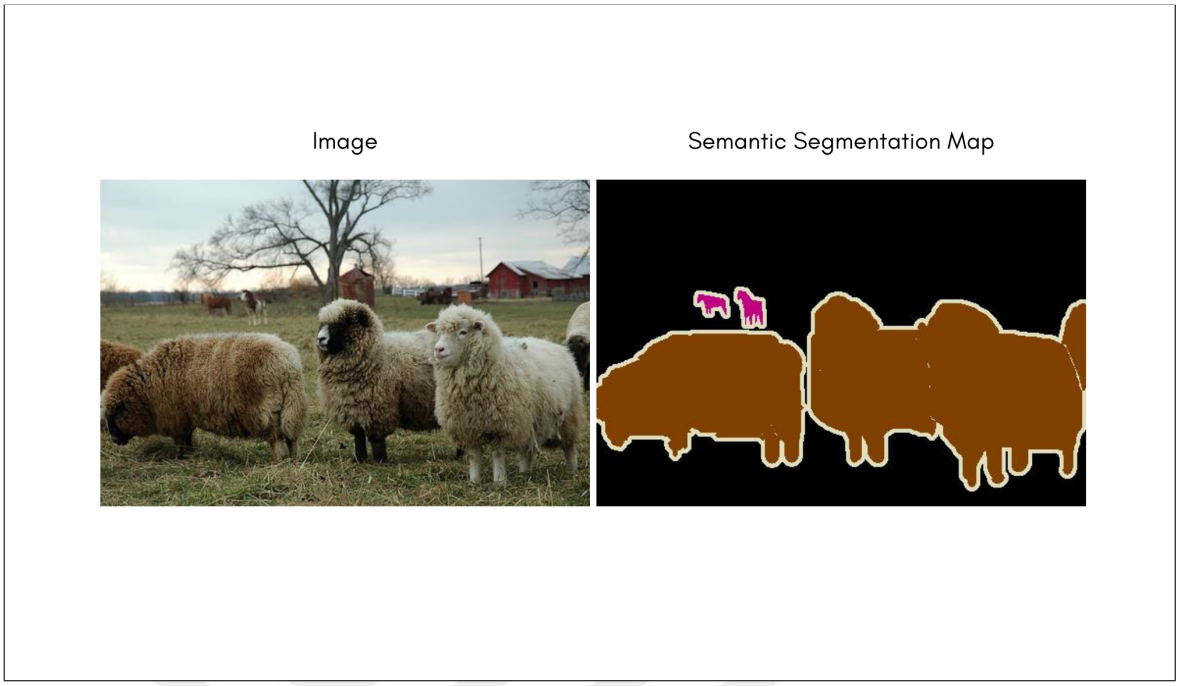


Figure 3.1. An example input output from PASCAL VOC 2012 semantic segmentation dataset

calculating the ratio of the intersection to the union of the predicted and ground truth masks for each class, and then averaging across classes. It is particularly useful for balancing the influence of large and small objects. Similar to IoU, the Dice coefficient measures the overlap between the predicted segmentation and the ground truth. It is calculated as twice the area of overlap divided by the total number of pixels in both the prediction and the ground truth.

To optimize these metrics, semantic segmentation models rely on various loss functions during training, which guide the learning process by quantifying the difference between the predicted and true labels. A common choice for pixel-wise classification in semantic segmentation is cross-entropy loss. It calculates the loss for each pixel independently and averages over all pixels, effectively encouraging the model to match the ground truth labels. Derived from the Dice Coefficient, Dice loss function is particularly effective for dealing with class imbalance by focusing on the overlap between the predicted and actual segments. Focal loss modifies the standard cross-entropy loss to place more focus on hard-to-classify pixels. This adjustment helps in handling imbalances by reducing the relative loss for well-classified examples, allowing the model to focus more on difficult cases.

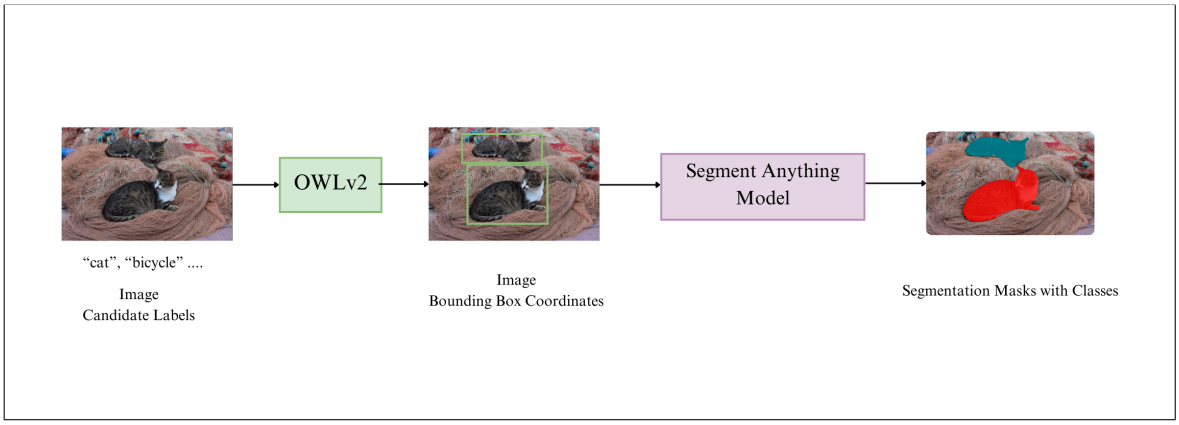


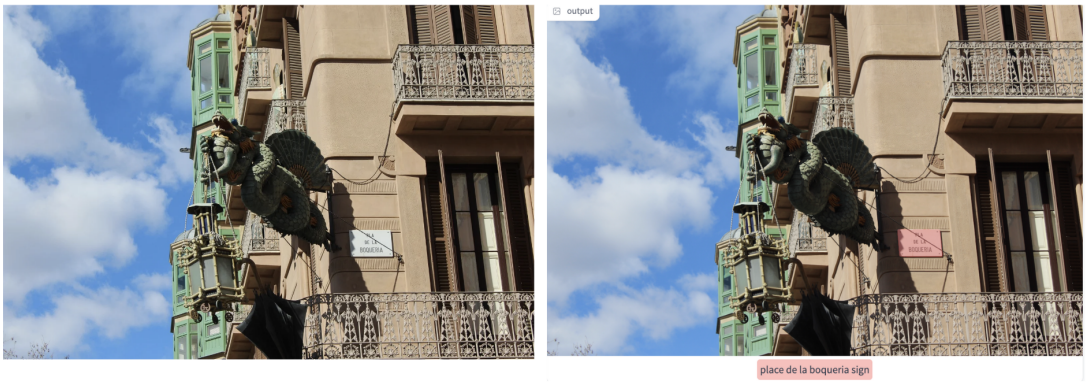
Figure 3.2. Overview of the OWLSAM Model

