

THE UNIVERSITY of ADELAIDE

Low-shot Learning based on Pre-trained Model

ZIQIN ZHOU

A thesis submitted for the degree of
DOCTOR OF PHILOSOPHY
The University of Adelaide

November 24, 2024

Contents

Abstract xiii

Declaration of Authorship xv

Acknowledgements xvii

1 Introduction 1
1.1 Problem Statement and Research Motivation 1
1.2 Thesis Organization and Contributions 2

2 Literature Review on Exploring Low-shot Learning via Pre-trained Model 5
2.1 Introduction 5
2.2 Concepts and Preliminaries of Low-shot Learning 6
2.3 Few-shot Classification 8
2.3.1 Episodic meta-training 9
2.3.2 Representation learning 10
2.4 Low-shot Semantic Segmentation 11
2.4.1 Semantic Segmentation 11
2.4.2 Zero-shot Semantic segmentation 12
2.4.3 Open Vocabulary Semantic segmentation 12
2.4.4 Few-shot Semantic segmentation 13
2.5 Popular Pre-trained Models 13
2.5.1 CNN-based Pre-trained Models 14
2.5.2 Transformer-based Pre-trained Models 14
2.5.3 Tuning Methods on Pre-trained Models 16

3 Exploring Few-shot Classification via Distribution Prediction 17
3.1 Introduction 17
3.2 Background 17
3.3 Method 20
3.3.1 Problem Formulation 20
3.3.2 Training stage 1: Representation learning 20
3.3.3 Extension of Ridge Regression to Incorporate Sample Distribution Estimate 22
3.3.4 Training stage 2: Distribution Learning 23
3.3.5 Meta-testing on novel classes 25
3.4 Experiments 26
3.4.1 Implementation Details 26
3.4.2 Datasets 26
3.4.3 Evaluation Metric 26

3.4.4 Implementation Details . . . 27
3.4.5 Comparison with State-of-the-art methods . . . 28
3.4.6 The advantage of the proposed classifier . . . 28
3.4.7 Comparison to related classifier combined augmentation methods . . . 30
3.4.8 Evaluation on CIFAR-FS and FC100 dataset . . . 31
3.4.9 Evaluation on miniImageNet and tieredImageNet dataset . . . 31
3.4.10 Ablation Study . . . 34
3.5 Conclusion . . . 37

4 Towards Adapting CLIP for Zero-shot Semantic Segmentation 39
4.1 Introduction . . . 39
4.2 Background . . . 40
4.3 Method . . . 42
4.3.1 Problem Formulation . . . 42
4.3.2 Baseline: One-stage Text-Patch Matching . . . 43
4.3.3 Design 1: Deep Prompt Tuning (DPT) . . . 44
4.3.4 Design 2: Non-mutually Exclusive Loss (NEL) . . . 45
4.3.5 Design 3: Relationship Descriptor (RD) . . . 46
4.4 Experiments . . . 47
4.4.1 Datasets . . . 47
4.4.2 Evaluation Metrics . . . 47
4.4.3 Implementation Details . . . 47
4.4.4 Comparison with State-of-the-art methods . . . 48
4.4.5 Transfer Zero-shot model to Open-Vocabulary Semantic Segmentation . . . 50
4.4.6 Ablation Study . . . 53
4.5 Conclusion . . . 57

5 Unlocking the Potential of Pre-trained Vision Transformers for Few-Shot Semantic Segmentation through Relationship Descriptors 61
5.1 Introduction . . . 61
5.2 Background . . . 63
5.2.1 Few-shot Semantic Segmentation . . . 63
5.2.2 Relationship Descriptor (RD) in ZegCLIP [234] . . . 63
5.2.3 Semantic Grouping Capability in Pre-trained Vision Transformers . . . 64
5.3 Method . . . 64
5.3.1 Few-shot Segmentation via Prototype Embeddings and Relationship Descriptors . . . 64
5.3.2 Relationship Descriptors Design for Few-shot Segmentation 66
5.3.3 Multi-layer Relationship Descriptors . . . 66
5.3.4 Training and Inference . . . 67
5.4 Experiments . . . 68
5.4.1 Datasets and Setup . . . 68
5.4.2 Evaluation Metrics . . . 69
5.4.3 Implementation Details . . . 69
5.4.4 Comparison with Existing GFSS Methods . . . 71

5.4.5 Apply Our Pre-trained Model to FSS Setting 72
5.4.6 Ablation study 73
5.5 Implementation Details 75
5.5.1 The detailed architecture of the decoder 75
5.5.2 Details of applying our trained GFSS model to (binary) FSS setting 76
5.5.3 Effect of various pre-trained vision transformer model for FSS task 76
5.6 Conclusion 76

6 Conclusions and Future Directions **81**
6.1 Conclusions 81
6.2 Future Directions 82

A Detailed zero-shot settings **83**

B Detailed augmented templates **85**

C Visualization of semantic masks of different text query embeddings. **87**

D Brief frameworks of related Two-stage and our One-stage method **89**

E Pseudo code of our approach **91**

Bibliography **93**

List of Figures

1.1 Structure of research problem categorization of this thesis. This structure provides a comprehensive overview of how low-shot learning is approached in this thesis, leveraging pre-trained models and task-specific adaptations. Each chapter of the thesis corresponds to different components of this diagram, as indicated by the color-coded labels (orange: Chapter 3, green: Chapter 4, blue: Chapter 5). 4

3.1 A toy illustration of our proposed method which can solve sampling over-bias problem via class-wise distribution prediction and refinement. 18

3.2 Briefly comparison with very related works. (a) represents few-shot classification baseline with traditional augmentation operations, it includes representation learning in fully-supervised training phase on base dataset and meta-classifier developing in meta-testing phase on novel dataset; (b) demonstrates introducing generative model to process data augmentation in (a); (c) describes applying distribution calibration algorithm guided by seen classes to sample sufficient support images in novel tasks. Our proposed methods (d) is not modelling data augmentation but class-wise distribution to improve the adaptation ability in novel meta-classifier. 19

3.3 Overall of our proposed method. The training strategy consists of two stages: (1) Stage 1 - Representation Learning: We follow related works to pre-train a feature extractor as backbone with fully supervised learning with classification and rotation prediction tasks on base dataset. (2) Stage 2 - Distribution Learning: We explore the ability to predict feature distribution from limited samples. We fix the parameters in feature extractor and develop an Approximated Ridge Regression (ARR) as meta-classifier with different objective functions to train distribution predictor. In the meta-testing phase, we compute the solution of ARR based on the features of support set and apply the solution as weights to predict classification results on query samples. Note that, both feature extractor and distribution predictor are frozen while testing. 21

3.4 Illustration of the few-shot classification problem formulation. We follow typical setting which includes (a) pretraining feature extractor via fully supervised representation learning on base dataset and (b) meta-training on base tasks and then meta-testing on novel tasks. . . 22

3.5 t-SNE visualization of few-shot classification with traditional augmentation operations and our proposed distribution prediction. Different colors represent different classes. '$\times$' denotes the extracted features of limited *support set* and their augmented counterparts, while translucent '●' denotes the large amounts of *query set*. '▲' in (b) represents the predicted class-wise mean vectors. (a) illustrates the sampling bias on randomly sampled support set and the weakness of using augmentation operations. (b) proves that our proposed method can alleviate over-bias by predicting class-wise distribution. Note that, we only show the predicted mean vector of each class. '$\rightarrow$' represents moving from the sampled support feature which is close to class boundary to our predicted class-wise mean as centre. 27
3.6 Quantitative comparison of the effect of λ in the Ridge Regression classifier during the training and meta-testing stages. Note that λ' in the legend represents the value set during the distribution learning stage. The symbol $\star$ denotes the best performance. 36

4.1 Quantitative improvements achieved by our proposed designs on VOC dataset. (1)(2) represents our one-stage **Baseline** model of different versions (Fix or Fine-Tune CLIP image encoder), while (3)-(5) shows the effectiveness of applying our proposed designs, i.e., **Deep Prompt Tuning (DPT)**, **Non-mutually Exclusive Loss (NEL)**, **Relationship Descriptor (RD)**, on baseline model step by step. We highlight that our designs can dramatically increase the segmentation performance on unseen classes. 41
4.2 Overall of our proposed **ZegCLIP**. Our method modifies a **One-Stage Baseline** framework of matching text and patch embeddings from CLIP to generate semantic masks. The key contribution of our work is three simple-but-effective designs (labeled as the red circles 1,2,3 in the figure). Incorporating these three designs into our proposed one-stage baseline framework can upgrade the poorly performed baseline method to a strong zero-shot segmentation model. . 42
4.3 The detailed architecture of text-path matching decoder. 44
4.4 The architecture of deep prompt tuning. 45
4.5 Qualitative results on COCO-Stuff 164K. (a) are the original testing images; (b) represent the performance of our proposed one-stage baseline (fine-tuning the image encoder); (c) are the visualization results of our proposed ZegCLIP; (d) are the ground truths of each image. Note that the white and red tags represent seen and unseen classes separately. 51
4.6 Visualization results of open-vocabulary segmentation on the most challenging dataset ADE20k with 847 categories. 58
4.7 Detailed performance on unseen classes of COCO datasets. Note that "ST" represents self-training in "transductive" setting. 58
4.8 The quantitative results of applying the different numbers of deep prompt tokens. Note that "S" and "U" represent seen and unseen classes separately. 59

5.1 Visualized hidden (1) intra- and (2) cross- image matching capability of pre-trained ViT-B/16 and ResNet50 based on ImageNet. $< p, \mathbf{H} >$ denotes the cosine similarity between class-wise prototypes p and local feature $\mathbf{H}$ of Image. Prototypes p represent the average of embeddings belonging to the object-of-interest for each image, for example, the "people" and "bicycle" object of the Image in (1). It illuminates that the pre-trained transformer model has certain semantic grouping capabilities which may be helpful for few-shot semantic segmentation. 62
5.2 Overall training framework of our proposed few-shot semantic segmentation method. 64
5.3 The detailed design of our Relationship Descriptor (RD) Generator module. 65
5.4 The framework of fusing multiple relationship descriptors from different transformer layers. 67
5.5 Visualization results on PASCAL-5^i compared with other methods on GFSS setting. Note that (b)-(c) are the methods without test-time tuning, while (d)-(f) are the methods with test-time tuning on novel support images. 73
5.6 Quantitative results of (1) generalized and (2) binary few-shot semantic segmentation. (a) are the testing images, (b) and (c) are the results of our method with single and multiple relationship descriptors, (d) denotes ground truths. After enriched RDs, (c) shows better segmentation and recognition ability. 74
5.7 Quantitative results on PASCAL VOC 2012 of fusing relationship descriptors from different vision transformer layers. 75

C.1 Visualization of semantic masks of different text query embeddings. 88

D.1 Brief frameworks of related **two-stage methods**, our **one-stage baseline**, and our proposed **ZegCLIP** model. The happy face in (b)(c) means good performance in seen or unseen classes, while the sad face in (b) represents that the baseline model achieves poor performance in unseen classes. 90

List of Tables

3.1 Comparison with different traditional meta-classifiers on miniImageNet with WRN-28-10 as feature extractor. 29
3.2 Comparison with different combinations of augmentation operations and competitive meta-classifiers. 30
3.3 Comparison to prior work on CIFAR-FS. 32
3.4 Comparison to prior work on FC100. 32
3.5 Comparison to prior work on miniImageNet and tieredImageNet. . . 33
3.6 Comparison with different combined meta-classifiers. 34
3.7 Quantitative comparison of computing W via different ways 35
3.8 Quantitative comparison of the effect of β_2 35

4.1 Differences between our approach and related zero-shot semantic segmentation methods based on CLIP. 41
4.2 Comparison with the state-of-the-art methods on PASCAL VOC 2012, COCO-Stuff 164K, and PASCAL Context datasets. "ST" represents applying self-training via generating pseudo labels on all unlabeled pixels, while "*"+"ST" denotes that pseudo labels are merely annotated on unseen pixels excluding the ignore part. 49
4.3 Efficiency comparison with different metrics. All models are evaluated on a single 1080Ti GPU. #Params represents the number of learnable parameters in the whole framework. 50
4.4 Comparison with the state-of-the-art open-vocabulary methods in terms of mIoU. 52
4.5 Effect of different formats of text queries $\hat{\mathbf{t}}$. 53
4.6 Quantitative results on PASCAL VOC 2012 dataset to demonstrate the effectiveness of our proposed three designs. 54
4.7 Quantitative results on COCO dataset to demonstrate the effectiveness of our proposed three designs. 55
4.8 Comparison of introducing advanced loss function. Note that "plain" represents merely Binary Cross Entropy (BCE), while "plus" means adding focal loss on BCE and dice loss 55
4.9 Generalization ability to other datasets. 56
4.10 Effect of the depth of deep prompt tuning on VOC. 56
4.11 Comparison of using single and multiple templates on COCO-Stuff 164K and PASCAL Context datasets. 56

5.1 Comparison of our proposed method with previous works on the PASCAL-5^i dataset under the GFSS setting. mIoU(N) and mIoU(B) denote the mean mIoU of the novel(N) and base(B) classes among four folders, while F0-F3 shows the detailed mIoU results of each novel fold. Note that the rows with gray background represent the results obtained after test-time finetuning on novel classes. 70
5.2 Qualitative results on COCO-20^i datasets under GFSS setting in terms of mIoU(%) on base and novel classes and hIoU(%). Note that the rows with gray background represent the results obtained after test-time finetuning on novel classes. 71
5.3 Using an alternative design for relationship descriptor (RD) on the PASCAL -5^i dataset. [CLS] indicates a design that directly calculates RD via the [CLS] and the prototype embedding, as directly borrowed from ZegCLIP [234]. Our method achieves better performance especially on novel classes, highlighting the contribution of our customized RD. 72
5.4 Using an alternative design for relationship descriptor (RD) on the COCO -20^i dataset. [CLS] indicates a design that directly calculates RD via the [CLS] and the prototype embedding, as directly borrowed from ZegCLIP [234]. Our method achieves better performance especially on novel classes, highlighting the contribution of our customized RD. 72
5.5 Comparison of our proposed method with the state-of-the-art FSS methods. Note that our method has not been trained on binary segmentation as well as test-time tuning on novel classes. 78
5.6 Effectiveness of our approach with an unsupervised pre-trained vision transformer model (i.e., ViT-B/16 pre-trained with DINO) on the PASCAL-5^i dataset. "Tuning" denotes the test-time tuning on the novel support set before inference under the generalized few-shot segmentation (GFSS) setting. 79

A.1 The details of unseen class names. 83

University of Adelaide

Abstract

Low-shot Learning based on Pre-trained Model

by ZIQIN ZHOU

In the field of computer vision, deep learning has demonstrated remarkable accomplishments. However, the dependence on extensive annotated datasets poses significant challenges, particularly when such data is scarce, costly, or ethically sensitive. This thesis addresses the crucial issue of low-shot learning by aiming to equip learning models with the capability to generalize effectively from limited examples. It explores various low-supervision methodologies, including self-supervised, unsupervised, semi-supervised, and active learning, with the goal of reducing the reliance on labeled data.

One potential solution to this challenge is leveraging pre-trained models trained on large-scale datasets and transferring them to downstream tasks and unknown classes. Nevertheless, directly applying pre-trained models in a frozen state often fails to meet the requirements of specific tasks. Fine-tuning these models on base classes with limited training samples and then transferring them to novel domains presents obstacles, as it may disrupt the general knowledge encoded in the model, leading to overfitting on the base classes and suboptimal performance on unseen categories.

The central focus of this research is to fully exploit the knowledge encoded in large pre-trained models and enhance their capacity to transfer to novel classes. This thesis introduces a systematic approach to balance generalization and task-specific adaptation, ensuring that fine-tuning preserves pre-trained knowledge while effectively adapting to new data. These models encapsulate extensive knowledge from diverse data sources, enabling swift adaptation to new tasks and classes with minimal additional training. The downstream tasks investigated in this doctoral work range from the low-shot problem for few-shot image classification to more intricate endeavors such as zero-shot and few-shot pixel-level semantic segmentation. The exploration encompasses various pre-trained model architectures, including convolutional and transformer-based models, spanning both pure vision and cross-modal vision-language paradigms.

By harnessing the representational capabilities of state-of-the-art pre-trained models, this research aims to push the boundaries of what is achievable with limited data. The novel methodologies proposed strive to maximize performance, maintain generalization across domains, and improve inference efficiency, paving the way for broader adoption in low-data scenarios.

Declaration of Authorship

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I acknowledge that copyright of published works contained within this thesis resides with the copyright holder(s) of those works.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

I acknowledge the support I have received for my research through the provision of an Australian Government Research Training Program Scholarship.

Ziqin Zhou

July 2024

Acknowledgements

First and foremost, I express my deepest gratitude to my supervisor, Professor Lingqiao Liu. His guidance has been invaluable throughout my academic journey, from the time I embarked on my Master's studies to the completion of my doctoral research. Professor Liu has not only provided extensive academic and theoretical instruction but has also shown remarkable patience and unwavering emotional support, encouraging me to persevere during challenging times. I am profoundly grateful for his meticulous mentorship and steadfast belief in my potential.

I also honor the cherished memory of my co-supervisor, Dr. Yifan Liu, whose untimely passing has left an irreplaceable void. I will always treasure the moments we shared—strolling along the seaside, skateboarding in parking lots, attending Pilates classes, and engaging in thoughtful discussions over meals. Dr. Liu was an extraordinary researcher whose brilliance, diligence, and the perfect harmony of intelligence and grace inspired everyone around her. Her insights and encouragement were integral to my research, and she remains in my heart as a model of perseverance, wisdom, and kindness. She will be profoundly missed but never forgotten.

My heartfelt thanks go out to my colleagues at AIML, whose camaraderie and collaborative spirit have made this journey both fulfilling and memorable. Their support during academic hurdles and personal challenges has been invaluable. Together, we have shared countless priceless moments—watching the stars from mountain tops, squid fishing by the sea, and celebrating the Spring Festival. These experiences have fostered a deep sense of belonging, making us more than just colleagues but a close-knit family. I will forever cherish these memories.

I am equally grateful to my friends in Adelaide, whose warmth and companionship have enriched my life outside the lab. They introduced me to the beauty of the outdoors and helped me find balance between work and leisure through activities like bouldering, free diving, tennis, and hiking. Their willingness to offer rides, lend a helping hand, and share their time has been a source of tremendous support, for which I am deeply thankful.

Lastly, I extend my deepest appreciation to my parents, my father Bing Zhou, my mother Jianjun Su, and all my family members. Their unconditional love, trust, and sacrifices have been the foundation of my strength and psychological well-being throughout my time abroad. I regret not being able to spend more time with them, yet their constant care and encouragement have been my greatest source of motivation. Without their unwavering support and the values they instilled in me, I would not be who I am today. For all they have done, I am eternally grateful.

Completing my PhD marks both the end of one journey and the beginning of a new adventure, where I will continue to pursue knowledge, embrace challenges, and strive to make a meaningful impact, always carrying a heart full of gratitude for everyone who has supported me along the way.

Chapter 1

Introduction

1.1 Problem Statement and Research Motivation

Recent years, deep learning methods have demonstrated remarkable performance across various computer vision tasks, especially when there are large amounts of annotated training data. However, this dependency on large datasets is not always practical or feasible, highlighting the importance of exploring data-efficient learning strategies. In scenarios where data are scarce, expensive to label, or require specialist knowledge, these approaches are indispensable. They enable models to generalize from limited examples, drawing on previously acquired knowledge, and making intelligent inferences about new, unseen data. This is particularly crucial for tasks where rapid model deployment is necessary or when dealing with rare events or categories that inherently lack extensive data. The pervasive challenges of label scarcity and the substantial costs associated with data annotation necessitate the creation of lean models that retain efficacy with limited data. Such models are not only economically and operationally advantageous but also essential for applications in fields where data collection is limited by practical or ethical constraints.

The demand for low-shot learning is driven by the need to develop models that can quickly adapt to new tasks or classes with only a minimal number of training samples. Addressing this challenge involves a variety of low-supervised settings, such as self-supervised [5, 19, 116], unsupervised [131, 140, 170], semi-supervised [169, 212], and active learning [49, 157], which aim to leverage the innate structure of data without heavy reliance on labels.

Within computer vision, downstream tasks are categorized into per-image evaluations, such as classification [28, 193] and retrieval [12, 208], and per-pixel prediction tasks, including semantic segmentation [95, 189], object detection [213, 223], tracking [50, 126] and etc. This thesis began with few-shot learning on image-level classification challenge, subsequently broadening to the zero-shot and few-shot pixel-level segmentation domains. By leveraging the impressive representational capabilities of state-of-the-art pre-trained models, this research aims to significantly advance the field of data-efficient learning and transfer to novel classes. This involves a multifaceted exploration of various settings and downstream applications, investigating how pre-trained models can be effectively utilized with limited annotated on base samples, a question that is currently under-explored.

Thanks to the pre-trained large model showing strong ability, leveraging the knowledge of pre-trained model and transfer the capability to novel classes or downstream tasks provide another solution. There is growing interest in the potential of

pre-trained models on large datasets to bolster transfer learning, thus enabling effective adaptation to downstream tasks and cross-domain datasets. Despite the clear importance of this research, the journey toward creating highly performative models under tight data constraints is replete with challenges and has yet to be fully explored. A common barrier to data-efficient learning is the tendency of models to overfit on seen categories, which compromises their ability to generalize to new classes in the open world.

This thesis focus on the exceptional abilities of large pre-trained models being harnessed to tackle the issue of limited data. These models have been trained on a vast array of data and have learned to identify a wide range of features. By using what these pre-trained models already know, we can apply their understanding to new classes or different tasks without needing a lot of additional data. This method of knowledge transfer allows us to make the most of the complex patterns the models have already learned, making it easier to adapt to new situations where data is scarce. This thesis investigates how we can make the best use of these pre-trained models when we only have a few or even none examples to learn from, as well as alleviate the overbias problem on base classes while transferring. The goal is to find ways to use the power of these models to improve our ability to learn from small amounts of data, opening up possibilities for using AI in areas where collecting large datasets is difficult or impossible.

1.2 Thesis Organization and Contributions

To explore low-shot learning problem based on pre-trained model, this thesis conduct experiments on few-shot and zero-shot setting on image-level classification task and pixel-level segmentation task. This research topics include different downstream tasks, different setting and different pre-trained models inspecting of architecture and objective loss function. The following thesis first introduce the related works in Chapter 2, then demonstrate three low-supervised learning tasks based on pre-trained models and conduct under these topics: few-shot classification in Chapter 3, zero-shot semantic segmentation in Chapter 4, and few-shot semantic segmentation in Chapter 5. The detailed research topic and related techniques are organized in Figure 1.1.

This figure outlines the hierarchical structure of the research problems addressed in this thesis, focusing on low-shot learning strategies based on pre-trained models. The diagram begins with the categorization of low-shot learning approaches, which include N-way K-shot tasks (e.g., image-level few-shot classification, pixel-level zero-shot, and few-shot semantic segmentation) and cross-domain tasks (e.g., cross-dataset classification and open vocabulary semantic segmentation). The second branch highlights the role of pre-trained models, further divided into pure vision pre-trained models (trained solely on vision tasks) and vision-language pre-trained models (jointly trained on vision and language tasks). Key methods for utilizing pre-trained models include classification pretraining, masked prediction pretraining, and self-supervised pretraining, as well as cross-modal strategies like image-text matching and contrastive learning. The figure also introduces tuning methods applied to pre-trained models, which are categorized into fine-tuning, partial-based tuning

(modifying specific layers or components while freezing others), and addition-based tuning (adding task-specific modules without altering the pre-trained parameters).

The contributions of each work can be summarized as follows:

- In the realm of **few-shot learning on classification**, the objective is to architect models that can discern novel classes from a limited compendium of samples. Conventional meta-learning strategies, while previously prevalent, are now recognized as suboptimal, particularly when compared with methods that directly train feature extractors fortified by data augmentation techniques. This dissertation proposes an integrated approach that combines the core principles of meta-learning with direct training methods. This fusion is realized through the innovative application of distribution prediction, which forges a closed-form linkage between the support set exemplars and the anticipated behavior of classifiers derived from the distribution. By employing Ridge Regression (RR) coupled with Gaussian distribution modeling, the proposed model mitigates the overfitting tendencies endemic to FSL and demonstrably elevates classification performance.

- In another vein, this research has pioneered the application of **zero-shot semantic segmentation** based on pre-trained vision-language model CLIP, developing an simple but efficient one-stage framework named ZegCLIP. It proposed three designs to signigicantly elliviate the overbias problem on seen classes: Use deep prompt tuning for efficient tuning on CLIP image encoder, apple non-mutually Exclusive Loss to generate the posterior probability of one class independently and design a relationship descriptor (RD) module to incorporate the image-level prior into text embedding before predicting semantic masks. It demonstrates superior performance, outperforming the state-of-the-art methods by a large margin under both “inductive” and “transductive” zero-shot settings and achieves a speedup of about 5 times faster compared with previous two-stage methods during inference.

- Furthermore, this thesis conduct experiments on **few-shot semantic segmentation** the advent of pre-trained vision transformers. This thesis explores to harnesses this emergent feature to tackle few-shot semantic segmentation, a task focused on classifying pixels in a test image with a few example data. As our main discovery, we find that the concept of “relationship descriptors", initially conceived for enhancing the CLIP model for zero-shot semantic segmentation, offers a potential solution. We adapt and refine this concept to craft a relationship descriptor construction tailored for few-shot semantic segmentation, extending its application across multiple layers to enhance performance. This work demonstrates a clear advantage of our method in diverse few-shot semantic segmentation scenarios, and a range of pre-trained vision transformer models. The findings clearly show that our method significantly outperforms current state-of-the-art techniques, highlighting the effectiveness of harnessing the emerging capabilities of vision transformers for few-shot semantic segmentation.

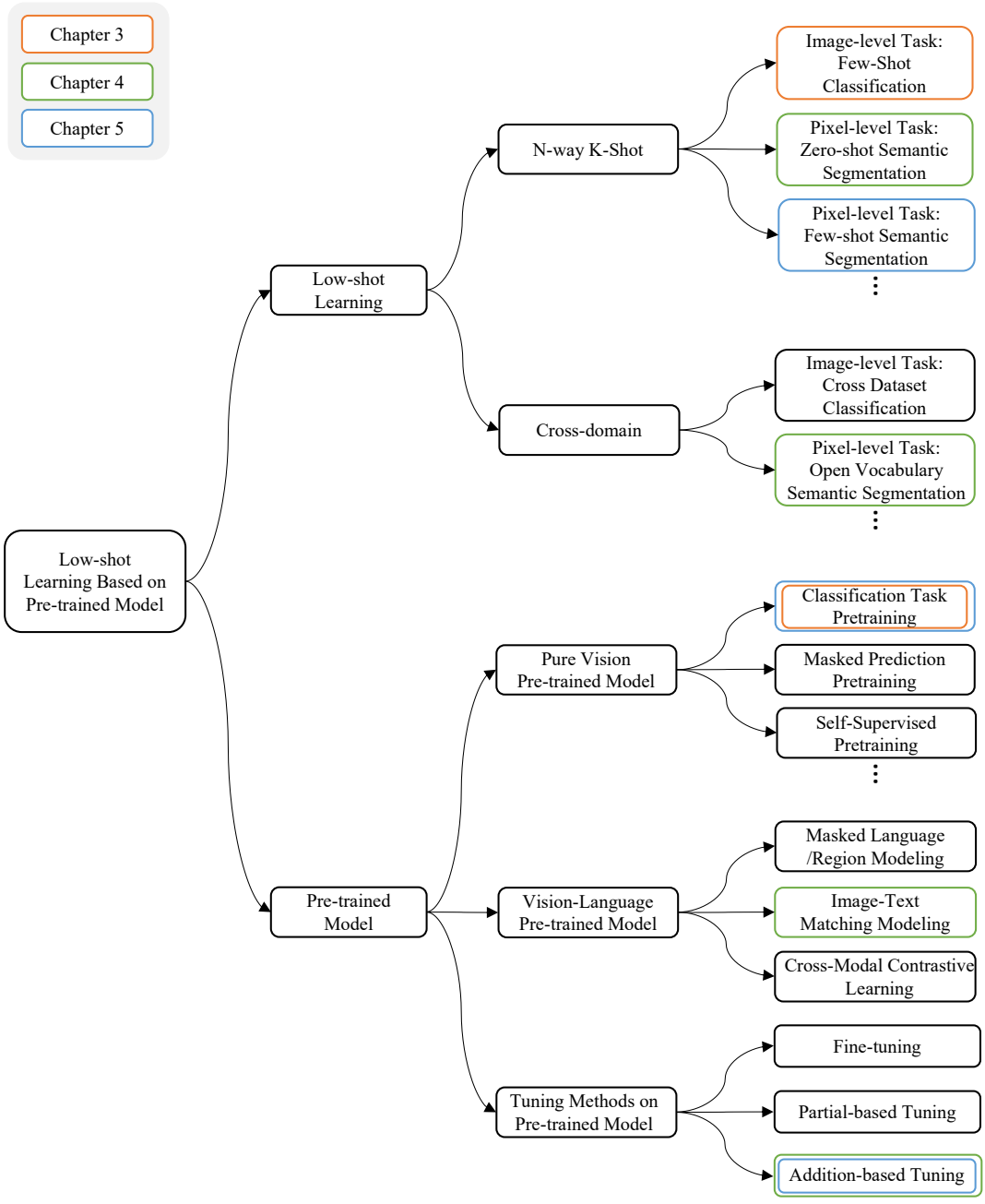


FIGURE 1.1. Structure of research problem categorization of this thesis. This structure provides a comprehensive overview of how low-shot learning is approached in this thesis, leveraging pre-trained models and task-specific adaptations. Each chapter of the thesis corresponds to different components of this diagram, as indicated by the color-coded labels (orange: Chapter 3, green: Chapter 4, blue: Chapter 5).

Chapter 2

Literature Review on Exploring Low-shot Learning via Pre-trained Model

2.1 Introduction

The field of deep learning [90] has seen remarkable progress in recent years, with applications ranging from computer vision [186] to natural language processing [29]. However, a common challenge persists: the need for large amounts of labeled data to train effective models [2]. Traditional machine learning approaches often struggle in scenarios where data is scarce, which is a common situation in many practical applications. In numerous real-world scenarios, the collection of extensive datasets is either impractical or prohibitively expensive. This limitation has spurred interest in low-shot learning [168, 175], a paradigm that aims to enable models to perform well with limited training examples. It is a challenging research topic aimed at enabling models to perform well even when only a small number of training examples are available.

Exploring low-shot learning is essential for several reasons:

- Addressing Data Scarcity: Helps develop models that generalize well from limited data.
- Enhancing Model Efficiency: Reduces the dependency on large datasets, lowering computational costs and accelerating model development.
- Improving Generalization Capabilities: Encourages the creation of adaptable and robust models for new tasks, crucial for applications like autonomous driving.
- Advancing Transfer Learning: Pushes the boundaries of utilizing pre-trained models in low-data scenarios.
- Facilitating Interdisciplinary Research: Applicable across various fields, fostering innovation and collaboration.
- Promoting Ethical and Inclusive AI: Reduces biases and ethical concerns in data collection, leading to more equitable AI systems.

Although important and practically, low-shot is difficult due to the limited training data, high variability within the data, susceptibility to overfitting, lack of representational diversity, complexity in adapting models to new tasks, class imbalance, challenges in reliable evaluation, domain-specific issues, and the sophisticated, resource-intensive algorithms required to effectively generalize from few examples. Fortunately, in recent developments, the use of pre-trained models [51] for low-shot learning has emerged as a promising strategy. Pre-trained models are initially trained on large datasets with fundamental objectives, inherently incorporating powerful and generalizable knowledge. These models have shown impressive performance improvements in various downstream applications. The key idea is to transfer the general knowledge acquired during the pre-training phase to new, specific tasks domain with limited training data and extend this capability from known classes to unknown categories, thereby facilitating low-shot learning.

This literature review chapter aims to provide a comprehensive overview of the current state of research on low-shot learning via pre-trained models. The following sections will delve into the fundamental concepts of low-shot learning, explore key methodologies in low-shot classification and semantic segmentation, and discuss the role of pre-trained models in enhancing low-shot learning performance. By synthesizing existing research, this review seeks to identify gaps in the literature and highlight promising directions for future research in this rapidly evolving field.

This thesis introduces the previous works related to my three research topics following the organization map as shown in Figure 1.1. More specifically, in this chapter, I first introduce the problem definition of low-shot learning in Subsection 2.2 and then describe the downstream tasks includes few-shot classification in Subsection 2.3, and few-supervised segmentation in Subsection 2.4. Then I introduce polular pre-trained models in Subsection 2.5 and demonstrate traditional methods related to low-shot learning on pre-trained model in Subsection 2.5.3.

2.2 Concepts and Preliminaries of Low-shot Learning

The motivation behind low-shot learning (LSL) stems from the limitations of conventional deep learning models, which typically require extensive labeled datasets to achieve robust performance. In many practical applications, such as medical diagnosis, rare species identification, and personalized recommendation systems, acquiring large-scale labeled data is either impractical or prohibitively expensive. This constraint necessitates the development of models capable of learning and generalizing from a limited number of examples, thereby broadening the applicability of machine learning to diverse and resource-constrained domains.

Low-shot learning draws inspiration from the human cognitive ability to learn new concepts with minimal examples. Humans can generalize from a few instances, adapting quickly to new tasks with limited data. Replicating this capability in machine learning models is not only a theoretical challenge but also carries significant practical implications, including reducing the costs associated with data collection and labeling, enhancing model robustness, and enabling rapid adaptation to new, unseen tasks.

Wang et al. [191] provided a comprehensive definition of LSL from the dimensions of experience, task, and performance in machine learning. According to this

definition, if the performance of a computer program improves on task T as measured by performance metric P through experience E, then the program is said to have learned some class of tasks T and performance metric P from experience E. Importantly, in the context of LSL, E is significantly limited, highlighting the necessity for models to operate effectively with minimal data.

For a deeper understanding of low-shot learning, it is crucial to introduce two fundamental concepts: N-way K-shot problem and cross-domain low-shot learning.

- **N-way K-shot** is a widely-used framework to describe particular low-shot learning scenarios, including few-shot, one-shot, and zero-shot learning. Few-shot learning aims to enable models to perform well on new tasks with only a few labeled examples. Formally, FSL can be defined as the task of learning a function $f : X \rightarrow Y$ that generalizes well from a limited number of training examples $\{x_i, y_i\}$ from the target domain. A typical FSL setup involves two datasets: a base dataset D_{base} with abundant labeled data and a target dataset D_{novel} with very few labeled examples, known as the "support set."

