THE UNIVERSITY OF SYDNEY

PH.D. THESIS

Deep Visual Learning with Less Labeled Data

Author:
Zhen ZHAO

Supervisor:
A/Prof. Luping ZHOU
Co-Supervisor:
A/Prof. Wanli OUYANG

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

School of Electrical and Information Engineering
Faculty of Engineering

November 9, 2023

Authorship Attribution Statement

Chapter 3 of this thesis is published as,

Zhen Zhao, Luping Zhou*, Lei Wang, Yinghuan Shi* and Yang Gao, “LaSSL: Label-guided Self-training for Semi-supervised Learning", *AAAI*, 2022. (**Oral**)

Chapter 4 of this thesis is published as,

Zhen Zhao, Luping Zhou*, Yue Duan, Lei Wang, Lei Qi and Yinghuan Shi*, “DC-SSL: Addressing Mismatched Class Distribution in Semi-supervised Learning", *CVPR*, 2022.

Chapter 5 of this thesis is published as,

Zhen Zhao, Sifan Long, Jimin Pi, Jingdong Wang* and Luping Zhou*, “Instance-specific and Model-adaptive Supervision for Semi-supervised Semantic Segmentation", *CVPR*, 2023.

Chapter 6 of this thesis is published as,

Zhen Zhao, Ye Liu, Meng Zhao, Di Yin, Yixuan Yuan and Luping Zhou, “Rethinking Data Perturbation and Model Stabilization for Semi-supervised Medical Image Segmentation", arxiv:2308.11903.

Chapter 2 of this thesis comprises discussions that draw upon the aforementioned publications as well as an additional paper,

Zhen Zhao, Lihe Yang, Sifan Long, Jimin Pi, Luping Zhou* and Jingdong Wang*, “Augmentation Matters: A Simple-yet-Effective Approach to Semi-supervised Semantic Segmentation", *CVPR*, 2023.

In all these publications, I proposed the methods and wrote the paper, and my primary supervisor, Associate Professor Luping Zhou, is the co-corresponding author.

In addition to the statements above, in cases where I am not the corresponding author of a published item, permission to include the published material has been granted by the corresponding author.

Student Name: Zhen Zhao

Date: November 9, 2023

As supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct.

Supervisor Name: Luping Zhou

Date: November 9, 2023

Abstract of thesis entitled

Deep Visual Learning with Less Labeled Data

Submitted by

Zhen ZHAO

for the degree of Doctor of Philosophy

at The University of Sydney

in November, 2023

The rapid development of deep learning has revolutionized various vision tasks, but the success relies heavily on supervised training with large-scale labeled datasets, which can be costly and laborious to acquire. In this context, semi-supervised learning (SSL) has emerged as a promising approach to facilitating deep visual learning with less labeled data. Despite numerous research endeavours in SSL, some technical issues, *e.g.*, the low unlabeled utilization and instance-discriminating, have not been well studied. This thesis emphasizes the cruciality of these issues and proposes new methods for semi-supervised classification (SSC) and semantic segmentation (SSS).

In SSC, recent studies are limited in excluding samples with low-confidence predictions and underutilization of label information. Hence, we propose a Label-guided Self-training approach to SSL, which exploits label information to employ a class-aware contrastive loss and buffer-aided label propagation algorithm to fully utilize all unlabeled data. Furthermore, most SSC assumes labeled and unlabeled datasets share an identical class distribution, which is hard to meet in practice. The distribution mismatch between the two sets causes severe bias and performance degradation. We thus propose the Distribution Consistency SSL to address the mismatch from a distribution perspective.

In SSS, most studies treat all unlabeled data equally and barely consider different training difficulties among unlabeled instances. We highlight instance differences and propose instance-specific and model-adaptive

supervision for SSS. We also study semi-supervised medical image segmentation, where labeled data is scarce. Unlike current increasingly complicated methods, we propose a simple yet effective approach that applies data perturbation and model stabilization strategies to boost performance.