3.2. PROPOSED APPROACH

In this thesis, we introduce two significant contributions. Firstly, we propose OWLSAM, an open-vocabulary segmentation model that combines OWLv2 and the Segment Anything Model. The input output schema of OWLSAM can be seen in "Figure 3.2" OWLv2 model is used primarily to add natural language interface, adding semantic information to SAM output masks. Second and primary contribution of this thesis involves investigating the use of OWLSAM logits as soft labels in task-specific knowledge distillation. By using OWLSAM logits as teacher signals, we explore how this approach can guide the distillation process better. For student model, we have used MobileNetV2 architecture. For the task-specific knowledge distillation experiment setup, we use the PASCAL Visual Object Classes (VOC) [39] dataset, a benchmark in semantic segmentation, to evaluate and demonstrate the effectiveness of our proposed method. This method also unlocks using OWLSAM as a pseudolabeller for semantic segmentation, reducing the need for human annotations for semantic segmentation.

3.3. TEACHER MODEL

In our distillation experiments, we have used OWLSAM as teacher model and MobileNetV2 with DeepLabV3+ segmentation head as the student model [30]. Overview of the distillation process can be seen in "Figure 3.5". OWLv2 model used here is ensemble of the pre-trained OWLv2 base model and OWLv2 fine-tuned on LVIS [40] dataset which already includes the classes of PASCAL-VOC 2021. The rationale behind choosing this model how ensembling



Text query:“place de la boqueria sign”

Figure 3.3. Open-vocabulary Example for OWLSAM Text Detection

Text query:“bitten apple”

Figure 3.4. Open-vocabulary Example for OWLSAM

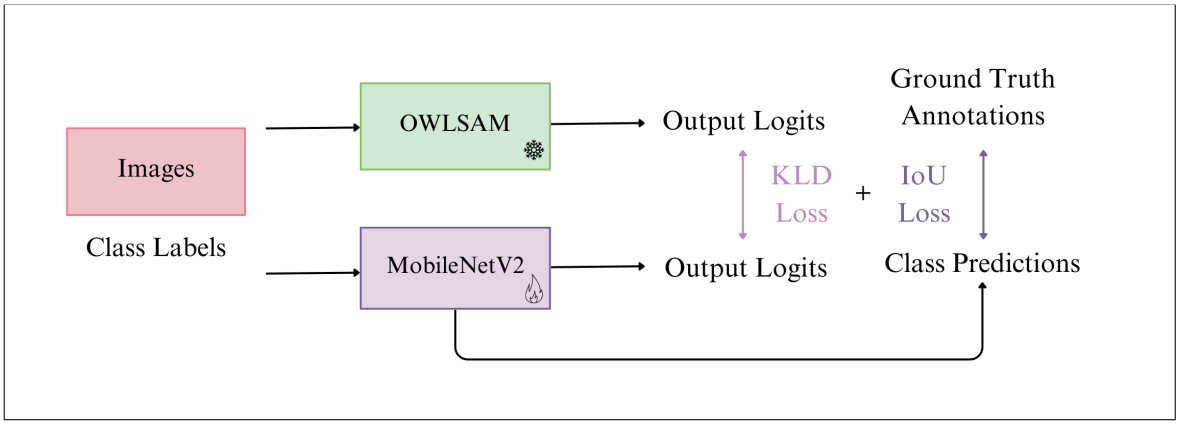


Figure 3.5. Distillation Process

a fine-tuned model with the base model provides a good trade-off in average precision of detections in the wild, compared to only using a base model or only using fine-tuned model [10]. The authors have ensembled the models through averaging the model weights. Base model performs better for open-vocabulary queries, meanwhile ensemble and fine-tuned models are more suitable for labelling tasks for pre-defined set of labels of everyday objects. We have combined this model with Segment Anything Model ViT-B checkpoint. We have first passed training split of PASCAL-VOC 2012 through OWLSAM model and saved each predicted labels along with their output logits and segmentation masks.

3.4. IMPLEMENTATION DETAILS

This section includes technical lower level details on implementation: the dataset, experimentation hyperparameters for distillation and experimentation hyperparameters for the fine-tuning for comparison.

3.4.1. PASCAL-VOC Dataset

The PASCAL Visual Object Classes (VOC) dataset is a widely used benchmark for object detection, image classification, semantic segmentation, and action classification. It was introduced in 2005 and ran challenges from 2005 to 2012. The dataset consists of photographs collected from Flickr, covering a wide range of scene categories. The PASCAL VOC 2012

dataset, specifically, contains a total of 11,540 images with objects from 20 different classes: aeroplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dining table, dog, horse, motorbike, person, potted plant, sheep, sofa, train, and TV/monitor. For the segmentation task, a subset of the images also includes pixel-wise annotations. In addition to above classes, there are two more classes: background and bordering region. Background here indicates all the pixels except for aforementioned classes and bordering region. Bordering region are the pixels that are on the edges between background, on top of aforementioned classes (pixels in between the objects and pixels between background and the objects). These bordering regions are excluded from loss calculations and bordering region pixels are often omitted in the final model evaluations. In the PASCAL VOC 2012 dataset, the segmentation train set contains 1,464 images and the validation set contains 1,449 images. These segmentation annotations provide detailed object boundaries and enable evaluation of semantic segmentation algorithms. We have utilized this part of the dataset for our experiments. We have split the train split into two splits with split ratio of 0.2, to training and evaluation split. We employ evaluation split to check the Intersection-over-Union and loss throughout the training. We use the dataset's own validation split to conduct the final validation of the model.

3.4.2. Knowledge Distillation Details

Knowledge distillation losses consist of two parts, a loss based on KL-divergence between teacher logits and student logits called "soft target", and a task-specific metric between student labels and ground truth targets called "hard target". Hard targets are the actual class labels used during standard training. In this case, semantic segmentation masks from training dataset is our hard target.