  In this framework, the support set is a small dataset used during the training phase to provide reference information. The query set, on the other hand, contains the tasks that the model is required to predict, with the stipulation that the classes in the query set are not present in the support set. In N-way K-shot learning, "N" refers to the number of classes, and "K" refers to the number of examples per class in the support set. For example, in a 5-way-5-shot learning scenario, the model is provided with 5 classes, and for each class, there are 5 examples in the support set. The support set would thus contain a total of 25 examples (5 classes × 5 examples per class). The goal of the model is to use this limited support set to correctly classify new, unseen examples in the query set. **Few-shot learning** generally refers to scenarios where K is small, such as 5 or 10. While **One-shot learning** is a special case of few-shot learning where K is exactly 1.

  Zero-shot learning was first introduced in [87] and it differs from the typical N-way K-shot framework, as it involves predicting classes that were not seen during training by using auxiliary information like semantic embeddings or descriptions. The model must predict classes that it has never encountered during training. This is achieved by leveraging high-dimensional semantic features instead of relying on low-dimensional raw data. Techniques such as embedding representations and autoencoders are utilized to create intermediate semantic spaces that capture attributes more comprehensively defining the categories. This semantic approach allows the model to discern previously unobserved categories, making zero-shot learning one of the closest methods to human intelligence in recognizing new classes. Essentially, one-shot learning and few-shot learning can be viewed as specific instances of zero-shot learning, where the model's ability to generalize from sparse data is put to the test.

  Overall, the N-way K-shot framework is fundamental in few-shot learning as it provides a structured way to test the model's ability to generalize from a small number of examples to new, unseen data.

- **Cross-domain** problem is heightened by combining the complexities of transfer learning and few-shot learning. Typically, in transfer learning, the target

domain classes have a sufficient number of samples. However, when there is a significant domain shift between the source and target domains, transferring knowledge becomes particularly difficult. In cross-domain few-shot learning, there is the additional challenge of very limited samples per class in the target domain, with no class overlap between source and target domains. This makes it exceptionally difficult for the model to generalize effectively using only the source domain data. Current research in this area focuses on adapting feature transformations and constructing auxiliary datasets to bridge domain gaps, recognizing cross-domain few-shot learning as one of the most challenging settings in the field of few-shot learning today.

Low-shot learning is applicable across various downstream tasks, particularly in the field of computer vision, spanning both high-level tasks like classification and low-level tasks such as semantic segmentation. High-level vision tasks aim to classify images into predefined categories with only a few examples per category, presenting a challenge in leveraging limited labeled data effectively. Conversely, low-level tasks like semantic segmentation involve delineating objects within an image, a process that becomes particularly demanding when only a small number of annotated examples are available. This thesis conducts a detailed review of the methods employed in low-shot learning for these downstream tasks in Subsections 2.3 and 2.4. The techniques reviewed in these subsections aim to maximize the utility of limited data, enabling models to generalize effectively from sparse examples to unseen, real-world scenarios.

Another promising approach related to LSL is low-resource learning, which offers a complementary solution by leveraging **learning from synthetic data**. This approach addresses the scarcity of labeled examples by generating artificial data for training. Synthetic data generation methods utilize techniques such as generative adversarial networks (GANs) [55], variational autoencoders (VAEs) [82], and diffusion models [68] to create diverse and realistic datasets. These methods have found successful applications in areas such as medical imaging [16], autonomous driving [40], and rare species classification [8]. By augmenting real-world data with synthetic examples, models can improve their generalization and robustness in low-shot learning scenarios. Additionally, synthetic data provides the opportunity to generate controlled variations and edge-case scenarios that are often underrepresented in real-world datasets, enhancing the model's ability to handle a broader range of tasks. Combining synthetic and real data offers a practical solution for overcoming data scarcity and enriching the training process. However, synthetic data also presents challenges, such as ensuring data realism and mitigating biases introduced during generation. In this thesis, the focus is on exploring the knowledge encapsulated in well pre-trained models to address the challenges of low-supervised learning.

2.3 Few-shot Classification

With the development of deep learning, neural networks have achieved significant performance through fully supervised training in a large amounts of labeled samples. However, there is still a huge gap between machine and human beings when training samples are limited. Few-shot learning (FSL) is a challenging topic that requires

target models to adapt to novel classes very quickly. It includes various tasks, such as NLP [9, 79, 216], classification [31, 117, 193], segmentation [38, 114, 189], detection [80, 213, 222], 3D point cloud [190][230] etc. In this work, we focus on the fundamental task, classification, to explore the possibility and effectiveness of handling few-shot learning problem via distribution learning. In this section, we restrain the review to the few-shot classification approaches and augmentation methods closely related to this work. More specifically, we summarize two paradigms of few-shot classification into episodic meta-training strategy described and representation learning with a simple meta-classifier strategy.

2.3.1 Episodic meta-training

Many researchers address the few-shot learning problem by generating amounts of episodic few-shot tasks from the base dataset. Episodic-based methods simulate meta-testing in the training phase that can encourage the target model to quickly adapt to novel tasks after updating each iteration on base samples.

The objective of episodic meta-training can be formulated as follows:

$$\theta^* = \arg\min_{\theta} \mathbb{E}\mathcal{T} \sim p(\mathcal{T}) \left[\mathcal{L}\mathcal{T}(f_\theta)\right], \tag{2.1}$$

where θ represents the model parameters to be optimized, $\mathcal{T}$ denotes a task sampled from the task distribution $p(\mathcal{T})$, and $\mathcal{L}_\mathcal{T}$ is the loss function associated with task $\mathcal{T}$. The goal is to find the optimal model parameters θ^* that minimize the expected loss over the distribution of tasks. By training the model on a variety of tasks, the model learns to generalize and perform well on new, unseen tasks.

Optimization-based methods design more suitable optimizers for few-shot learning. Ravi [154] introduced a meta-learner to optimize training hyper-parameters while the base learner updates network weights for training tasks. MAML [46] attempted to learn good initialized weights which can be adapted to novel tasks after several gradient updates. Similarly, Reptile [139] proposed a simpler approximation of MAML and dramatically saved computation and memory cost. MetaOptNet [91] followed MAML and formulated an end-to-end architecture by introducing differentiable Quadratic Programming (QP) and Support Vector Machine (SVM).

Metric-based approaches [70, 143, 219] tend to embed samples of different categories into discriminative feature space. Koch [84] developed a siamese network to extract distinguished features. Matching Network [185] applied Long Short-Term Memory (LSTM) with attention kernel and augmented with memory to match conditions between support and query samples rapidly. Prototypical Network [168] creatively represented each class by prototype and retrieved classification results by Bregman divergence. Sung [175] formulated Relation Network to concatenate support features and each query feature to compare the similarity between each category.

Model-based methods aim to quickly adapt to limited data by designing better architecture. Inspired by Neural Turing Machines (NTMs) [56], Santoro [162] investigated memory-augmented neural networks to remember and predict the data that only appear once. Munkhdalai [134] proposed a novel Meta-learning network to improve the ability of rapid generalization via fast parametrization.

2.3.2 Representation learning

Recently, many works propose that learning a good feature representation in training stage and generating augmented support set in meta-testing stage can improve performance significantly. Beyond that, feature extraction and selection are important for model convergence and improve the robustness while transferring [177][178].

The objective of representation learning can be formulated as:

$$\theta^* = \arg\min_{\theta} \mathbb{E}(x, y) \sim p\text{train}\left[\mathcal{L}(f_\theta(x), y)\right], \tag{2.2}$$

where θ represents the model parameters, (x, y) denotes the input-output pair sampled from the training distribution p_{train}, and $\mathcal{L}$ is the loss function that measures the discrepancy between the predicted and true labels. The aim is to learn model parameters θ^* that minimize the expected loss over the training data distribution.

Furthermore, the learned representation can be utilized to generate an augmented support set in the meta-testing stage, thereby improving the model's ability to generalize to new tasks. This process can be mathematically expressed as:

$$\tilde{\mathcal{S}} = g_\phi(\mathcal{S}), \tag{2.3}$$

where $\mathcal{S}$ is the original support set, g_ϕ is the augmentation function parameterized by ϕ, and $\tilde{\mathcal{S}}$ is the augmented support set.

By combining these objectives, representation learning aims to achieve robust and transferable feature representations that facilitate better performance on few-shot learning tasks. Related works are summarised as:

Representation learning based methods replace traditional episodic meta-training strategy by fully supervised training on base dataset to pre-train a good feature extract which can be transfer to unseen classes directly. [20] shown that a deeper backbone leads to a slight performance gap between previous representative methods. Indeed, the baseline model with fine-tuning can achieve significant performance. Tian [179] also found that learning a good embedding module is crucial in few-shot learning problem. In addition, then applied self-distillation operation to further improve performance.

Hallucination based methods become more and more popular recently. As we know that, the basic problem of few-shot learning is limited training samples, hallucination based methods are similar to data augmentation but propose to learn how to synthesize support features [63]. Yang [210] utilized the similarity of mean and variance between base classes and novel classes to calibrate the distributions of unseen categories. After that, sufficient support features can be sampled randomly to enrich the training set. Lazarou [89] firstly generated support feature tensors instead of vectors with prototypical classification by a tensor feature hallucinator.

Self-supervised learning (SSL) methods have achieved reasonable success recently. Predicting image rotation [54] is a simple but effective pretext task to encourage target model capture semantic information and has been widely used in few-shot learning [127]. Rizve [158] proposed a novel architecture to predict transformation style on training tasks to learn equivariance or invariance jointly. Contrastive learning is widely used in few-shot learning, Gao [51] proposed a novel CPLAE by combining augmented rotated embeddings and contrastive prototype learning. Liu

[107] proposed a novel info-patch as noise and trained the embedding with InfoNCE loss function. In this way, they attempted to exploit the local relationship between samples and improve the representative ability of the target model.

2.4 Low-shot Semantic Segmentation

2.4.1 Semantic Segmentation

Semantic segmentation is a fundamental dense prediction task in computer vision that involves annotating each pixel of an input image with a class label. This task is crucial for various applications, such as autonomous driving, medical imaging, and scene understanding. Previous work in semantic segmentation generally follows two main principles:

- **Per-pixel Classification** Per-pixel classification treats semantic segmentation as a pixel-wise classification task. Models predict the probability distribution over all possible categories for every pixel in an image, where the distribution lies within a K-dimensional probability simplex. The per-pixel classification can be formally expressed as:

$$y = \{p_i | p_i \in \Delta^K\}_{i=1}^{H \times W}, \tag{2.4}$$

  where $H \times W$ is the image size, and p_i denotes the probability distribution for pixel i across K classes. The model is trained using the ground truth category labels $y^{gt} = \{y_i^{gt}\}_{i=1}^{H \times W}$ and applies a per-pixel cross-entropy loss:

$$\mathcal{L}_{\text{pixel-cls}}(y, y^{gt}) = \sum_{i=1}^{H \times W} -\log p_i(y_i^{gt}), \tag{2.5}$$

- **Mask Classification Formulation** Mask classification splits the segmentation task into two main parts: 1) partitioning the image into N regions represented with binary masks $\{m_i | m_i \in [0,1]^{H \times W}\}_{i=1}^{N}$, and 2) associating each region as a whole with a distribution over K categories. The approach can be formulated as finding the probability-mask pairs $z = \{(p_i, m_i)\}_{i=1}^{N}$. Unlike per-pixel class probability prediction, this method includes an auxiliary "no object" label ($\emptyset$).

  To train a mask classification model, a matching σ is required between the set of predicted masks and the set of ground truth segments $z^{gt} = \{(c_i^{gt}, m_i^{gt}) | c_i^{gt} \in \{1, \ldots, K\}, m_i^{gt} \in \{0,1\}^{H \times W}\}_{i=1}^{N^{gt}}$, where c_i^{gt} is the ground truth class of the ith segment. The matching involves a bipartite assignment that allows for handling different numbers of predicted and ground truth segments.

  The loss function for mask classification, $\mathcal{L}_{\text{mask-cls}}$, is a combination of a cross-entropy classification loss and a binary mask loss, formulated as:

$$\mathcal{L}_{\text{mask-cls}}(\mathbf{z}, \mathbf{z}^{\text{gt}}) = \sum_{j=1}^{N} \left[-\log p_\sigma(j)(c_j^{\text{gt}}) + \mathbb{1}_{c_j^{\text{gt}} \neq \emptyset} \mathcal{L}_{\text{mask}}(m_\sigma(j), m_j^{\text{gt}}) \right], \tag{2.6}$$

where $\mathcal{L}_{\text{mask}}$ is a binary mask loss, and $\mathbb{1}$ is an indicator function.

Both methods have shown remarkable advancements with a pre-defined closed set of semantic classes. However, in few-shot learning scenarios, the challenge lies in adapting these techniques to handle new, previously unseen classes with limited annotated examples. This requires innovative strategies to transfer learned knowledge from base classes to novel classes efficiently, ensuring robust segmentation performance even in the presence of sparse data.

2.4.2 Zero-shot Semantic segmentation

Zero-shot Semantic Segmentation remains an important but challenging task due to the inevitable imbalance problem in seen classes. The target model is required to segment unseen classes after training on seen classes with annotated labels. Previous works like SPNet[197], ZS3[10], CaGNet[59], SIGN[27], Joint[3] and STRICT[146] follow the strategy that improves the generalization ability of semantic mapping from seen classes to unseen classes.

As the popular pre-trained vision-language model CLIP shows the powerful ability of zero-shot classification, it has been also applied in zero-shot semantic segmentation recently. Zegformer[36] and zsseg[207] develop an extensive proposal generator and use CLIP to classify each region and then ensemble the predicting results. Although it remains CLIP's zero-shot ability at the image level, the computational cost increases inevitably due to classifying each proposal. MaskCLIP+[232] creatively applies CLIP to generate pseudo annotations on novel classes ("transductive") for self-training. It has achieved competitive performance but will be invalid in "inductive" setting where the names of unseen classes in inference are unavailable while training.

Self-training in Zero-shot Segmentation introduces another setting of zero-shot semantic segmentation called "transductive". Unlike the traditional "inductive" setting where the novel class names and annotations are both unavailable in the training stage, [146] proposed that self-training via pseudo labels on unlabeled pixels benefits solving the imbalance problem. In such a "transductive" situation, both the ground truth of the seen classes, as well as the pseudo labels of unseen classes, will be used to supervise the target model for self-training [59, 207, 232] that can achieve better performance.

2.4.3 Open Vocabulary Semantic segmentation

Open Vocabulary Semantic segmentation is very related to the zero-shot setting but further increases the requirements for improving the generalization ability of pre-trained models. LSeg+ [93] developed a language-driven semantic segmentation model based on CLIP and per-pixel segmentation results are generated by matching the text and image features. OpenSeg [52] predicted the semantic masks by generating class-agnostic regions while learning the visual-semantic alignment relationships to implement region-word grounding by introducing class-agnostic segmentation loss. OvSeg [102] also proposed a two-stage method and introduced mask prompt tuning for adapting CLIP for identifying masked regions and retaining its

generalization ability. Similarly, two-stage method MaskCLIP [37] introduced learnable mask class tokens and designed Relative Mask Attention (RMA) modules beside the CLIP image encoder to classify the class-agnostic proposals. SAN [205] modified the two-stage zero-shot semantic segmentation method SimSeg [206] into a one-stage way via applying side-adapter networks and fusing the patch embeddings extracted from CLIP to avoid extensive encoding on cropped proposals.

2.4.4 Few-shot Semantic segmentation

Binary Few-shot Semantic Segmentation (FSS) was initially proposed in the work [189, 221] which designed a two-branch architecture to provide a classifier from support to query. Subsequent class-wise prototype methods [180, 189, 221, 229], influenced by [168], notably improved performance. To further improve performance, democratized graph attention mechanisms have been introduced in [188, 220], while [211] applied a multi-scale correlation module, and ASGNet [95] proposed a superpixel-guided clustering approach. Recent work by Lang et al. [88] leveraged the base learner to refine semantic masks of novel classes, while MSANet [74] developed a similarity and attention module to generate visual correspondence information between support and query. The latest developments by [88] and [74] focus on refining semantic masks and enhancing visual correspondence in few-shot segmentation. Despite these advances, previous methods mainly focus on recognizing the novel objects from the corresponding support set after manual selection but fail to handle the more practical situation where both base and novel classes need to be segmented simultaneously.

Generalized Few-shot Semantic Segmentation (GFSS) introduced in [181], segments images of both base and novel classes without prior inference knowledge to overcome the limitation of the FSS setting. Since previous binary FSS methods failed on few-shot multiple dense predictions, they proposed context-aware prototype learning (CAPL) to enrich contextual classifiers from novel support images dynamically but performed poorly on novel classes. FineTune [136] claimed that a test-time fine-tuning approach on classifier is the key to GFSS and then introduced triplet loss to improve the performance on novel classes extremely. CCA [113] proposed a new-class contrastive loss to further consider the relationship between base and novel categories while updating the novel prototypes. DIaM [60] inspired by InfoMax principle that maximized the mutual information to optimize the novel classes classifier updating processing and introduced knowledge distillation to retain the ability on base classes. POP [112] introduced orthogonal constrain to improve the generalized ability and proposed enrichment strategy delicately to alleviate the unbalanced problem of base and novel classes during test-time fine-tuning.

2.5 Popular Pre-trained Models

This section explores several of the most popular and widly used pre-trained models in deep learning as well as few-shot learning, which have significantly advanced the field by providing robust foundations for various downstream tasks. We categorize these models based on their underlying architectures—namely, Convolutional Neural Networks (CNNs) and Transformer-based models—and further classify them

according to the specific pre-training tasks they were designed for. In addition, this section discuss different tuning methods on pre-trained model.

2.5.1 CNN-based Pre-trained Models

Convolutional Neural Networks (CNNs) [101] have been a cornerstone in the field of computer vision, excelling in tasks such as image classification, object detection, and segmentation. CNNs are designed with convolutional layers that are adept at capturing spatial hierarchies in images. The fundamental operation in a CNN is the convolution, which produces feature maps as follows:

$$\text{Feature Map} = f(W * X + b), \tag{2.7}$$

where W represents the filter or kernel, X is the input image or feature map from the previous layer, b is the bias term, f is a non-linear activation function (e.g., ReLU), $*$ denotes the convolution operation. The layers in a CNN typically consist of convolutional layers, pooling layers, and fully connected layers, forming a deep network that captures both local and global patterns in the data. Below are some of the most influential CNN-based pre-trained models:

AlexNet [86]: Introduced by Krizhevsky et al., AlexNet was groundbreaking for its performance in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012. It demonstrated the efficacy of deep CNNs and the importance of GPU acceleration.

VGGNet [167]: Developed by the Visual Geometry Group at Oxford, VGGNet is known for its simplicity and depth, using 16-19 layers of convolutional networks. It achieved top results in ILSVRC 2014 and is widely used for transfer learning.

ResNet [66]: The Residual Network (ResNet) introduced by He et al. introduced the concept of residual learning, allowing networks to be trained with much greater depth (up to 152 layers). ResNet has become a go-to model for various vision tasks due to its robustness and ease of optimization.

GoogLeNet [176]: Developed by Google, the Inception architecture employs a module that approximates sparse networks with dense components, allowing for more efficient computation. It won the ILSVRC 2014 competition.

DenseNet [72]: The Dense Convolutional Network (DenseNet) connects each layer to every other layer in a feed-forward fashion, which mitigates the vanishing gradient problem and encourages feature reuse, leading to more compact and efficient models.

2.5.2 Transformer-based Pre-trained Models

Transformer [184] is originally designed for natural language processing, have been adapted for vision tasks. They rely on self-attention mechanisms, allowing them to model long-range dependencies in the data. The self-attention mechanism in a Transformer is described by:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{2.8}$$

where Q (Query), K (Key), and V (Value) are matrices derived from the input, d_k is the dimension of the key vectors, the softmax function normalizes the output. In

vision transformer architecture, an image is divided into patches, which are linearly embedded and combined with positional encodings before being fed into the Transformer model.

Pre-trained Vision Transformer has been highly valued in the research community due to its scalability [39] which is being fully supervised on ImageNet-21K [32] and exceptional feature representation capabilities [137]. Various downstream tasks have benefited from building upon pre-trained vision transformers for better performances, such as object detection [100, 171] and image segmentation [24, 99, 234]. For the few-shot semantic segmentation task, multiple works [123, 163, 226] have achieved great performance gain through using a pre-trained vision transformer backbone. These models can be categorized based on the self-supervised tasks performed during training. The MAE [64] and BEiT family [4, 148] reconstruct the missing region from masked images, while the MOCO family [22, 65] and SimCLR [18] apply different augmentation operations to conduct unsupervised contrastive learning. DINO [11] simplifies self-supervised learning by introducing self-distillation without any label. Moreover, it has been discovered that pre-trained vision transformers have demonstrated the property of grouping regions with the same semantic together. This can be observed by examining the similarity between embeddings or from the attention map. Such observations indicate that pre-trained vision transformers can be used for semantic segmentation tasks by employing a decoder network on top of the transformer output. In this work, we further investigate the semantic grouping capability of the pre-trained vision transformer and introduce it to the few-shot segmentation task for better novel class generalization.

Pre-trained Vision Language Models [76, 96, 152, 172] which connects the image representation with text embedding have achieved significant performance on various downstream tasks, like image retrieval [119], dense prediction [153], visual referring expression [192], visual question answering [78] and so on. It can be categorized by cross-modal pre-training tasks. Cross-modal Masked Language Modeling (MLM) inspired by BERT [34] model. ViLT [81] proposed the Whole Word Masking strategy and InterBERT [103] further makes the predictions more difficult by masking consecutive text tokens. Cross-modal Masked Region Prediction (MRP) is similar to MLM but leverages RoI features to predict other image tokens. SOHO [73] proposed a method to predicted grid-based tokens based on surrouding features. Image-Text Matching (ITM) aligns language and vision tokens with amounts of text-image pairs [23, 94]. Cross-modal Contrastive Learning (CMCL) presents one of the most popular pre-training task in vision-language models. ALIGN [76] calculate the large-scale image-text similarity for zero-shot classification.

Similarly, the Contrastive Language–Image Pre-training (CLIP) [152] model is a state-of-the-art framework designed to bridge vision and language tasks. Developed by OpenAI, CLIP utilizes large-scale datasets consisting of image and text pairs to learn a unified embedding space. This shared space allows for the alignment of visual and textual information, enabling robust zero-shot learning capabilities across a wide range of visual classification tasks. CLIP is trained using a contrastive learning approach, which is central to its ability to link images with relevant textual descriptions. The core idea is to learn image and text embeddings that are close to each other if they correspond to the same content, and far apart if they do not. The model achieves this through the following contrastive loss function:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\left[\log\frac{\exp(\text{sim}(z_i,t_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(z_i,t_j)/\tau)} + \log\frac{\exp(\text{sim}(z_i,t_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(z_j,t_i)/\tau)}\right], \tag{2.9}$$

where z_i and t_i are the embeddings for the i-th image and its corresponding text, respectively. $sim(z_i, t_i)$ represents the cosine similarity between the image and text embeddings, calculated as:

$$\text{sim}(z_i,t_i) = \frac{z_i \cdot t_i}{\|z_i\|\|t_i\|}, \tag{2.10}$$

where τ is a temperature parameter that controls the concentration of the softmax distribution. This loss function is computed over a batch of N image-text pairs. It comprises two symmetrical terms: the first term aligns each image with its corresponding text, and the second term aligns each text with its corresponding image. By maximizing the cosine similarity for matching pairs and minimizing it for non-matching pairs, the model learns a joint embedding space that facilitates effective cross-modal retrieval and zero-shot transfer learning.

2.5.3 Tuning Methods on Pre-trained Models

.

Adapting pre-trained models to downstream tasks, especially in few-shot learning scenarios, necessitates meticulous tuning to fully exploit the rich representations these foundational models offer and alleviate the overfitting problem. This tuning process is vital for improving the model's performance on specific tasks with limited data. Various methods are employed to update the parameters of pre-trained models when adapting them to new tasks, each presenting distinct trade-offs in terms of computational efficiency, flexibility, and performance.

Intuitively, fine-tuning [67, 187] the whole mooel is able to achieve significant performance in fully supervised learning tasks but is challenged by transfer learning. Recently, parameters-efficient tuning methods can be divided into partial-based tuning and addition-based tuning. Partial-based approaches only tune a part of pre-trained parameters. For example, Linear Probe [85] adjust the linear layers, AdapterBias [48] only refine the bias weights, and LoRA [71] modify the output of attention layer. Addition-based approaches include introduce extra learnable parameters, like using adapter modules [17, 145], side-by networks [24, 224] and deep prompts [77]. Among them, prompting tuning that freezes the pre-trained model while introducing a set of learnable prompts [9] provides an alternative way. It has achieved satisfied performance in both NLP tasks [97, 115, 151] and CV tasks [201, 214, 233]. Visual prompt tuning [77] proposed an effective solution that inserts trainable parameters in each layer of the transformer and yielded notable gains, especially in zero-shot learning tasks [164, 235].

Chapter 3

Exploring Few-shot Classification via Distribution Prediction

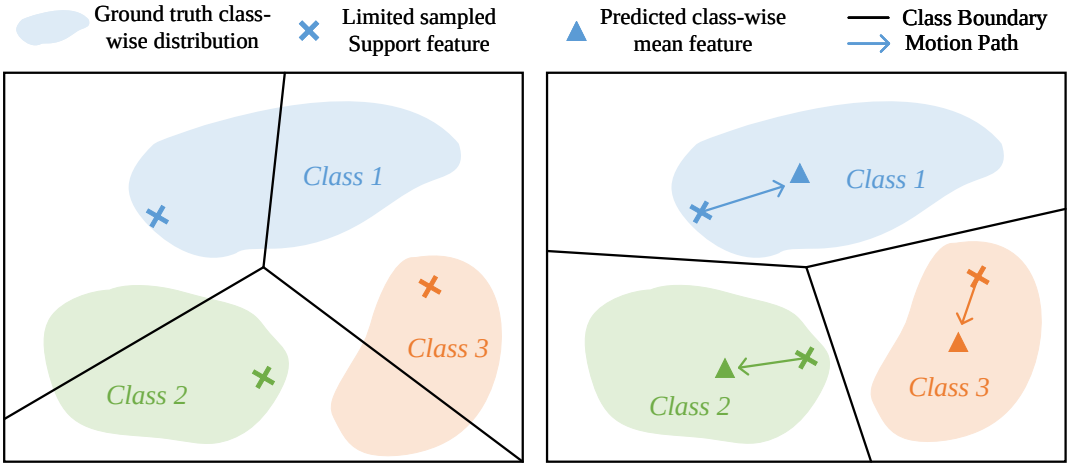
3.1 Introduction

Few-shot learning (FSL) remains an important but challenging task in machine learning. In FSL, the target model is required to learn how to classify based on a small number of support set samples. In this situation, the model tends to over-fit the sampled support images and underperform on the query set. Meta-learning used to be the most popular scheme to handle this problem by learning to learn from a large number of episodic tasks in the meta-training phase. Meanwhile, several recent works show that directly training a feature extractor from the base class samples with data augmentation can achieve competitive performance. In this work, we propose a method that unifies both ideas. The key to our approach is distribution prediction, which directly predicts the distribution parameters from the support set samples rather than relying on domain-specific input augmentation or challenging sample feature generation. By adopting the Ridge Regression (RR) as the classifier and Gaussian distribution to model the feature distribution, we establish a closed-form connection from the support set sample to the expectation of classifiers learned from samples drawn from the predicted distribution. We further leverage this formulation by developing loss functions from the perspective of distribution prediction and meta-classifier training. Training jointly with those two objectives, our final model achieves competitive performance over the existing solutions.

3.2 Background

Deep neural networks achieve significant performance on various tasks especially training on a large number of annotated samples. However, unlike human learners, networks usually fail to cope with the problem of few-shot learning (FSL). It remains challenging for the train meta-classifier for unseen classes that may only have only 1 or 5 samples. The major difficulty for FSL is that a few support samples can not represent the true feature distributions. It will lead to a sampling bias problem and could induce incorrect decision boundaries, as shown in Figure 3.1-(a). Thus training with limited labeled data often results in poor performance on the test set.

To fill this gap, many previous FSL methods recourse to meta-learning. In meta-learning, a large number of episodic tasks are sampled from a meta-training dataset with many base classes, simulating the few-shot learning scenario at the test stage.



(a) Classifier trained with a few randomly sampled support features

(b) Classifier trained with support features after class-wise distribution prediction

FIGURE 3.1. A toy illustration of our proposed method which can solve sampling over-bias problem via class-wise distribution prediction and refinement.

A model is trained to excel on those episodic tasks and is expected to generalize to unseen tasks. More recently, several works [138, 180, 210] argue that a good feature extractor with augmented support samples is very effective for few-shot classification and can achieve excellent performances even without meta-training. Tian [180] introduced a simple yet competitive baseline, illustrated in Figure 3.2-(a). This approach utilizes standard supervised learning on the base dataset to train a backbone, which is then reused as a fixed feature extractor during the meta-testing phase. In this phase, a Logistic Regression (LR) classifier is employed to learn from the support set samples. To enhance the effectiveness of the support set, data augmentation techniques can be applied, providing additional diversity and improving overall performance. However, traditional data augmentation operations may not be able to create samples distinguishing from the original samples and provide limited extra information. Another way is learning to synthesize training samples through an additional learning process. Inspired by Generative Adversarial Networks (GANs) [55], previous methods has developed generators to create image-level augmentations [227] or feature-level [89, 125, 196, 225] augmentations. Such methods can be summarized and illustrated as Figure 3.2-(b).

Nevertheless, synthesizing samples from a generative model is a nontrivial task. Not only the quality of the generated samples but also the diversity and distribution of the generated samples have a significant impact on the few-shot learning performance. An alternative solution is to directly model the data distribution rather than generating individual samples. Recently, Yang [210] observed that similar classes tend to exhibit similar feature distributions. They proposed modeling each class's feature distribution as a Gaussian, characterized by a mean vector and a covariance matrix. Building on this assumption, they calibrated the distributions of novel classes by leveraging the distributions of similar base classes. This distribution calibration

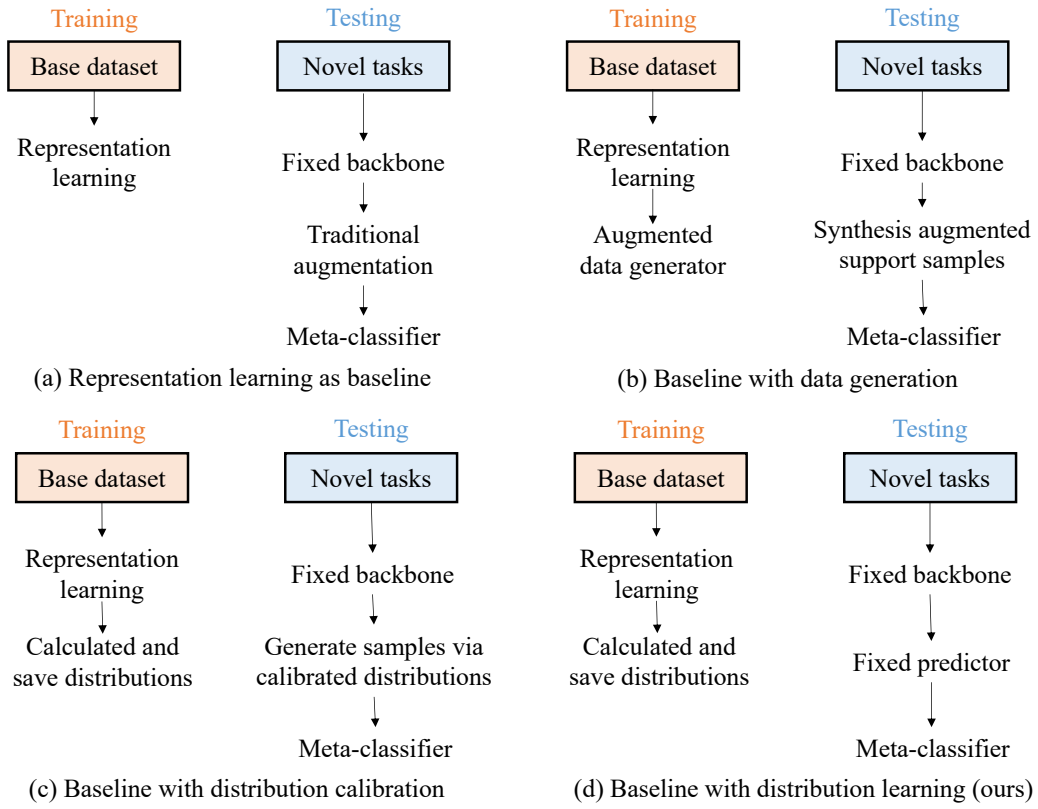


FIGURE 3.2. Briefly comparison with very related works. (a) represents few-shot classification baseline with traditional augmentation operations, it includes representation learning in fully-supervised training phase on base dataset and meta-classifier developing in meta-testing phase on novel dataset; (b) demonstrates introducing generative model to process data augmentation in (a); (c) describes applying distribution calibration algorithm guided by seen classes to sample sufficient support images in novel tasks. Our proposed methods (d) is not modelling data augmentation but class-wise distribution to improve the adaptation ability in novel meta-classifier.

process is shown in Figure 3.2-(c). Motivated by their work, we step further to propose distribution learning: we adopt an additional training process to learn the mapping function from a sample to its class-wise feature distribution, as demonstrated in Figure 3.1-(b).

Two issues need to be solved to realize this idea: (1) how to learn the distribution estimator and (2) how to use the estimated distribution for few-shot learning. We solve those two issues by taking the Gaussian distribution assumption of the normalized features and modifying a ridge regressor to seamlessly incorporate the mean and covariance estimated from the distribution estimator, making the mapping from the support set samples to the final classifier a differentiable process. This design allows us learn the distribution estimator from two objectives: (1) fitting base-class samples to their associated class-wise statistics (2) ensuring the resulted classifier performs well on the unseen data, which could be cast as a standard meta-training process.

More specifically, we made the following key contributions in this work:

- Instead of modeling data generation, we explored the possibility of few-shot classification via learning a class-wise distribution estimator.
- We experiment with two formulations for the class-wise feature distribution, i.e., complete and simplified versions, and introduce novel objective functions to train the distribution estimator.
- We extend the Ridge Regression classifier to seamlessly incorporate distribution estimation results, resulting in a closed-form mapping function from samples to the meta-testing classifier.
- We conduct extensive experiments on four benchmark datasets and achieve competitive performance.