Extensive experiments and ablation studies are conducted to verify the superiority of proposed methods on SSC and SSS benchmarks.

Deep Visual Learning with Less Labeled Data

by

Zhen Zhao

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of

Doctor of Philosophy

at

University of Sydney
November, 2023

Statement of Originality

I, Zhen ZHAO, declare that this thesis titled, "Deep Visual Learning with Less Labeled Data", which is submitted in fulfilment of the requirements for the Degree of Doctor of Philosophy, represents my own work except where due acknowledgement have been made. I further declared that it has not been previously included in a thesis, dissertation, or report submitted to this University or to any other institution for a degree, diploma or other qualifications.

Student Name: Zhen Zhao

Date: November 9, 2023

For My Big Family

Acknowledgements

This road to pursue my PhD was a difficult and trying one, but the journey was also incredibly rewarding and has left an indelible mark on my life. I am grateful for all the individuals and circumstances that have contributed to my growth and development during this experience, especially the challenging ones. I never forgot that these unexpected things, like the VISA, and unprofessional people, like my VISA agent and some paper reviewers, were almost killing me and causing me to struggle with self-doubt and uncertainty. I would not survive the process without the exceptional support of my supervisors, family, colleagues, friends, and all those who stood by me through thick and thin.

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Luping Zhou, for her invaluable advice and continuous support during my PhD study. I have to admit that my impatience and playful nature can cause significant obstacles for mentors in their efforts to supervise me. Working with a compassionate and encouraging professor like her has been a great privilege. In addition to this valuable studying opportunity at the University of Sydney, Prof. Zhou taught me to concentrate more on critical issues in the research and helped me revise papers with patience and precision. I feel fortunate that Prof. Zhou gave me the space and freedom to conduct my own research in the field of Computer Vision. Her suggestions regarding my research attitude and outlook on life will be instrumental in shaping my future path.

I would also like to thank my progress evaluation committee members, Prof. Craig Jin and Prof. Sinan Li, for their valuable feedback and constructive suggestions. In addition, my sincere thanks go to all the exceptional mentors who have provided me with invaluable guidance throughout this journey, including but not limited to Prof. Lei Wang from the University of Wollongong, Prof. Yinghuan Shi from the Nanjing University, Prof. Jingdong Wang from the Baidu VIS, Dr. Ye Liu from the Tencent Youtu Lab, Dr. Longyue Wang from the Tencent AI Lab, and Prof. Yixuan Yuan from the Chinese University of Hong Kong. I am also grateful to my colleagues, a group of brilliant scholars whom I have had the fortune to work with, for their insightful comments and stimulating suggestions during our discussions.

Last but not least, I must express my profound gratitude to my big family for their constant encouragement and genuine dedication. This thesis could not have been possible without their unconditional support. Their boundless love has been a constant source of strength for me, and I pledge to spend the rest of my life being a good father, husband, and son as a way of honouring the love.

Zhen ZHAO
University of Sydney
November 9, 2023

List of Publications

CONFERENCES:

[1] **Zhen Zhao***, Sifan Long*, Jimin Pi, Jingdong Wang, and Luping Zhou “Instance-specific and Model-adaptive Supervision for Semi-supervised Semantic Segmentation", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[2] **Zhen Zhao**, Lihe Yang, Sifan Long, Jimin Pi, Luping Zhou and Jingdong Wang, “Augmentation Matters: A Simple-yet-Effective Approach to Semi-supervised Semantic Segmentation", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023

[3] **Zhen Zhao**, Luping Zhou, Yue Duan, Lei Wang, Lei Qi, and Yinghuan Shi, “DC-SSL: Addressing Mismatched Class Distribution in Semi-supervised Learning", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[4] **Zhen Zhao**, Meng Zhao, Ye Liu, Di Yin, and Luping Zhou, “Entropy-based Optimization on Individual and Global Predictions for Semi-Supervised Learning", ACM *Multimedia*, 2023.