Soft targets are the class probabilities output by the teacher model. Instead of providing a hard binary decision, the teacher model gives a probability distribution over all classes. These probabilities often contain more information than hard targets because they capture the relative confidence of the teacher model in each class. To estimate the difference between output probabilities (logits) of student model and teacher model, KL-divergence is used. The Kullback-Leibler (KL) divergence is a measure of how one probability distribution diverges

Figure 3.6. Example Images and their masks from PASCAL-VOC 2012 Dataset

from a second, expected probability distribution.

It is calculated as follows.

$$\text{KL-Divergence between } P \text{ and } Q: \quad D_{KL}(P \parallel Q) = \sum_i P(i) \log\left(\frac{P(i)}{Q(i)}\right) \tag{3.1}$$

During the distillation process, the goal is to train the student model to replicate the behavior of the teacher model. This is achieved by minimizing the KL divergence between the output probability distributions of the teacher model and the student model.

The loss function used in typical knowledge distillation problems combines the cross-entropy loss on the hard targets and the KL divergence on the soft targets. In our problem, we use Intersection-over-Union on hard targets and KL-divergence on soft targets.

$$\text{Loss} = \alpha \cdot \left(\frac{1}{N}\sum_{i=1}^{N}\left(1 - \text{IoU}(Y_i, \hat{Y}_i)\right)\right) + \beta \cdot T^2 \cdot \left(\sum_{i=1}^{N} q_i \log\left(\frac{q_i}{p_i}\right)\right) \tag{3.2}$$

First term of the loss is reserved for IoU calculation, where:

- Y_i represents the ground truth segmentation mask for the i-th sample.

- $\hat{Y}_i$ represents the predicted segmentation mask for the i-th sample.

- $(1 - \text{IoU})$ is IoU subtracted by 1 to incorporate it over loss. We take the average over N samples.

Second term in the loss is reserved for KL-Divergence: $\beta \cdot T^2 \cdot \left(\sum_{i=1}^{N} q_i \log\left(\frac{q_i}{p_i}\right)\right)$ where:

- q_i represents the teacher logits (soft targets).

- p_i represents the student logits.

- T is the temperature parameter.

The temperature parameter is used to soften the output probabilities of the teacher model, making the differences between probabilities less pronounced. The softmax function used to convert logits (raw model outputs) into probabilities is adjusted by this temperature parameter

as follows.

The temperature-adjusted softmax function is given by:

$$\text{softmax}(z_i) = \frac{e^{z_i/T}}{\sum_j e^{z_j/T}} \tag{3.3}$$

where T is the temperature parameter.

$$\text{loss} = \alpha \cdot L_{\text{KL}} + 0.5 \cdot \big(1 - \text{IoU}(\text{gt}_i, s_i)\big) \tag{3.4}$$

The loss function combines two terms as seen in Equation 3.2. First part of the loss function, L_{KL} is the KL Divergence based soft target term, where α is the weight applied to the KL divergence loss term L_{KL}. The second term involves the Intersection over Union (IoU) metric, which measures the overlap between the ground truth targets (gt_i) and the student model's labels (s_i).
The term $1 - \text{iou}(\text{gt}_i, s_i)$ penalizes the loss based on the IoU, with the coefficient 0.5 scaling its contribution to the total loss.

KL-divergence tends to decrease very easily even when signals are still different, so we have used a technique called KL-divergence warm-up to assign dynamic weights of importance to KL-divergence instead of keeping weights for hard and soft targets constant. The KL-divergence warm-up weight coefficient swings between 0.1 and 0.5, meanwhile the weight coefficient for hard targets are fixed at 0.5. We have used a fixed temperature of 5 for our experiments.

3.4.3. Other Technical Details

Since the Segment Anything Model (SAM) is trained to generate instance masks, we converted logits and segmentation masks to repurpose them for the semantic segmentation task. In instance segmentation, the goal is to detect and delineate each object instance within an image, whereas semantic segmentation aims to classify each pixel into a predefined class without distinguishing between different instances of the same class. To convert instance

Algorithm 3.1. KL-divergence Warm-up and Combined Loss

Input:
    $lr\_adjustment\_factor = 0.05$
    $max\_kl = 1.0$
    $min\_kl = 0.1$
    $student\_probs$
    $teacher\_probs$
    $gt\_targets$
    $student\_labels$
Output: Loss

$kl\_loss \leftarrow kl\_div(student\_probs, teacher\_probs)$
if $kl\_loss < min\_kl$ **then**
    $kl\_weight \leftarrow kl\_weight + lr\_adjustment\_factor$
else
    $kl\_weight \leftarrow kl\_weight - lr\_adjustment\_factor$
end if
$kl\_weight \leftarrow \max(kl\_weight, 0.05)$

$kl\_loss\_term \leftarrow kl\_weight \cdot kl\_loss$
$kl\_loss\_term \leftarrow \max(\min(kl\_loss, max\_kl), min\_kl)$
$loss \leftarrow kl\_loss\_term + 0.5 \cdot (1 - \text{iou}(gt\_targets, student\_labels))$

masks to semantic masks, we simply aggregate instance masks for each class and convert them to binary masks by filtering for any non-zero pixel. This ensures that all pixels belonging to a particular class are grouped together. To convert instance logits to semantic logits, for each image, we multiply the binary mask with segmentation logits and sum the resulting matrices for each class. This aggregation process helps in transforming the detailed instance-level information into class-level information suitable for semantic segmentation. By aligning the logit values and segmentation outputs in this manner, we make the teacher outputs compatible with student outputs, ensuring that the training process for semantic segmentation is coherent and effective.

Another trick applied to make teacher outputs compatible with the student is that we resize the segmentation masks and logits. We apply bilinear interpolation to resize the logit matrix. Bilinear interpolation works by taking the four nearest pixel values to the target location and computing the weighted average based on the distance of these pixels to the target point. This method considers both the horizontal and vertical directions, ensuring a smoother and more accurate resizing process. This technique helps in maintaining the spatial coherence of the logits.

We label every pixel with no label as "background" and ignore background pixels during loss calculation. This is one limitation of OWLSAM, since each class has to be semantically meaningful, text queries like "background" cannot be inferred as they do not hold any meaning. In semantic segmentation, most datasets consist of limited number of classes and any pixel that is not classified within those classes get classified as "background". Meanwhile, foundation models built on image and text encoders require the classes to be semantically meaningful, and background would simply be misclassified.