3.3 Method

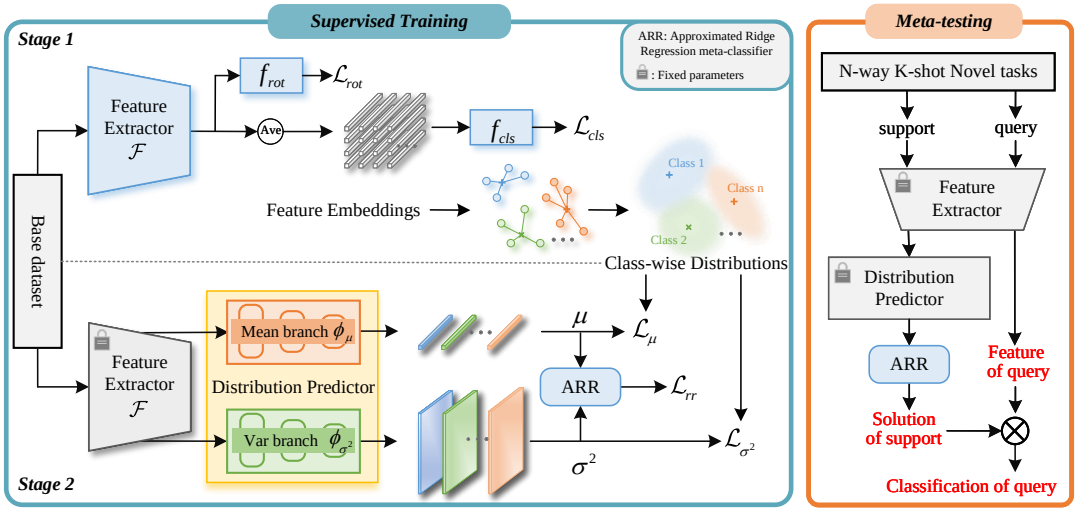
In this section, we first give a brief introduction to the problem formulation of few-shot classification in Subsection 3.3.1; then, we describe our method in multiple stages: "training stage 1" aims to learn the feature representation, which is described in Subsection 3.3.2; our extension to Ridge Regression classifier is introduced in Subsection 3.3.3, which will be subsequently used in our method; next, we describe the details of how to train the distribution estimator as "training stage 2" in Subsection 3.3.4; finally, we present meta-testing process on novel classes in Subsection 3.3.5. The overall of our proposed method is illustrated in Figure 3.3.

3.3.1 Problem Formulation

Our proposed method follows the typical few-shot classification setting shown in Figure 3.4. The dataset $D := \{(x_j, \hat{y}_j)\}_j^I$ consists of data-label pairs, where x_j is the jth input image and $\hat{y}_j$ denotes its class label. A set of base class samples D_b forms a meta-training set that could be used to either train the feature extractor or the meta-classifier. Once trained, the model can be applied to the meta-testing stage. For meta-testing, a classifier is generated from the *support set* $\mathcal{S}$ with novel classes. The generated classifier is evaluated on the *query set*. More specifically, we randomly select N classes and K images per class as *support set* (N-way K-shot problem) $\mathcal{S} := \{(x_j, \hat{y}_j)\}_j^{N*K}$ while Q images per class as *query set* $\mathcal{Q} := \{x_j\}_j^{N*Q}$.

3.3.2 Training stage 1: Representation learning

Recent works [210] demonstrate that a good embedding is a competitive baseline for few-shot classification. Motivated by this observation, we pre-train the feature extractor $\mathcal{F}$ on base dataset D_b with the standard supervised training strategy instead of using the episodic training strategy as in meta-learning approaches. Besides the classification loss, we add the rotation prediction [33, 45] as an auxiliary task, which has shown to be effective in improving the quality of the feature representation. More



(a) Two-stage supervised training strategy

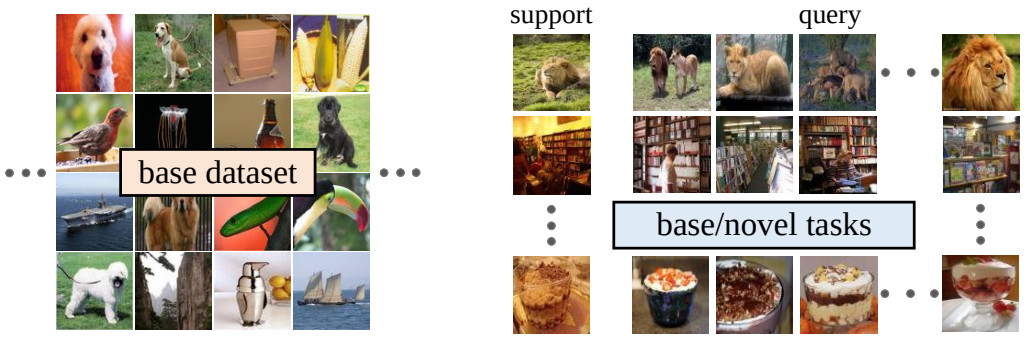
(b) Meta-testing strategy

FIGURE 3.3. Overall of our proposed method. The training strategy consists of two stages: (1) Stage 1 - Representation Learning: We follow related works to pre-train a feature extractor as backbone with fully supervised learning with classification and rotation prediction tasks on base dataset. (2) Stage 2 - Distribution Learning: We explore the ability to predict feature distribution from limited samples. We fix the parameters in feature extractor and develop an Approximated Ridge Regression (ARR) as meta-classifier with different objective functions to train distribution predictor. In the meta-testing phase, we compute the solution of ARR based on the features of support set and apply the solution as weights to predict classification results on query samples. Note that, both feature extractor and distribution predictor are frozen while testing.

specifically, each sample in D_b will be rotated with 0, 90, 180, and 270 degrees and annotated with rotation label r. Therefore, the format of training dataset can be represented as $D_b := \{(x_j, \hat{y}_j, \hat{r}_j)\}_j^I$, where $\hat{y}_j \in \{C_b\}, \hat{r}_j \in \{0, 1, 2, 3\}$. We develop two fully connected layers as classifier denoted as $\{f_{cls}, f_{rot}\}$ to predict the classification results $\{y\}_j^I$ and rotation degree $\{r\}_j^I$ respectively. To train our proposed model, we apply cross entropy $\mathcal{L}_{CE}$ to measure the discrepancy between prediction and groundtruth (class label or rotation angle). The overall loss function is:

$$\mathcal{L}_1 = \mathcal{L}_{CE}(P(y|x), \hat{y}) + \alpha \cdot \mathcal{L}_{CE}(P(r|x), \hat{r}), \tag{3.1}$$

where $P(y|x)$ and $P(r|x)$ are posterior probability estimation for class label and rotation label. α is the hyper-parameter to balance those two terms. The parameters of feature extractor $\mathcal{F}$ will be fixed once the first training stage is done.



(a) representation learning (b) meta-trianing/testing

FIGURE 3.4. Illustration of the few-shot classification problem formulation. We follow typical setting which includes (a) pretraining feature extractor via fully supervised representation learning on base dataset and (b) meta-training on base tasks and then meta-testing on novel tasks.

3.3.3 Extension of Ridge Regression to Incorporate Sample Distribution Estimate

Ridge Regression (RR) is a classic linear regression model which can also be used for multi-class classification. By properly encoding class label information into c dimensional label vector $\mathbf{y}_i$, e.g., $y_{ik} = 1$, if dimension k corresponds to the true class label, otherwise $y_{ik} = -1$. Denoting a set of samples $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ as $\mathbf{X} \in \mathbb{R}^{n \times d}$ and their associated class labels as $\mathbf{Y} \in \mathbb{R}^{n \times C}$, Ridge Regression Classifier learns a linear classifier $\mathbf{W} \in \mathbb{R}^{d \times C}$ by optimizing the following problem:

$$J_{\mathrm{W}} = \min\{||\mathbf{XW} - \mathbf{Y}||^2 + \lambda||\mathbf{W}||^2\}, \tag{3.2}$$

where λ is a hyper-parameter to balance the mean square error and the regularization item. One benefit of using RR for classification is that the solution can be obtained via a closed-form:

$$\mathbf{W} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y}, \tag{3.3}$$

This closed-form solution makes the mapping from the training samples $\{\mathbf{X}, \mathbf{Y}\}$ to the classifier $\mathbf{W}$ differentiable. This property has been explored in the recent few-shot learning work [180, 193].

This paper aims to estimate the data distribution directly from a sample by taking the assumption that the distribution follows the Gaussian distribution. Formally, suppose the mean and covariance estimated for the j-th sample are $\{\mu(\mathbf{x}_j), \Sigma(\mathbf{x}_j)\}$, then the expectation of $\mathbf{X}^\top\mathbf{X}$ can be calculated as

$$E(\mathbf{X}^\top\mathbf{X}) = \sum_j^n E(\mathbf{x}_j^\top\mathbf{x}_j) = \sum_j^n (\mu(\mathbf{x}_i)\mu(\mathbf{x}_i)^\top + \Sigma(\mathbf{x}_j)), \tag{3.4}$$

Algorithm 1: Two-stage supervised training

Input: base dataset D_b
Input: feature extractor $\mathcal{F}$, classifier $\{f_{cls}, f_{rot}\}$
Input: distribution predictor $\{\phi_\mu, \phi_{\sigma^2}\}$
Output: fixed pre-trained $\mathcal{F}$ and $\{\phi_\mu, \phi_{\sigma^2}\}$

Stage 1 - Representation learning;
while *not done* **do**
  Sample a mini-batch images from base dataset D_b;
  Obtain prediction of classification and rotation of each sample;
  Update parameters of $\mathcal{F}$ by Equation 3.1;
end
Save features $\{z_j\}, j \in D_b$ of base samples;
Calculate real distribution $\{\hat{\mu}_i, \hat{\sigma}_i^2\}, i \in C_b$ as ground truth of training stage 2;
Stage 2 - Distribution meta-learning;
while *not done* **do**
  Sample meta-training tasks $\mathcal{T}_b = \{\mathcal{S}, \mathcal{Q}\}$ from $\{z_j\}, j \in D_b$;
  Generate the specific meta-classifier via predicting class-wise distributions of $\mathcal{S}$;
  Calculate the classification results on $\mathcal{Q}$ and update the parameters of $\{\phi_\mu, \phi_{\sigma^2}\}$ by objective function Equation 3.15;
end

similarly, we can calculate the expectation of $\mathbf{X}^\top\mathbf{Y}$ and this gives us the following equation to calculating $\mathbf{W}$:

$$\mathrm{W} = (\sum_{j}^{n}(\mu(\mathbf{x}_j)\mu(\mathbf{x}_j)^\top + \Sigma(\mathbf{x}_j)) + \lambda I)^{-1}(\sum_{j}^{n}\mu(\mathbf{x}_j)^\top\mathbf{y}_i), \tag{3.5}$$

3.3.4 Training stage 2: Distribution Learning

As aforementioned, our method aims to estimate the same-class distribution parameters from a feature instance directly. There are two treatments chosen in our approach. First, we apply a data preprocessing step to make the feature distributed more similar to a Guassian distribution. Second, we take an assumption that the feature dimensions are uncorrelated with each other, and therefore the covariance matrix is a diagonal matrix. Empirically, we find such an approximation can alleviate the difficulty of estimating the covariance matrix that might contain too many parameters and will not compromise the final classification performance. More specifically, the details about distribution learning are elaborated as follows:

Data preprocessing: We apply Tukey's Ladder of Powers [47] transformation on extracted features x to change the shape of a skewed distribution and make it nearly normal:

$$z = \begin{cases} x^{\tau} & \text{if } \tau > 0 \\ \log(x) & \text{if } \tau = 0 \\ -(x^{\tau}) & \text{if } \tau < 0, \end{cases} \tag{3.6}$$

where τ is the hyper-parameter to adjust the shape of a skewed distribution. We set $\tau = 0.5$ as the same as [210].

Prediction targets: At the training stage, the class-wise distribution parameters of each sample can be directly estimated from the training set, such as:

$$\begin{aligned} \hat{\mu}_i &= \frac{\sum_{j=1}^{N_i} \mathbf{x}_j}{N_i} \\ \hat{\sigma}_i^2 &= \frac{1}{N}(\mathbf{x}_i - \hat{\mu}_i) \odot (\mathbf{x}_i - \hat{\mu}_i), \end{aligned} \tag{3.7}$$

where $\odot$ denotes the element-wise product.

First, we calculate the ground distributions of each base class denoted as $\{\hat{\mu}_i, \hat{\sigma}_i^2\}, i \in C_b$. More specifically, we obtain embeddings of base dataset by fixed feature extractor and calculate the ground feature distribution of each base class. The target class-wise feature distribution can be calculated as:

$$\hat{\mu}_i = \frac{\sum_{j=1}^{N_i} z_j}{N_i}, \tag{3.8}$$

$$\hat{\sigma}_i^2 = E[(Z_i - \hat{\mu}_i)], \tag{3.9}$$

where N_i is the number of training samples Z_i of base class i, z_j is the feature vector of j-th sample extractor by backbone $\mathcal{F}$ and Tukey's Ladder of Power Transform.

To improve the generalization ability of distribution prediction model, amounts of meta-training tasks are generated from base features and donated as $\mathcal{T}_b = \{\mathcal{S}, \mathcal{Q}\}$. More specifically, for each episodic while training, we randomly sample N classes with K samples as support set $\mathcal{S}$ to generate class-wise distributions. While selecting other Q features per class to evaluation the classification performance as query set $\mathcal{Q}$. The feature embeddings of support and query set are denoted as $\{Z_{\mathcal{S}}, Z_{\mathcal{Q}}\}$ separately.

Model design: As shown in Figure C.1, a dedicated network with two branches is proposed to predict the distribution of each sample. Since the value of variance should be greater than zero, we predict $\log(\sigma_{\mathrm{j}}^2)$ instead. Each branch is MLP-like architecture combined with several fully connected layers and ReLU activation functions. The output of distribution predictor of jth sample can be formulated as:

$$\mu_j = \phi_{\mu}(\mathbf{x}_j), \tag{3.10}$$

$$\log(\sigma_{\mathrm{j}}^2) = \phi_{\sigma^2}(\mathbf{x}_j), \tag{3.11}$$

where $\mathbf{x}_j$ represents the features of support set.

Objective Function: To train our proposed distribution predictor, we introduce different loss items to measure the discrepancy between prediction and ground truth.

For mean vectors, we use Mean Squared Error (MSE) as:

$$\mathcal{L}_{\mu} = \frac{1}{NK}\sum_{j}^{NK}(\mu_j - \hat{\mu}_j)^2, \tag{3.12}$$

where NK is the number of support samples. To compute the loss of variance, we apply exponential operation on $\log(\sigma_j^2)$ to obtain σ_j^2, similarly, the loss function of variance is:

$$\mathcal{L}_{\sigma^2} = \frac{1}{NK}\sum_{j}^{NK}|\frac{\sigma_j^2}{\hat{\sigma}_j^2} - 1|, \tag{3.13}$$

The predicted distribution parameters will be used to learn the classifier from the support set features. Therefore, in addition to requiring them to be close to the parameters estimated from the training set, we also hope their estimation can lead to better few-shot learning performance. This motivates us to use meta-learning to further regularize the parameter estimators. Specifically, we follow the convention of few-shot learning to sample few-shot learning tasks that consist of a support set $\mathcal{S}$ and a query set $\mathcal{Q}$. For each $\mathcal{S}$, we use the Equation 3.5 to calculate the classifier. Then classifier will incur losses on the query set as an indication of the quality of the estimated classifier. Formally, this leads to the following loss function

$$\mathcal{L}_{rr} = \mathcal{L}_{CE}(\mathbf{W}_{\mathcal{S}}\mathbf{X}_{\mathcal{Q}}, \hat{\mathbf{Y}}_{\mathcal{Q}}; \mu, \sigma^2|\mathbf{Z}_{\mathcal{S}}), \tag{3.14}$$

where $\mathbf{W}_{\mathcal{S}}$ is the solution of support, $\hat{\mathbf{Y}}_{\mathcal{Q}}$ is target class-labels of query features $\mathbf{X}_{\mathcal{Q}}$ and $\{\mu, \sigma^2\}$ represents the predicted class-wise distribution from support features $\mathbf{X}_{\mathcal{S}}$.

Since the relationship between the distribution parameter estimator and the classifier is differentiable, backpropagation can be directly applied to optimize the parameters within the distribution predictor. Combining all the above loss terms with proper balancing parameters (β_1, β_2), we arrive at the following overall objective function:

$$\mathcal{L}_2 = \mathcal{L}_{\mu} + \beta_1 \cdot \mathcal{L}_{\sigma^2} + \beta_2 \cdot \mathcal{L}_{rr}, \tag{3.15}$$

3.3.5 Meta-testing on novel classes

After two-stage training, we obtain the feature extractor $\mathcal{F}$ and the distribution predictor $\{\phi_{\mu}, \phi_{\sigma^2}\}$ whose parameters are frozen in the meta-testing stage. For each meta-testing task $\mathcal{T}_n = \{\mathcal{S}, \mathcal{Q}\}$ of novel classes, we first perform data preprocessing on features extracted from the support set $\mathcal{S}$ and query set $\mathcal{Q}$, denoted as $\mathbf{X}_{\mathcal{S}}$ and $\mathbf{X}_{\mathcal{Q}}$. Next, we feed support features $\mathbf{X}_{\mathcal{S}}$ into distribution predictor $\{\phi_{\mu}, \phi_{\sigma^2}\}$ to generate distribution parameters for each sample. Then, we calculate $\mathrm{W_S}$ by Equation 3.5 and obtain the classification results on the query set. The detailed steps of meta-testing are illustrated on the right of Figure C.1-(b) and Algorithm 2.

Algorithm 2: Meta-testing

Input: novel support set images $\mathcal{S}$ and labels $\mathcal{Y}_{\mathcal{S}}$
Input: novel query set images $\mathcal{Q}$
Input: fixed feature extractor $\mathcal{F}$
Input: fixed distribution predictor $\{\phi_{\mu}, \phi_{\sigma^2}\}$
Output: classification prediction of query set $P_{\mathcal{Q}}$

1 Obtain features of support and query samples: $\mathbf{X}_{\mathcal{S}} = \text{trans}(\mathcal{F}(\mathcal{S})), \mathbf{X}_{\mathcal{Q}} = \text{trans}(\mathcal{F}(\mathcal{Q}))$;
2 Get distributions from support samples: $\mu_{\mathcal{S}} = \phi_{\mu}(\mathbf{X}_{\mathcal{S}}), \log(\sigma^2_{\mathcal{S}}) = \phi_{\sigma^2}(\mathbf{X}_{\mathcal{S}})$;
3 Calculate the approximated solution of Ridge Regression Classifier via predicted distribution: $\mathrm{W}_{\mathcal{S}} = ((\mu_{\mathcal{S}}\mu_{\mathcal{S}}^{\mathrm{T}} + \text{diag}(\sigma^2_{\mathcal{S}})) + \lambda I)^{-1}\mu_{\mathcal{S}}^{\mathrm{T}}\mathbf{Y}_{\mathcal{S}}$;
4 Apply coefficient matrix $\mathrm{W}_{\mathcal{S}}$ on query features: $P_{\mathcal{Q}} = \mathrm{W}_{\mathcal{S}} Z_{\mathcal{Q}}$;

3.4 Experiments

3.4.1 Implementation Details

In order to demonstrate the effectiveness of our proposed method, we conduct experiments on four public datasets: CIFAR-FS, FC100, miniImageNet, and tieredImagenet. We first introduce the experimental setups, including the details of benchmark datasets in Subsection 3.4.2 and then evaluation metrics we used in Subsection 3.4.3. Finally, we present the implementation details of our proposed method in Subsection 3.4.4.

3.4.2 Datasets

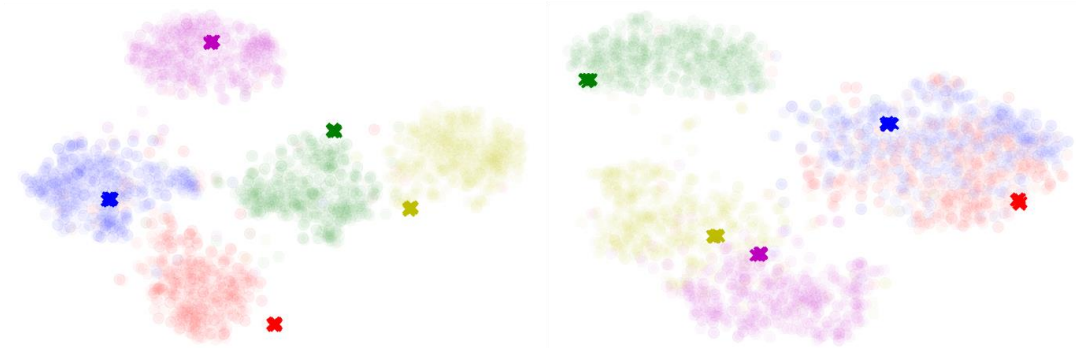
CIFAR-FS [6] and FC100 [143] are both derived from CIFAR-100 but split in different ways. CIFAR-FS dataset consists of 64 classes for training, 16 for validation, and 20 for testing. While FC100 dataset has 60 classes for training, 20 for validation, and 20 for testing. Each class has 600 samples and the image size is $32 \times 32 \times 3$.

The miniImageNet [185] and tieredImageNet dataset [156] are subsets of ILSVRC-2012 dataset (ImageNet) [160]. The miniImageNet dataset consists of 100 classes and 64 classes for meta-training , 16 for meta-validation and 20 for meta-testing. The tieredImageNet dataset is much larger than miniImageNet which consists of 608 classes and belongs to 34 higher-level categories. It has 351 classes for meta-training, 97 classes for validation and 160 classes for meta-testing. Each class has 600 samples and the image size is $84 \times 84 \times 3$.

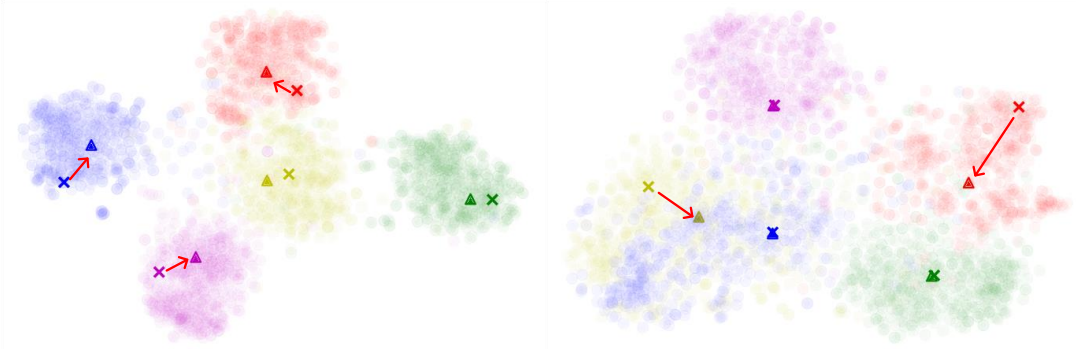
Same as previous works, we train our proposed model on base classes and conduct meta-testing on novel classes. To make a fair comparison with previous work, we drop all validation subsets.

3.4.3 Evaluation Metric

Following the state-of-the-arts [179][89], we conduct experiments with mean top-1 accuracy and 95% confidence interval over 2,000 meta-testing tasks. Each N-way K-shot classification task consists of N unseen classes and each class randomly samples



(a) t-SNE visualization of the sampling bias and the weakness of using traditional augmentation operations.



(b) t-SNE visualization of the effectiveness of our class-wise distribution prediction.

FIGURE 3.5. t-SNE visualization of few-shot classification with traditional augmentation operations and our proposed distribution prediction. Different colors represent different classes. '$\times$' denotes the extracted features of limited *support set* and their augmented counterparts, while translucent '●' denotes the large amounts of *query set*. '▲' in (b) represents the predicted class-wise mean vectors. (a) illustrates the sampling bias on randomly sampled support set and the weakness of using augmentation operations. (b) proves that our proposed method can alleviate over-bias by predicting class-wise distribution. Note that, we only show the predicted mean vector of each class. '$\rightarrow$' represents moving from the sampled support feature which is close to class boundary to our predicted class-wise mean as centre.

K images as support set $\mathcal{S}$ while other Q images as query set $\mathcal{Q}$. We follow previous works and set $N = 5$, $K = \{1, 5\}$ and $Q = 15$ in all experiments.

3.4.4 Implementation Details

Our proposed method is implemented based on open-source Pytorch 1.2.0 toolbox with the python 3.6 platform.

Data augmentation: As the same as previous work [179], we follow traditional data augmentation operations in public open-source code. More specifically, we randomly apply cropping, color jittering, and horizontal flipping on input mini-batch images.

Architecture of feature extractor: For a fair comparison with other methods, we develop ResNet12 and ResNet18 as backbone for all four datasets and additional WRN-28-10 network for miniImageNet and tieredImageNet. ResNet12 and ResNet18 consist of 4 residual blocks, a global average pooling with ReLU activation function, a fully connected (FC) layer to predict classification and an auxiliary FC branch to predict rotation degree. WRN-28-10 is a wide ResNet network with 28 layers and a width factor of 10. Previous works [149][35][210] shown that WRN-28-10 performances better on the miniImageNet and tieredImageNet. Note that the feature dimension is set to 640 in all above backbone architectures.

Architecture of distribution predictor: We develop a very simple but effective MLP-like network to learn distribution from base features. As shown in Figure C.1, the mean branch consists of 3 FC layers with ReLU activation functions, while the var branch has similar architecture but drops the last activation layer to satisfy the range of values of the log function.

Training hyperparameters: While in the training stage 1, representation learning, we apply SGD optimizer with a momentum of 0.9 and the initial learning rate is 0.05 and set the maximum epoch is 90 for CIFAR-FS and FC100. While we use Adam optimizer [83] with an initial learning rate is 0.001 and set the maximum epoch as 100 for miniImageNet and 60 for tieredImageNet. The setting of more training hyper-parameters refers to the open-source in [179, 210].

In the training stage 2, distribution learning, the maximum iteration is set to 2,000 for CIFAR-FS, FC100 while 3,000 for miniImageNet and 5,000 for tieredImageNet. The initial learning rate is set to 0.1 and will decay with a factor of 0.75 every 100, 150, 250 iterations for CIFAR-FS/FC100,s miniImageNet and tieredImageNet separately . The hyper-parameters of Equation 3.15 is set to $\{\beta_1 = 1, \beta_2 = 0.01\}$. Note that, we save the feature embeddings of base samples and calculate the ground class-wise distributions for each base class after stage 1.

3.4.5 Comparison with State-of-the-art methods

In this section, we present results from various experimental evaluations. The first evaluation is to verify our argument that the classifier obtained from ridge regression with the proposed simple-wise distribution estimation is beneficial in 3.4.6. Then, we investigate the relationship between distribution prediction and data augmentation in 3.4.7. After that, we systematically compare the proposed method with state-of-the-art approaches on four benchmark datasets CIFAR-FS, FC100, miniImagenNet, and tieredImageNet in Subsection 3.4.8 and 3.4.9.

3.4.6 The advantage of the proposed classifier

To compare our method with traditional deep learning and popular machine learning methods, we conduct extensive experiments on different meta-classifiers [179][182], including support vector machines (SVM), MLP-based classifier (MLP), Logistic Regression (LR), Prototypes-based classifier (Proto), Cosine similarity (Cosine) and Ridge Regression (RR) described as follows:

- **SVM**: Support Vector Machines can be implemented via scikit-learn toolkit while the default setting of kernel is "linear", gamma is set to "auto" and penalty $C = 1$.

- **MLP**: MLP based meta-classifier contains two fully connected layers with ReLU activation function. For each novel task, we train the target model on support set with Cross Entropy loss function and update the parameters 100 times. The optimizer is SGD, and the learning rate for initialization is 0.1 while decreases with factor 0.1 every 50 iterations.

- **LR**: Similar to Linear Regression, Logistic Regression models the data via the sigmoid function and it can also be simply implemented via scikit-learn toolkit. The default setting of penalty is L2 Norm, the solver is "lbfgs", while the maximum iteration is 1,000.

- **Proto**: Considering the mean vectors of support set as the class-wise prototypes and generating the classification results via computing the Euclidean Distance between query samples and all prototypes.

- **NN**: In Nearest Neighbour meta-classier, each query sample is assigned to the class of its nearest support sample.

- **Cosine**: Cosine meta-classifier calculates the cosine similarity between support features and query samples after unified normalization.

- **RR**: We fit the traditional Ridge Regression meta-classifier on support set and test on query samples. It can also be implemented via scikit-learn toolkit and the default α is set to 1.0 while solver is "auto".

- **Ours**: For our proposed method, we compute the approximated close-form solution of RR via predicted class-wise distributions from support set and apply the solution to predict classification of query.

TABLE 3.1. Comparison with different traditional meta-classifiers on miniImageNet with WRN-28-10 as feature extractor.

n	method	5-way 1-shot	5-way 5-shot
1	SVM	57.37 ± 0.45	82.45 ± 0.62
2	MLP	65.11 ± 0.45	83.05 ± 0.30
3	LR	66.40 ± 0.44	83.78 ± 0.30
4	Proto	65.11 ± 0.45	83.51 ± 0.30
5	NN	65.26 ± 0.45	77.86 ± 0.35
5	Cosine	65.02 ± 0.46	78.45 ± 0.33
6	RR	66.56 ± 0.44	83.03 ± 0.30
7	**Ours**	**69.06 ± 0.54**	**85.58 ± 0.42**

The results are reported in Table 3.1. Note that, for fair comparison, all meta-classifiers have the same feature embeddings of base classes as training data. It shows that applying "SVM" as meta-classifier failed to handle the over-bias problem, while "NN" and "Cosine" underperform in 5-way 5-shot few-shot setting with an obvious disparity, some machine learning based methods like "LR" and "RR" tends to achieve stable classification performance. By calibrating class-wise feature distributions, our proposed method achieves significant performance.

3.4.7 Comparison to related classifier combined augmentation methods

To further illustrate the effectiveness of our proposed method, we evaluate our method with different combinations of related augmentation operations with competitive meta-classifiers as shown in Table 3.2. For a fair comparison, we apply the same pre-trained backbone ResNet-12 as feature extractor on CIFAR-FS.

TABLE 3.2. Comparison with different combinations of augmentation operations and competitive meta-classifiers.

n	method	5-way 1-shot	5-way 5-shot
1	LR	72.04 ± 0.85	85.93 ± 0.62
2	Aug + LR	72.13 ± 0.90	86.26 ± 0.67
3	RR	72.23 ± 0.86	85.28 ± 0.66
4	Aug + RR	73.21 ± 0.94	85.66 ± 0.68
5	DC + Sampling w/cov + LR	73.34 ± 0.95	85.25 ± 0.63
6	DC + Sampling w/cov + RR	73.53 ± 0.88	85.66 ± 0.61
7	DC + Sampling w/var + LR	72.71 ± 0.91	84.68 ± 0.65
8	DC + Sampling w/var + RR	73.65 ± 0.92	85.45 ± 0.59
9	**DP + RR w/var(ours)**	**75.72 ± 0.68**	**86.53 ± 0.49**

Method 1 and 2 are implemented as the baseline [179]. They demonstrate that even traditional augmentation operations can improve performance to some extent. Method 3 and 4 are similar to Methods 1 and 2 but use Ridge Regression as the meta-classifier instead of Logistic Regression. It proves that the improvement of our method does not rely on the different meta-classifier. The reason why we choose RR is that we hope to obtain the solution of meta-classifier via support set and apply it to query set instead of modeling data generation. Method 5 is the implementation of distribution calibration [210] with sampling hundreds of augmented features driven by the base dataset. Method 6 is similar to 5 but uses RR as meta-classifier. It shows that calibrated class-wise distribution can replace augmented data sampling operation.

To evaluate the impact of covariance and variance while describing class-wise feature distribution, we conduct simplified distribution in methods 5-6 and report the results in 7-8. 7 shows that synthesizing augmented features method requires the

complete Gaussian-like representation. However, 8 demonstrates that biased estimation classifier RR focuses more on main feature components and can perform better when sampling synthetic features via variance.

This observation supports our assumption and also provides the possibility of approximated implementation of RR. The last item 9 is our proposed method which applies a pre-trained distribution predictor and can outperform other methods.

3.4.8 Evaluation on CIFAR-FS and FC100 dataset

For a fair comparison, we conducted extensive experiments with different backbones and reported the 5-way 1-shot and 5-way 5-shot classification results in Table 3.3 and Table 3.4. It shows that our proposed approach achieves superior classification performance on both datasets while using ResNet-12 and ResNet-18 as feature backbone. Note that the parameters in pre-trained model are fixed in the whole meta-testing stage.

The results shows that our method has achieved significant improvement, especially in 5-way 1-shot learning setting which is the worst-case scenario. In this situation, it is more likely to encounter the problem of over-bias once randomly selecting the support sample which is located at the edge of the belonging class distribution. Unlike traditional augmentation operations, our method calibrate the class-wise feature distributions via predicting model and shows a strong generalization ability in novel classes.

In addition to quantitative results, we also visualize the predicted mean vectors of meta-testing tasks to compare the qualitative performance with traditional augmentation operations as shown in Figure 4.5. From where we can see in Figure 4.5-(a), there is a model collapse problem that augmented samples are almost overlapping. However, our predicted mean vectors are much closer to the center of feature distribution as shown in Figure 4.5-(b). Meanwhile, the variance will further describe the class-wise distribution. Although GANs or contrastive learning can synthesize augmented samples with discriminative features, it need extensive computation. Our work solves the problem in another way, and shows the probability and effectiveness of modeling data distribution instead of data generation.

3.4.9 Evaluation on miniImageNet and tieredImageNet dataset

We conduct the similar meta-testing evaluation experiments on miniImageNet and tieredImageNet datasets. The results are reported in Table 3.5. For a fair comparison, we use ResNet-12, ResNet-18 and a larger model WRN-28-10 as the backbone to get feature embeddings. It shows that our proposed approach achieves superior performance than the state-of-the-arts on miniImageNet based on different backbone networks. However it underperformed on tieredImageNet slightly when compared with the competitive method DC w/Sampling [210]. We suppose that since tieredImageNet contains much more categories compared with the other three datasets, the base dataset may have very complicated class-wise distribution in feature space which affect the effectiveness of distribution learning further. Even though, our work can still outperform most of previous works.