[5] **Zhen Zhao**, Luping Zhou, Lei Wang, Yinghuan Shi and Yang Gao, “LaSSL: Label-guided Self-training for Semi-supervised Learning", *AAAI Conference on Artificial Intelligence (AAAI)*, 2022. (**Oral**)

[6] Sifan Long*, **Zhen Zhao***, Jimin Pi, Shengsheng Wang, and Jingdong Wang, “Beyond Attentive Tokens: Incorporating Token Importance and Diversity for Efficient Vision Transformers", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[7] Sifan. Long*, **Zhen Zhao***, Junkun Yuan*, Zichang Tan, Jiangjiang Liu, Luping Zhou, Shengsheng Wang, and Jindong Wang, "Task-Oriented Multi-Modal Mutual Leaning for Vision-Language Models", IEEE *International Conference on Computer Vision (ICCV)*, 2023.

[8] Guan Gui, **Zhen Zhao**, Lei Qi, Luping Zhou, Lei Wang, and Yinghuan Shi, "Enhancing Sample Utilization through Sample Adaptive Augmentation in Semi-Supervised Learning", IEEE *International Conference on Computer Vision (ICCV)*, 2023 (**Oral**).

[9] Zicheng Wang, **Zhen Zhao**, Luping Zhou, Dong Xu, Xiaoxia Xing and Xiangyu Kong, "Conflict-Based Cross-View Consistency for Semi-Supervised Semantic Segmentation", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[10] Lihe Yang, **Zhen Zhao**, Lei Qi, Yu Qiao, Yinghuan Shi, Hengshuang Zhao, "Shrinking Class Space for Enhanced Certainty in Semi-Supervised Learning", IEEE *International Conference on Computer Vision (ICCV)*, 2023.

[11] Yue Duan, **Zhen Zhao**, Lei Qi, Luping Zhou, Lei Wang, Yinghuan Shi, "Class Transition Tracking Based Pseudo-Rectifying Guidance for Semi-supervised Learning with Non-random Missing Labels", IEEE *International Conference on Computer Vision (ICCV)*, 2023.

[12] Guan Gui, **Zhen Zhao**, Lei Qi, Luping Zhou, Lei Wang, and Yinghuan Shi, "Improving Barely Supervised Learning by Discriminating Unlabeled Samples with Super-Class", *Thirty-Sixth Conference on Neural Information Processing Systems (NeurIPS)*, 2022. (**Spotlight**)

[13] Ziyuan Wang, **Zhen Zhao**, Lei Qi, Yinghuan Shi, and Yang Gao, "Adaptive Weight of Unreliable Relation Module for Semi-supervised Multi-label Image Recognition", IEEE *International Conference on Cloud Computing and Intelligent Systems (CCIS)*, 2022.

[14] **Zhen Zhao**, Ye Liu, Meng Zhao, Di Yin, Yixuan Yuan and Luping Zhou, "Rethinking Data Perturbation and Model Stabilization for Semi-supervised Medical Image Segmentation", arxiv:2308.11903.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we first introduce the problem setting of semi-supervised learning (SSL) in Section 1.1. We then discuss the main challenges of SSL studies and present the motivations of our research in Section 1.2. Section 1.3 describes the contributions and overall structure of this thesis.

1.1 Problem Statement

In the past several years, many remarkable breakthroughs have been achieved in various computer vision tasks thanks to fast developments of deep learning [46, 46]. However, such a big success is closely dependent on constructing large-scale labeled datasets which are increasingly costly and even infeasible in some professional areas (e.g., medical and astronomical fields). In practice, unlabeled data is readily available and easy to collect, while accurately labeled data is typically hard and time-consuming to obtain. To mitigate the demand for labeled data, Semi-supervised learning (SSL) [138, 111, 114] has been proposed as a powerful approach to facilitating visual learning with less labeled data.