Hyperparameters used for our experiments are given in Table Table 3.1. We haven't applied any augmentations to the images. We employ MobileNetV2 with DeepLabV3+ head with upsampling head of size 256, 256, normally the model outputs 65, 65 feature maps. We initially process the inputs to 513, 513, since this is the resolution the pre-trained MobileNetV2 model takes in. The learning utilizes an initial learning rate of 0.00005 due to usage of low batch sizes, as low batch size prevents overfitting. Moreover, this is due to how loss calculation takes place in GPU, and our teacher logits take up space. We train for 50 epochs on L4 GPU,

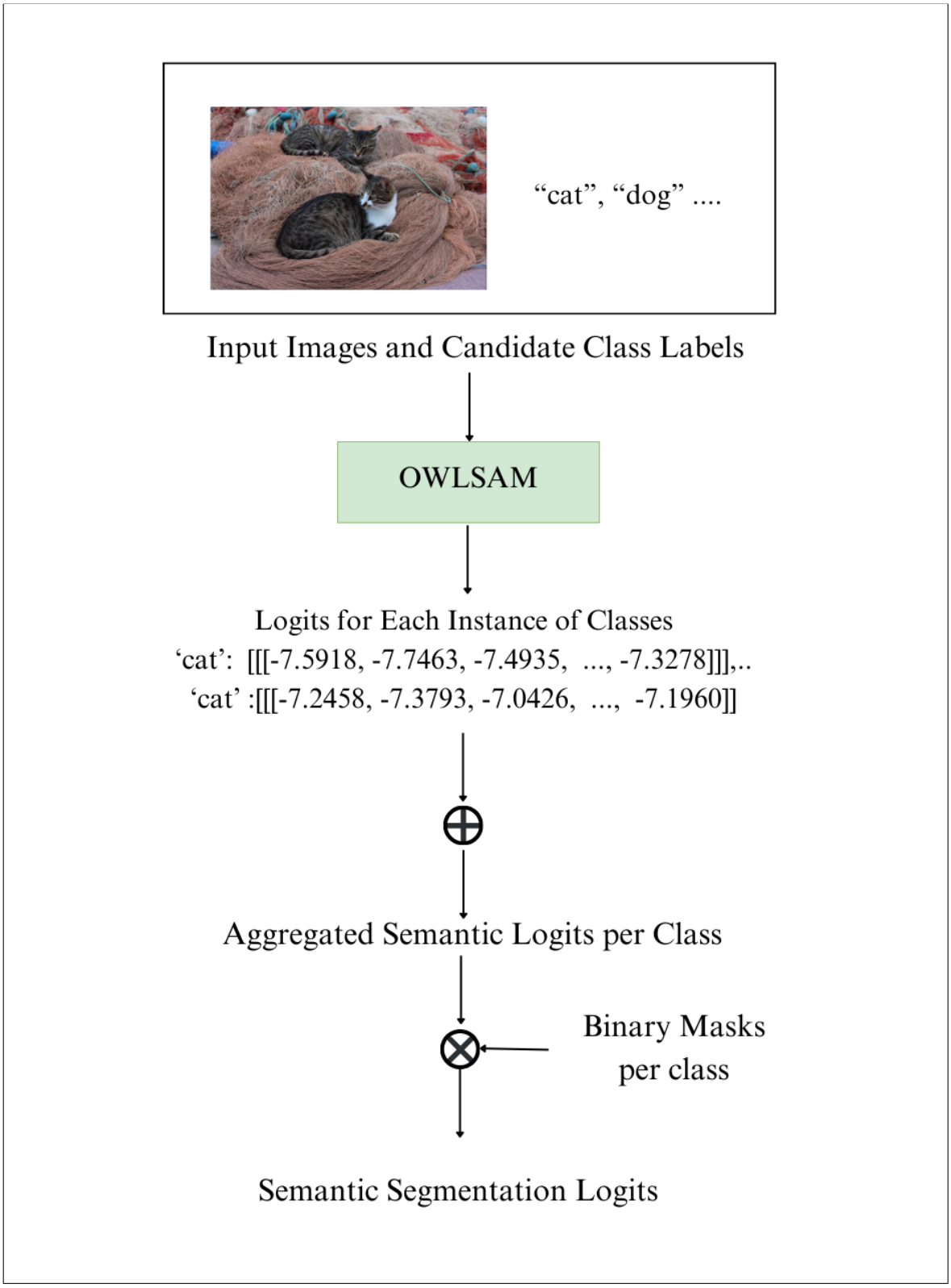


Figure 3.7. Preprocessing Pipeline to Obtain Soft Targets

reason why we picked was that it has a high RAM where we can compute the KL-divergence and take teacher logits which consume a lot of memory and cannot be taken to the VRAM. We employ cosine scheduler for learning rate, with an initial warm-up phase that spans 10 percent of the total training steps. During this warm-up period, the learning rate gradually increases from a near-zero value to the specified maximum of 0.0001, allowing the model to establish stable gradient directions before full-scale optimization begins. Following the warm-up phase, the learning rate follows a cosine decay curve, smoothly decreasing from its peak value in a cyclical manner. This scheduling strategy serves multiple purposes, the warm-up period helps prevent early training instability caused by large gradient updates on randomly initialized parameters, while the subsequent cosine decay provides an adaptive balance between exploration and exploitation throughout the training process. The gradual reduction in learning rate enables fine-grained parameter adjustments as training progresses, while the periodic nature of the cosine function allows the optimization to escape poor local minima. We picked the Adam Weight Decay optimizer, which enhances the adaptive learning rate capabilities of Adam with effective weight decay regularization. We have frozen every layer until layer 8 and rest of the parameters, including complete segmentation head (DeepLabV3+ and classifier layer with upsampling) are trainable. We have realized the model as is doesn't yield good IoU score in the earlier steps, specifically during warm-up it yields average of 0.003, so we wanted to only benefit from the early layers as is. Finally, we apply mixed-precision training using 16-bit floating-point (FP16) arithmetic, a technique that significantly optimizes computational efficiency and memory utilization. This approach maintains model weights in 32-bit precision while performing forward and backward passes using 16-bit precision, effectively halving the memory requirements for activations and gradients. FP16 training not only reduces memory consumption but also accelerates training speed by up to 2-3 times compared to single-precision training, while maintaining comparable model accuracy.