TABLE 3.3. Comparison to prior work on CIFAR-FS.

method	**backbone**	**CIFAR-FS**	
		5-way 1-shot	**5-way 5-shot**
MAML[46]	32-32-32-32	58.9 ± 1.9	71.5 ± 1.0
PrototypicalNet[168]	64-64-64-64	55.5 ± 0.7	72.0 ± 0.6
RelationNet[175]	64-64-128-256	55.0 ± 1.0	69.3 ± 0.8
R2D2[6]	96-192-384-512	65.3 ± 0.2	79.4 ± 0.1
Shot-Free[155]	ResNet-12	69.2 ± N/A	84.7 ± N/A
TEWAM[150]	ResNet-12	70.4 ± N/A	81.3 ± N/A
PrototypicalNet[168]	ResNet-12	72.2 ± 0.7	83.5 ± 0.5
MetaOptNet[91]	ResNet-12	72.6 ± 0.7	84.3 ± 0.5
Distillation[179]	ResNet-12	73.9 ± 0.8	86.9 ± 0.5
TFH w/ Finetune[89]	ResNet-18	74.7 ± 0.9	86.8 ± 0.6
DP (ours)	ResNet-12	75.7 ± 0.7	86.5 ± 0.5
DP (ours)	**ResNet-18**	**76.1 ± 0.6**	**87.1 ± 0.5**

TABLE 3.4. Comparison to prior work on FC100.

method	**backbone**	**FC100**	
		5-way 1-shot	**5-way 5-shot**
PrototypicalNet[168]	64-64-64-64	35.3 ± 0.6	48.6 ± 0.6
TADAM[143]	ResNet-12	40.1 ± 0.4	56.1 ± 0.4
PrototypicalNet[168]	ResNet-12	37.5 ± 0.6	52.5 ± 0.6
MetaOptNet[91]	ResNet-12	41.1 ± 0.6	55.5 ± 0.6
Distillation[179]	ResNet-12	44.6 ± 0.7	60.9 ± 0.6
DP (ours)	ResNet-12	44.9 ± 0.6	60.6 ± 0.6
DP (ours)	**ResNet-18**	**45.4 ± 0.6**	**60.9 ± 0.6**

TABLE 3.5. Comparison to prior work on miniImageNet and tieredImageNet.

method	backbone	miniImageNet		tieredImageNet	
		5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
MAML[46]	32-32-32-32	48.70 ± 1.84	63.11 ± 0.92	51.67 ± 1.81	70.30 ± 1.75
MatchingNet[185]	64-64-64-64	43.56 ± 0.84	55.31 ± 0.73	-	-
IMP[1]	64-64-64-64	49.20 ± 0.70	64.70 ± 0.70	-	-
PrototypicalNet[168]	64-64-64-64	49.42 ± 0.78	68.20 ± 0.66	53.31 ± 0.89	72.69 ± 0.74
TAML[75]	64-64-64-64	51.77 ± 1.86	66.05 ± 0.85	-	-
SAML[61]	64-64-64-64	52.22 ± N/A	66.49 ± N/A	-	-
GCR[92]	64-64-64-64	53.21 ± 0.80	72.34 ± 0.64	-	-
KTN[147]	64-64-64-64	54.61 ± 0.80	71.21 ± 0.66	-	-
PARN[195]	64-64-64-64	55.22 ± 0.84	71.55 ± 0.66	-	-
Dynamic[53]	64-64-128-128	56.20 ± 0.86	73.00 ± 0.64	-	-
RelationNet[175]	64-64-128-256	50.44 ± 0.82	65.32 ± 0.70	54.48 ± 0.93	71.32 ± 0.78
R2D2[6]	96-192-384-512	51.20 ± 0.60	68.80 ± 0.10	-	-
SNAIL[132]	ResNet-12	55.71 ± 0.99	68.88 ± 0.92	-	-
AdaResNet[135]	ResNet-12	56.88 ± 0.62	71.94 ± 0.57	-	-
TADAM[143]	ResNet-12	58.50 ± 0.30	76.70 ± 0.30	-	-
Shot-Free[155]	ResNet-12	59.04 ± N/A	77.64 ± N/A	63.52 ± N/A	82.59 ± N/A
TEWAM[150]	ResNet-12	60.07 ± N/A	75.90 ± N/A	-	-
MTL[173]	ResNet-12	61.20 ± 1.80	75.50 ± 0.80	-	-
AM3[106]	ResNet-12	65.21 ± 0.49	75.20 ± 0.36	67.23 ± 0.34	78.95 ± 0.22
Neg-Cosine[200]	ResNet-12	63.85 ± 0.81	81.57 ± 0.56	-	-
Variation FSL[225]	ResNet-12	61.23 ± 0.26	77.69 ± 0.17	-	-
MetaOptNet[91]	ResNet-12	62.64 ± 0.61	78.63 ± 0.46	65.99 ± 0.72	81.56 ± 0.53
Diversity[41]	ResNet-18	59.48 ± 0.65	75.62 ± 0.48	-	-
Fine-tuning[35]	WRN-28-10	57.73 ± 0.62	78.17 ± 0.49	66.58 ± 0.70	85.55 ± 0.48
LEO-trainval[161]	WRN-28-10	61.76 ± 0.08	77.59 ± 0.12	66.33 ± 0.05	81.44 ± 0.09
Distillation[179]	ResNet-12	64.82 ± 0.60	82.14 ± 0.43	71.52 ± 0.69	86.03 ± 0.49
DSN-MR[166]	ResNet-12	64.60 ± 0.72	79.51 ± 0.50	67.39 ± 0.82	82.85 ± 0.56
DeepEMD[219]	ResNet-12	65.91 ± 0.82	82.41 ± 0.56	71.16 ± 0.87	86.03 ± 0.58
FEAT[215]	ResNet-12	66.78 ± 0.20	82.05 ± 0.14	70.80 ± 0.23	84.79 ± 0.16
TFH w/ Finetune[89]	ResNet-18	65.07 ± 0.82	80.81 ± 0.61	-	-
DC w/ Sampling[210]	WRN-28-10	68.57 ± 0.55	82.88 ± 0.42	**78.19 ± 0.25**	**89.90 ± 0.41**
CPLAE[51]	ResNet-12	67.46 ± 0.44	83.22 ± 0.29	72.23 ± 0.50	87.35 ± 0.34
DP (ours)	ResNet-12	67.82 ± 0.65	82.83 ± 0.46	75.37 ± 0.52	87.45 ± 0.45
DP (ours)	ResNet-18	68.04 ± 0.60	83.17 ± 0.48	75.86 ± 0.55	88.04 ± 0.49
DP (ours)	WRN-28-10	**69.06 ± 0.54**	**83.58 ± 0.42**	76.62 ± 0.54	89.36 ± 0.48

3.4.10 Ablation Study

In this subsection, we conduct more experiments to further analyse our method with very related methods and evaluate the effects of different modules and hyper-parameters setting in our method. For fair comparison, the following results are all evaluated on novel tasks sampled from miniImageNet dataset, while the pre-trained backbone is WRN-28-10 network with rotation prediction as auxiliary task. It is believed that, the other CNN backbones share similar performance trend.

A. Effect of hyper-parameter N while training:

As we mentioned above, to further improve the performance on novel classes, we adapt meta-learning strategy to train the distribution predictor. More specificity, we randomly sample N_b base classes for training and select $N = 5$ classes from novel dataset to evaluation the few-shot ability of proposed model. The number of N_b in each training episodic can effect the final results as we shown in Table 3.6. We find that 25-way training episodes for 5-way 1-shot setting and 20-way training episodes for 5-way 5-shot setting can achieve better performances. As it shown that training with harder classification tasks could be helpful to force the target model to learn to distinguish more complex categories boundaries which can further improve both generalization and discriminative ability while testing on novel classes. We figure that different combinations of base classes can encourage our model to update parameters with a more general direction, however, too many categories contain both more information and noise may hinder the model to find a local solution in each episodic.

TABLE 3.6. Comparison with different combined meta-classifiers.

N_b	**5-way 1-shot**	**5-way 5-shot**
5	65.74 ± 0.66	85.03 ± 0.42
10	66.83 ± 0.60	85.26 ± 0.44
15	67.03 ± 0.56	85.38 ± 0.41
20	67.91 ± 0.55	**85.58 ± 0.42**
25	**69.06 ± 0.54**	85.49 ± 0.41
30	69.03 ± 0.54	85.53 ± 0.44

B. Effect of using average function while obtaining class-wise feature distributions: To further improve the classification performance, we combine our methods with traditional augmentation operations, such as color jittering, flip, clip, as previous works did. Therefore, we have two options to obtain class-wise distribution while calculating the solution of our approximated ridge regression classifier. The details are as follows:

I. Calculate the solution W via each predicted distributions from all training samples directly:

$$\begin{aligned} \mathrm{W_A} = (\tfrac{1}{NK}\textstyle\sum_i^{NK}(\mu(x_i)\mu(x_i)^\top + \mathrm{diag}(\sigma^2(x_i)) + \lambda I)^{-1} \\ [\mu(x_1), \mu(x_2), ..., \mu(x_{NK})]^{\mathrm{T}} Y, \end{aligned} \tag{3.16}$$

where NK is the number of samples in support set while training.

II. Before calculating the solution of approximated meta-classifier, applying average function on mean and variant vectors at first:

$$\bar{\mu_i} = \frac{1}{K}\sum_{j}^{K}\mu_i^j, \tag{3.17}$$

$$\bar{\sigma_i}^2 = \frac{1}{K}\sum_{j}^{K}(\bar{\sigma_i}^2)_j. \tag{3.18}$$

Then the W can be formulated as:

$$\begin{aligned} \mathrm{W_B} = (\tfrac{1}{N}\textstyle\sum_i^N(\bar{\mu_i}\bar{\mu_i} + \mathrm{diag}(\bar{\sigma_i}^2) + \lambda I)^{-1} \\ [\bar{\mu_1}, \bar{\mu_2}, ..., \bar{\mu_C}]^{\mathrm{T}}Y, \end{aligned} \tag{3.19}$$

where N is the number of categories in each meta-task. The comparison of using W_1 and W_2 has reported in Table 3.7. It shows that considering each sample individually can achieve improvement slightly especially in 1-shot situation due to it can provide more information and benefits the target model adapt to various representations of each class.

TABLE 3.7. Quantitative comparison of computing W via different ways

option	5-way 1-shot	5-way 5-shot
I	**69.06 ± 0.54**	**86.58 ± 0.42**
II	68.18 ± 0.49	86.45 ± 0.40

C. Effect of hyper-parameter β_2 of objective function : Table 3.8 shows that adding the last classification loss item in Equation 3.15 with an appropriate weight is helpful to better generalize to novel classes. It constrains the meta-classifier should not only regress truth class-wise distribution but also complete few-shot learning task and focus on categorize correctly. However, a large weight $\beta_2 = 0.1$ encourages to recognize base samples while training but hinders the convergence on real mean an variance. Note that we set $\beta_1 = 1$ due to ensure that mean loss and variance loss are of the same order of magnitude.

TABLE 3.8. Quantitative comparison of the effect of β_2

β_1	β_2	5-way 1-shot	5-way 5-shot
1	0	74.15 ± 0.81	85.43 ± 0.54
1	0.01	**75.72 ± 0.68**	**86.53 ± 0.49**
1	0.1	72.62 ± 0.70	84.46± 0.48

D. Effect of hyper-parameter λ when solving the closed-form of Ridge Regression Classifier: As we described in Subsection 3.3.3, there is a weighted identity

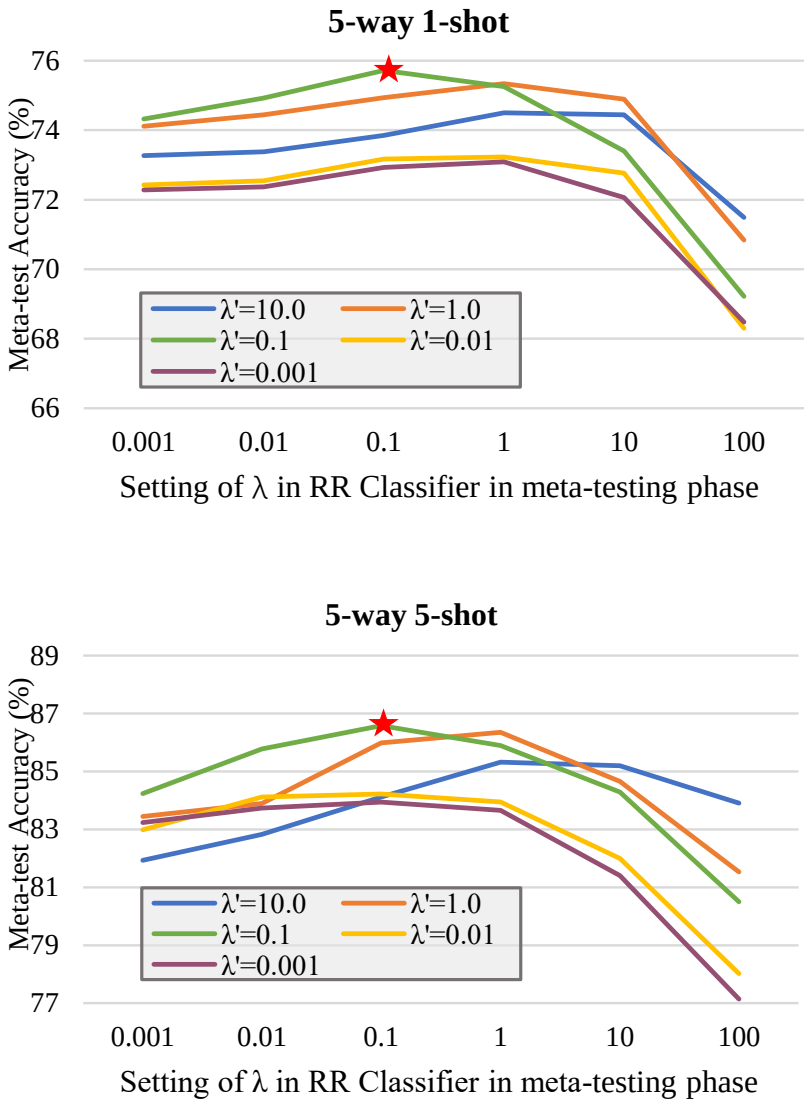


FIGURE 3.6. Quantitative comparison of the effect of λ in the Ridge Regression classifier during the training and meta-testing stages. Note that λ' in the legend represents the value set during the distribution learning stage. The symbol $\star$ denotes the best performance.

matrix when deriving the solution. It plays an important role in both training stage and meta-testing stage. Figure 3.6 detailed illustrates the impact of hyper-parameters λ setting. As result, it performances best when the value is set to 1.

3.5 Conclusion

In this section, we propose a novel paradigm to generalize class-wise distribution learning instead of modeling data generation from base classes to novel classes and develop a simple but strong baseline. In addition, we redefine a simplified distribution format and propose the mathematical approximated implementation of Ridge Regression as meta-classifier. To evaluate the effectiveness of our proposed method, we conduct extensive quantitative comparisons with state-of-the-arts and recombine closely related methods. To display our improvement intuitively, we further plot t-SNE visualization results that show our work can refine data distribution effectively, especially at the 1-shot situation.

The future work can be improved from many respects: (1) Exploring specific network architecture and objective function to generalize distribution more effectively. (2) In this work, we predict distribution via images including target and background, will distribution learning performs better only via the pixels of correlated objects? (3) Investigating distribution learning from a larger pre-trained networks even cross-domain models. Our method is an exploration of few-shot classification via distribution learning, we believe that it will be a potential solution in further study and transfer to other few-shot tasks, like segmentation, detection and semi-supervised learning setting tasks.

Statement of Authorship

Title of Paper	Exploring few-shot classification via feature disctribution learning
Publication Status	☐ Published ☐ Accepted for Publication ☐ Submitted for Publication ☒ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	

Principal Author

Name of Principal Author (Candidate)	Ziqin Zhou		
Contribution to the Paper	Developed the conceptual framework, completed the coding, conducted experiments, and wrote the manuscript for the paper.		
Overall percentage (%)	70%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	25.07.2024

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Lingqiao Liu		
Contribution to the Paper	Idea discussions and contributed to writing revisions.		
Signature		Date	25.07.2024

Name of Co-Author			
Contribution to the Paper			
Signature		Date	

Please cut and paste additional co-author panels here as required.

Chapter 4

Towards Adapting CLIP for Zero-shot Semantic Segmentation

4.1 Introduction

In recent advancements, CLIP has been applied to pixel-level zero-shot learning tasks through a two-stage scheme. This approach involves generating class-agnostic region proposals and then feeding the cropped regions into CLIP to leverage its image-level zero-shot classification capabilities. While effective, this method necessitates two image encoders—one for proposal generation and another for CLIP—resulting in a complex pipeline and increased computational cost.

To address these challenges, we propose a simpler and more efficient one-stage solution that extends CLIP's zero-shot prediction capability directly from image to pixel level. Our baseline approach generates semantic masks by comparing the similarity between text and patch embeddings extracted from CLIP. However, this method tends to overfit seen classes, hindering generalization to unseen classes.

To mitigate this issue, we introduce three key designs:

Deep Prompt Tuning (DPT): Instead of fine-tuning the entire CLIP model, we employ prompt tuning, which adjusts only the input prompts while keeping the model parameters fixed. This strategy preserves CLIP's inherent zero-shot capabilities and enhances generalization to unseen classes.

Non-mutually Exclusive Loss (NEL): Traditional cross-entropy loss assumes mutually exclusive classes, which can be restrictive in zero-shot settings. NEL allows for overlapping class probabilities, enabling the model to better handle ambiguous or novel classes.

Relationship Descriptor (RD): We incorporate additional textual descriptors that capture relationships between classes. This provides contextual information, aiding the model in distinguishing between similar classes and improving segmentation accuracy.

Integrating these designs results in ZegCLIP, an efficient zero-shot semantic segmentation system. Extensive experiments on three public benchmarks demonstrate that ZegCLIP outperforms state-of-the-art methods by a significant margin in both "inductive" and "transductive" zero-shot settings. Moreover, compared to the two-stage approach, our one-stage ZegCLIP achieves approximately five times faster inference speed.

4.2 Background

Semantic segmentation is one of the fundamental tasks in the computer vision field, which aims to predict the category of each pixel of an image [21, 44, 128, 183]. Extensive works have been proposed [25, 108, 121], e.g., Fully Convolutional Networks [120], U-net [159], DeepLab family [13–15] and more recently Vision Transformer based methods [58, 228, 236]. However, the success of the deep semantic segmentation models heavily relies on the availability of a large amount of annotated training images, which involves a substantial amount of labor. This gives rise to a surging interest in low-supervision-based semantic segmentation approaches, including semi-supervised [21], weakly-supervised [204], few-shot [198], and zero-shot semantic segmentation[10, 146, 197].

Among them, zero-shot semantic segmentation is particularly challenging and attractive since it is required to directly produce the semantic segmentation results based on the semantic description of a given class. Recently, the pre-trained vision-language model CLIP [152] has been adopted into various dense prediction tasks, such as referring segmentation [192], semantic segmentation [144], and detection [42]. It also offers a new paradigm and has made a breakthrough for zero-shot semantic segmentation. Initially built for matching text and images, CLIP has demonstrated a remarkable capability for image-level zero-shot classification.

However, zsseg [207] and Zegformer[36] follow the common strategy that needs **two-stage** processing that first generates region proposals and then feeds the cropped regions to CLIP for zero-shot classification. Such a strategy requires two image encoding processes, one for generating proposals and one for encoding each proposal via CLIP. This design creates additional computational overhead and cannot leverage the knowledge of the CLIP encoder at the proposal generation stage. Besides, MaskCLIP+[232] utilizes CLIP to generate pseudo labels of novel classes for self-training but will be invalid if the unseen class names in inference are unknown in the training stage ("inductive" zero-shot setting).

This paper pursues simplifying the pipeline by directly extending the zero-shot capability of CLIP from image-level to pixel-level. The basic idea of our method is straightforward: we use a lightweight decoder to match the text prompts against the local embeddings extracted from CLIP, which could be achieved via the self-attention mechanism in a transformer-based structure. We train the vanilla decoder and fix or fine-tune the CLIP image encoder on a dataset containing pixel-level annotations from a limited number of classes, expecting the text-patch matching capability can generalize to unseen classes. Unfortunately, this basic version tends to overfit the training set: while the segmentation results for seen classes generally improve, the model fails to produce reasonable segments on unseen classes. Surprisingly, we discover such an overfitting issue can be dramatically alleviated by incorporating three modified design choices and report the quantitative improvements in Figure 4.1. The key designs in our proposed method include: using deep prompt tuning (DPT), applying non-mutually exclusive loss (NEL) and introducing relationship descriptor (RD).

The motivation behind these designs is to address specific limitations observed in the baseline approach. DPT aims to prevent overfitting to seen classes by maintaining the model's zero-shot capabilities, thereby improving generalization to unseen

TABLE 4.1. Differences between our approach and related zero-shot semantic segmentation methods based on CLIP.

Methods	Need an extra image encoder?	CLIP as an image -level classifier?	Can do inductive?
zsseg[207]	✓	✓	✓
ZegFormer[36]	✓	✓	✓
MaskCLIP+[232]	✓	✗	✗
ZegCLIP (Ours)	✗	✗	✓

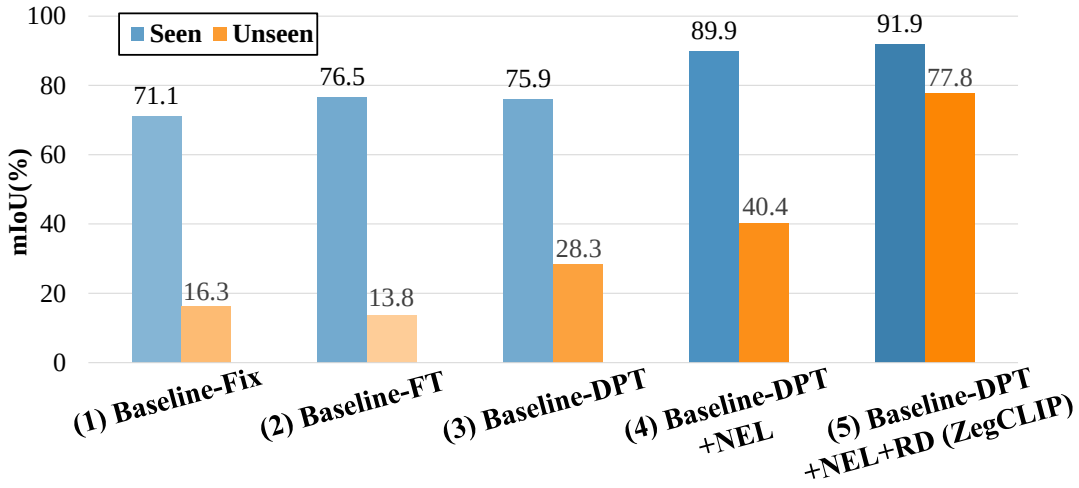


FIGURE 4.1. Quantitative improvements achieved by our proposed designs on VOC dataset. (1)(2) represents our one-stage **Baseline** model of different versions (Fix or Fine-Tune CLIP image encoder), while (3)-(5) shows the effectiveness of applying our proposed designs, i.e., **Deep Prompt Tuning (DPT)**, **Non-mutually Exclusive Loss (NEL)**, **Relationship Descriptor (RD)**, on baseline model step by step. We highlight that our designs can dramatically increase the segmentation performance on unseen classes.

classes. NEL addresses the restrictive nature of traditional loss functions in zero-shot settings, allowing the model to accommodate overlapping class probabilities and better handle novel classes. RD provides contextual information that helps the model distinguish between similar classes, enhancing segmentation accuracy.

By incorporating those three designs into our **one-stage** baseline, we create a simple-but-effective zero-shot semantic segmentation model named **ZegCLIP**. Table 4.1 summarizes the differences between our proposed method and existing approaches based on CLIP. More details can be found in Appendix. We conduct extensive experiments on three public datasets and show that our method outperforms the state-of-the-art methods by a large margin in both the "inductive" and "transductive" settings.

4.3 Method

4.3.1 Problem Formulation

Our proposed method follows the generalized zero-shot semantic segmentation (GZLSS) [197], which requires to segment both seen classes $\mathcal{C}^s$ and unseen classes $\mathcal{C}^u$ after only training on a dataset with pixel-annotations of seen part. In the training stage, the model generates per-pixel classification results from the semantic description of all seen classes. In the testing stage, the model is expected to produce segmentation results for both known and novel classes. Note that $\mathcal{C}^s \cap \mathcal{C}^u = \oslash$ and the labels of $\mathcal{C}^u$ are unavailable while training. The key problem of zero-shot segmentation is that merely training on seen classes inevitably leads to overbias on known categories while inference.

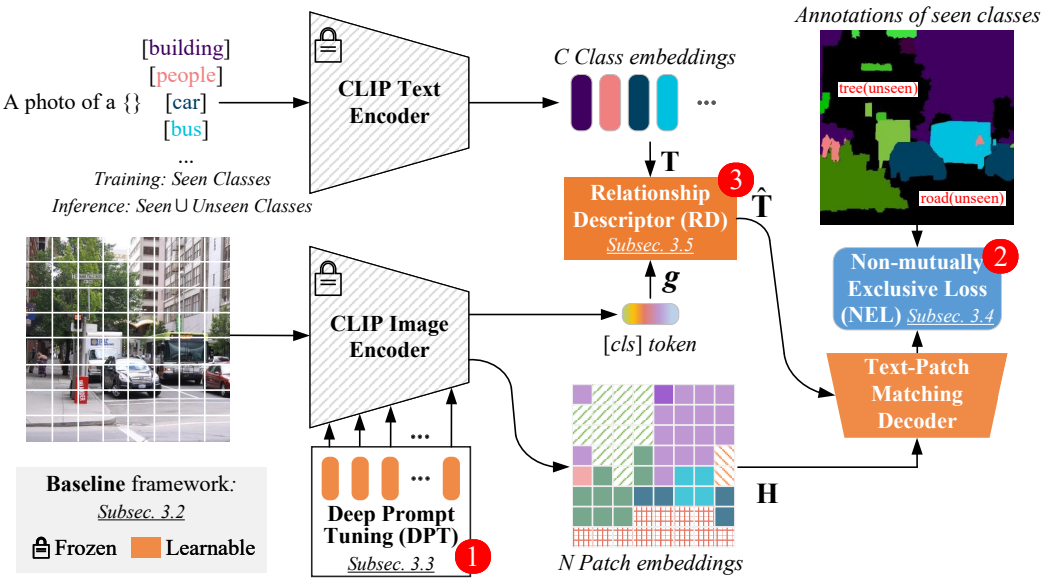


FIGURE 4.2. Overall of our proposed **ZegCLIP**. Our method modifies a **One-Stage Baseline** framework of matching text and patch embeddings from CLIP to generate semantic masks. The key contribution of our work is three simple-but-effective designs (labeled as the red circles 1,2,3 in the figure). Incorporating these three designs into our proposed one-stage baseline framework can upgrade the poorly performed baseline method to a strong zero-shot segmentation model.

This naturally corresponds to the "inductive" zero-shot segmentation setting, in which both unseen class names and images are not accessible during training. Besides "inductive" zero-shot segmentation, there is a "transductive" zero-shot learning setting, which assumes that the names of unseen classes are known before the testing stage. They [59, 232] suppose that the training images include the unseen objects, and only ground truth masks for these regions are not available. Our method can easily be extended to both settings and achieve excellent performance.

4.3.2 Baseline: One-stage Text-Patch Matching

As the large-scale pre-trained model CLIP shows impressive zero-shot classification ability, recent methods explore applying CLIP to zero-shot segmentation by proposing a **two-stage** paradigm. In stage 1, they train a class-agnostic generator and then leverage CLIP as a zero-shot image-level classifier by matching the similarity between text embeddings and [cls] token of each proposal in stage 2. While effective, such a design requires two image encoding processes and brings expensive computational overhead.

To simplify the two-stage pipeline when adapting CLIP to zero-shot semantic segmentation, in this work, we aim to cope with the critical problem that how to transfer CLIP's powerful generalization capability from image to pixel level classification effectively. Motivated by the observation of recent work [232] that the text embedding can be implicitly matchable to patch-level image embeddings, we build a **one-stage** baseline by adding a vanilla light-weight transformer as a decoder inspired by [39, 218]. Then we formulate semantic segmentation as a matching problem between a representative class query and the image patch features.

Formally, let's denote the C class embeddings as $\mathbf{T} = [\mathbf{t}^1, \mathbf{t}^2, ..., \mathbf{t}^C] \in \mathbb{R}^{C\times d}$, with d is the feature dimension of CLIP model, $\mathbf{t}^i$ representing the ith class, and the N patch tokens of an image as $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_N] \in \mathbb{R}^{N\times d}$, with $\mathbf{h}_j$ denoting the jth patch.

Then we apply linear projections ϕ to generate $\mathbf{Q}(query)$, $\mathbf{K}(key)$ and $\mathbf{V}(value)$ as:

$$\begin{aligned} \mathbf{Q} &= \phi_q(\mathbf{T}) \in \mathbb{R}^{C\times d} \\ \mathbf{K} = \phi_k(\mathbf{H}) \in \mathbb{R}^{N\times d}&, \mathbf{V} = \phi_v(\mathbf{H}) \in \mathbb{R}^{N\times d}. \end{aligned} \tag{4.1}$$

The semantic masks could be calculated by the scaled dot-product attention which is the intermediate product of the multi-head attention model (MHA):

$$\text{Masks} = \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} \in \mathbb{R}^{C\times N}, \tag{4.2}$$

where $\sqrt{d_k}$ is the dimension of the keys as a scaling factor, and the final segmentation results are obtained by applying the Argmax operation on the class dimension of Masks. The detailed architecture of the decoder has been shown on the right of Figure 4.3, which consists of three layers of transformers.

How to update the CLIP image encoder: The patch feature representation is generated by the CLIP image encode. How to update the CLIP image encoder, e.g., how to calculate $\mathbf{H}$, is an important factor. In our baseline approaches, we consider $\mathbf{H}$ is obtained from a parameter fixed CLIP or a parameter tunable CLIP, denoted as Baseline-Fix and Baseline-FT separately. Later in SubSection 4.3.3, we will discuss Deep Prompt Tuning (DPT), which turns out to be a better way to adapt the CLIP for zero-shot segmentation.

How to train the segmentation model: To properly train the decoder and (optionally) the CLIP model, we apply the commonly used softmax operator to convert the logits calculate from Equation 5.10 to the posterior probability. Exclusive Loss (EL) like Cross-entropy is then used as the objective function. Later in SubSection 4.3.4, we will point out this seemly straightforward strategy can be potentially harmful for generalization.

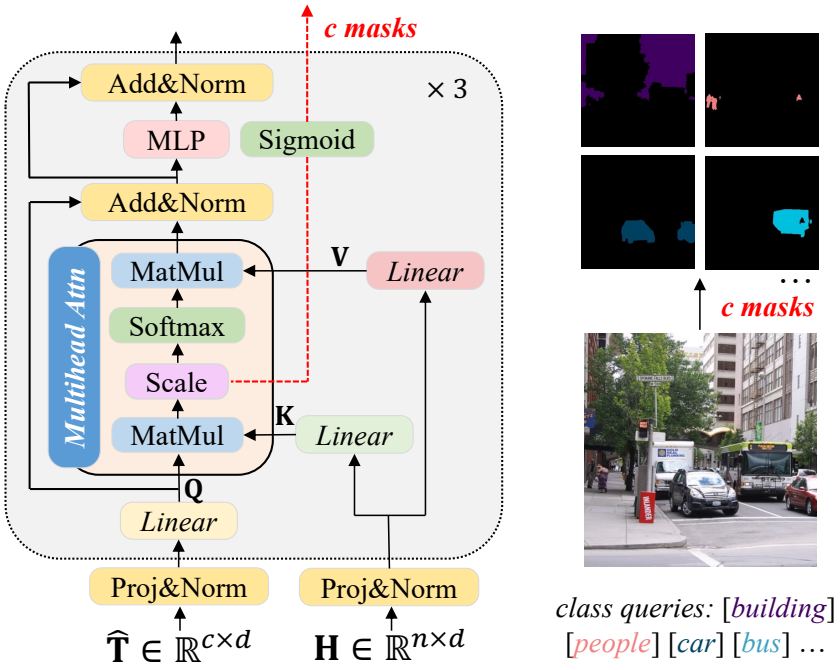


FIGURE 4.3. The detailed architecture of text-path matching decoder.

Design of query embedding T: The query embeddings $\mathbf{T}$ are the key to our approach. In our baseline model, we use the embeddings from the CLIP text encoder. However, as will be pointed out in SubSection 4.4.6, such a choice might cause severe overfitting and we propose a relationship descriptor by using the relationship between text and image token as class queries. In SubSection 4.4.6-A, we further explore other choices of $\mathbf{T}$. Please refer to the relevant sections for more details.

As shown in Figure 4.1-(1)(2), Table 4.6 and Table 4.7, our baseline model turns out to perform poorly in practice, especially for the unseen classes. It seems that the model overfits severely to the seen classes and forgets the zero-shot learning capability of CLIP during training. Fortunately, we identify three simple-but-effective designs that can dramatically alleviate this issue, as will be described in the subsequent sections.

4.3.3 Design 1: Deep Prompt Tuning (DPT)

The first design modification is to use deep prompt tuning (DPT) [233] for the CLIP backbone rather than fine-tuning CLIP. Prompt tuning [201] is a recently proposed scheme for adapting a pre-trained transformer model to a target domain. It has become a competitive alternative to fine-tuning in the transfer learning setting [77]. In this work, we explore deep prompt tuning to adapt CLIP for zero-shot image segmentation. Prompt tuning fixes the original parameters of CLIP and adds learnable prompt tokens as additional input for each layer. Since the zero-shot segmentation model is trained on a limited number of seen classes, directly fine-tuning the model tends to overfit the seen classes as the model parameters are adjusted to optimize the loss only for the seen classes. Consequently, knowledge learned for vision concepts

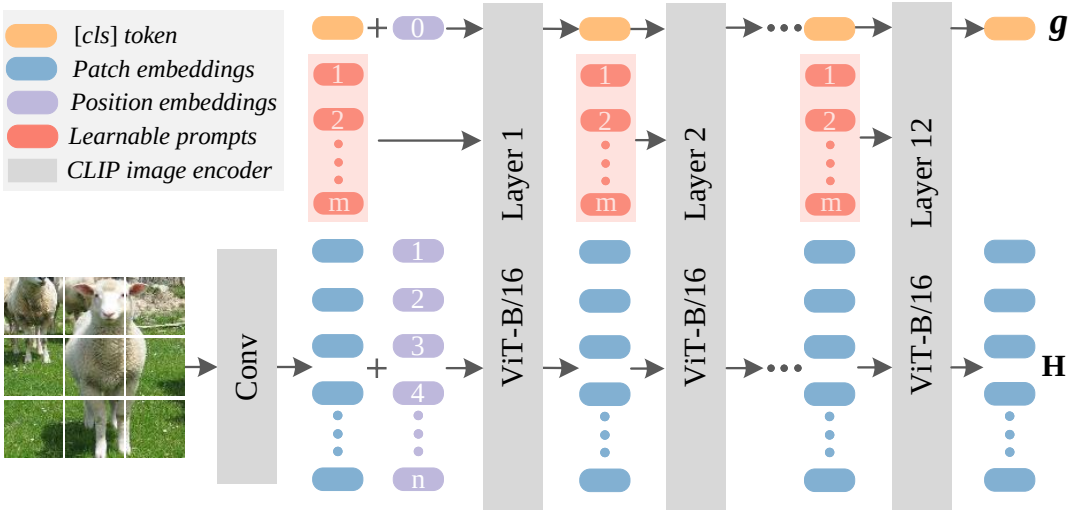


FIGURE 4.4. The architecture of deep prompt tuning.

unseen from the training set might be discarded in fine-tuning. Prompt tuning could potentially alleviate this issue since the original parameters are intact during training.