In standard SSL setting, we are given a partially-labeled dataset D containing a small portion of labeled samples D_l and a large number of unlabeled samples D_u, commonly, $|D_u| >> |D_l|$. Our goal in SSL is to effectively leverage the unlabeled samples to train a better performing model than the one only trained on the labeled samples. For instance, as

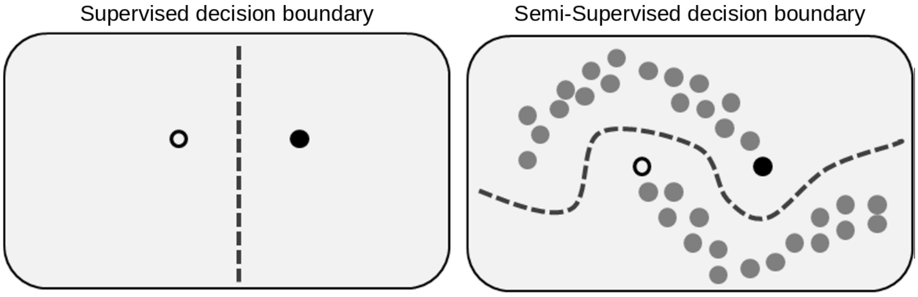

Supervised decision boundary
Semi-Supervised decision boundary

Figure 1.1: An example of the influence of unlabeled data in semi-supervised learning. From *https://en.wikipedia.org/wiki/Semi-supervised_learning*.

shown in Figure 1.1, large amounts of unlabeled data can help us estimate the data distribution, such that the decision boundary can be modified to differentiate distinct classes better. Intuitively, a lower performance bound exists when the model is only trained on the small portion of labeled data, and a desired upper performance bound that is obtained by using all accurate labels for the whole dataset D.

1.2 Challenges and Motivations

Semi-supervised learning has been researched for decades, and the essential idea is to learn from the unlabeled data to enhance the training process. However, we have no accurate label information on unlabeled data, such that we cannot directly train models on unlabeled data in a supervised fashion. To this end, many SSL methods aim to dig guidance information, *e.g.*, pseudo-labels, for the unlabeled data and cooperate with less labeled data to train models.

The main challenges lie in two aspects. On one hand, the accuracy of generated guidance information for unlabeled data, *i.e.*, high quality. It is straightforward that the accuracy pose a significant impact on the ultimate SSL performance. Noisy and even wrong guidance will hurt the training and largely degrade the performance. Many filtering strategies, *e.g.*, high-confidence threshold [83], entropy minimization [47], uncertainty estimation [163], are carefully designed to select a portion of unlabeled data with relatively reliable guidance information. On the other hand, thorough exploitation of unlabeled data, *i.e.*, high quantity. This aspect is usually ignored in many SSL studies. As shown in Figure 1.1, sufficient unlabeled data can be a strong support to seek a better decision boundary. In fact, high quantity becomes increasingly important when dealing with more challenging visual tasks, *e.g.*, classifying ImageNet [37], and segmenting Cityscapes [34]. These two aspects are commonly in conflict with each other. For instance, all the generated pseudo-labels can be directly employed to train models on unlabeled data (which constitutes the highest quantity). However, this trivial approach can prove exceedingly detrimental to the model's performance due to the low quality of guidance. Thus our objective is to seek an optimal trade-off in specific downstream tasks. In summary, the primary challenges of SSL reside in the effective and comprehensive utilization of unlabeled data.

Specifically, the motivation of this thesis mainly concentrates on the following two aspects.

In conventional semi-supervised classification (SSC), we find that current methods are limited in excluding samples with low-confidence pseudo-labels and under-utilization of label information. In the literature, a high-threshold mask is widely adopted to alleviate the confirmation bias[4], but excluding samples with low-confidence pseudo-labels results in severe inefficiencies in exploiting unlabeled data and consumes a longer training time. Besides, the label information in existing studies only contributes as a supervised loss, but its direct effects on pseudo-label generations are not explicitly considered. Furthermore, we observe that the success of SSC studies largely depends on the assumption that the labeled and unlabeled data share an identical class distribution, which is hard to meet in real practice. The distribution mismatch between the labeled and unlabeled sets can cause severe bias in SSL and result in significant performance degradation.