3.4.4. Model Fine-tuning Details

We compare our method to model fine-tuning. Here, we only fine-tune the same architecture on the same dataset with minor differences. For transformations, we only apply random

Table 3.1. Hyperparameters for Distillation

Hyperparameter	**Value**
Initial KL-Divergence Weight	0.1
Maximum KL-Divergence Weight for KL Warm-up	0.1
Minimum KL-Divergence Weight Value	0.01
KL Adjustment Factor	0.05
IoU Weight	0.5
Learning Rate Scheduler	Linear
Initial Learning Rate	1e-3
Number of Epochs	50

horizontal and vertical flipping augmentation, synchronized with the labels. The training configuration for the bare model fine-tuning employs a comprehensive set of hyperparameters designed to optimize the model's learning process, which can be found in Table 3.2. We employ MobileNetV2, the same architecture, with upsampling head of size (375, 500). This is picked particularly considering most of the PASCAL VOC dataset keeps this aspect ratio. We initially process the inputs to (513, 513), since this is the resolution the pre-trained MobileNetV2 model takes in. The learning utilizes an initial learning rate of 0.005, which is found through our experiments, to balance training stability and convergence speed the best. We train 60 epochs, with batch sizes of 128 samples for training batches and 128 samples for evaluation batches, to effectively utilize single A100 with 80 GB of VRAM, as we wanted to particularly not spend as much time on this as the main experiment. Just like the main experiment we have frozen every layer until layer 8 and rest of the parameters, including complete segmentation head (DeepLabV3+ and classifier layer with upsampling) are trainable. We have experimented with completely freezing everything but the segmentation head, and not freezing any of the layers. For this one, there's a trade-off of fitting in larger batches against increasing training capacity of the model. With freezing all but the head, we could fit in 128 examples per training batch in an A100 40GB, and with not freezing, we were able to fit in 32 items per batch. With unfreezing selectively, we can fit in 64 items per batch, and the model learns better. Unfortunately the pre-trained model lacks good IoU performance out of the box (comparing first evaluation IoU for each training setup) per starting point for transfer learning, thus we had to unfreeze later layers. Lastly, after

600 steps the model started to overfit, so we freeze everything except the head and resume training with a lower learning rate (precisely 5e-4) and similar scheduler setup for 20 more epochs. We used cross entropy loss, where we only ignore the bordering lines class. This is a common practice with segmentation tasks as these pixels are often ambiguous. Lastly, we apply mixed-precision training using 16-bit floating-point (FP16) arithmetic as well.

Table 3.2. Hyperparameters for Training

Hyperparameter	**Value**
Initial Learning Rate	0.005
Number of Epochs	70
Train Batch Size	128
Evaluation Batch Size	128
Gradient Accumulation Steps	2
Learning Rate Scheduler	Cosine
Warm-up Ratio	0.1
Optimizer	Adam Weight Decay
Loss	Cross Entropy Loss

4. RESULTS

This chapter includes the results of our experiments.

4.1. OWLSAM RESULTS

In this work, we contribute a new open-vocabulary segmentation model called OWLSAM, and we benchmark OWLSAM with different combinations of OWLv2 and SAM models on Segmentation in the Wild [41] benchmark and compare to other open-vocabulary segmentation models.

To conduct final evaluation for both OWLSAM and Distil-OWLSAM, we have used mean average precision (mAP) and mean Intersection-over-Union (mIoU), which is defined as:

$$\text{mean IoU} = \frac{1}{N}\sum_{i=1}^{N}\frac{|A_i \cap B_i|}{|A_i \cup B_i|} \tag{4.1}$$

where A_i and B_i represent the predicted and ground truth sets, respectively, and N is the total number of sets. You can see an explanatory figure on Intersection-over-Union in Figure 4.1.

Mean Average Precision (mAP) is calculated by first determining the Average Precision (AP) for each class, which integrates the precision-recall curve. To compute AP, predictions are ranked by confidence scores, and precision and recall are evaluated at various thresholds. The precision values are interpolated over the range of recall from 0 to 1, and the area under this

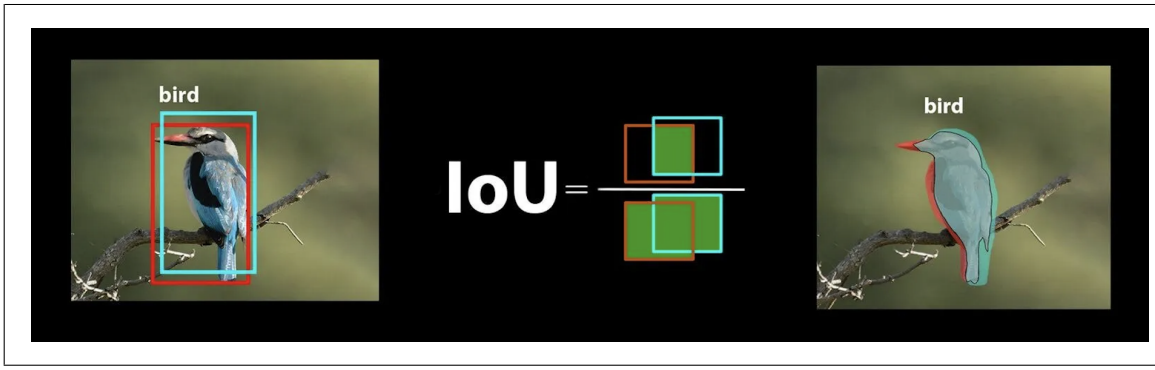


Figure 4.1. Visualized Intersection-over-Union from [3]