Formally, we denote the input embeddings from the l-th MHA module of the ViT-based image encoder in CLIP as $\{\mathbf{g}^l, \mathbf{h}_1^l, \mathbf{h}_2^l, \cdots, \mathbf{h}_N^l\}$, where $\mathbf{g}^l$ denotes the [CLS] token embedding and $\mathbf{H}^l = \{\mathbf{h}_1^l, \mathbf{h}_2^l, \cdots, \mathbf{h}_N^l\}$ denotes the image patch embeddings. Deep prompt tuning appends learnable tokens $\mathbf{P}^l = \{\mathbf{p}_1^l, \mathbf{p}_2^l, \cdots, \mathbf{p}_M^l\}$ to the above token sequence in each ViT layer of CLIP image encoder. Then the l-th MHA module process the input token as:

$$[\mathbf{g}^l, _, \mathbf{H}^l] = \text{Layer}^l([\mathbf{g}^{l-1}, \mathbf{P}^{l-1}, \mathbf{H}^{l-1}]), \tag{4.3}$$

where the output embeddings of $\{\mathbf{p}_1^l, \cdots, \mathbf{p}_M^l\}$ are discarded (denoted as _) and will not feed into the next layer. Therefore, $\{\mathbf{p}_1^l, \mathbf{p}_2^l, \cdots, \mathbf{p}_M^l\}$ merely act as a set of learnable parameters to adapt the MHA model.

As shown in Figure 4.1-(3), Table 4.6 and Table 4.7, compared with fine-tuning, deep prompt tuning on CLIP image encoder can achieve similar performance on seen classes but improve the segmentation results on unseen significantly.

4.3.4 Design 2: Non-mutually Exclusive Loss (NEL)

The general practice of semantic segmentation models treats it as a per-pixel multi-way classification problem, and Softmax operation is used to calculate the posterior probability, followed by using mutually Exclusive Loss (**EL**) like Cross Entropy as the loss function. However, Softmax essentially assumes a mutually exclusive relationship between the to-be-classified classes: a pixel has to belong to one of the classes of interest. Thus only the relative strength of logits, i.e., the ratio of logits, matters for posterior probability calculation. However, when applying the model to unseen classes, the class space will be different from the training scenario, making the logit of an unseen class poorly calibrated with the other unseen classes.

To handle this issue, we suggest avoiding the mutual-exclusive mechanism during training time and using Non-mutually Exclusive Loss (**NEL**), more specifically,

Sigmoid and Binary Cross Entropy (BCE) loss to ensure the segmentation result for each class is independently generated. In addition, we use the focal loss [104] variation of the BCE loss and combine it with an extra dice loss [129, 231] as previous work [26]:

$$\mathcal{L}_{\texttt{focal}} = -\frac{1}{\text{hw}}\sum_{i=1}^{\text{hw}}(1-y_i)^{\gamma}\times\hat{y}\log(y_i) + y_i^{\gamma}\times(1-\hat{y}_i)\log(1-y_i), \tag{4.4}$$

$$\mathcal{L}_{\texttt{dice}} = 1 - \frac{2\sum_{i=1}^{\text{hw}} y_i\hat{y}_i}{\sum_{i=1}^{\text{hw}} y_i^2 + \sum_{i=1}^{\text{hw}}\hat{y}_i^2}, \tag{4.5}$$

$$\mathcal{L} = \alpha\cdot\mathcal{L}_{\texttt{focal}} + \beta\cdot\mathcal{L}_{\texttt{dice}}, \tag{4.6}$$

where $\gamma = 2$ balances hard and easy samples and $\{\alpha, \beta\}$ are coefficients to combine focal loss and dice loss. As shown in Figure 4.1-(4), Table 4.6 and Table 4.7, compared with CE, BCE performs better on unseen classes, and the focal loss and dice loss on BCE [26] boost the performance furthermore as demonstrated in Sub-Section 4.4.6-C.

4.3.5 Design 3: Relationship Descriptor (RD)

In the above design, the class embeddings extracted from the CLIP text encoder will match against the patch embeddings from the CLIP image encoder in the decoder head. While being quite intuitive, we find this design could lead to severe overfitting. We postulate that this is because the matching capability between the text query and image patterns is only trained on the seen-class datasets.

This motivates us to incorporate the matching capability learned from the original CLIP training into the transformer decoder. Specifically, we noticed that CLIP calculates the matching score between a text prompt and an image by

$$\mathbf{t}^{c\top}\mathbf{g} = \sum_{j}^{d} t_j^c g_j = \sum_{j}^{d} r_j^c, \tag{4.7}$$

where $\mathbf{t}^c \in \mathbb{R}^d$ denotes a text embedding for the cth class and t_j is its j-th dimension; $\mathbf{g}$ is the image embedding ([cls] token); $r_j^c = t_j^c g_j$. We postulate that $\mathbf{r}^c = [r_1^c, r_2^c, \cdots, r_d^c]$ characterizes how the image and text, i.e., the text prompt for representing class c, are matched, and call it text-image **Relationship Descriptor (RD)** denoted as $\mathbf{R} \in \mathbb{R}^{C\times d}$ for all C classes in this work. Then we concatenate RDs with the original text embedding $\mathbf{T} \in \mathbb{R}^{C\times d}$ as image-specific text queries $\hat{\mathbf{T}} = \{\hat{\mathbf{t}}^1, \hat{\mathbf{t}}^2, ..., \hat{\mathbf{t}}^C\} \in \mathbb{R}^{C\times 2d}$ for the transformer decoder. Specifically, with this scheme, the input text query of transformer decoder for each class becomes:

$$\hat{\mathbf{t}} = \text{concat}[\mathbf{r}, \mathbf{t}] = \text{concat}[\mathbf{t}\odot\mathbf{g}, \mathbf{t}], \tag{4.8}$$

where $\odot$ is the Hadamard product. Note that we apply learnable linear projection layers on both $\hat{\mathbf{T}}$ and $\mathbf{H}$ to make them have the same feature dimension.

We compare the effectiveness of applying the relationship description shown in Figure 4.1-(5) as well as in Table 4.6 and Table 4.7. As seen, it can dramatically improve the segmentation results on both seen and unseen categories. In SubSection 4.4.6-A, we further discuss the effect of using different combination formats of element-wise operations for text-image matching in the relationship descriptor and text queries $\hat{\mathbf{T}}$.

4.4 Experiments

4.4.1 Datasets

To evaluate the effectiveness of our proposed method, we conducted extensive experiments on three public benchmark datasets, including PASCAL VOC 2012, COCO-Stuff 164K, and PASCAL Context. The unseen classes of each dataset are shown in the Appendix A according to previous works [36, 59, 207, 232]. The details of the datasets are elaborated as follows:

PASCAL VOC 2012 contains 10,582 augmented images for training and 1,449 for validation. We also ignore the "background" category and use 15 classes as the seen part and 5 classes as the unseen part.

COCO-Stuff 164K is a large-scale dataset that contains 171 categories with 118,287 images for training and 5,000 for testing. The whole dataset is divided into 156 seen classes and 15 unseen classes.

PASCAL Context includes 60 classes with 4,996 for training and 5,104 for testing. The dataset is divided into 50 known classes (including "background") and the rest 10 classes as used as unseen classes in the test set.

4.4.2 Evaluation Metrics

Following previous works, we measure pixel-wise classification accuracy (pAcc) and the mean of class-wise intersection over union (mIoU) on both seen and unseen classes, denoted as $mIoU(S)$ and $mIoU(U)$, respectively. We also evaluate the harmonic mean IoU ($hIoU$) among seen and unseen classes, which is formulated as:

$$hIoU = \frac{2 * mIoU(S) * mIoU(U)}{mIoU(S) + mIoU(U)}, \tag{4.9}$$

4.4.3 Implementation Details

Our proposed method is implemented based on the open-source toolbox MMSegmentation[30] with PyTorch 1.10.1. All experiments we provided are based on the pre-trained CLIP ViT-B/16 model and conducted on 4 Tesla V100 GPUs, and the batch size is set to 16 with 512x512 as the resolution of images. For "inductive" zero-shot learning, the total training iterations are 20K for PASCAL VOC 2012, 40K for PASCAL Context, and 80K for COCO-Stuff 164K. In the "transductive" setting, we train our ZegCLIP model on seen classes in the first half of training iterations and then apply self-training via generating pseudo labels in the rest of iterations. The optimizer is set to AdamW with the default training schedule in the MMSeg toolbox.

4.4.4 Comparison with State-of-the-art methods

To demonstrate the effectiveness of our method, the evaluation results compared with previous state-of-the-art methods are reported in Table 4.2. We also provide the fully supervised learning results as the upper bound to show the performance gap between fully-supervised segmentation and zero-shot segmentation results on unseen classes. The qualitative results on COCO-Stuff 164K are shown in Figure 4.5 and more visualization results are provided in Appendix C. The detailed difference between previous two-stage and one-stage are demonstrated in Appendix D.

TABLE 4.2. Comparison with the state-of-the-art methods on PASCAL VOC 2012, COCO-Stuff 164K, and PASCAL Context datasets. "ST" represents applying self-training via generating pseudo labels on all unlabeled pixels, while "*"+"ST" denotes that pseudo labels are merely annotated on unseen pixels excluding the ignore part.

Methods	PASCAL VOC 2012				COCO-Stuff 164K				PASCAL Context			
	pAcc	**mIoU(S)**	**mIoU(U)**	**hIoU**	**pAcc**	**mIoU(S)**	**mIoU(U)**	**hIoU**	**pAcc**	**mIoU(S)**	**mIoU(U)**	**hIoU**
Inductive												
SPNet[197]	-	78.0	15.6	26.1	-	35.2	8.7	14.0	-	-	-	-
ZS3[10]	-	77.3	17.7	28.7	-	34.7	9.5	15.0	52.8	20.8	12.7	15.8
CaGNet[59]	80.7	78.4	26.6	39.7	56.6	33.5	12.2	18.2	-	24.1	18.5	21.2
SIGN[27]	-	75.4	28.9	41.7	-	32.3	15.5	20.9	-	-	-	-
Joint[3]	-	77.7	32.5	45.9	-	-	-	-	-	33.0	14.9	20.5
ZegFormer[36]	-	86.4	63.6	73.3	-	36.6	33.2	34.8	-	-	-	-
zsseg[207]	90.0	83.5	72.5	77.5	60.3	39.3	36.3	37.8	-	-	-	-
ZegCLIP (Ours)	**94.6**	**91.9**	**77.8**	**84.3**	**62.0**	**40.2**	**41.4**	**40.8**	**76.2**	**46.0**	**54.6**	**49.9**
Transductive												
SPNet+ST[197]	-	77.8	25.8	38.8	-	34.6	26.9	30.3	-	-	-	-
ZS5 [10]	-	78.0	21.2	33.3	-	34.9	10.6	16.2	49.5	27.0	20.7	23.4
CaGNet+ST[59]	81.6	78.6	30.3	43.7	56.8	35.6	13.4	19.5	-	-	-	-
STRICT[146]	-	82.7	35.6	49.8	-	35.3	30.3	34.8	-	-	-	-
zsseg+ST[207]	88.7	79.2	78.1	79.3	63.8	39.6	43.6	41.5	-	-	-	-
ZegCLIP+ST (Ours)	**95.1**	**91.8**	**82.2**	**86.7**	**68.8**	**40.6**	**54.8**	**46.6**	**77.2**	**46.6**	**65.4**	**54.4**
*MaskCLIP+[232]	-	88.8	86.1	87.4	-	38.1	54.7	45.0	-	44.4	66.7	53.3
***ZegCLIP+ST (Ours)**	**96.2**	**92.3**	**89.9**	**91.1**	**69.2**	**40.7**	**59.9**	**48.5**	**77.3**	**46.8**	**68.5**	**55.6**
Fully Supervised												
ZegCLIP (Ours)	**96.3**	**92.4**	**90.9**	**91.6**	**69.9**	**40.7**	**63.2**	**49.6**	**77.5**	**46.5**	**78.7**	**56.9**

TABLE 4.3. Efficiency comparison with different metrics. All models are evaluated on a single 1080Ti GPU. #Params represents the number of learnable parameters in the whole framework.

Datasets	Methods	#Params(M) ↓	Flops(G) ↓	FPS ↑
VOC	ZegFormer[36]	60.3	1829.3	1.7
	ZegCLIP	**13.8**	**110.4**	**9.0**
COCO	ZegFormer[36]	60.3	1875.1	1.5
	ZegCLIP	**14.6**	**123.9**	**6.7**

From Table 4.2, we can see that our proposed method achieves significant performance under the "inductive" setting and outperforms previous works, especially for unseen classes. This clearly demonstrates the superior generalization capability of our method over the previous approaches. In addition, we also find our approach excels in the "transductive" setting, although our method is not specifically designed for that setting. As seen, our model dramatically improves the performance on unseen classes while maintaining excellent performance on seen part after self-training.

Figure 4.5 shows the segmentation results of the Baseline-Fix version and our proposed ZegCLIP on seen and unseen classes. After applying our designs, ZegCLIP shows impressive segmentation ability on both seen and unseen classes and can clearly distinguish similar unseen categories, for example, the unseen "tree", "grass" and "playingfield" categories in (1).

In addition, to demonstrate the efficiency of our proposed method, we compare the number of learnable parameters and inference speed between our one-stage ZegCLIP and typical two-stage method Zegformer[36] in Table 4.3. Our proposed method achieves significant performance while only requiring approximately 14M learnable parameters with 117 Flops(G) on average which is only **23%** and **6%** of [36], respectively. In terms of frames Per Second (FPS), our method can achieve a speedup of about **5 times** faster during inference compared with two-stage method.

4.4.5 Transfer Zero-shot model to Open-Vocabulary Semantic Segmentation

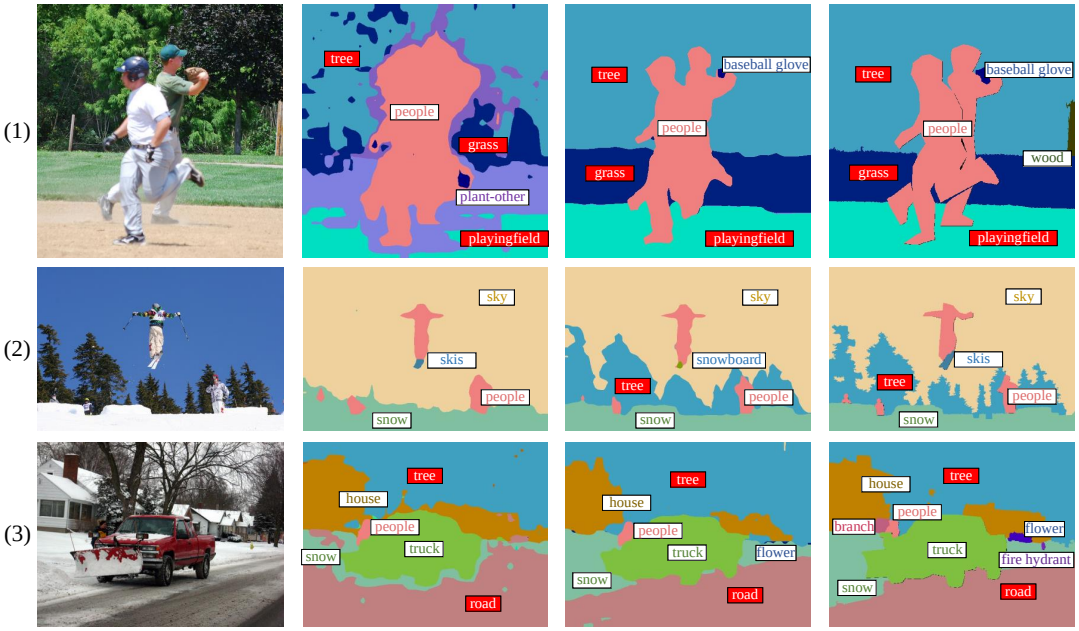
To further demonstrate the effectiveness of our method, we also conduct our zero-shot method in a similar open-vocabulary setting where the model is pre-trained on COCO-Stuff with 171 classes and then conducted extensive experiments on five public benchmark datasets, and the details are elaborated as follows:

PASCAL VOC 2012 contains 10,582 augmented images for training and 1,449 for validation. Following previous works [52, 102], we also ignore the "background" category.

PASCAL Context-59 includes 59 classes (exclude "background" category) with 4,996 images for training and 5,104 for testing.

PASCAL Context-459 is a challenging dataset for open-vocabulary segmentation which has the same training and testing images as PASCAL Context-59 but annotated with fine-grained 459 classes.

ADE20K-150 is also widely used in open-vocabulary that contains 150 categories with 20,210 images for training and 2,000 for testing.



(a) Image (b) Baseline-FT (c) Ours (ZegCLIP) (d) Ground truth

FIGURE 4.5. Qualitative results on COCO-Stuff 164K. (a) are the original testing images; (b) represent the performance of our proposed one-stage baseline (fine-tuning the image encoder); (c) are the visualization results of our proposed ZegCLIP; (d) are the ground truths of each image. Note that the white and red tags represent seen and unseen classes separately.

ADE20K-847 is the most challenging open-vocabulary dataset that has the same images as ADE20K-150K but has 847 annotated categories.

We apply the pre-trained model of the source dataset via supervised learning on seen classes and evaluate the segmentation results on both seen and unseen classes of target datasets. Our method shows better cross-domain generalization capability compared with previous methods. From Table 4.4 we can see that our proposed method can also achieve impressive improvement. We also provide the visualization results on the most challenging open-vocabulary dataset ADE20k with 847 classes in Figures 4.6.

TABLE 4.4. Comparison with the state-of-the-art open-vocabulary methods in terms of mIoU.

Method	VLM	Training Dataset	ensemble?	ADE-847	PC-459	ADE-150	PC-59	VOC-20
GroupViT [203]	random initial	CC12M+YFCC	✗	-	-	-	22.4	52.3
SegCLIP [124]	random initial	CC12M+COCO	✗	-	-	-	19.1	33.3
LSeg+ [93]	ALIGN RN101	COCO Panoptic	✗	2.5	5.2	13.0	36.0	59.0
OpenSeg [52]	ALIGN RN101	COCO Panoptic	✗	4.0	6.5	15.3	36.9	60.0
LSeg+ [93]	ALIGN EN-B7	COCO Panoptic	✗	3.8	7.8	18.0	46.5	-
OpenSeg [52]	ALIGN EN-B7	COCO Panoptic	✗	6.3	9.0	21.1	42.1	-
SegCLIP [124]	CLIP ViT-B/16	CC12M+COCO	✗	-	-	-	24.7	52.6
SimSeg [206]	CLIP ViT-B/16	COCO Stuff 171	✓	6.9	9.7	21.1	51.9	91.8
OvSeg [102]	CLIP ViT-B/16	COCO Stuff 171	✓	7.1	11.0	24.8	53.3	92.6
ZegCLIP	CLIP ViT-B/16	COCO Stuff 171	✗	6.0	10.1	21.7	52.4	97.2
ZegCLIP	CLIP ViT-B/16	COCO Stuff 171	✓	8.1	12.4	23.9	54.8	97.4

4.4.6 Ablation Study

A. Effect of different formats of text query $\hat{\mathbf{t}}$

In this work, we propose an important design called Relationship Descriptor (RD) as described in SubSection 4.3.5. In this module, we combine the queried text embeddings $\mathbf{T}$ with image [cls] token $\mathbf{g}$ extract from CLIP to generate text-image description $\hat{\mathbf{T}}$ before feeding the queries into the segment decoder. Such image-specific queries can dramatically improve the zero-shot segmentation performance on unseen classes. To further explore the effect of different element-wise operations in the relationship descriptor, we conduct various formats of $\hat{\mathbf{T}}$ on VOC and report the results in Table 4.5. For the image-specific text query of each class $\hat{\mathbf{t}}$, it can be formulated as shown in the second column.

We can see, the dot product and the absolute difference between text embedding $\mathbf{t}$ and [cls] token $\mathbf{g}$ can provide more general information, but the sum and concatenate operations perform poorly on both seen and unseen classes. This is understandable since dot product and absolute difference characterize the relationship between the image and text encoding, which is in line with the interpretation made in SubSection 4.3.5. The red line in Table 4.5 is the format of $\hat{\mathbf{t}}$ we finally chose in our ZegCLIP model.

TABLE 4.5. Effect of different formats of text queries $\hat{\mathbf{t}}$.

dim	format of $\hat{\mathbf{t}}$	pAcc	mIoU(S)	mIoU(U)	hIoU
512	**t**	86.8	89.5	33.7	49.0
	t⊙g	93.1	90.2	68.4	77.8
	|t-g|	92.4	90.6	64.2	75.1
	t-g	88.7	87.9	46.5	60.8
	t+g	82.2	89.9	13.9	24.1
512*2	**[t, g]**	88.9	88.8	39.3	54.5
	[t⊙g, t]	**94.6**	**91.9**	**77.8**	**84.3**
	[|t-g|, t]	90.9	91.5	54.2	68.1
	[t⊙g, t+g]	88.3	90.0	38.0	53.4
	[t+g, t]	82.8	89.4	20.7	33.6
	[t⊙g, |t-g|]	94.1	91.2	73.9	81.6
512*3	**[t⊙g, |t-g|, t]**	93.4	91.6	67.3	77.6

B. Detailed results of applying designs on baseline

To demonstrate the effectiveness of our proposed designs, we further report the improvements of applying designs on our Baseline model in Table 4.6 and Table 4.7. It shows that fixed CLIP (Baseline-Fix) with a learnable segment decoder fails to achieve satisfactory performance on both seen and unseen classes due to weak image representation for the dense prediction task. Meanwhile, fine-tuning CLIP (Baseline-FT) performs better on seen classes but deteriorates on unseen classes dramatically. It seems that the zero-shot transfer learning ability of CLIP is destroyed after updating on seen classes.

Changing objective functions from mutual exclusive into **Non-mutually Exclusive Loss (NEL, Design 2)** also leads to significant performance boost both in seen

TABLE 4.6. Quantitative results on PASCAL VOC 2012 dataset to demonstrate the effectiveness of our proposed three designs.

method	PASCAL VOC 2012			
	pAcc	mIoU(S)	mIoU(U)	hIoU
Baseline-Fix	69.3	71.1	16.3	26.5
Baseline-Fix + NEL	85.5	85.2	36.6	51.2
Baseline-Fix + RD	86.0	82.5	46.6	59.6
Baseline-Fix + NEL + RD	89.6	83.3	66.4	73.9
Baseline-FT	77.3	76.5	13.8	23.4
Baseline-FT + NEL	83.8	84.1	27.5	41.4
Baseline-FT + RD	79.4	77.8	20.7	32.7
Baseline-FT + NEL + RD	89.6	90.2	42.4	57.7
Baseline-DPT	76.2	75.9	28.3	41.2
Baseline-DPT + NEL	89.2	89.9	40.4	55.7
Baseline-DPT + RD	85.5	81.0	55.2	65.7
Baseline-DPT + NEL + RD (ZegCLIP)	**94.6**	**91.9**	**77.8**	**84.3**

and unseen scenarios. The most dramatic improvement comes from **Relationship Descriptor (RD, Design 3)** which almost doubles the unseen performance in many cases.

Finally, we noticed that **Deep Prompt Tuning (DPT, Design 1)** works well with NEL (Design 2) and RD (Design 3). If we replace DPT by fine-tuning CLIP and combine it with NEL and RD, the issue of overfitting seen classes remain.

In conclusion, our proposed method can ensure semantic segmentation performance while maintaining the powerful ability of CLIP and achieving excellent performance on seen and unseen classes simultaneously.

C. Effect of advanced loss function

As we described in Subsec 4.3.4, we propose that applying non-mutually exclusive objective functions like using binary cross entropy (BCE) with sigmoid performs better on zero-shot semantic segmentation. To better handle the imbalance problem among categories, we further combine BCE with focal loss[104] and dice loss[98] according to [26, 218] as Equation 4.6, and the results of applying such two versions of the loss function, denoted as "plain" and "plus", are reported separately in Table 4.8. We can see that the "plus" loss function achieves better performance on both seen and unseen classes on VOC, COCO, and Context datasets.

D. Generalization ability to other datasets

To further explore the generalization ability of our proposed method, we conduct extra experiments in Table 4.9. We apply the pre-trained model of the source dataset via supervised learning on seen classes and evaluate the segmentation results on both seen and unseen classes of target datasets. Our method shows better cross-domain generalization capability compared with the latest related work Zegformer which is also based on the CLIP model.

TABLE 4.7. Quantitative results on COCO dataset to demonstrate the effectiveness of our proposed three designs.

method	COCO-Stuff 164K			
	pAcc	mIoU(S)	mIoU(U)	hIoU
Baseline-Fix	33.3	17.1	15.4	16.2
Baseline-Fix + NEL	52.4	31.7	20.8	25.1
Baseline-Fix + RD	41.0	23.3	23.4	23.3
Baseline-Fix + NEL + RD	53.7	32.3	32.5	32.4
Baseline-FT	48.4	32.4	17.5	22.7
Baseline-FT + NEL	56.5	39.9	25.4	31.0
Baseline-FT + RD	54.0	39.6	22.4	28.6
Baseline-FT + NEL + RD	60.2	**42.7**	22.3	29.3
Baseline-DPT	39.0	22.5	17.5	19.7
Baseline-DPT + NEL	58.5	38.0	27.4	31.8
Baseline-DPT + RD	46.4	28.4	27.8	28.1
Baseline-DPT + NEL + RD (ZegCLIP)	**62.0**	40.2	**41.4**	**40.8**

TABLE 4.8. Comparison of introducing advanced loss function. Note that "plain" represents merely Binary Cross Entropy (BCE), while "plus" means adding focal loss on BCE and dice loss

dataset	loss	pAcc	mIoU(S)	mIoU(U)	hIoU
VOC	plain	93.4	89.7	73.6	80.9
	plus	**94.6**	**91.9**	**77.8**	**84.3**
COCO	plain	59.8	38.8	39.0	38.9
	plus	**62.0**	**40.2**	**41.4**	**40.8**
Context	plain	75.3	43.5	50.0	46.5
	plus	**76.2**	**46.0**	**54.6**	**49.9**

E. Effect of the number of deep prompt tokens

We figure out that the best number of deep prompt tokens varies for different datasets and the detailed results are shown in Figure 4.8. For the PASCAL VOC 2012 dataset (VOC) which contains fewer training samples and categories, 10 tokens are enough to obtain significant performance on both seen and unseen classes. However, for large-scale datasets, more deep prompts, i.e., 100 for COCO-Stuff 164K (COCO) and 35 for PASCAL Context (Context), are beneficial to achieve better segmentation performance. In general, the best number of deep prompt tokens increases with the scale of the dataset and the complexity of the per-pixel classification task increases. Meanwhile, using too many visual prompts may be detrimental to our model instead.

F. Effect of the depth of deep prompt tokens

Except that the number of deep prompt tokens can impact the performance of zero-shot semantic segmentation, we also conduct extensive experiments on PASCAL VOC 2012 (VOC) to explore the effect of inserting the learnable prompts in different layers of the CLIP image encoder. The quantitative results are reported in Table 4.10. For a better explanation, we number the total 12 vision transformer layers in the CLIP image encoder from 1 ("bottom") to 12 ("top") and the layers of

TABLE 4.9. Generalization ability to other datasets.

source	target	method	pAcc	mIoU	mAcc
COCO	Context	Zegformer[36]	56.8	36.1	64.0
		ZegCLIP	60.9	41.2	68.4
		***ZegCLIP+ST**	**68.4**	**45.8**	**70.9**
	VOC	Zegformer[36]	92.8	85.6	92.7
		ZegCLIP	96.9	93.6	96.4
		***ZegCLIP+ST**	**97.2**	**94.1**	**96.7**

TABLE 4.10. Effect of the depth of deep prompt tuning on VOC.

layer	pAcc	mIoU(S)	mIoU(U)	hIoU
1	91.4	87.5	67.8	76.4
1→3	91.7	86.7	70.2	77.6
1→6	92.7	87.8	75.3	81.1
1→9	93.3	88.9	72.4	79.8
1→12	**94.6**	**91.9**	**77.8**	**84.3**
10→12	92.5	88.3	70.9	78.6
7→12	92.5	89.0	68.0	77.1
4→12	93.6	91.5	66.9	77.3

inserting prompts are as denoted in the first column of Table 4.10. We figure out that adding prompt tokens on “bottom" layers generally tends to perform better than on “top" layers. Meanwhile, inserting learnable prompt tokens in each ViT layer (layer=1→12) achieves the best performance which is also the default setting in our experiments.

G. Effect of single and multiple text templates

Following the training details of CLIP, we apply a single template “*A photo of a* {}" on PASCAL VOC 2012 (VOC) and multiple templates (more details provided in Appendix B) on large-scale datasets, i.e., COCO-Stuff 164K (COCO) and PASCAL Context (Context), when obtaining the class embeddings from CLIP text encoder. We provide the quantitative results of using single and multiple templates in Table 4.11 where we can see that multiple descriptions achieve reasonable improvements on both two datasets.

TABLE 4.11. Comparison of using single and multiple templates on COCO-Stuff 164K and PASCAL Context datasets.

dataset	template	pAcc	mIoU(S)	mIoU(U)	hIoU
COCO	single	61.4	39.5	40.6	40.0
	multiple	**62.0**	**40.2**	**41.4**	**40.8**
Context	single	75.8	45.1	52.1	48.3
	multiple	**76.2**	**46.0**	**54.6**	**49.9**

4.5 Conclusion

In this work, we propose an efficient one-stage straightforward zero-shot semantic segmentation method based on the pre-trained vision-language CLIP. To transfer the image-wise classification ability to dense prediction tasks while maintaining the advanced zero-shot knowledge, we figure out three designs to achieve competitive results on seen classes while extremely improving the performance on novel classes. Our proposed method relay on text embeddings as queries that are very flexible to cope with both "inductive" and "transductive" zero-shot settings. To demonstrate the effectiveness of our method, we conduct extensive performance on three public benchmark datasets and outperform previous state-of-the-art methods. Meanwhile, our one-stage framework shows about 5 times faster compared with two-stage methods while inference. In general, our work explores how to leverage the pre-trained vision-language model CLIP into semantic segmentation and successfully utilize its zero-shot knowledge in downstream tasks which may provide inspiration for future research.



FIGURE 4.6. Visualization results of open-vocabulary segmentation on the most challenging dataset ADE20k with 847 categories.

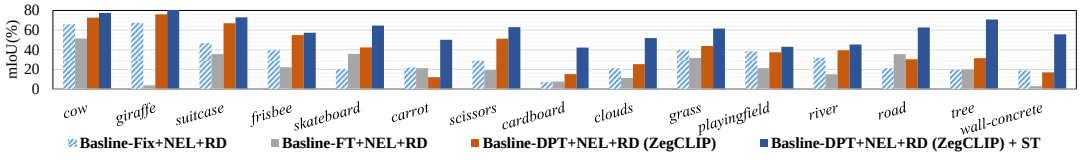


FIGURE 4.7. Detailed performance on unseen classes of COCO datasets. Note that "ST" represents self-training in "transductive" setting.

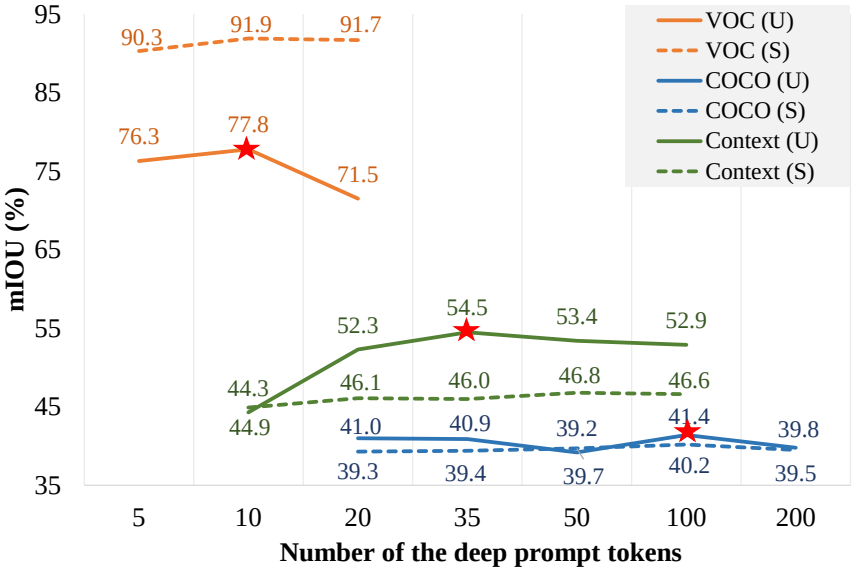


FIGURE 4.8. The quantitative results of applying the different numbers of deep prompt tokens. Note that "S" and "U" represent seen and unseen classes separately.

Statement of Authorship

Title of Paper	ZegCLIP: Towards Adapting CLIP for Zero-shot Semantic Segmentation
Publication Status	[X] Published [ ] Accepted for Publication [ ] Submitted for Publication [ ] Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Accepted and published in CVPR 2023.

Principal Author

Name of Principal Author (Candidate)	Ziqin Zhou
Contribution to the Paper	Developed the conceptual framework, completed the coding, conducted experiments, and wrote the manuscript for the paper.
Overall percentage (%)	70%
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.
Signature	Date: 25.07.2024

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Lingqiao Liu, Yifan Liu
Contribution to the Paper	Idea discussions and contributed to writing revisions.
Signature	Date: 25.07.2024

Name of Co-Author	Bowen Zhang, Yinjie Lei
Contribution to the Paper	Idea discussions.
Signature	Date: 25.07.2024

Please cut and paste additional co-author panels here as required.

Chapter 5

Unlocking the Potential of Pre-trained Vision Transformers for Few-Shot Semantic Segmentation through Relationship Descriptors

5.1 Introduction

The emergence of pre-trained vision transformers constitutes a pivotal advancement within the computer vision field. The pioneering model ViT [39] demonstrated competitive performance on various image classification tasks. Subsequent studies [11, 64, 142] introduced different pre-training tasks to improve the perception ability further and widely applied on various downstream tasks. Unlike pre-trained ResNet models [66] which are based on conventional convolutional neural networks, vision transformers employ self-attention mechanisms to model global dependencies in images, affording them the capacity to hierarchically group semantically related visual concepts [11]. This means that they can learn to recognize and understand complex visual patterns by associating meaningful concepts within the image and enabling the potential for dense semantic prediction. As shown in Figure 5.1, we visualize that without further learning, the pre-trained transformer already has the capability of group patches with similar semantic meanings together, which could be a valuable property for segmentation tasks.