In semi-supervised semantic segmentation (SSS), we observe that existing studies treat all unlabeled data equally and barely consider the differences and training difficulties among unlabeled instances. In fact, requiring more dedicated supervision in segmentation tasks than SSC, further enlarges the weakness of such instance-indiscriminating strategies. Thus we emphasize the cruciality of instance differences and propose the instance-specific and model-adaptive supervision for SSS. In addition, we studied semi-supervised medical image segmentation (SSMIS) where the labeled information is even scarcer. Despite their promising performance, we find that recent studies tend to integrate increasingly complicated techniques to improve SSMIS in an indirect manner, *e.g.*, using auxiliary self-supervised tasks, employing additional feature-label pruning strategies. In contrast, our emphasis lies in the SSL problem itself, as we strive to propose a simple yet highly effective method to enhance SSMIS.

1.3 Thesis Contribution and Outline

In this section, we present major contributions and the organization of this thesis. In Chapter 2, we provide a comprehensive literature review on the deep visual learning with less labeled data and discuss how existing works are related to our methods in this thesis. Chapters 3 to 6 present our proposed methods in semi-supervised classification and semantic segmentation in detail, which are listed below. In the end, Chapter 7 concludes the thesis and discusses the possible future directions.

Chapter 3. Label-Guided Self-Training for Semi-supervised Learning In this chapter, we emphasize the cruciality of the label information in SSL and propose a Label-guided Self-training approach to Semi-supervised Learning (LaSSL), which improves pseudo-label generations from two mutually boosted strategies. First, with the ground-truth labels and iteratively-polished pseudo-labels, we explore instance relations among all samples and then minimize a class-aware contrastive loss to learn discriminative feature representations that make same-class samples gathered and different-class samples scattered. Second, on top of improved feature representations, we propagate the label information to the unlabeled samples across the potential data manifold at the feature-embedding level, which can further improve the labelling of samples with reference to their neighbours. These two strategies are seamlessly integrated and mutually promoted across the whole training process. We evaluate LaSSL on several classification benchmarks under partially labeled settings and demonstrate its superiority over the state-of-the-art approaches.

- The contributions in this part are included in:

> **Zhen Zhao**, Luping Zhou, Lei Wang, Yinghuan Shi and Yang Gao, "LaSSL: Label-guided Self-training for Semi-supervised Learning", *AAAI*, 2022. (**Oral**)

Chapter 4. DC-SSL: Addressing Mismatched Class Distribution in Semi-supervised Learning In this chapter, we discuss the distribution mismatch between the labeled and unlabeled datasets, which can cause severe bias in the pseudo-labels of SSL, resulting in significant performance degradation. To bridge this gap, we put forward a new SSL learning framework, named Distribution Consistency SSL (DC-SSL), which rectifies the pseudo-labels from a distribution perspective. The basic idea is to directly estimate a reference class distribution (RCD), which is regarded as a surrogate of the ground truth class distribution about the unlabeled data, and then improve the pseudo-labels by encouraging the predicted class distribution (PCD) of the unlabeled data to approach RCD gradually. To this end, we first revisit the Exponentially Moving Average (EMA) model and utilizes it to estimate RCD in an iteratively improved manner, which is achieved with a momentum-update scheme throughout the training procedure. On top of this, two strategies are proposed for RCD to rectify the pseudo-label prediction, respectively. They correspond to an efficient training-free scheme and a training-based alternative that generates more accurate and reliable predictions. DC-SSL is evaluated on multiple SSL benchmarks and demonstrates remarkable performance improvement over competitive methods under matched- and mismatched-distribution scenarios.