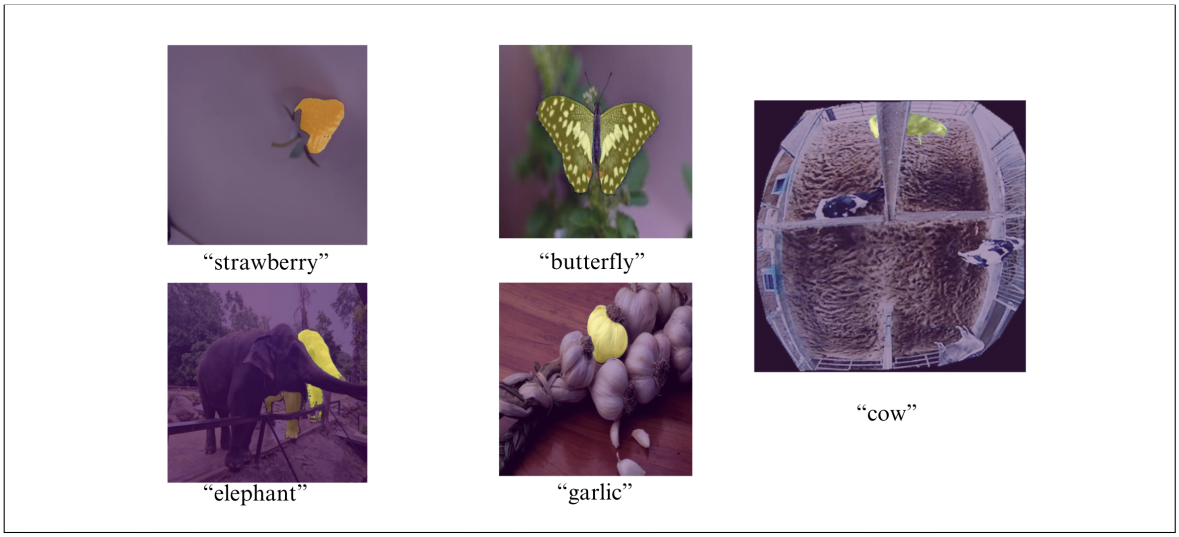


Figure 4.2. OWLSAM Outputs on SegInW

curve gives the AP. The mAP is then obtained by averaging the AP values across all classes. This metric assesses both localization accuracy (using IoU thresholds) and classification confidence.

The results of other models are taken from Grounded-SAM [22] paper. We compare against X-Decoder [41] as well. For benchmarking, we have used a confidence of 0.1 for OWLv2 model and gradually decrease to 0.05 and finally 0.01 in case no bounding boxes are found in the image.

Table 4.1. Mean Average Precision for SegInW

Model	**Mean IoU**
Grounded-SAM (L+H)	46.0
X-Decoder-T	22.6
X-Decoder-L-IN22K	26.6
OpenSeeD-L	36.7
OWLSAM-H*	41.2

* OWLSAM with OWL and SAM ViT-H checkpoint.

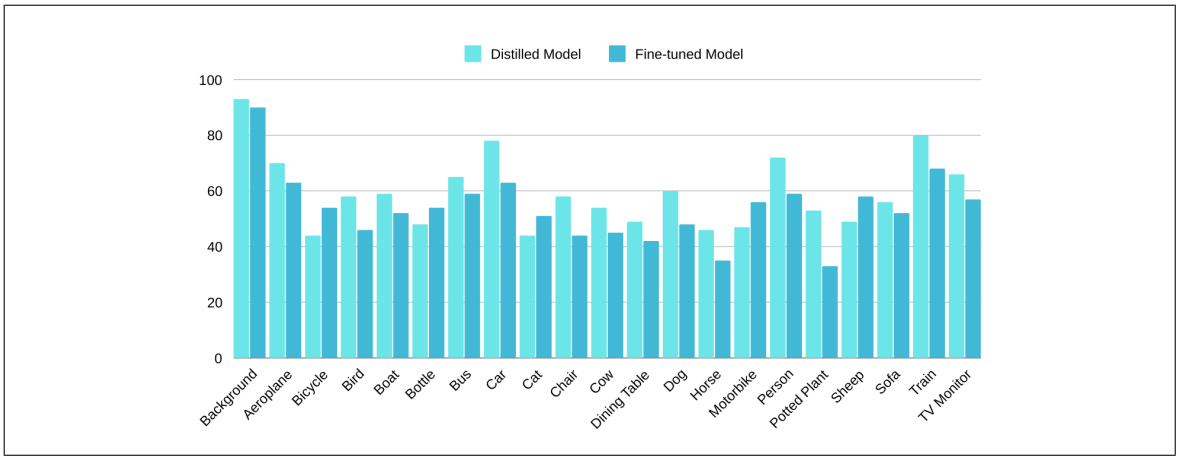


Figure 4.3. Intersection-over-Union distributions per class

4.2. DISTILLATION RESULTS

The results of distilled model on PASCAL-VOC 2012 validation set are a given in Table 4.2.

Table 4.2. Mean IoU IoU on PASCAL-VOC 2012 Validation Set

Model	**Mean IoU in Percentage**
Distilled MobileNetV2	58.4
Fine-tuned MobileNetV2	54.5

We provide the percentage Intersection-over-Union per class values in Table 4.3.

Percentage Intersection-over-Union per class distribution over validation set of PASCAL VOC 2012 is visualized in Figure 4.3.

Table 4.3. Comparison of Distilled and Fine-tuned Model Performance in Intersection-over-Union for Each Class

Labels	**Distilled Model**	**Fine-tuned Model**
Background	91	90
Aeroplane	70	63
Bicycle	44	54
Bird	58	46
Boat	59	52
Bottle	48	54
Bus	65	59
Car	78	63
Cat	44	51
Chair	58	44
Cow	54	45
Dining Table	49	42
Dog	60	48
Horse	46	35
Motorbike	47	56
Person	72	59
Potted Plant	53	33
Sheep	49	58
Sofa	56	52
Train	80	68
TV Monitor	66	57

5. DISCUSSION

In this chapter there is findings about the distillation process and the foundation model. One trick required to combine OWL and SAM was to filter their outputs. Both models generate multiple outputs and have confidence thresholds. OWL model has a confidence threshold to filter output boxes and SAM model has an Intersection-over-Union (IoU) threshold. SAM model predicts IoU scores for each output, which can be used to filter outputs with less IoU. The original codebase of SAM uses 0.88 as the output. In this work, we have discovered that using a low confidence threshold for OWL and a high IoU threshold for SAM makes more sense to maximize IoU of the predicted outputs.

We have discovered that specifically tuning confidence thresholds for OWL matters the most compared to tuning IoU threshold for SAM as per sensitivity. For OWL, having a lower confidence threshold either means wrongly predicted bounding boxes or having larger bounding boxes, resulting in false positive pixels. In Figure 5.1, it is seen that having a higher threshold reduces false positives, increasing false negatives per pixel, resulting in more conservative outputs. In the same figure, lower confidence threshold shows multiple bounding boxes, thus masks over one instance. One has to tune the confidence threshold to find correct bounding boxes, thus masks. In Figure 5.2, we demonstrate that so long as OWL threshold is proper, there is not a lot of change with SAM output masks.