Few-shot semantic segmentation task aims to achieve precise segmentation of novel classes with a limited number of examples. A common paradigm involves utilizing a dedicated training dataset to acquire this capability. A pivotal obstacle in this learning process lies in ensuring that the few-shot segmentation capability extends effectively to classes that were not encountered in the training set. Utilizing pre-trained models, specifically pre-trained vision transformers, presents a promising avenue for mitigating this challenge. This is due to the fact that these pre-trained models are typically trained on large-scale datasets, encompassing a diverse array of visual concepts, even though their training is not specifically tailored for segmentation tasks. Consequently, one might anticipate that by employing a segmentation training dataset consisting of some base classes, it becomes possible for the few-shot segmentation capability to generalize effectively to classes that were not included in the training set.

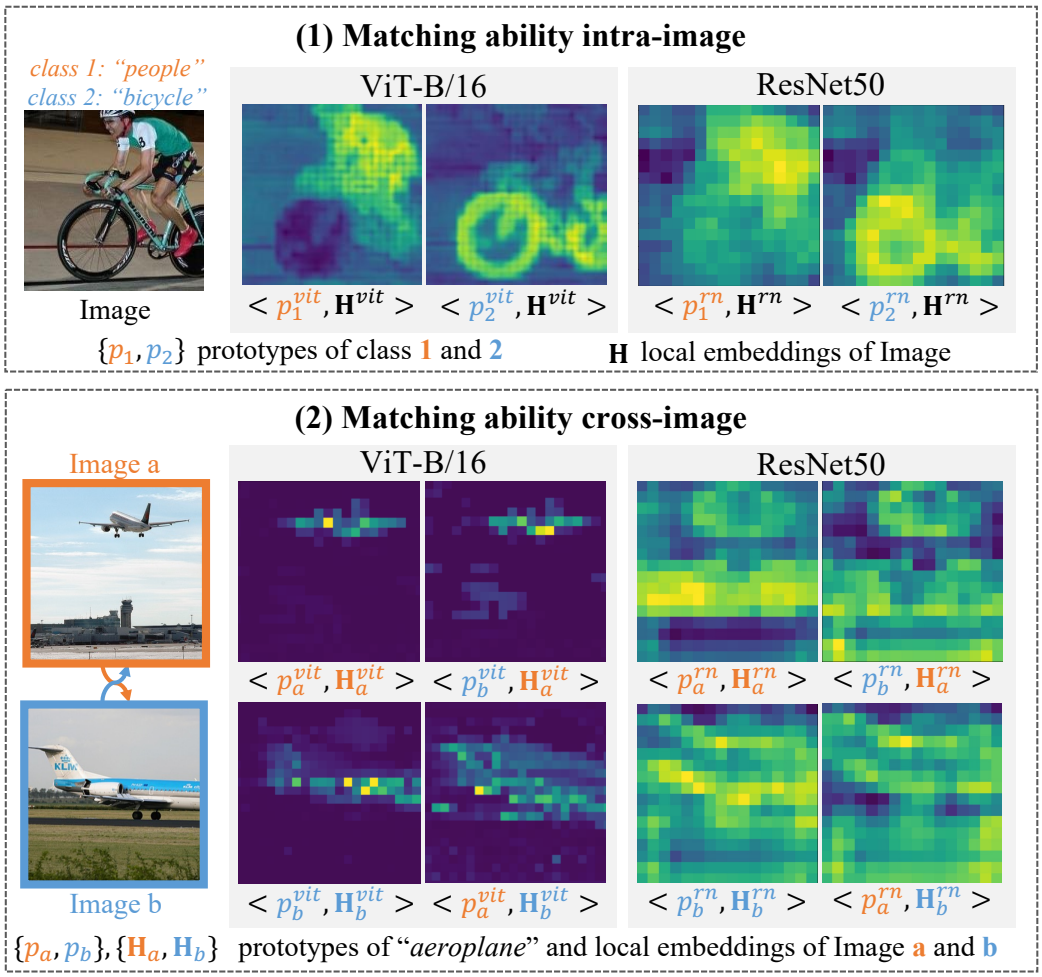


FIGURE 5.1. Visualized hidden (1) intra- and (2) cross- image matching capability of pre-trained ViT-B/16 and ResNet50 based on ImageNet. $< p, \mathbf{H} >$ denotes the cosine similarity between class-wise prototypes p and local feature $\mathbf{H}$ of Image. Prototypes p represent the average of embeddings belonging to the object-of-interest for each image, for example, the "people" and "bicycle" object of the Image in (1). It illuminates that the pre-trained transformer model has certain semantic grouping capabilities which may be helpful for few-shot semantic segmentation.

Nonetheless, fine-tuning a pre-trained transformer directly with a decoder for segmentation often results in significant overfitting to the base classes. This occurs because the parameters introduced in the decoder, initially designed to produce optimal segmentation results for base classes, tend to be solely optimized for base class segmentation performance. This can lead to valuable knowledge about unseen classes learned from the pre-trained transformer being overlooked during the training process. To mitigate this issue, our approach draws inspiration from the recently introduced concept of a relationship descriptor (RD), originally developed for extending the CLIP model for zero-shot semantic segmentation. We adapt and refine the design of the relationship descriptor construction to tailor it specifically for few-shot semantic segmentation. Furthermore, we extend the construction of the RD to

incorporate information from multiple layers. Building upon these adaptations, we propose a few-shot semantic segmentation framework that is not only easy to implement and train but also exhibits effective scalability with the number of support examples and categories.

We carried out extensive experiments on two well-known datasets, namely PASCAL-5^i and COCO-20^i, to evaluate the effectiveness of our proposed method. The results demonstrate that our method successfully tackles the issue of overfitting to base classes, and exhibits superior performance compared to state-of-the-art methods in various settings: generalized (GFSS) and binary (FSS) few-shot semantic segmentation.

5.2 Background

5.2.1 Few-shot Semantic Segmentation

Few-shot Semantic Segmentation task aims to segment the object-of-interest in a query image using only a few user-provided examples, where each example consists of an image paired with its corresponding object mask. The ability to perform few-shot segmentation is typically learned from a training set with well-annotated examples from a set of "base" classes, denoted as $\mathcal{C}^B$. The challenge of few-shot segmentation is how to ensure the model learned from "base" classes can be generalized to the novel classes during the test time. In the literature, there are two primary settings for few-shot segmentation:

Binary Few-shot Semantic Segmentation (FSS): This conventional setting evaluates the model exclusively on novel classes ($\mathcal{C}^N$, where $\mathcal{C}^N \cap \mathcal{C}^B = \emptyset$) and often focuses on object-background segmentation. The object mask only contains binary values indicating the object's presence.

Generalized Few-shot Semantic Segmentation (GFSS): The generalized setting extends beyond the standard framework to better mirror real-world scenarios, necessitating the recognition of both base and novel classes ($\mathcal{C}^B \cup \mathcal{C}^N$) as well as background within the same image. Also, this setting considers multiple object classes in the support examples, which requires the FSS method to scale well with the number of object categories.

5.2.2 Relationship Descriptor (RD) in ZegCLIP [234]

Recently, researchers have shown interest in adapting the image-level zero-shot classification ability of CLIP [152] model to the per-pixel prediction task, e.g., zero-shot semantic segmentation [36, 207]. One work ZegCLIP [234] proposed a novel relationship descriptor module (RD) to sufficiently utilize the text-image matching capability learned during the CLIP pretraining stage and effectively alleviate the base class overfitting problem. Specifically, the RD is the element-wise product between the image encoder embedding from the [CLS] token and the text embedding from the text encoder. Essentially, such a mechanism encodes the dimension-wise contribution to the image-text matching score calculated from a pre-trained CLIP model. It's important to note that the RD calculation takes place before applying any transformations to the text and image embeddings using newly introduced decoder parameters.

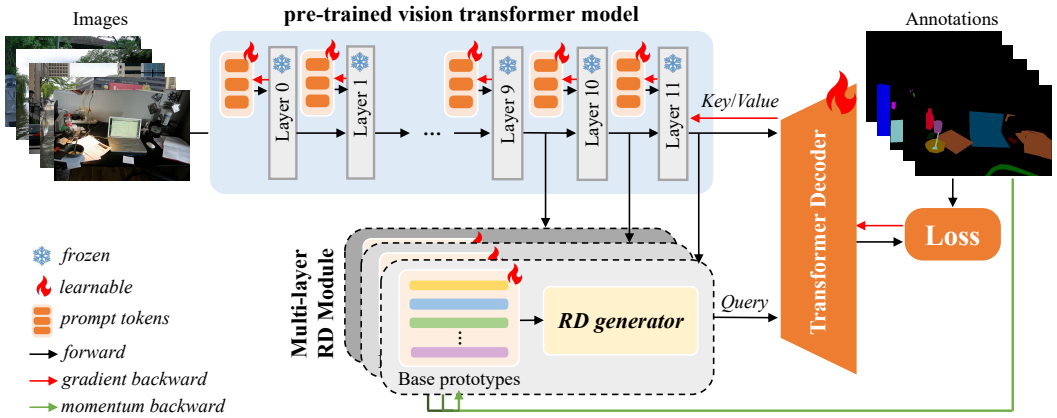


FIGURE 5.2. Overall training framework of our proposed few-shot semantic segmentation method.

This ensures that the matching capability of these embeddings is preserved, even for classes that are not part of the base classes. Therefore, using RDs can potentially alleviate the issue of overfitting to the based classes.

Our work is motivated by its effectiveness of leveraging the inherent matching capability of CLIP to alleviate the overfiting-to-base-class issue, despite our method not trying to leverage any text-image matching capability and needs a further adaption and extension of the idea of RD for few-shot segmentation.

5.2.3 Semantic Grouping Capability in Pre-trained Vision Transformers

Akin to the text-image matching capability of CLIP, the inherent capability we are interested in utilizing is the semantic grouping capability in a pre-trained vision transformer. This capability is well demonstrated in Figure 5.1. We can see that even before any fine-tuning, the patch embedding extracted from a pew-trained visual transformer is similar to patches sharing the same semantic concept, i.e., the "person" riding bicycle or the "bicycle". In particular, we find this semantic grouping capability is quite prominent for vision transformers than CNNs. This motivates us to explore pre-trained vision transformers for few-shot segmentation tasks. The key research problem in this work is thus how to extend the idea of RD to fully leverage this semantic grouping capability to improve generalization.

5.3 Method

5.3.1 Few-shot Segmentation via Prototype Embeddings and Relationship Descriptors

Without loss of generality, we can express a Few-shot Semantic Segmentation model in the form of $F(\mathbf{I}_q, \mathcal{S})$, where $\mathbf{I}_q$ represents the query image and $\mathcal{S}$ denotes the support set examples. To make the few-shot segmentation model scalable to the number

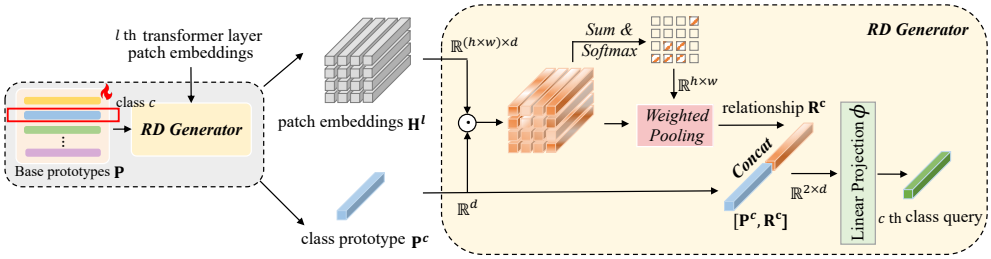


FIGURE 5.3. The detailed design of our Relationship Descriptor (RD) Generator module.

of the classes and support set examples, recent work [181] advocates representing each support example as a prototype embedding vector, with one vector per object category. Specifically for one support image, we calculate the prototype embedding for class c as:

$$\mathbf{p}^c = \frac{1}{\sum_{i,j} \mathbf{M}^c[i,j]} \sum_{i,j} \mathbf{M}^c[i,j]\mathbf{H}_{i,j}, \tag{5.1}$$

where $\mathbf{H}_{i,j} \in \mathbb{R}^d$ is the patch embedding extracted from a pre-trained vision transformer and located at the (i,j) grid of the support image. $\mathbf{M}^c[i,j]$ is the value at the corresponding mask for object category c. It can be either "1" or "0", indicating foreground and background respectively. When there is more than one support example, we simply average the prototype across all examples. Once support examples are represented by $\mathbf{P} = [\mathbf{p}^1, \mathbf{p}^2, \cdots, \mathbf{p}^c]$, predicting semantic masks is performed by using a decoder $F(\mathbf{I}_q, \mathbf{P})$.

A straightforward idea to design $F(\mathbf{I}_q, \mathbf{P})$ is to use a matching function to compare $\mathbf{p}^c$ against patch embedding $\mathbf{H}_{i,j}$, producing how likely patch (i,j) is compatible with object category c and thus achieving semantic segmentation. However, this naive idea often leads to overfitting the base classes in the training set.

As mentioned in [234], the matching function inevitably introduces new parameters for achieving good matching quality[1]. However, those newly introduced parameters are optimized solely for the existing base classes, potentially disregarding valuable knowledge gained during the earlier transformer pretraining stage that could be beneficial for novel classes. In our experiments, we indeed observe this issue if we directly use prototypes to match patch embedding as shown in Table 5.3 and Table 5.4. Thus motivated by [234], we use an RD-augmented decoder to perform few-shot segmentation.

More specifically, the decoder consists of three transformer layers with a multi-head cross-attention model. The overall architecture is shown in Figure C.1. The transformer takes two types of inputs. **The first one** is the patch embedding from a pre-trained vision transformer, denoted as $\mathbf{H}$; and **the second one** is the concatenation of the prototypes $\mathbf{P}$ and the relationship descriptor $\mathbf{R}$ which is derived from the prototypes. The former will be used to produce the key and value vectors of the transformer and the latter will be used for producing the query vectors of the transformer.

[1]Note that in ZegCLIP, the matching process is between text embedding and patch embedding, while in our case is matching between prototype and patch embedding.

The final layer of attention, between the query for a given class and the patch embeddings, is then used to produce the segmentation mask for that particular class. The high-level calculation can be written as $\boldsymbol{Masks} = \text{Trans}(\phi([\mathbf{P}, \mathbf{R}]), \varphi(\mathbf{H}))$, where $[\cdot]$ denotes concatenation and $\{\phi, \varphi\}$ represent the projection layers on class queries and patch features respectively. Please refer to [234] and Chapter 4 for more details about the decoder design. For better comprehension, the complete pseudo code detailing the training and inference processes of our unified framework is presented in Appendix E.

5.3.2 Relationship Descriptors Design for Few-shot Segmentation

Unlike the ZegCLIP [234] model, which utilizes the inherent matching ability between image and text embeddings, the inherent matching capability we wish to leverage is on the matching between patch embeddings. This could further boil down to the matching between a class prototype and patch embeddings in a query image.

First, we define a relationship descriptor for each patch that can be done by performing an element-wise multiplication between a patch embedding and the class prototype. Considering the potentially large number of patch embeddings and the need to scale the method across multiple classes, it's more efficient to consolidate these relationship descriptors into a single descriptor per class. This consolidation can be effectively done by placing greater emphasis on descriptors whose corresponding patch embedding shows a higher matching score with the class prototype. The approach to achieve this involves a weighted summation method, which prioritizes descriptors based on their matching scores:

$$\mathbf{r}^c = \sum_{i,j} \alpha_{i,j} \mathbf{H}_{i,j} \odot \mathbf{p}^c, \qquad \alpha_{i,j} = \frac{\exp(\mathbf{H}_{i,j}^\top \mathbf{p}^c / \tau)}{\sum_{m,n} \exp(\mathbf{H}_{m,n}^\top \mathbf{p}^c / \tau)}, \tag{5.2}$$

where τ is a temperature hyperparameter and is set to 0.1. $\mathbf{H}_{i,j} \odot \mathbf{p}^c \in \mathbb{R}^d$ with $\odot$ denoting the elementwise product. $\mathbf{H}_{i,j}^\top \mathbf{p}^c$ is the inner product (a scalar) between $\mathbf{H}_{i,j}$ and $\mathbf{p}^c$, indicating how $\mathbf{H}_{i,j}$ is related to class c. The final class-query procedure based on relationship descriptor generation is detailed presented in Figure 5.3 and the collection of relationship descriptor can be formulated as $\mathbf{R} = [\mathbf{r}^1, \mathbf{r}^2, \cdots, \mathbf{r}^c]$.

5.3.3 Multi-layer Relationship Descriptors

Since semantic grouping capability exists in different layers of a vision transformer, it is possible to extend the relationship descriptor definitions to multiple layers. We use the l_ith layer patch embeddings of query and support images to perform the calculation in Equation 5.2. Similar augmentation can be made to patch embeddings and prototypes. Specifically when multi-layer relationship descriptors are used, the enriched class query of the decoder transformers are $[\mathbf{P}^{l_1}, \mathbf{R}^{l_1}, \mathbf{P}^{l_2}, \mathbf{R}^{l_2}, \cdots]$ and the mask is calculated via:

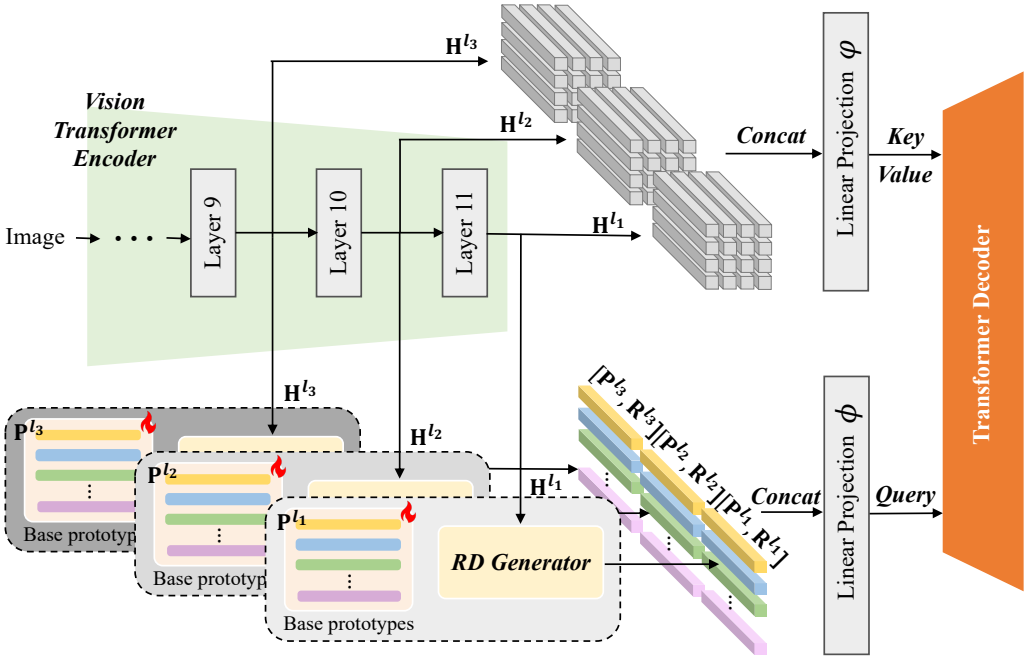


FIGURE 5.4. The framework of fusing multiple relationship descriptors from different transformer layers.

$$\boldsymbol{Masks} = \text{Trans}\left(\phi([\mathbf{P}^{l_1}, \mathbf{R}^{l_1}, \mathbf{P}^{l_2}, \mathbf{R}^{l_2}, \cdots]), \varphi([\mathbf{H}^{l_1}, \mathbf{H}^{l_2}, \cdots])\right) \tag{5.3}$$

5.3.4 Training and Inference

Training algorithm: The aim of the training is to empower the network with the capability to conduct Few-Shot Semantic Segmentation using class prototypes derived from the support examples. In this work, we adopt a simple yet effective training procedure to learn the few-shot semantic segmentation capability from the training set. Specifically, our method maintains C prototypes for all objects that appear in the training set and updates the prototype dynamically using a momentum-style update equation:

$$\mathbf{p}^c \leftarrow (1-\eta) * \mathbf{p}_i^c + \eta * \mathbf{p}^c, \tag{5.4}$$

where $\mathbf{p}^c$ is the prototype embedding for the object in class c and $\mathbf{p}_i^c$ is the prototype embedding calculated from the i-th image. $\eta \in [0, 1)$ is the momentum coefficient to update parameters smoothly instead of learning linear classifiers as class-wise prototypes in the previous GFSS methods [181]. Following [57, 65], we initialize η with a relatively large value of 0.996 and increase η gradually with iteration up to 1.0.

At every training step, we utilize the up-to-date prototypes to compute RDs and generate queries for the decoder transformer. It's worth noting that because $\mathbf{p}^c$ is

dynamically updated, it effectively introduces variations in the prototypes since $\mathbf{p}^c$ will differ at each iteration.

Training loss: Following previous works [218, 234], we apply the combination of the focal loss and dice loss to train the network, namely,

$$\mathcal{L}_{mask} = \lambda \cdot \mathcal{L}_{focal}(\mathbf{M}, \mathbf{M}^{gt}) + \beta \cdot \mathcal{L}_{dice}(\mathbf{M}, \mathbf{M}^{gt}), \tag{5.5}$$

where $\{\lambda, \beta\}$ is set to 20.0 and 1.0 to balance two loss items.

Our training method avoids the episodic training strategy of traditional few-shot learning and a pre-calculated prototype will be supplied at each training step, which is akin to the standard fully-supervised semantic segmentation at each training step. As shown in our experimental study, this method successfully adapts to both generalized and traditional binary few-shot segmentation settings.

Learnable parameters: Following ZegCLIP [234], we fix the parameters of the backbone vision transformer and introduce a few deep prompt tokens to make it adapt to the segmentation task. So the to-be-trained parameters are only those learnable deep prompts and the parameters in the transformer decoder.

Inference: In the context of Generalized Few-Shot Semantic Segmentation (GFSS), the class-wise prototypes are momentum-updated by using the training samples from the base classes. Meanwhile, we determine the prototypes for novel classes based on the examples in the support set. In the binary FSS setting that focuses on distinguishing foreground from background, the representation of the background can be achieved using prototypes from the designated background class, as well as prototypes from base classes. This approach is adopted because, during the inference stage in conventional FSS settings, base classes are treated as part of the background. More details are provided in the Appendix.

It's important to highlight that, in contrast to the techniques outlined in [60, 112, 113, 136], our approach has the capability to directly execute Generalized Few-Shot Semantic Segmentation using the support set examples, without requiring any additional fine-tuning on those examples. For a fair comparison, we also provide the results after a fast update on learnable parameters on the few annotated novel support set to further improve the performance.

5.4 Experiments

5.4.1 Datasets and Setup

To evaluate the effectiveness of our proposed method, we conducted extensive experiments on two public benchmark datasets for GFSS and FSS settings: **PASCAL-**5^i and **COCO-**20^i. The PASCAL-5^i dataset is created from PASCAL VOC 2012 [43] with additional annotations from SDS [62] and contains a total of 20 classes with an extra "background" category. COCO-20^i dataset is more difficult which is derived from MSCOCO [105] and includes 80 categories with "background".

For a fair comparison, we followed the evaluation protocol used in [60, 112] and divided the classes of each dataset into four non-overlapping folds. For each fold, we considered the selected classes as the novel part (5 classes for PASCAL-5^i and 20 classes for COCO-20^i), while the remaining classes (15 for PASCAL-5^i and 60

for COCO-20^i) were considered as base categories. In the training stage, we learn the segmentation ability on the base dataset, and then randomly selected either 1 or 5 samples as support sets for each novel class. In the testing stage, we used the original testing set which contains both base and novel classes for evaluation without any pair-wise input in the generalized few-shot segmentation task. We randomly selected different samples as the novel support set several times by controlling the seeds, as done in [181]. The averaged results of each fold are provided in this work.

5.4.2 Evaluation Metrics

Following previous works, we measure pixel-wise classification accuracy (pAcc) and the mean of class-wise intersection over union (mIoU) on both base and novel classes, denoted as $mIoU(B)$ and $mIoU(N)$ respectively. We also evaluate the harmonic mean of the IoU ($hIoU$) among base and novel classes which is formulated as:

$$hIoU = \frac{2 * mIoU(B) * mIoU(N)}{mIoU(B) + mIoU(N)}. \tag{5.6}$$

5.4.3 Implementation Details

Our proposed method is implemented based on the open-source toolbox MMSegmentation [30] with PyTorch 1.10.1. All experiments we provided are based on the pre-trained vision transformers with the ViT-B/16 architecture and conducted on 4 Tesla V100 GPUs. The batch size is set to 16 with 512x512 as the resolution of images. The total training iterations on base datasets are 10K for PASCAL-5^i and 40K for COCO-20^i. The number of prompt tokens is set to 10 and 50 for PASCAL-5^i and COCO-20^i according to [234]. During the testing stage, similar to prior works [60, 112], we perform quick adaptation by fine-tuning the learned parameters for 100 iterations on PASCAL-5^i and 400 iterations on COCO-20^i, respectively. The optimizer is set to AdamW with the default training schedule in the MMSeg toolbox. Similar to the architecture proposed in [218, 234], we employ three plain vision transformer layers serving as the lightweight decoder network.

TABLE 5.1. Comparison of our proposed method with previous works on the PASCAL-5^i dataset under the GFSS setting. mIoU(N) and mIoU(B) denote the mean mIoU of the novel(N) and base(B) classes among four folders, while F0-F3 shows the detailed mIoU results of each novel fold. Note that the rows with gray background represent the results obtained after test-time finetuning on novel classes.

Method	**1-shot**							**5-shot**						
	mIoU(N)	F0	F1	F2	F3	**mIoU(B)**	**hIoU**	**mIoU(N)**	F0	F1	F2	F3	**mIoU(B)**	**hIoU**
CANet[221]	2.4	-	-	-	-	8.7	3.8	1.5	-	-	-	-	9.1	2.6
PFENet [180]	2.7	-	-	-	-	8.3	4.0	1.9	-	-	-	-	8.8	3.1
SCL [217]	2.4	-	-	-	-	8.9	3.8	1.8	-	-	-	-	9.1	3.1
PANet [189]	11.3	-	-	-	-	31.9	16.6	15.3	-	-	-	-	33.0	20.9
CAPL+PANet [181]	15.0	-	-	-	-	63.1	24.2	19.7	-	-	-	-	63.8	30.1
CAPL+DeeplabV3 [181]	15.1	-	-	-	-	65.7	24.6	23.3	-	-	-	-	67.0	34.6
CAPL+PSPNet [181]	17.5	11.5	26.0	20.3	12.0	66.1	27.7	24.6	16.7	34.6	27.4	19.6	66.9	36.0
CAPL-ViT [181]	23.2	16.5	32.4	25.3	18.6	72.6	35.2	28.5	20.2	37.8	31.2	24.5	73.4	41.1
Ours-single	45.4	45.6	53.5	38.8	43.5	74.2	56.3	51.2	55.7	61.9	40.9	46.3	74.5	60.7
Ours-multiple	**51.0**	58.1	60.4	40.5	44.8	**76.3**	**61.1**	**55.3**	63.0	66.5	42.5	49.2	**76.5**	**64.2**
FineTune [136]	19.7	-	-	-	-	66.4	30.4	50.5	-	-	-	-	71.3	59.1
CCA [113]	22.6	18.0	34.1	22.8	15.5	68.4	34.0	32.1	27.6	46.0	30.1	24.7	70.5	44.1
DIaM [60]	35.1	29.4	46.7	27.1	37.3	70.9	47.0	55.3	53.7	63.6	54.0	50.2	70.9	62.1
POP [112]	35.5	-	-	-	-	73.9	48.0	55.9	-	-	-	-	75.0	64.1
Ours-single	48.9	49.3	57.7	41.3	47.1	76.4	59.6	52.1	55.8	59.3	42.9	50.5	76.6	62.0
Ours-multiple	**52.6**	58.6	60.4	42.5	48.9	**76.5**	**62.3**	**56.7**	64.0	65.9	44.3	52.4	**76.9**	**65.3**

TABLE 5.2. Qualitative results on COCO-20^i datasets under GFSS setting in terms of mIoU(%) on base and novel classes and hIoU(%). Note that the rows with gray background represent the results obtained after test-time finetuning on novel classes.

Method	mIoU(N)	F0	F1	F2	F3	mIoU(B)	hIoU
1-shot							
CAPL [181]	7.6	5.3	9.2	6.9	9.1	44.4	13.0
CAPL-ViT [181]	9.7	6.5	10.8	11.2	10.2	46.1	16.0
Ours-single	17.2	13.4	18.5	22.0	14.7	46.5	25.1
Ours-multiple	**21.1**	17.1	21.3	25.2	20.9	49.5	**29.6**
FineTune [136]	9.2	-	-	-	-	43.6	15.2
CCA [113]	8.8	6.6	10.0	9.3	9.4	46.9	14.1
DIaM [60]	17.2	15.9	19.5	16.9	16.6	48.3	25.4
POP [112]	15.3	-	-	-	-	**54.7**	23.9
Ours-single	20.6	17.6	21.8	24.1	19.0	48.4	28.9
Ours-multiple	**23.2**	20.3	22.3	27.5	22.7	49.6	**31.5**
5-shot							
CAPL [181]	11.0	6.5	14.0	10.6	13.0	44.9	17.7
CAPL-ViT [181]	11.9	7.2	13.5	14.1	12.9	46.9	19.0
Ours-single	22.0	19.4	22.6	25.7	20.5	47.2	30.0
Ours-multiple	**27.0**	22.6	28.3	30.4	26.5	50.1	**35.1**
FineTune [136]	28.8	-	-	-	-	46.6	35.6
CCA [113]	12.7	9.2	15.3	12.1	14.1	47.1	20.0
DIaM [60]	28.7	24.9	33.9	27.2	29.0	48.4	36.0
POP [112]	30.0	-	-	-	-	**54.9**	38.8
Ours-single	25.1	25.3	25.4	26.6	23.1	49.6	33.3
Ours-multiple	**33.1**	32.1	36.9	34.2	29.3	50.2	**39.9**

5.4.4 Comparison with Existing GFSS Methods

We first compare our methods against existing ones in Table 5.1 and Table 5.2. Quantitative improvements achieved by our proposed are shown in Fig 5.5. We consider using single-layer RDs (our-single) and multi-layer RDs (our-multiple). In addition to the inference approaches mentioned previously, we also explore using the test-tuning strategy, that is, fine-tuning the model several iterations on the novel support examples, to achieve a fair comparison to methods [60, 112, 113, 136]. The results obtained are marked in gray background. As seen from Table 5.1 and Table 5.2, our method shows a clear advantage over existing methods when no test-tuning strategy is used. In particular, using multi-layer RDs leads to consistent improvement over single layer RDs, achieving state-of-the-art performance in GFSS. After test-tuning, our methods gains further improve and outperform the comparable methods.

Moreover, we reported the results by replacing the pre-trained ResNet backbone with ViT-B/16 in CAPL [181] and using the same decoder in our proposed method. The results (CAPL-ViT) demonstrate that even though the CAPL method with ViT achieved better segmentation performance on base classes, it still fails to generalize

TABLE 5.3. Using an alternative design for relationship descriptor (RD) on the PASCAL -5^i dataset. [CLS] indicates a design that directly calculates RD via the [CLS] and the prototype embedding, as directly borrowed from ZegCLIP [234]. Our method achieves better performance especially on novel classes, highlighting the contribution of our customized RD.

RD	format	PASCAL -5^i					
		1-shot			5-shot		
		mIoU(N)	mIoU(B)	hIoU	mIoU(N)	mIoU(B)	hIoU
N/A	-	15.9	68.6	25.8	16.1	71.0	26.2
single	[cls]	41.9	73.8	53.5	48.8	74.2	58.9
	ours	45.4	74.2	56.3	51.2	74.5	60.7
		3.5(↑)	**(0.4↑)**	**(2.8↑)**	**(2.4↑)**	**(0.3↑)**	**(1.8↑)**
multiple	[cls]	48.6	74.6	58.9	53.9	75.3	62.8
	ours	51.0	76.3	61.1	55.3	76.5	64.2
		(2.4↑)	**(1.7↑)**	**(2.2↑)**	**(1.4↑)**	**(1.2↑)**	**(1.4↑)**

TABLE 5.4. Using an alternative design for relationship descriptor (RD) on the COCO -20^i dataset. [CLS] indicates a design that directly calculates RD via the [CLS] and the prototype embedding, as directly borrowed from ZegCLIP [234]. Our method achieves better performance especially on novel classes, highlighting the contribution of our customized RD.

RD	format	COCO -20^i					
		1-shot			5-shot		
		mIoU(N)	mIoU(B)	hIoU	mIoU(N)	mIoU(B)	hIoU
N/A	-	7.2	42.8	12.3	7.8	43.4	13.2
single	[cls]	13.7	45.4	21.0	16.8	46.5	24.7
	ours	17.2	46.5	25.1	22.0	47.2	30.0
		(3.5↑)	**(1.1↑)**	**(4.1↑)**	**(5.2↑)**	**(0.7↑)**	**(5.3↑)**
multiple	[cls]	17.5	49.9	25.9	20.6	50.2	29.2
	ours	21.1	49.5	29.6	27.0	50.1	35.1
		(3.6↑)	**(0.4↓)**	**(3.7↑)**	**(6.4↑)**	**0.1(↓)**	**(5.9↑)**

to novel categories. It validates that the competitive performance of our proposed method is not coming from the use of ViT, but the effectiveness of using relationship descriptor. Besides, the visualization in Figure 5.6 indicates that enriched multiple RDs further improve the generalized recognition ability while achieving better segmentation boundaries.

5.4.5 Apply Our Pre-trained Model to FSS Setting

In accordance with previous studies, we evaluated the binary Few-Shot Semantic Segmentation (FSS) performance on novel pair-wise support-query sets. Although our model was not specifically designed for binary segmentation, it demonstrated remarkable results on PASCAL-5^i and COCO-20^i, as reported in Table 5.5 and qualitative performance is presented in Figure 5.6.