- The contributions in this part are included in:

> **Zhen Zhao**, Luping Zhou, Yue Duan, Lei Wang, Lei Qi, and Yinghuan Shi, “DC-SSL: Addressing Mismatched Class Distribution in Semi-supervised Learning", *CVPR*, 2022.

Chapter 5. Instance-specific and Model-adaptive Supervision for Semi-supervised Semantic Segmentation In this chapter, we highlight the significance of instance differences and propose an instance-specific and model-adaptive supervision for semi-supervised semantic segmentation, named iMAS. Relying on the model’s performance, iMAS employs a class-weighted symmetric intersection-over-union to evaluate quantitative hardness of each instance and supervises the training on unlabeled data in a model-adaptive manner. Specifically, iMAS learns from unlabeled instances progressively by weighing their corresponding consistency losses based on the evaluated hardness. Besides, iMAS dynamically adjusts the augmentation for each instance such that the distortion degree of augmented instances is adapted to the model’s generalization capability across the training course. Not integrating additional losses and training procedures, iMAS can obtain remarkable performance gains against current state-of-the-art approaches on segmentation benchmarks under different semi-supervised partition protocols.

- The contributions in this part are included in:

> **Zhen Zhao**, Sifan Long, Jimin Pi, Jingdong Wang, and Luping Zhou “Instance-specific and Model-adaptive Supervision for Semi-supervised Semantic Segmentation", *CVPR*, 2023.

Chapter 6. Boosting Semi-supervised Medical Image Segmentation with Data Perturbation and Model stabilization In this chapter, we study the semi-supervised Medical Image Segmentation where the labeled data is even scarce. We argue that, while current state-of-the-art methods exhibit promising performance, they usually come at the cost of introducing increasingly complex algorithms and loss terms. Differently, we tend to focus more on the semi-supervised learning itself, and we propose DPMS, a simple yet effective approach that applies strong data perturbation and model stabilization strategies to boost SSMIS performance. Specifically, it follows a clean Siamese framework with a standard supervised loss and unsupervised consistency loss. On the one hand, DPMS perturbs the unlabeled data via strong augmentations to enlarge prediction disagreements considerably. On the other hand, it utilizes a forwarding-twice and momentum updating strategies for normalization statistics to stabilize the training on unlabeled data effectively. Despite its simplicity, DPMS can obtain new state-of-the-art performance on the public ACDC and Pancreas datasets under various semi-supervised settings.

- The contributions in this part are included in:

> **Zhen Zhao**, Ye Liu, Meng Zhao, Di Yin, Yixuan Yuan and Luping Zhou, "Rethinking Data Perturbation and Model Stabilization for Semi-supervised Medical Image Segmentation", arxiv:2308.11903.
>
> **Zhen Zhao**, Lihe Yang, Sifan Long, Jimin Pi, Luping Zhou and Jingdong Wang, "Augmentation Matters: A Simple-yet-Effective Approach to Semi-supervised Semantic Segmentation", *CVPR*, 2023.

Chapter 2

Literature Review

In this chapter, we first give a brief introduction of deep supervised learning for different visual tasks. Then we present the label-efficient studies on unsupervised self-supervised learning and various semi-supervised learning methods. In the end, we introduce the popular benchmark datasets used for semi-supervised classification and semi-supervised semantic segmentation in the literature.

2.1 Supervised Deep Visual Learning

In the past several years, many remarkable breakthroughs have been achieved in different computer vision tasks thanks to the rapid development of machine learning, especially the deep learning [46]. Starting from the Alexnets [74] that achieved their record-breaking results in image classification, there have been many deep learning based methods proposed in the field of image classification and semantic segmentation.