For distillation, we have figured OWLSAM outputs can sometimes contain noise depending on the OWL threshold and SAM threshold. SAM model outputs are very precise which helps keeping a high resolution with the pseudo-annotations. However, to make our experiments

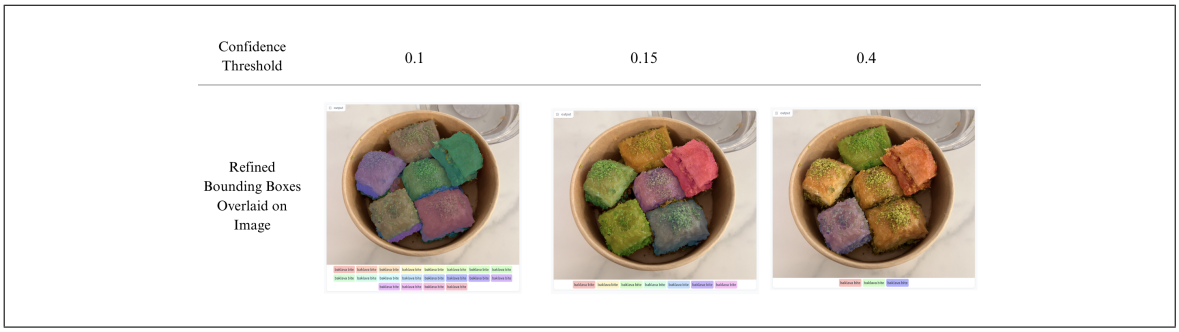


Figure 5.1. Illustrative Example for Tuned OWL Confidence Thresholds

Figure 5.2. Illustrative Example for Tuned OWLSAM Outputs

work in PASCAL VOC validation dataset (ground truth), we had to ground the OWLSAM outputs and get rid of noise, we have leveraged the segmentation masks from the PASCAL VOC 2012 segmentation. We have multiplied the logits of the model with the segmentation masks and tried to reduce the signal on rest of the pixels through putting a very small value epsilon (1e-5). Without these tricks in the preprocessing pipeline, the model was generating noisier output in the initial epochs. Moreover, in the initial steps of the training, because teacher and student outputs are too different, the KL-Divergence is different as well. Because we employ KL-Divergence warm-up, initial weight for KL-divergence was the minimum value threshold for nearly two epochs.

6. CONCLUSIONS

In this thesis, we explored the distillation of an open-vocabulary segmentation model, OWLSAM, which combines the OWL and SAM models, into a more efficient MobileNetV2 architecture. Through the distillation process, we aimed to preserve the robustness and flexibility of OWLSAM while significantly reducing the model's computational complexity, making it more suitable for deployment on resource-constrained devices.

Our experiments demonstrated that the distilled MobileNetV2 model, is more compact and efficient. Additionally, we compared the distilled model with a MobileNetV2 trained from scratch, showing that our approach yields a model that not only benefits from the distilled knowledge of OWLSAM but also offers superior segmentation capabilities in open-vocabulary tasks.

Success in our experiments proves the potential of leveraging advanced models like OWLSAM in conjunction with efficient architectures like MobileNetV2, enabling more practical applications of state-of-the-art segmentation models in real-world scenarios. Moreover, this also unlocks the use of open-vocabulary models as robust pseudolabelers.

6.1. FUTURE WORK

Future work could explore multiple additions to this training pipeline and conduct ablation studies on swapping teacher and student models. One promising direction is to fine-tune the OWL model on the task-specific dataset before distillation and compare its performance to the out-of-the-box OWL model. This fine-tuning step could potentially improve the quality of the distilled model. Furthermore, quantizing the fine-tuned OWL model to reduce the precision of the logits could make the training more efficient in terms of memory and time. The quantized OWL model can then be integrated into the teacher model for distillation. Another area of exploration is the use of alternative student architectures. Investigating the use of smaller models, such as SlimSAM [35], as a replacement for the SAM model in the student architecture could further reduce the model size and computational requirements while maintaining performance. Additionally, experimenting with different architectures to replace

MobileNetV2 in the student model could provide insights into the trade-offs between model size, computational efficiency, and performance. With the recent release of the SAM2.1 [42] model, it would be worthwhile to explore its use as a replacement for the original SAM model used in this study. SAM2.1 was not available at the time of this research but could potentially bring improvements to the distillation process. Moreover, distilling the GroundingSAM [43] model, which combines object detection and segmentation capabilities, could extend the applicability of the distillation approach to a wider range of tasks. Lastly, we would also like to investigate use of only pseudolabels from OWLSAM model for training, instead of a complete knowledge distillation setup with soft targets and ground truth labels. Post-inference tuning is another avenue for future research. Investigating post-inference tuning techniques for both the open-vocabulary detection model and the mask generation model could involve fine-tuning the models on additional data or adapting them to specific domains or use cases. This could further optimize the models' performance and adaptability to real-world scenarios. Lastly, future work can contribute to the advancement of model distillation in the context of open-vocabulary object detection and segmentation. The exploration of fine-tuning and quantization techniques, alternative architectures, upgraded models, and post-inference tuning can lead to more efficient and effective distilled models, expanding the possibilities for deploying these models in resource-constrained environments.

REFERENCES

1. Kirillov A, Mintun E, Ravi N, Mao H, Rolland C, Gustafson L, et al. Segment Anything. *arXiv preprint arXiv:230402643*. 2023;abs/2304.02643.

2. Minderer M, Gritsenko A, Stone A, Neumann M, Weissenborn D, Dosovitskiy A, et al. Simple Open-Vocabulary Object Detection with Vision Transformers. *arXiv preprint arXiv:220506230*. 2022;abs/2205.06230.

3. LearnOpenCV. Intersection over Union (IoU) in Object Detection and Segmentation. 2022. Accessed: 2024-11-08. Available from: https://learnopencv.com/intersection-over-union-iou-in-object-detection-and-segmentation/.

4. Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, et al. Learning Transferable Visual Models From Natural Language Supervision. *arXiv preprint arXiv:210300020*. 2021;abs/2103.00020.