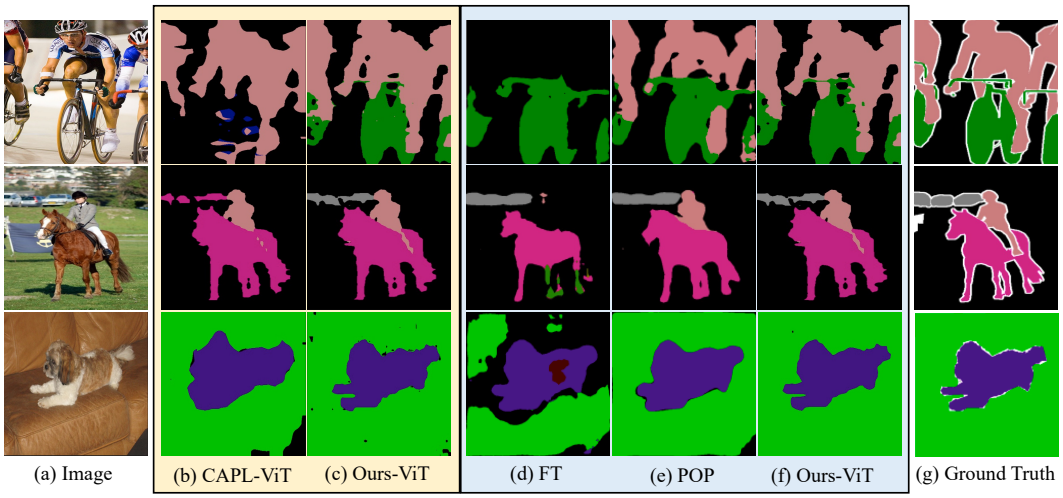


FIGURE 5.5. Visualization results on PASCAL-5^i compared with other methods on GFSS setting. Note that (b)-(c) are the methods without test-time tuning, while (d)-(f) are the methods with test-time tuning on novel support images.

5.4.6 Ablation study

A. How about the results without applying RD?

RD (relationship descriptor) was first proposed in ZegCLIP [234] to unleash the latent matching potential of the pre-trained transformer. By design, it only applies to the CLIP model that matches text and image modalities and it does not support the multiple layers design as [CLS] from other layers do not match across modalities in the pre-trained CLIP model. In fact, how to generalize the idea of RD to other domain or alternative construction of RD are not clear before this work. To experimentally verify this argument, we devised a baseline approach that exclusively employed class-wise prototypes without RDs as queries for the segment decoder. The results are shown in the first row of Table 5.3 and Table 5.4. Evidently, when RD is not utilized, a significant performance drop is observed. This outcome clearly shows the indispensable role played by RD in our method for mitigating overfitting to the training set.

B. Effect of our proposed design of RD

ZegCLIP [234] constructed relationship descriptors based on [cls] token, leveraging the inherent matching capability of [cls] tokens with CLIP text embedding. Our work is based on the vision transformer and we propose to leverage the inherent semantic grouping (matching) capability of patch tokens. Thus, we have a different design of RD, as shown in Equation 5.2, than the one in ZegCLIP. Moreover, we propose a multi-layer strategy and fuse RD from different layers in Equation 5.3. One alternative solution is to straightforwardly borrow the approach of RD construction from ZegCLIP by using the element-wise product between the [CLS] embedding and prototype embedding. We investigate this variant (denoted as [cls]), and report the results in Table 5.3 and Table 5.4. Interestingly, we find this approach can

also achieve quite good performance than the baseline without using RDs. The significantly lower performance compared to our RD design clearly demonstrates the unique contribution of our customized RD approach.

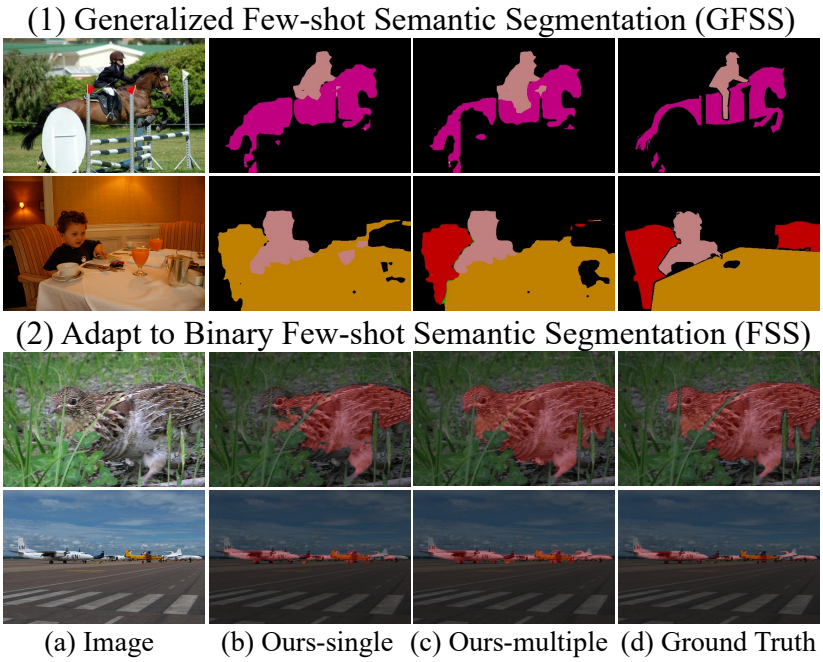


FIGURE 5.6. Quantitative results of (1) generalized and (2) binary few-shot semantic segmentation. (a) are the testing images, (b) and (c) are the results of our method with single and multiple relationship descriptors, (d) denotes ground truths. After enriched RDs, (c) shows better segmentation and recognition ability.

C. The choice of layers for multi-layer RDs

Our proposed multi-layer relationship descriptors (RDs) have exhibited a clear advantage over the use of single-layer RDs, as evident in Table 5.3 and Table 5.4 and the quantitative enhancements demonstrated in Figure 5.6. To further investigate, we conducted comparisons among various choices of layers for constructing multi-layer RDs, with the experimental results presented in Figure 5.7. As seen, the incorporation of the last layer is a main factor for achieving competitive performance. In addition to that, we identify the best combination from our investigation is utilizing the last three layers.

In this supplementary material, we provide more details to complement the manuscript, including the implementation details in Section 5.5 and additional experimental results in Section 5.5.

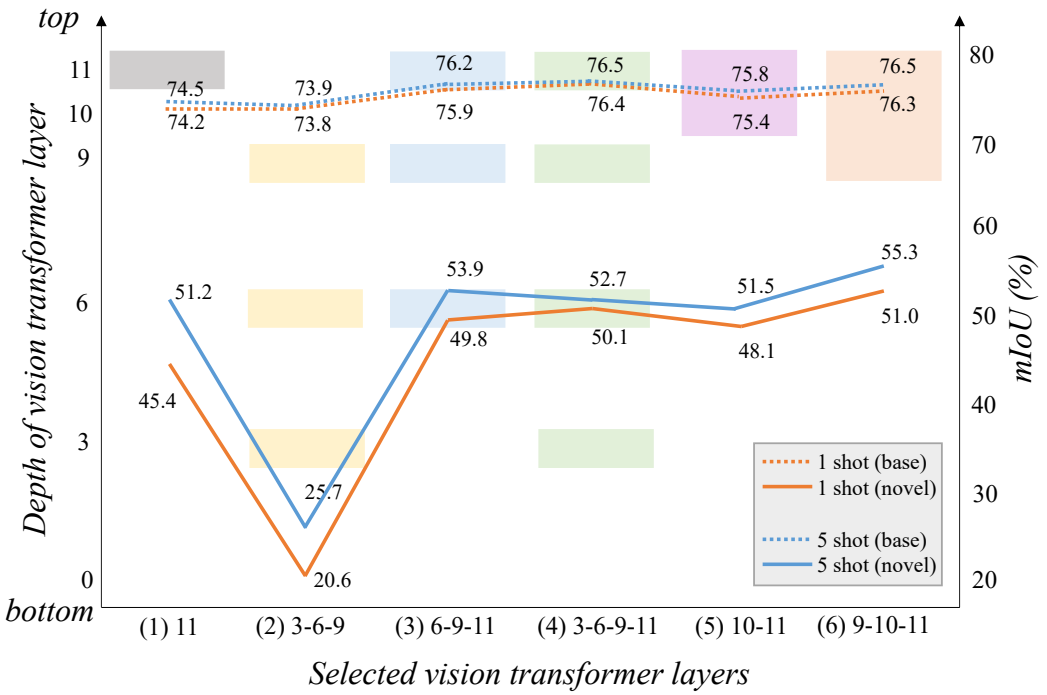


FIGURE 5.7. Quantitative results on PASCAL VOC 2012 of fusing relationship descriptors from different vision transformer layers.

5.5 Implementation Details

5.5.1 The detailed architecture of the decoder

In this work, we propose a unified framework to generate semantic predictions by matching the class embeddings and visual embeddings in a vanilla transformer-based decoder. In this section, we provide more details of the decoder architectur.

Specifically, there are two inputs for the transformer-based decoder: the one input is $[\mathbf{P}, \mathbf{R}] \in \mathbb{R}^{C\times(2*d)}$, where $\mathbf{P} \in \mathbb{R}^{C\times d}$ and $\mathbf{R} \in \mathbb{R}^{C\times d}$ are class-wise prototype embedding and our proposed relationship descriptor (RD) embeddings respectively, and d is the feature dimension. The other input is $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_N] \in \mathbb{R}^{N\times d}$, where N is the number of patch tokens of an image and $\mathbf{h}_j$ denoting the jth patch. We can apply linear layers $\{\psi_q, \psi_k, \psi_v\}$ to generate $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ for query, key and value embeddings, respectively

$$\mathbf{Q} = \psi_q(\phi([\mathbf{P}, \mathbf{R}])) \in \mathbb{R}^{C\times d}, \tag{5.7}$$

$$\mathbf{K} = \psi_k(\varphi(\mathbf{H})) \in \mathbb{R}^{N\times d}, \tag{5.8}$$

$$\mathbf{V} = \psi_v(\varphi(\mathbf{H})) \in \mathbb{R}^{N\times d} \tag{5.9}$$

where $\{\phi, \varphi\}$ are the projection layers. The semantic masks could be obtained by calculating the scaled dot-product attention which is the intermediate product of the multi-head attention model (MHA):

$$\texttt{Masks} = \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} \in \mathbb{R}^{C \times N}. \tag{5.10}$$

where d_k is the dimension of the keys as a scaling factor. The final semantic segmentation results are obtained by applying `Argmax` operation on the class dimension of `Masks` logits.

5.5.2 Details of applying our trained GFSS model to (binary) FSS setting

We have successfully demonstrated the application of our optimized model, originally developed for the Generalized Few-Shot Segmentation (GFSS) setting, to the binary Few-Shot Segmentation (FSS) context. In this section, we offer further insights into the methods and processes we employed to accomplish this. Specifically, the binary FSS setting requires the model to segment out the target class objects and treat all other pixels as non-target (background) for a given testing image.

Therefore, we first adapt our optimized model to the FSS task by creating the novel class prototype from the target class objects in the support set and accumulating non-target-region features of the support set into the existing background class prototype. Moreover, since our optimized model has accommodated the knowledge for base classes, we propose to reinterpret predictions for these base classes as background class decisions in the binary FSS setting. This straightforward yet effective modification enabled our model to deliver remarkable results in the binary FSS setting. The complete results on both PASCAL-5^i and COCO-20^i datasets are provided in Table 5.5.

5.5.3 Effect of various pre-trained vision transformer model for FSS task

In this work, we demonstrate that our proposed relationship descriptor (RD) module can unlock the potential of the supervised pre-trained ViT-B/16 model and improve the generalization ability for FSS tasks. In this section, we further present the effectiveness of the proposed method on the unsupervised pre-trained transformer model DINO [11]. As shown in Table 5.6, our method can consistently achieve better performance on both 1-shot and 5-shot settings.

5.6 Conclusion

This work presents a novel approach to few-shot semantic segmentation that leverages the hidden matching capability of a pre-trained vision transformer. The proposed method focuses on a more practical but challenging generalized few-shot segmentation setting and addresses the potential overfitting to base classes in the segmentation training set by using the relationship descriptor technique. In addition, our work can handle arbitrary input to conduct both generalized and traditional settings of the few-shot semantic segmentation tasks. We demonstrate the effectiveness and simplicity of our approach through extensive experimentation on various pre-trained

vision transformers, achieving superior generalization performance and providing insights for choosing the appropriate transformer model. This work introduces a new paradigm for designing few-shot dense prediction models and is expected to contribute significantly to the research community.

Limitation Limited by the identification ability and resolution of ViT, similar objects may be misclassified and small objects may be ignored in complex scenes.

TABLE 5.5. Comparison of our proposed method with the state-of-the-art FSS methods. Note that our method has not been trained on binary segmentation as well as test-time tuning on novel classes.

Method	Backbone	PASCAL-5^i		COCO-20^i	
		1-shot	5-shot	1-shot	5-shot
PANet [189]	RN-50	48.1	55.7	20.9	29.7
PFENet [180]	RN-50	60.1	61.4	32.4	37.4
SCL [217]	RN-50	61.8	62.9	-	-
RePri [7]	RN-50	59.1	66.8	34.0	42.1
MMNet [194]	RN-50	61.8	63.4	37.5	38.2
CMN [199]	RN-50	62.8	63.7	39.3	43.1
DPCN [110]	RN-50	66.7	69.9	43.0	49.8
BAM [88]	RN-50	67.8	70.9	46.2	51.2
MSANet [74]	RN-50	69.1	74.0	51.1	56.8
SVF [174]	RN-50	69.0	72.3	48.5	53.9
SiGCN [111]	RN-50	65.3	68.5	41.4	48.0
FECANet [109]	RN-50	69.3	74.9	50.9	58.3
ASGNet [95]	RN-101	59.3	63.9	34.5	42.5
SAGNN [198]	RN-101	62.1	62.8	37.2	42.7
CWT [122]	RN-101	58.0	64.7	32.4	42.0
Mining [209]	RN-101	62.6	68.8	36.4	44.4
HSNet [130]	RN-101	66.2	70.4	41.2	49.5
CAPL [181]	RN-101	63.6	68.9	42.8	50.4
IPMT [118]	RN-101	66.1	69.2	42.6	47.9
HM [133]	RN-101	67.8	70.9	45.9	50.6
VAT [69]	RN101	67.9	72.0	41.3	47.9
DACM [202]	RN-101	69.1	73.3	43.0	49.2
CLIPSeg [123]	CLIP-ViT/B	52.3	-	33.2	-
CLIPSeg+ [123]	CLIP-ViT/B	59.3	-	33.2	-
PGMA-Net [165]	CLIP-ViT/B	74.1	74.6	-	-
PGMA-Net [165]	CLIP-RN50	74.1	75.2	54.3	57.1
PGMA-Net [165]	CLIP-RN101	77.6	78.6	59.4	**61.8**
FPTrans [226]	ViT-B/16	64.7	73.7	42.0	53.8
FPTrans [226]	DeiT-B/16	68.8	78.0	47.0	58.9
HSNet [163]	Swin-B	67.3	71.6	47.3	55.1
DCAMA [163]	Swin-B	69.3	74.9	50.9	58.3
IPRNet [141]	ResNet101	67.5	70.9	46.9	53.3
Ours-single	ViT-B/16	77.7	78.0	57.1	59.2
Ours-multiple	ViT-B/16	**78.9**	**80.3**	**60.1**	61.2

TABLE 5.6. Effectiveness of our approach with an unsupervised pre-trained vision transformer model (i.e., ViT-B/16 pre-trained with DINO) on the PASCAL-5^i dataset. "Tuning" denotes the test-time tuning on the novel support set before inference under the generalized few-shot segmentation (GFSS) setting.

RD	**Tuning**	**1-shot**			**5-shot**		
		mIoU(N)	**mIoU(B)**	**hIoU**	**mIoU(N)**	**mIoU(B)**	**hIoU**
N/A	✗	14.2	68.2	23.5	16.5	68.3	26.6
single	✗	24.3	60.4	34.7	24.6	61.0	35.1
	✓	41.5	65.7	50.9	42.6	66.2	51.8
multiple	✗	34.4	69.4	46.0	37.1	70.3	48.6
	✓	47.8	68.8	56.4	49.0	69.6	57.5

Statement of Authorship

Title of Paper	Unlocking the Potential of Pre-trained Vision Transformers for Few-Shot Semantic Segmentation through Relationship Descriptors
Publication Status	[X] Published [ ] Accepted for Publication [ ] Submitted for Publication [ ] Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Accepted and published in CVPR 2024.

Principal Author

Name of Principal Author (Candidate)	Ziqin Zhou
Contribution to the Paper	Developed the conceptual framework, completed the coding, conducted experiments, and wrote the manuscript for the paper.
Overall percentage (%)	70%
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.
Signature	Date: 25.07.2024

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Lingqiao Liu, Hai-ming Xu
Contribution to the Paper	Idea discussions and contributed to writing revisions.
Signature	ate: 25.07.2024

Name of Co-Author	Yangyang Shu
Contribution to the Paper	Idea discussions.
Signature	Date: 25.07.2024

Please cut and paste additional co-author panels here as required.

Chapter 6

Conclusions and Future Directions

In this section, I conclude methods proposed in this thesis about data-efficiency learning based on pre-trained model and present multiple potential research directions for future exploration.

6.1 Conclusions

Chapter3 This work introduces a new approach to generalize class-wise distribution learning, focusing on modeling data distribution across classes rather than individual data generation. The authors propose a straightforward yet effective baseline and redefine a simplified distribution format, implementing Ridge Regression as a meta-classifier. Extensive quantitative comparisons with existing methods are conducted to demonstrate the effectiveness of the proposed approach, including recombination with closely related methods. Visualization results using t-SNE plots illustrate the method's ability to refine data distribution, particularly in 1-shot scenarios.
Chapter4 This study introduces a novel one-stage zero-shot semantic segmentation approach leveraging the pre-trained vision-language model CLIP. By transferring image-wise classification capabilities to dense prediction tasks while retaining advanced zero-shot knowledge, the method achieves competitive results on known classes and significantly improves performance on new classes. Text embeddings serve as flexible queries adaptable to both "inductive" and "transductive" zero-shot settings. Extensive evaluations on three benchmark datasets demonstrate superior performance over previous state-of-the-art methods. Additionally, the one-stage framework offers approximately five times faster inference compared to two-stage methods. Despite limitations such as misclassification of similar classes and overlooking small objects in complex scenes, the study provides insights into integrating CLIP into semantic segmentation, offering inspiration for future research endeavors.
Chapter5 This work introduces an innovative approach to few-shot semantic segmentation utilizing the latent matching capability of a pre-trained vision transformer. Addressing the challenge of generalized few-shot segmentation, the method mitigates potential overfitting to base classes in the training set by employing the relationship descriptor technique. Notably, the proposed approach accommodates arbitrary inputs for both generalized and traditional few-shot segmentation tasks. Extensive experimentation across diverse pre-trained vision transformers demonstrates the effectiveness and simplicity of the method, achieving superior generalization performance and providing guidance for selecting suitable transformer models. This

research presents a new paradigm for developing few-shot dense prediction models, poised to make significant contributions to the research community.

6.2 Future Directions

For future directions in low-shot downstream tasks, like semantic segmentation, it is crucial to address existing weaknesses such as poor performance on the long-tailed problem prevalent in real-world scenarios and model pretraining stage. Additionally, exploring more challenging tasks like panoptic segmentation, which involves identifying each instance and their corresponding categories, holds promise for advancing the field.

Recent advancements, such as Segment Anything Model (SAM) and its variations incorporating semantics (e.g., SAM with an object detector), have gained significant traction in the domain of semantic segmentation. These models offer remarkable generalization capabilities and versatility across various segmentation tasks. Future research could explore how SAM and similar approaches can be tailored or extended for low-shot learning scenarios, enabling improved segmentation performance with minimal data.

Another promising direction is cross-domain model adaptation. Developing methods to adapt segmentation models effectively across diverse domains (e.g., medical, aerial, and autonomous driving imagery) by fine-tuning on minimal target-domain data would address the growing need for generalized segmentation models. These models must be robust and capable of performing in varied, real-world environments, overcoming domain gaps and achieving high performance in data-scarce settings.

In conlusion, researchers should prioritize fully leveraging pre-trained models and exploring the latest and most advanced pre-trained models available. By integrating cross-domain adaptation techniques and innovations from models like SAM, researchers can maximize the utility of pre-trained models and improve their robustness across tasks and domains. By maximizing the utilization of these models, researchers can harness the extensive knowledge encoded within these models to enhance performance across various tasks. Furthermore, staying abreast of advancements in pre-trained models ensures that the field remains at the forefront of innovation, enabling the development of more powerful and effective segmentation techniques. This emphasis on leveraging and exploring cutting-edge pre-trained models will be instrumental in driving continued progress and innovation in semantic segmentation research.

Appendix A

Detailed zero-shot settings

For fair comparison, here we provide the detailed unseen class names of PASCAL VOC 2012 (VOC), COCO-Stuff 164K (COCO), and PASCAL Context (Context) dataset in TableA.1.

TABLE A.1. The details of unseen class names.

Dataset	**The name of unseen classes**
VOC	*pottedplant, sheep, sofa, train, tvmonitor*
COCO	*cow, giraffe, suitcase, frisbee, skateboard* *carrot, scissors, cardboard, clouds, grass* *playingfield, river, road, tree, wall concrete*
Context	*cow, motorbike, sofa, cat, boat, fence* *bird, tv monitor, keyboard, aeroplane*

Appendix B

Detailed augmented templates

The details of the 15 augmented templates we used on COCO-Stuff 164K and PASCAL Context datasets are:

‘A photo of a {}.’
‘A photo of a small {}.’
‘A photo of a medium {}.’
‘A photo of a large {}.’
‘This is a photo of a {}.’
‘This is a photo of a small {}.’
‘This is a photo of a medium {}.’
‘This is a photo of a large {}.’
‘A {} in the scene.’
‘A photo of a {} in the scene.’
‘There is a {} in the scene.’
‘There is the {} in the scene.’
‘This is a {} in the scene.’
‘This is the {} in the scene.’
‘This is one {} in the scene.’

Appendix C

Visualization of semantic masks of different text query embeddings.

To further demonstrate the effectiveness of our designs on the one-stage baseline (Baseline-FT version), we provide more visualizations including the predicted segmentation results and the semantic masks of different class queries via decoder in FigureC.1. Note that the class names in red are the novel categories.

We can see, after applying our proposed designs, the segmentation performance of (b) ZegCLIP has improved on both seen and unseen classes compared with (a) Baseline-FT. Meanwhile, similar unseen classes can be more clearly classified by our ZegCLIP model as shown in the heat maps. For example, in the "COCO-000000079188" testing image, although Baseline-FT can classify "grass" and "tree" (both are unseen classes) correctly in the semantic masks, our ZegCLIP can distinguish these novel classes discriminatively.

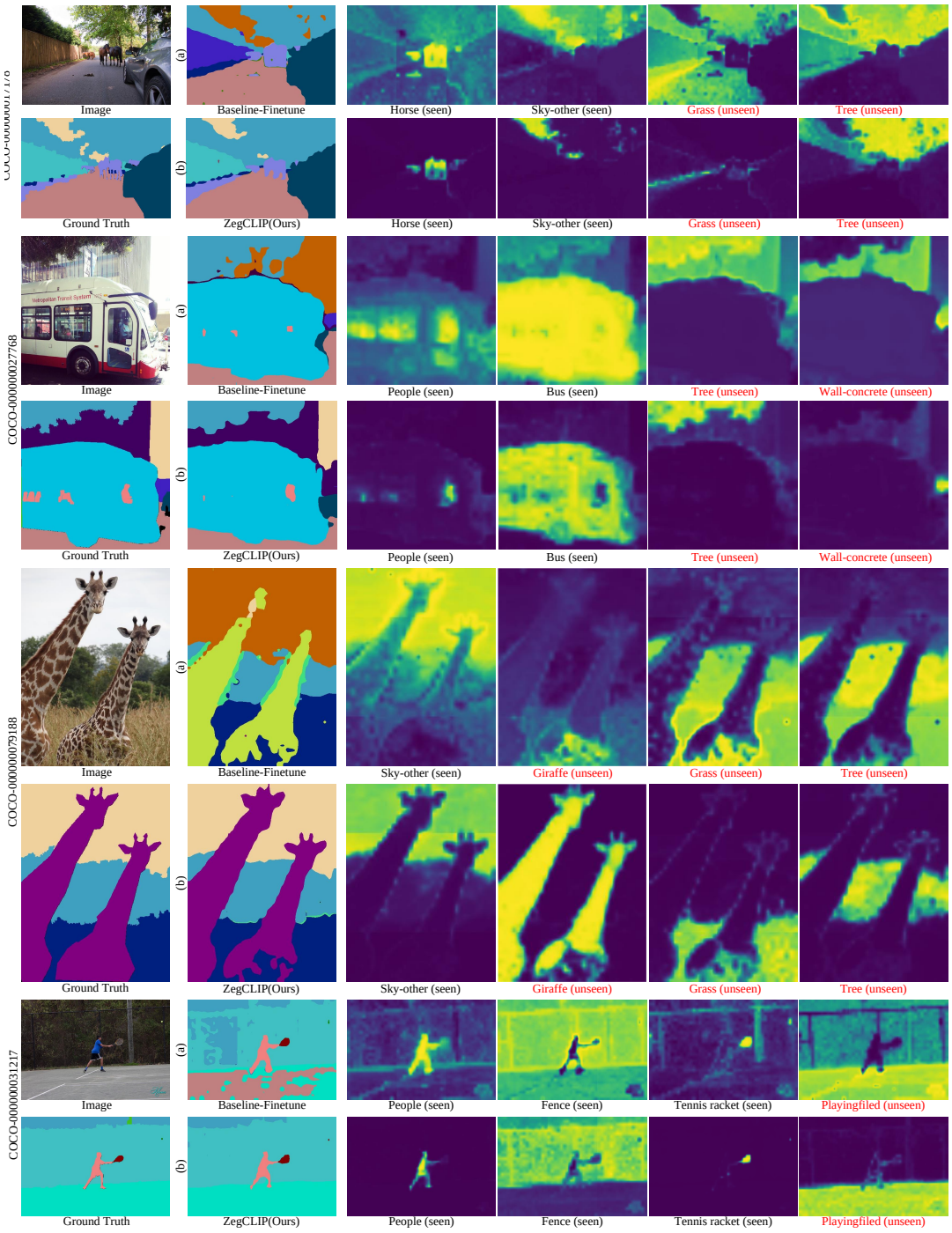


FIGURE C.1. Visualization of semantic masks of different text query embeddings.

Appendix D

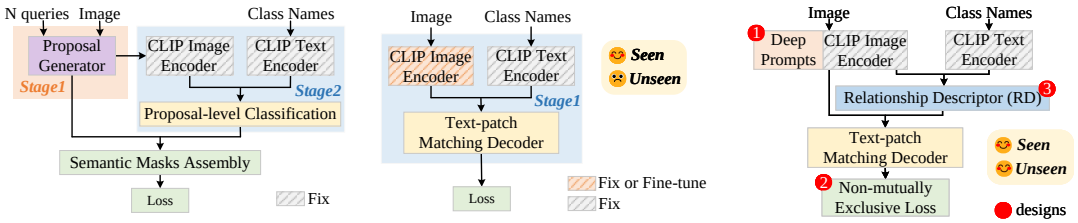
Brief frameworks of related Two-stage and our One-stage method

As we described above, previous zero-shot semantic segmentation methods based on CLIP follow a **two-stage** paradigm as shown in Figure D.1-(a). In stage 1, they need to generate abundant class-agnostic region proposals according to the learnable queries. In stage 2, each cropped proposal region will be encoded via CLIP image encoder to utilize its powerful image-level zero-shot classification capability. The CLIP is still used as a zero-shot image-level classifier.

Instead, we propose a simpler-and-efficient **one-stage** solution as our baseline that directly leverages the feature embedding from CLIP and extends CLIP from image-level classification into pixel-level (or patch-level) as shown in Figure D.1-(b). We introduce a light transformer-based decoder to generate semantic masks by computing the similarities between text-wise and patch-wise embeddings extracted from CLIP. In the baseline, the CLIP text encoder is frozen and the CLIP image encoder can be fixed (Baseline-Fix) or fine-tuned (Baseline-FT).

However, our baseline model still faces the overfitting problem on seen classes. To improve the generalization ability to unseen classes, we propose three important designs on our baseline as shown in D.1-(c). After combining three designs, our model ZegCLIP can transfer the zero-shot ability of CLIP from image-level to pixel-level and achieve significant performance on both seen and unseen classes.

In conclusion, in the two-stage methods, N cropped class-agnostic images will be fed into CLIP for image-wise classification which may heavily increase the computational cost. Our proposed one-stage paradigm is simple-but-efficient due to the original image will be encoded only once. The inference speed has been compared in Table3. Our one-stage method ZegCLIP can achieve a speedup of about **5 times faster** than the two-stage method in the inference stage.



(a) Related **Two-Stage** methods. (b) Our **Baseline**: **One-Stage** method. **(c) ZegCLIP:** Baseline with designs.

FIGURE D.1. Brief frameworks of related **two-stage methods**, our **one-stage baseline**, and our proposed **ZegCLIP** model. The happy face in (b)(c) means good performance in seen or unseen classes, while the sad face in (b) represents that the baseline model achieves poor performance in unseen classes.

Appendix E

Pseudo code of our approach

For better comprehension, the complete pseudo code detailing the training and inference processes of our unified framework is presented as follows:

Algorithm 3: Pseudo code of our framework

// Train on base & background classes C_B
Input: Dataset $\mathcal{D}_B$; Encoder with learnable prompts E, Decoder F, Relationship Descriptor Generator G;
Input: Initialized base class prototypes $\mathbf{P}_B$.

for sampled minibatch $\{\mathbf{I}, \mathbf{M}^{gt}\}_n^{bs}$ from $\mathcal{D}_B$ **do**
  $\mathbf{h}_{cls}, \mathbf{H} = E(\mathbf{I})$;
  // Generate Relationship Descriptors
  $\mathbf{R} = G(\mathbf{P}_B, \mathbf{H})$;
  $\mathbf{M} = sigmoid(F([\mathbf{P}_B, \mathbf{R}], \mathbf{H}))$;
  // Adamw update: E, F
  Updata parameters via $\mathcal{L}_{mask}(\mathbf{M}, \mathbf{M}^{gt})$;
  for class $c (c \in C_B)$ in $\{\mathbf{M}^{gt}\}_n^{bs}$ **do**
    // Momentum update: $\mathbf{P}_B$
    $\mathbf{P}^c = \frac{1}{\sum_{i,j}(\mathbf{M}^{gt})^c_{i,j}} \sum_{i,j} (\mathbf{M}^{gt})^c_{i,j} \mathbf{H}^c_{i,j}$ $\mathbf{P}^c_B \leftarrow (1-\eta) * \mathbf{P}^c + \eta * \mathbf{P}^c_B$
  end
end

// Register novel classes C_N, $C_B \cap C_N = \oslash$
Input: K-shot novel support set $\mathcal{S}_N = \{\mathbf{I}_\mathcal{S}, \mathbf{M}^{gt}_\mathcal{S}\}_n^{C_N * K}$
$\mathbf{h}^{cls}_\mathcal{S}, \mathbf{H}_\mathcal{S} = E(\mathbf{I}_\mathcal{S})$;
for class $c (c \in C_N)$ in $\{\mathbf{M}^{gt}_\mathcal{S}\}_n^{C_N * K}$ **do**
  $\mathbf{P}^c_N += \frac{1}{K} \frac{1}{\sum_{i,j}(\mathbf{M}^{gt}_\mathcal{S})^c_{i,j}} \sum_{i,j} (\mathbf{M}^{gt}_\mathcal{S})^c_{i,j} (\mathbf{H}_\mathcal{S})^c_{i,j}$
end
$\mathbf{P} = concat[\mathbf{P}_B, \mathbf{P}_N]$;
/* If test-time tuning: */
Adamw Updata $\{E, F\}$ via $\mathcal{L}_{mask}(\mathbf{M}_\mathcal{S}, \mathbf{M}^{gt}_\mathcal{S})$;
Momentum update: $\mathbf{P}$;
/* End if */

// Generalized segmentation on $C_B \cup C_N$
Input: Sampled testing image $\mathbf{I}$
$\mathbf{h}_{cls}, \mathbf{H} = E(\mathbf{I})$;
$\mathbf{M} = argmax(F(\phi([\mathbf{P}, G(\mathbf{P}, \mathbf{H})]), \varphi(\mathbf{H})))$;

Bibliography

[1] Kelsey R. Allen, Hanul Shin, and Joshua B. Tenenbaum. “Infinite mixture prototypes for few-shot learning”. In: *Proceedings of the International Conference on Machine Learning (ICML)*. 2019, pp. 232–241.

[2] Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper, and Naama Zwerdling. “Do not have enough data? Deep learning to the rescue!” In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. Vol. 34. 05. 2020, pp. 7383–7390.

[3] Donghyeon Baek, Youngmin Oh, and Bumsub Ham. “Exploiting a joint embedding space for generalized zero-shot semantic segmentation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9536–9545.

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. “Beit: Bert pre-training of image transformers”. In: *arXiv preprint arXiv:2106.08254* (2021).

[5] Ido Ben-Shaul, Ravid Shwartz-Ziv, Tomer Galanti, Shai Dekel, and Yann LeCun. “Reverse engineering self-supervised learning”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 36 (2023), pp. 58324–58345.

[6] Luca Bertinetto, Joao F. Henriques, Philip H.S. Torr, and Andrea Vedaldi. “Meta-learning with differentiable closed-form solvers”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2018.

[7] Malik Boudiaf, Hoel Kervadec, Ziko Imtiaz Masud, Pablo Piantanida, Ismail Ben Ayed, and Jose Dolz. “Few-shot segmentation without meta-learning: A good transductive inference is all you need?” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 13979–13988.

[8] Christopher Bowles, Liang Chen, Ricardo Guerrero, Paul Bentley, Roger Gunn, Alexander Hammers, David Alexander Dickie, Maria Valdés Hernández, Joanna Wardlaw, and Daniel Rueckert. “Gan augmentation: Augmenting training data using generative adversarial networks”. In: *arXiv preprint arXiv:1810.10863* (2018).

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. “Language models are few-shot learners”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), pp. 1877–1901.

[10] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. “Zero-shot semantic segmentation”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 32 (2019).

[11] 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 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9650–9660.

[12] Hui Chen, Guiguang Ding, Xudong Liu, Zijia Lin, Ji Liu, and Jungong Han. "Imram: Iterative matching with recurrent attention memory for cross-modal image-text retrieval". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 12655–12663.

[13] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 40.4 (2017), pp. 834–848.

[14] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. "Semantic image segmentation with deep convolutional nets and fully connected crfs". In: *arXiv preprint arXiv:1412.7062* (2014).

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

[16] Richard J Chen, Ming Y Lu, Tiffany Y Chen, Drew FK Williamson, and Faisal Mahmood. "Synthetic data in machine learning for medicine and healthcare". In: *Nature Biomedical Engineering* 5.6 (2021), pp. 493–497.

[17] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. "Adaptformer: Adapting vision transformers for scalable visual recognition". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 35 (2022), pp. 16664–16678.