The early LeNet [82] was the first convolutional neural network (CNN) in computer vision problems and was successfully applied to classify handwritten numbers. Despite its simplicity, LeNet has already introduced essential network components in later deep nets, like convolutional layers, pooling layers and fully connected classification layers. AlexNet [74] was the first truly deep network and achieved the breakthrough in classification tasks on ImageNet. It utilizes stacked convolutional layers and pooling layers to significantly improve classification accuracy. The Rectified Linear Units (ReLU) were introduced as an effective activation function. Since then, AlexNet has motivated a lot of following studies in this field. For example, two follow-up studies, GoogLeNet [133] and VGGNet [129], won the first and the second-ranked networks in 2014, respectively. The former has a top-5 classification error rate of 6.7%. Its core lies in the Inception Module, which uses a parallel approach. The latter includes two versions with 16 and 19 layers, containing a total of about 550 M parameters. Its convolutional neural network structure is simplified by using all 3×3 convolutional kernels and 2×2 max-pooling kernels. In 2015, ResNet [54] won the classification task championship. It outperformed human recognition with an error rate of 3.57% and set a new model record with a 152-layer network architecture. ResNet adopts cross-layer connectivity (residual connection) to successfully alleviate the gradient vanishing problem in deep neural networks. The following-up ResNeXt citeresnext adopts grouped convolution and achieves the comparable accuracy of ResNet152 with half of the complexity. The success of transformer [139] in language models has inspired many recent studies to explore the Transformer blocks to solve the vision problems [19, 40, 95]. All these models have significantly contributed to deep learning's progress in image classification tasks.

Unlike classification tasks in which the whole image belongs to a single semantic, semantic segmentation must classify each pixel, known as dense prediction tasks in computer vision. In the era of deep learning, the pioneering work in this field is FCN [96], which introduced an encoder-decoder architecture based on fully convolutional networks for pixel-wise semantic segmentation. This groundbreaking approach has inspired numerous subsequent methods employing similar architectures, including the SegNet [5], ENet [118], DenseNet [64], the successful DeepLab series [25, 24, 26, 27]. The Deeplab series have significantly advanced the field of semantic segmentation by effectively capturing fine-grained details and producing high-quality segmentation results. Using the ResNet as the encoder, DeepLab v1 [25] further introduced the idea of atrous (dilated) convolutions to capture multi-scale information while maintaining computational efficiency. DeepLab v2 [24] carefully designed the atrous spatial pyramid pooling (ASPP) model and also incorporated a fully connected conditional random field (CRF) as a post-processing step to refine the segmentation boundaries. DeepLab v3 [26] further designed the "asymmetric" atrous convolutions, which have different dilation rates in the horizontal and vertical directions. DeepLab v3+ [27] inherited the success of former explorations, dropped the post-processing CRF, and adopted the encoder-decoder structure to improve the segmentation performance further. In recent years, the Transformer model has achieved remarkable success [139, 40, 19], leading researchers to explore the potential of attention mechanisms in semantic segmentation to capture long-range contextual information. SETR [178] tended to process semantic segmentation as a sequence-to-sequence prediction perspective. Its transformer-based design effectively solves the limited receptive field challenge of FCN-based methods. SegFormer [149] prevented complex designs in previous methods and proposed a simple yet effective solution for semantic segmentation, which consists of a positional-encoding-free, hierarchical Transformer encoder and a lightweight AllMLP decoder. Recent SegViT [167] successfully integrated the spatial information in its Attention-to-mask (ATM) module and designed a plain ViT transformer-based method to further boost the segmentation performance. Despite their exceptional performance, as we discussed in Chapter 1, these methods commonly required large-scale, high-quality labeled datasets, which can be costly and even infeasible to obtain.