5. Jia C, Yang Y, Xia Y, Chen YT, Parekh Z, Pham H, et al. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. 2021.

6. Yuan Y, Chen X, Wang J. Florence: A New Foundation Model for Computer Vision. *arXiv preprint arXiv:211111432*. 2021.

7. Yu J, Wang Z, Vasudevan V, Yeung L, Seyedhosseini M, Wu Y. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:220501917*. 2022.

8. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *CoRR*. 2020;abs/2010.11929. Available from: https://arxiv.org/abs/2010.11929.

9. Liu Z, Mao H, Wu CY, Feichtenhofer C, Darrell T, Xie S. A ConvNet for the 2020s. *arXiv preprint arXiv:220103545*. 2022.

10. Minderer M, Gritsenko A, Houlsby N. Scaling Open-Vocabulary Object Detection. *arXiv preprint arXiv:230609683*. 2024;abs/2306.09683.

11. Hinton G, Vinyals O, Dean J. Distilling the Knowledge in a Neural Network. *arXiv*

preprint arXiv:150302531. 2015;abs/1503.02531.

12. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:201011929*. 2021;abs/2010.11929.

13. LeCun Y, Bengio Y. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*. 1995;3361(10):1995.

14. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention Is All You Need. *arXiv preprint arXiv:170603762*. 2023;abs/1706.03762.

15. Jeeveswaran K, Kathiresan S, Varma A, Magdy O, Zonooz B, Arani E. A Comprehensive Study of Vision Transformers on Dense Prediction Tasks. *arXiv preprint arXiv:220108683*. 2022;abs/2201.08683.

16. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet Large Scale Visual Recognition Challenge. *arXiv preprint arXiv:14090575*. 2015;abs/1409.0575.

17. Lin TY, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, et al. Microsoft COCO: Common Objects in Context. *arXiv preprint arXiv:14050312*. 2015;abs/1405.0312.

18. Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:181004805*. 2019;abs/1810.04805.

19. Long J, Shelhamer E, Darrell T. Fully Convolutional Networks for Semantic Segmentation. *arXiv preprint arXiv:14114038*. 2015;abs/1411.4038.

20. Sun P, Chen S, Zhu C, Xiao F, Luo P, Xie S, et al. Going Denser with Open-Vocabulary Part Segmentation. 2023.

21. Beyer L, Steiner A, Pinto AS, Kolesnikov A, Wang X, Salz D, et al. PaliGemma: A Versatile 3B VLM for Transfer. *arXiv preprint arXiv:240707726*. 2024;abs/2407.07726.

22. Ren T, Liu S, Zeng A, Lin J, Li K, Cao H, et al. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks. *arXiv preprint arXiv:240114159*. 2024;abs/2401.14159.

23. Liu S, Zeng Z, Ren T, Li F, Zhang H, Yang J, et al. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. *arXiv preprint arXiv:230305499*. 2023.

24. He L, Hua B. Semi-Supervised Knowledge Distillation Via Teaching Assistant. *Highlights in Science, Engineering and Technology*. 2023 12;72:429-36.

25. Romero A, Ballas N, Kahou SE, Chassang A, Gatta C, Bengio Y. FitNets: Hints for Thin Deep Nets. *arXiv preprint arXiv:14126550*. 2015;abs/1412.6550.

26. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv preprint arXiv:170404861*. 2017;abs/1704.04861.

27. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *arXiv preprint arXiv:180104381*. 2019;abs/1801.04381.

28. Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2018;40(4):834-48.

29. Chen LC, Papandreou G, Schroff F, Adam H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv preprint arXiv:170605587*. 2017;abs/1706.05587.

30. Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018:801-18.

31. Zhang C, Han D, Qiao Y, Kim JU, Bae SH, Lee S, et al. Faster Segment Anything: Towards Lightweight SAM for Mobile Applications. *arXiv preprint arXiv:230614289*. 2023;abs/2306.14289.

32. Xiong Y, Varadarajan B, Wu L, Xiang X, Xiao F, Zhu C, et al. EfficientSAM: Leveraged Masked Image Pretraining for Efficient Segment Anything. *arXiv*. 2023;abs/2312.00863.

33. Chen J, Zhu D, Qian G, Ghanem B, Yan Z, Zhu C, et al. Exploring Open-Vocabulary Semantic Segmentation without Human Labels. *arXiv preprint arXiv:230600450*. 2023;abs/2306.00450.

34. Julka S, Granitzer M. Knowledge Distillation with Segment Anything (SAM) Model for Planetary Geological Mapping. *arXiv preprint arXiv:230507586*. 2023;abs/2305.07586.

35. Chen Z, Fang G, Ma X, Wang X. SlimSAM: 0.1 Available from: https://arxiv.org/abs/2312.05284.

36. Tunstall L, Beeching E, Lambert N, Rajani N, Rasul K, Belkada Y, et al. Zephyr: Direct Distillation of LM Alignment. 2023. Available from: https://arxiv.org/abs/2310.16944.

37. Gandhi S, von Platen P, Rush AM. Distil-Whisper: Robust Knowledge Distillation via Large-Scale Pseudo Labelling. 2023. Available from: https://arxiv.org/abs/2311.00430.

38. Radford A, Kim JW, Xu T, Brockman G, McLeavey C, Sutskever I. Robust Speech Recognition via Large-Scale Weak Supervision. 2022. Available from: https://arxiv.org/abs/2212.04356.

39. Everingham M, Eslami SMA, Van Gool L, Williams CKI, Winn J, Zisserman A. The Pascal Visual Object Classes Challenge: A Retrospective. *International Journal of Computer Vision*. 2015 jan;111(1):98-136.

40. Gupta A, Dollar P, Girshick RB. LVIS: A Dataset for Large Vocabulary Instance Segmentation. *CoRR*. 2019;abs/1908.03195.

41. Zou X, Dou ZY, Yang J, Gan Z, Li L, Li C, et al. Generalized Decoding for Pixel, Image, and Language. *arXiv preprint arXiv:221211270*. 2022;abs/2212.11270.

42. Ravi N, Gabeur V, Hu YT, Hu R, Ryali C, Ma T, et al. SAM 2: Segment Anything in Images and Videos. *arXiv preprint*. 2024.

43. Li Y, Zhang Y, Wang Y, Wang Z, Zhang Y, Chen Y, et al. GroundedSAM: Grounding Segment Anything Model with Open-Vocabulary Object Detection. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023:1234-43.