[18] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. "A simple framework for contrastive learning of visual representations". In: *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR. 2020, pp. 1597–1607.

[19] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. "Big self-supervised models are strong semi-supervised learners". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), pp. 22243–22255.

[20] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. "A closer look at few-shot classification". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2019.

[21] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. "Semi-supervised semantic segmentation with cross pseudo supervision". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 2613–2622.

[22] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. "Improved baselines with momentum contrastive learning". In: *arXiv preprint arXiv:2003.04297* (2020).

[23] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. "Uniter: Universal image-text representation learning". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 104–120.

[24] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. "Vision transformer adapter for dense predictions". In: *arXiv preprint arXiv:2205.08534* (2022).

[25] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. "Masked-attention mask transformer for universal image segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 1290–1299.

[26] Bowen Cheng, Alex Schwing, and Alexander Kirillov. "Per-pixel classification is not all you need for semantic segmentation". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 34 (2021), pp. 17864–17875.

[27] Jiaxin Cheng, Soumyaroop Nandi, Prem Natarajan, and Wael Abd-Almageed. "SIGN: Spatial-information incorporated generative network for generalized zero-shot semantic segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9556–9566.

[28] Philip Chikontwe, Soopil Kim, and Sang Hyun Park. "Cad: Co-adapting discriminative features for improved few-shot classification". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 14554–14563.

[29] KR1442 Chowdhary and KR Chowdhary. "Natural language processing". In: *Fundamentals of Artificial Intelligence* (2020), pp. 603–649.

[30] MMSegmentation Contributors. *MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark*. https://github.com/open-mmlab/mmsegmentation. 2020.

[31] Rajshekhar Das, Yu-Xiong Wang, and José MF Moura. "On the importance of distractors for few-shot classification". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9030–9040.

[32] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Ieee. 2009, pp. 248–255.

[33] Weijian Deng, Stephen Gould, and Liang Zheng. "What does rotation prediction tell us about classifier accuracy under varying testing environments?" In: *Proceedings of the International Conference on Machine Learning Deep Learning (ICML)*. PMLR. 2021, pp. 2579–2589.

[34] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[35] Guneet S Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. "A baseline for few-shot image classification". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2020.

[36] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. "Decoupling Zero-Shot Semantic Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 11583–11592.

[37] Zheng Ding, Jieke Wang, and Zhuowen Tu. "Open-Vocabulary Panoptic Segmentation with MaskCLIP". In: *arXiv preprint arXiv:2208.08984* (2022).

[38] Nan-Qing Dong and Eric P Xing. "Few-shot semantic segmentation with prototype learning". In: *The British Machine Vision Conference (BMVC)*. Vol. 3. 4. 2018.

[39] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[40] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. "CARLA: An open urban driving simulator". In: *Conference on Robot Learning*. PMLR. 2017, pp. 1–16.

[41] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. "Diversity with cooperation: Ensemble methods for few-shot classification". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 3723–3731.

[42] Sepideh Esmaeilpour, Bing Liu, Eric Robertson, and Lei Shu. "Zero-Shot Out-of-Distribution Detection Based on the Pretrained Model CLIP". In: *Proceedings of the AAAI conference on artificial intelligence (AAAI)*. 2022.

[43] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. "The pascal visual object classes (VOC) challenge". In: *International Journal of Computer Vision (IJCV)* 88 (2009), pp. 303–308.

[44] Mingyuan Fan, Shenqi Lai, Junshi Huang, Xiaoming Wei, Zhenhua Chai, Junfeng Luo, and Xiaolin Wei. "Rethinking BiSeNet for real-time semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 9716–9725.

[45] Zeyu Feng, Chang Xu, and Dacheng Tao. "Self-supervised representation learning by rotation feature decoupling". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 10364–10374.

[46] Chelsea Finn, Pieter Abbeel, and Sergey Levine. "Model-agnostic meta-learning for fast adaptation of deep networks". In: *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR. 2017, pp. 1126–1135.

[47] David Florence N. and Tukey John W. "Exploratory data analysis". In: *Biometrics* 33.4 (1977), p. 768.

[48] Chin-Lun Fu, Zih-Ching Chen, Yun-Ru Lee, and Hung-yi Lee. "Adapterbias: Parameter-efficient token-dependent representation shift for adapters in nlp tasks". In: *arXiv preprint arXiv:2205.00305* (2022).

[49] Mingfei Gao, Zizhao Zhang, Guo Yu, Sercan Ö Arık, Larry S Davis, and Tomas Pfister. "Consistency-based semi-supervised active learning: Towards minimizing labeling cost". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 510–526.

[50] Shenyuan Gao, Chunluan Zhou, and Jun Zhang. "Generalized relation modeling for transformer tracking". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 18686–18695.

[51] Yi-Zhao Gao, Nan-Yi Fei, Guang-Zhen Liu, Zhi-Wu Lu, Tao Xiang, and Song-Fang Huang. "Contrastive prototype learning with augmented embeddings for few-Shot learning". In: *Uncertainty in Artificial Intelligence (UAI)*. 2021.

[52] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. "Scaling open-vocabulary image segmentation with image-level labels". In: *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI*. Springer. 2022, pp. 540–557.

[53] Spyros Gidaris and Nikos Komodakis. "Dynamic few-shot visual learning without forgetting". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 4367–4375.

[54] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. "Unsupervised representation learning by predicting image rotations". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2018.

[55] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 27. 5. 2014, pp. 2672–2680.

[56] Alex Graves, Greg Wayne, and Ivo Danihelka. "Neural turing machines". In: *arXiv preprint arXiv:1410.5401* (2014).

[57] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. "Bootstrap your own latent-a new approach to self-supervised learning". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), pp. 21271–21284.

[58] Jiaqi Gu, Hyoukjun Kwon, Dilin Wang, Wei Ye, Meng Li, Yu-Hsin Chen, Liangzhen Lai, Vikas Chandra, and David Z Pan. "Multi-scale high-resolution vision transformer for semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 12094–12103.

[59] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. "Context-aware feature generation for zero-shot semantic segmentation". In: *Proceedings of the ACM International Conference on Multimedia (ACMMM)*. 2020, pp. 1921–1929.

[60] Sina Hajimiri, Malik Boudiaf, Ismail Ben Ayed, and Jose Dolz. "A Strong Baseline for Generalized Few-Shot Semantic Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 11269–11278.

[61] Fu-Sheng Hao, Feng-Xiang He, Lei Wang, Jian-Zhong Cao, and Da-Cheng Tao. "Collect and select: Semantic alignment metric learning for few-shot learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019.

[62] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. "Semantic contours from inverse detectors". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE. 2011, pp. 991–998.

[63] Bharath Hariharan and Ross Girshick. "Low-shot visual recognition by shrinking and hallucinating features". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2017, pp. 3018–3027.

[64] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. "Masked autoencoders are scalable vision learners". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 16000–16009.

[65] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. "Momentum contrast for unsupervised visual representation learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 9729–9738.

[66] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778.

[67] Stefan Hinterstoisser, Vincent Lepetit, Paul Wohlhart, and Kurt Konolige. "On pre-trained image features and synthetic images for deep learning". In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*. 2018.

[68] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), pp. 6840–6851.

[69] Sunghwan Hong, Seokju Cho, Jisu Nam, Stephen Lin, and Seungryong Kim. "Cost aggregation with 4d convolutional swin transformer for few-shot segmentation". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2022, pp. 108–126.

[70] Rui-Bing Hou, Hong Chang, Bing-Peng Ma, Shi-Guang Shan, and Xi-Lin Chen. "Cross attention network for few-shot classification". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 27. 2019, pp. 4003–4014.

[71] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. "Lora: Low-rank adaptation of large language models". In: *arXiv preprint arXiv:2106.09685* (2021).

[72] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. "Densely connected convolutional networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 4700–4708.

[73] Zhicheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. "Seeing out of the box: End-to-end pre-training for vision-language representation learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 12976–12985.

[74] Ehtesham Iqbal, Sirojbek Safarov, and Seongdeok Bang. "MSANet: Multi-Similarity and Attention Guidance for Boosting Few-Shot Segmentation". In: *arXiv preprint arXiv:2206.09667* (2022).

[75] Muhammad Abdullah Jamal and Guo-Jun Qi. "Task agnostic meta-learning for few-shot learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 11719–11727.

[76] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. "Scaling up visual and vision-language representation learning with noisy text supervision". In: *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR. 2021, pp. 4904–4916.

[77] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. "Visual prompt tuning". In: *arXiv preprint arXiv:2203.12119* (2022).

[78] Jingjing Jiang, Ziyi Liu, and Nanning Zheng. "Finetuning Pretrained Vision-Language Models with Correlation Information Bottleneck for Robust Visual Question Answering". In: *arXiv preprint arXiv:2209.06954* (2022).

[79] Xi-Sen Jin, Mohammad Rostami, and Xiang Ren. "Lifelong learning of few-shot learners across nlp tasks". In: *arXiv preprint arXiv:2104.08808* (2021).

[80] Bing-Yi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jia-Shi Feng, and Trevor Darrell. "Few-shot object detection via feature reweighting". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 8420–8429.

[81] Wonjae Kim, Bokyung Son, and Ildoo Kim. "Vilt: Vision-and-language transformer without convolution or region supervision". In: *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR. 2021, pp. 5583–5594.

[82] Diederik P Kingma. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[83] Diederik P. Kingma and Jimmy Lei Ba. "Adam: A method for stochastic optimization". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2015.

[84] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. "Siamese neural networks for one-shot image recognition". In: *Proceedings of the International Conference on Machine Learning Deep Learning Workshop (ICML workshop)*. Vol. 2. Lille. 2015.

[85] Simon Kornblith, Jonathon Shlens, and Quoc V Le. "Do better imagenet models transfer better?" In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 2661–2671.

[86] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 25 (2012).

[87] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. "Learning to detect unseen object classes by between-class attribute transfer". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2009, pp. 951–958.

[88] Chunbo Lang, Gong Cheng, Binfei Tu, and Junwei Han. "Learning what not to segment: A new perspective on few-shot segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 8057–8067.

[89] Michalis Lazarou, Yannis Avrithis, and Tania Stathaki. "Tensor feature hallucination for few-shot learning". In: *arXiv preprint arXiv:2106.05321* (2021).

[90] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *Nature* 521.7553 (2015), pp. 436–444.

[91] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. "Meta-learning with differentiable convex optimization". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 10657–10665.

[92] Ao-Xue Li, Tian-Ge Luo, Tao Xiang, Wei-Ran Huang, and Li-Wei Wang. "Few-shot learning with global class representations". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 9715–9724.

[93] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. "Language-driven semantic segmentation". In: *arXiv preprint arXiv:2201.03546* (2022).

[94] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. "Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. Vol. 34. 07. 2020, pp. 11336–11344.

[95] Gen Li, Varun Jampani, Laura Sevilla-Lara, Deqing Sun, Jonghyun Kim, and Joongkyu Kim. "Adaptive Prototype Learning and Allocation for Few-Shot Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 8334–8343.

[96] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. "Visualbert: A simple and performant baseline for vision and language". In: *arXiv preprint arXiv:1908.03557* (2019).

[97] Xiang Lisa Li and Percy Liang. "Prefix-tuning: Optimizing continuous prompts for generation". In: *arXiv preprint arXiv:2101.00190* (2021).

[98] Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. "Dice loss for data-imbalanced NLP tasks". In: *arXiv preprint arXiv:1911.02855* (2019).

[99] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. "Exploring plain vision transformer backbones for object detection". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2022, pp. 280–296.

[100] Yanghao Li, Saining Xie, Xinlei Chen, Piotr Dollar, Kaiming He, and Ross Girshick. "Benchmarking detection transfer learning with vision transformers". In: *arXiv preprint arXiv:2111.11429* (2021).

[101] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. "A survey of convolutional neural networks: analysis, applications, and prospects". In: *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)* 33.12 (2021), pp. 6999–7019.

[102] 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: *arXiv preprint arXiv:2210.04150* (2022).

[103] Junyang Lin, An Yang, Yichang Zhang, Jie Liu, Jingren Zhou, and Hongxia Yang. "Interbert: Vision-and-language interaction for multi-modal pretraining". In: *arXiv preprint arXiv:2003.13198* (2020).

[104] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. "Focal loss for dense object detection". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2017, pp. 2980–2988.

[105] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. "Microsoft coco: Common objects in context". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2014, pp. 740–755.

[106] Bin Liu, Yue Cao, Yutong Lin, Qi Li, Zheng Zhang, Mingsheng Long, and Han Hu. "Negative margin matters: Understanding margin in few-shot classification". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 438–455.

[107] Chen Liu, Yan-Wei Fu, Cheng-Ming Xu, Si-Qian Yang, Ji-Lin Li, Cheng-Jie Wang, and Li Zhang. "Learning a few-shot embedding model with contrastive learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. Vol. 35. 10. 2021, pp. 8635–8643.

[108] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan L Yuille, and Li Fei-Fei. "Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 82–92.

[109] Huafeng Liu, Pai Peng, Tao Chen, Qiong Wang, Yazhou Yao, and Xian-Sheng Hua. "FECANet: Boosting few-shot semantic segmentation with feature-enhanced context-aware network". In: *IEEE Transactions on Multimedia (TMM)* (2023).

[110] Jie Liu, Yanqi Bao, Guo-Sen Xie, Huan Xiong, Jan-Jakob Sonke, and Efstratios Gavves. "Dynamic prototype convolution network for few-shot semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 11553–11562.

[111] Jie Liu, Yanqi Bao, Wenzhe Ying, Haochen Wang, Yang Gao, Jan-Jakob Sonke, and Efstratios Gavves. "Few-shot semantic segmentation with support-induced graph convolutional network". In: *arXiv preprint arXiv:2301.03194* (2023).

[112] Sun-Ao Liu, Yiheng Zhang, Zhaofan Qiu, Hongtao Xie, Yongdong Zhang, and Ting Yao. "Learning Orthogonal Prototypes for Generalized Few-Shot Semantic Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 11319–11328.

[113] Weide Liu, Zhonghua Wu, Yang Zhao, Yuming Fang, Chuan-Sheng Foo, Jun Cheng, and Guosheng Lin. "Harmonizing Base and Novel Classes: A Class-Contrastive Approach for Generalized Few-Shot Segmentation". In: *arXiv preprint arXiv:2303.13724* (2023).

[114] Weide Liu, Chi Zhang, Guosheng Lin, and Fayao Liu. "Crnet: Cross-reference networks for few-shot segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 4165–4173.

[115] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. "P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks". In: *Proceedings of the Association for Computational Linguistics (ACL)*. 2022, pp. 61–68.

[116] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. "Self-supervised learning: Generative or contrastive". In: *IEEE Transactions on Knowledge and Data Engineering (KDE)* 35.1 (2021), pp. 857–876.

[117] Yanbin Liu, Juho Lee, Linchao Zhu, Ling Chen, Humphrey Shi, and Yi Yang. "A multi-mode modulator for multi-domain few-shot classification". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 8453–8462.

[118] Yuanwei Liu, Nian Liu, Xiwen Yao, and Junwei Han. "Intermediate prototype mining transformer for few-shot semantic segmentation". In: *arXiv preprint arXiv:2210.06780* (2022).

[119] Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. "Image retrieval on real-life images with pre-trained vision-and-language models". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 2125–2134.

[120] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3431–3440.

[121] Xiankai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. "Video object segmentation with episodic graph memory networks". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 661–679.

[122] Zhihe Lu, Sen He, Xiatian Zhu, Li Zhang, Yi-Zhe Song, and Tao Xiang. "Simpler is better: Few-shot semantic segmentation with classifier weight transformer". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 8741–8750.

[123] 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 (CVPR)*. 2022, pp. 7076–7086.

[124] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. "SegCLIP: Patch Aggregation with Learnable Centers for Open-Vocabulary Semantic Segmentation". In: *arXiv preprint arXiv:2211.14813* (2022).

[125] Yuan-Bo Ma, Xing Xu, Fu-Min Shen, and Heng-Tao Shen. "Similarity preserving feature generating networks for zero-shot learning". In: *Neurocomputing* 406 (2020), pp. 333–342.

[126] Christoph Mayer, Martin Danelljan, Goutam Bhat, Matthieu Paul, Danda Pani Paudel, Fisher Yu, and Luc Van Gool. "Transforming model prediction for tracking". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 8731–8740.

[127] Pratik Mazumder, Pravendra Singh, and Vinay P Namboodiri. "Improving few-Shot learning using composite rotation based auxiliary task". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2021, pp. 2654–2663.

[128] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. "Deep Spectral Methods: A Surprisingly Strong Baseline for Unsupervised Semantic Segmentation and Localization". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 8364–8375.

[129] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. "V-net: Fully convolutional neural networks for volumetric medical image segmentation". In: *Proceedings of the International Conference on 3D Vision (3DV)*. IEEE. 2016, pp. 565–571.

[130] Juhong Min, Dahyun Kang, and Minsu Cho. "Hypercorrelation squeeze for few-shot segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 6941–6952.

[131] Yimeng Min, Yiwei Bai, and Carla P Gomes. "Unsupervised learning for solving the travelling salesman problem". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 36 (2024).

[132] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. "A simple neural attentive meta-learner". In: *arXiv preprint arXiv:1707.03141* (2017).

[133] Seonghyeon Moon, Samuel S Sohn, Honglu Zhou, Sejong Yoon, Vladimir Pavlovic, Muhammad Haris Khan, and Mubbasir Kapadia. "HM: Hybrid masking for few-shot segmentation". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2022, pp. 506–523.

[134] Tsendsuren Munkhdalai and Hong Yu. "Meta networks". In: *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR. 2017, pp. 2554–2563.

[135] Tsendsuren Munkhdalaim, Xingdi Yuan, Soroush Mehri, and Adam Trischler. "Rapid adaptation with conditionally shifted neuron". In: *Proceedings of the International Conference on Machine Learning (ICML)*. 2018, pp. 3661–3670.

[136] Josh Myers-Dean, Yinan Zhao, Brian Price, Scott Cohen, and Danna Gurari. "Generalized few-shot semantic segmentation: All you need is fine-tuning". In: *arXiv preprint arXiv:2112.10982* (2021).

[137] Muhammad Muzammal Naseer, Kanchana Ranasinghe, Salman H Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. "Intriguing properties of vision transformers". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 34 (2021), pp. 23296–23308.

[138] Khoi Nguyen and Sinisa Todorovic. "A Self-supervised GAN for Unsupervised Few-shot Object Recognition". In: *Proceedings of the International Conference on Pattern Recognition (ICPR)*. IEEE. 2021, pp. 3225–3231.

[139] Alex Nichol, Joshua Achiam, and John Schulman. "On first-order meta-learning algorithms". In: *arXiv preprint arXiv:1803.02999* (2018).

[140] Pedro O O Pinheiro, Amjad Almahairi, Ryan Benmalek, Florian Golemo, and Aaron C Courville. "Unsupervised learning of dense visual representations". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), pp. 4489–4500.

[141] Atsuro Okazawa. "Interclass prototype relation for few-shot segmentation". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2022, pp. 362–378.

[142] 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". In: *arXiv preprint arXiv:2304.07193* (2023).

[143] Boris N Oreshkin, Pau Rodriguez, and Alexandre Lacoste. "Tadam: Task dependent adaptive metric for improved few-shot learning". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 31. 2018, pp. 719–729.

[144] Daniil Pakhomov, Sanchit Hira, Narayani Wagle, Kemar E Green, and Nassir Navab. "Segmentation in style: Unsupervised semantic image segmentation with stylegan and CLIP". In: *arXiv preprint arXiv:2107.12518* (2021).

[145] Junting Pan, Ziyi Lin, Xiatian Zhu, Jing Shao, and Hongsheng Li. "St-adapter: Parameter-efficient image-to-video transfer learning". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 35 (2022), pp. 26462–26477.

[146] Giuseppe Pastore, Fabio Cermelli, Yongqin Xian, Massimiliano Mancini, Zeynep Akata, and Barbara Caputo. "A closer look at self-training for zero-label semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 2693–2702.

[147] Zhi-Mao Peng, Ze-Chao Li, Jun-Ge Zhang, Yan Li, Guo-Jun Qi, and Jin-Hui Tang. "Few-shot image recognition with knowledge transfer". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 441–449.

[148] Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. "Beit v2: Masked image modeling with vector-quantized visual tokenizers". In: *arXiv preprint arXiv:2208.06366* (2022).

[149] Mangla Puneet, Singh Mayank, Sinha Abhishek, Kumari Nupur, Balasubramanian Vineeth N., and Krishnamurthy Balaji. "Charting the right manifold: Manifold mixup for few-shot learning". In: *Winter Conference on Applications of Computer Vision (WACV)*. 2020, pp. 2218–2227.

[150] Li-Meng Qiao, Ye-Min Shi, Jia Li, Yong-Hong Tian, Tie-Jun Huang, and Yao-Wei Wang. "Transductive episodic-wise adaptive metric for few-shot learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 3603–3612.

[151] Guanghui Qin and Jason Eisner. "Learning how to ask: Querying lms with mixtures of soft prompts". In: *arXiv preprint arXiv:2104.06599* (2021).

[152] 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: *Proceedings of the International Conference on Machine Learning (ICML)*. 2021, pp. 8748–8763.

[153] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. "Denseclip: Language-guided dense prediction with context-aware prompting". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 18082–18091.

[154] Sachin Ravi and Hugo Larochelle. "Optimization as a model for few-shot learning". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2017.

[155] Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. "Few-shot learning with embedded class models and shot-free meta training". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 331–339.

[156] Meng-Ye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. "Meta-learning for semi-supervised few-shot classification". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2018.

[157] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. "A survey of deep active learning". In: *ACM computing surveys (CSUR)* 54.9 (2021), pp. 1–40.

[158] Mamshad Nayeem Rizve, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. "Exploring complementary strengths of invariant and equivariant representations for few-Shot learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 10836–10846.

[159] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*. 2015, pp. 234–241.

[160] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. "Imagenet large scale visual recognition challenge". In: vol. 115. 3. Springer, 2015, pp. 211–252.

[161] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. "Meta-learning with latent embedding optimization". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2018.

[162] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. "One-shot learning with memory-augmented neural networks". In: *arXiv preprint arXiv:1605.06065* (2016).

[163] Xinyu Shi, Dong Wei, Yu Zhang, Donghuan Lu, Munan Ning, Jiashun Chen, Kai Ma, and Yefeng Zheng. "Dense cross-query-and-support attention weighted mask aggregation for few-shot segmentation". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2022, pp. 151–168.

[164] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. "Test-time prompt tuning for zero-shot generalization in vision-language models". In: *arXiv preprint arXiv:2209.07511* (2022).

[165] Chen Shuai, Meng Fanman, Zhang Runtong, Qiu Heqian, Li Hongliang, Wu Qingbo, and Xu Linfeng. "Visual and Textual Prior Guided Mask Assemble for Few-Shot Segmentation and Beyond". In: *arXiv preprint arXiv:2308.07539* (2023).

[166] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. "Adaptive subspaces for few-shot learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 4136–4145.

[167] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

[168] Jake Snell, Kevin Swersky, and Richard S Zemel. "Prototypical networks for few-shot learning". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 30. 2017, pp. 4077–4087.

[169] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. "A simple semi-supervised learning framework for object detection". In: *arXiv preprint arXiv:2005.04757* (2020).

[170] Jeremy Speth, Nathan Vance, Patrick Flynn, and Adam Czajka. "Non-contrastive unsupervised learning of physiological signals from video". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 14464–14474.

[171] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. "Segmenter: Transformer for semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 7262–7272.

[172] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. "Vl-bert: Pre-training of generic visual-linguistic representations". In: *arXiv preprint arXiv:1908.08530* (2019).

[173] Qian-Ru Sun, Yao-Yao Liu, Tat-Seng Chua, and Bernt Schiele. "Meta-transfer learning for few-shot learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 403–412.

[174] Yanpeng Sun, Qiang Chen, Xiangyu He, Jian Wang, Haocheng Feng, Junyu Han, Errui Ding, Jian Cheng, Zechao Li, and Jingdong Wang. "Singular Value Fine-tuning: Few-shot Segmentation requires Few-parameters Fine-tuning". In: *arXiv preprint arXiv:2206.06122* (2022).

[175] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. "Learning to compare: Relation network for few-shot learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 1199–1208.

[176] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 1–9.

[177] Chang Tang, Xinwang Liu, Xinzhong Zhu, Jian Xiong, Miaomiao Li, Jingyuan Xia, Xiangke Wang, and Lizhe Wang. "Feature selective projection with low-rank embedding and dual Laplacian regularization". In: *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 32.9 (2019), pp. 1747–1760.

[178] Chang Tang, Xiao Zheng, Xinwang Liu, Wei Zhang, Jing Zhang, Jian Xiong, and Lizhe Wang. "Cross-view locality preserved diversity and consensus learning for multi-view unsupervised feature selection". In: *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2021).

[179] Yong-Long Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. "Rethinking few-shot image classification: a good embedding is all you need?" In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 266–282.

[180] Zhuo-Tao Tian, Heng-Shuang Zhao, Michelle Shu, Zhi-Cheng Yang, Rui-Yu Li, and Jia-Ya Jia. "Prior guided feature enrichment network for few-shot segmentation". In: *IEEE Transactions on Pattern Analysis & Machine Intelligence (TPAMI)* 01 (2020), pp. 1–1.

[181] Zhuotao Tian, Xin Lai, Li Jiang, Shu Liu, Michelle Shu, Hengshuang Zhao, and Jiaya Jia. "Generalized few-shot semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 11563–11572.

[182] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. "Meta-dataset: A dataset of datasets for learning to learn from few examples". In: *arXiv preprint arXiv:1903.03096* (2019).

[183] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. "Unsupervised semantic segmentation by contrasting object mask proposals". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 10052–10062.

[184] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)* 30 (2017).

[185] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. "Matching networks for one shot learning". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 29. 2016, pp. 3630–3638.

[186] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. "Deep learning for computer vision: A brief review". In: *Computational Intelligence and Neuroscience* 2018.1 (2018), p. 7068349.

[187] Deze Wang, Zhouyang Jia, Shanshan Li, Yue Yu, Yun Xiong, Wei Dong, and Xiangke Liao. "Bridging pre-trained models and downstream tasks for source code understanding". In: *Proceedings of the International Conference on Software Engineering (ICSE)*. 2022, pp. 287–298.

[188] Haochen Wang, Xudong Zhang, Yutao Hu, Yandan Yang, Xianbin Cao, and Xiantong Zhen. "Few-shot semantic segmentation with democratic attention networks". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 730–746.

[189] Kai-Xin Wang, Jun Hao Liew, Ying-Tian Zou, Da-Quan Zhou, and Jia-Shi Feng. "Panet: Few-shot image semantic segmentation with prototype alignment". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 9197–9206.

[190] Lingjing Wang, Xiang Li, and Yi Fang. "Few-shot learning of part-specific probability space for 3d shape segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 4504–4513.

[191] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. "Generalizing from a few examples: A survey on few-shot learning". In: *ACM Computing Surveys* 53.3 (2020), pp. 1–34.

[192] Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. "Cris: Clip-driven referring image segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 11686–11695.

[193] Davis Wertheimer, Luming Tang, and Bharath Hariharan. "Few-shot classification with feature map reconstruction networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 8012–8021.

[194] Zhonghua Wu, Xiangxi Shi, Guosheng Lin, and Jianfei Cai. "Learning meta-class memory for few-shot semantic segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 517–526.

[195] Zi-Yang Wu, Yu-Wei Li, Li-Hua Guo, and Kui Jia. "PARN: Position-aware relation networks for few-shot learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 6659–6667.

[196] Yong-Qin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. "Feature generating networks for zero-shot learning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 5542–5551.

[197] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. "Semantic projection network for zero-and few-label semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 8256–8265.

[198] Guo-Sen Xie, Jie Liu, Huan Xiong, and Ling Shao. "Scale-aware graph neural network for few-shot semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 5475–5484.

[199] Guo-Sen Xie, Huan Xiong, Jie Liu, Yazhou Yao, and Ling Shao. "Few-shot semantic segmentation with cyclic memory network". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 7293–7302.

[200] Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro O O Pinheiro. "Adaptive cross-modal few-shot learning". In: *Advances in Neural Information Processing Systems (NeurIPS)* 32 (2019).

[201] Yinghui Xing, Qirui Wu, De Cheng, Shizhou Zhang, Guoqiang Liang, and Yanning Zhang. "Class-Aware Visual Prompt Tuning for Vision-Language Pre-Trained Model". In: *arXiv preprint arXiv:2208.08340* (2022).

[202] Zhitong Xiong, Haopeng Li, and Xiao Xiang Zhu. "Doubly deformable aggregation of covariance matrices for few-shot segmentation". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. Springer. 2022, pp. 133–150.

[203] 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 (CVPR)*. 2022, pp. 18134–18144.

[204] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, Ferdous Sohel, and Dan Xu. "Leveraging auxiliary tasks with affinity learning for weakly supervised semantic segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 6984–6993.

[205] Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. "Side adapter network for open-vocabulary semantic segmentation". In: *arXiv preprint arXiv:2302.12242* (2023).

[206] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. "A Simple Baseline for Open-Vocabulary Semantic Segmentation with Pre-trained Vision-Language Model". In: *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIX*. Springer. 2022, pp. 736–753.

[207] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. "A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model". In: *arXiv preprint arXiv:2112.14757* (2021).

[208] Chenggang Yan, Biao Gong, Yuxuan Wei, and Yue Gao. "Deep multi-view enhancement hashing for image retrieval". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 43.4 (2020), pp. 1445–1451.

[209] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. "Mining latent classes for few-shot segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 8721–8730.

[210] Shuo Yang, Lu Liu, and Min Xu. "Free lunch for few-shot learning: Distribution calibration". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2021.

[211] Xianghui Yang, Bairun Wang, Kaige Chen, Xinchi Zhou, Shuai Yi, Wanli Ouyang, and Luping Zhou. "Brinet: Towards bridging the intra-class and inter-class gaps in one-shot segmentation". In: *arXiv preprint arXiv:2008.06226* (2020).

[212] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. "A survey on deep semi-supervised learning". In: *IEEE Transactions on Knowledge and Data Engineering (KDE)* 35.9 (2022), pp. 8934–8954.

[213] Ze Yang, Ya-Li Wang, Xian-Yu Chen, Jian-Zhuang Liu, and Yu Qiao. "Context-transformer: tackling object confusion for few-shot detection". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. Vol. 34. 07. 2020, pp. 12653–12660.

[214] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. "CPT: Colorful prompt tuning for pre-trained vision-language models". In: *arXiv preprint arXiv:2109.11797* (2021).

[215] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. "Few-shot learning via embedding adaptation with set-to-set functions". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 8808–8817.

[216] Qin-Yuan Ye, Bill Yu-Chen Lin, and Xiang Ren. "Crossfit: A few-shot learning challenge for cross-task generalization in nlp". In: *arXiv preprint arXiv:2104.08835* (2021).

[217] Bingfeng Zhang, Jimin Xiao, and Terry Qin. "Self-guided and cross-guided learning for few-shot segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 8312–8321.

[218] Bowen Zhang, Zhi Tian, Quan Tang, Xiangxiang Chu, Xiaolin Wei, Chunhua Shen, and Yifan Liu. "SegViT: Semantic segmentation with plain vision transformers". In: *arXiv preprint arXiv:2210.05844* (2022).

[219] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. "Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 12203–12213.

[220] Chi Zhang, Guosheng Lin, Fayao Liu, Jiushuang Guo, Qingyao Wu, and Rui Yao. "Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 9587–9595.

[221] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. "Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 5217–5226.

[222] Gong-Jie Zhang, Kai-Wen Cui, Rong-Liang Wu, Shi-Jian Lu, and Yong-Hong Tian. "PNPDet: Efficient few-shot detection without forgetting via plug-and-play sub-networks". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2021, pp. 3823–3832.

[223] Gongjie Zhang, Zhipeng Luo, Kaiwen Cui, Shijian Lu, and Eric P Xing. "Meta-DETR: Image-level few-shot detection with inter-class correlation exploitation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2022).

[224] Jeffrey O Zhang, Alexander Sax, Amir Zamir, Leonidas Guibas, and Jitendra Malik. "Side-tuning: a baseline for network adaptation via additive side networks". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 698–714.

[225] Jian Zhang, Cheng-Long Zhao, Bing-Bing Ni, Ming-Hao Xu, and Xiao-Kang Yang. "Variational few-shot learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 1685–1694.

[226] Jian-Wei Zhang, Yifan Sun, Yi Yang, and Wei Chen. "Feature-proxy transformer for few-shot segmentation". In: *arXiv preprint arXiv:2210.06908* (2022).

[227] Rui-Xiang Zhang, Tong Che, Zoubin Ghahramani, Yoshua Bengio, and Yang-Qiu Song. "MetaGAN: An Adversarial approach to few-Shot learning". In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 31. 2018, pp. 2371–2380.

[228] Wenqiang Zhang, Zilong Huang, Guozhong Luo, Tao Chen, Xinggang Wang, Wenyu Liu, Gang Yu, and Chunhua Shen. "TopFormer: Token Pyramid Transformer for Mobile Semantic Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 12083–12093.

[229] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas S Huang. "Sg-one: Similarity guidance network for one-shot semantic segmentation". In: *IEEE transactions on cybernetics* 50.9 (2020), pp. 3855–3865.

[230] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. "Few-shot 3d point cloud semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 8873–8882.

[231] Rongjian Zhao, Buyue Qian, Xianli Zhang, Yang Li, Rong Wei, Yang Liu, and Yinggang Pan. "Rethinking dice loss for medical image segmentation". In: *Proceedings of the IEEE International Conference on Data Mining (ICDM)*. IEEE. 2020, pp. 851–860.

[232] Chong Zhou, Chen Change Loy, and Bo Dai. "Extract free dense labels from CLIP". In: *Proceedings of the IEEE conference on European Conference on Computer Vision (ECCV)*. 2022.

[233] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. "Learning to prompt for vision-language models". In: *International Journal of Computer Vision (IJCV)* 130.9 (2022), pp. 2337–2348.

[234] Ziqin Zhou, Yinjie Lei, Bowen Zhang, Lingqiao Liu, and Yifan Liu. "Zegclip: Towards adapting clip for zero-shot semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 11175–11185.

[235] Ziqin Zhou, Bowen Zhang, Yinjie Lei, Lingqiao Liu, and Yifan Liu. "ZegCLIP: Towards Adapting CLIP for Zero-shot Semantic Segmentation". In: *arXiv preprint arXiv:2212.03588* (2022).

[236] Fangrui Zhu, Yi Zhu, Li Zhang, Chongruo Wu, Yanwei Fu, and Mu Li. "A unified efficient pyramid transformer for semantic segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 2667–2677.