Most of deep segmentation methods leverage a pre-trained ResNet [54] as the backbone encoder to extract semantic information and employ diverse decoders to generate dense predictions. Differently, medical images have some special properties like scarce labeled data, and fine-grained classes, and smooth boundaries. To address these challenges, numerous studies have devised specialized methods for medical image segmentation, which can be classified into two primary categories. The first focuses mainly on designing medical-specific network architecture, like the widely used UNet [127] and vnet [104], which designs a fully convolutional network that is trained end-to-end to capture multi-level semantic features to perform dense predictions. The second tends to propose medical-specific loss functions, like the Dice loss [104], which utilizes the dice coefficient to tackle the class imbalance problem. Nevertheless, the majority of these approaches heavily depend on extensively annotated medical image datasets, which necessitates significant effort from expert annotators to obtain precise annotations [141].

2.2 Self-supervised Deep Visual Learning

Self-supervised deep visual learning [93, 63] is a new learning approach that aims to train deep neural networks for visual tasks without relying on human-labeled supervision. Instead of using manually annotated labels, self-supervised learning leverages the inherent structure and information within the data itself to learn useful representations or features. There are roughly two popular strategies that are widely explored in label-efficient studies.

The first is to design pretext tasks, also known as proxy tasks or auxiliary tasks. These pretext tasks are constructed by creating surrogate supervisory signals from the input data, without the need for human annotations. It can then encourage the model to learn meaningful representations to solve these specific tasks, such as colorization [81, 80], clustering [150, 157], channel prediction [170], jigsaw puzzles [71, 109], image inpainting [119], *etc*. These tasks are commonly designed to help the model capture relevant and useful information about the data distribution. As a result, these methods learns to extract high-level features that capture important visual cues, which enables the model to generalize well to other tasks that are related to the original task.

The second is to employ contrastive learning. Recent studies along this line have presented promising results to directly leverage the unlabeled data [63, 53, 29, 30]. Such methods exploit the similarity and dissimilarity among different data instances for representation learning, which essentially encourage similar feature representations between two random crops from the same image and distinct representations among different images. By learning to distinguish between different augmentations of the same data, the model can capture the invariant and discriminative aspects of the data. However, these approaches rely heavily on the assumption of instance discrimination, where each image instance is considered to be a distinct class. The assumption limits its application in semi-supervised learning, and we improve the standard contrastive learning with a pseudo-label aided class-aware design in Chapter 3.

2.3 Large Foundation Models

Recently, we have witnessed rapid advancements in various large foundational models, particularly the emergence of large language models (LLMs) [112] and vision-language models (VLMs) [123]. Their impressive perceptual and reasoning capabilities bring new possibilities as well as formidable challenges to a wide range of artificial intelligence tasks. In this section, we briefly discuss how such foundation models can mutually benefit and affect semi-supervised visual studies.

Large language models, such as ChatGPT [116], Bloom [128], LLaMA [136] and PaLM [32], have significantly broadened the boundaries of language comprehension and generation, exhibiting remarkable human-like language capabilities in complex reasoning. On top of these text-only LLMs, recent multi-modal studies like LLava [91] and MiniGPT-4 [183], aim to leverage the strong reasoning ability of LLMs to solve the visual problems. These works [86, 84, 168, 88] typically follow a two-step training paradigm: 1) to train the visual converter to align the semantics of the vision encoder and LLM, 2) to construct an instruction-following dataset to further fine-tune the whole model. Their effectiveness is closely dependent on the generated instruction-following data. Furthermore, as we all know, Reinforcement Learning from Human Feedback (RLHF) [117] is crucial to enhancing the performance of LLMs. However, training a reliable reward model is not a straightforward task and typically requires a significant amount of accurate labeled data. In this regard, semi-supervised learning, being a data-efficient strategy, can serve as a potential solution to further improve the performance of LLMs or VLMs.

Recent large vision models, such as SAM [72] and DINO [113], have also achieved impressive improvements in visual tasks. Especially, SAM is the first foundation model for image segmentation, providing the powerful segment anything model as well as a 1-billion mask dataset. Utilizing a unified user interface prompt, SAM can segment any objects within both images and videos, eliminating the requirement for additional training. Acknowledging its exceptional segmentation capabilities, SAM has been widely applied in diverse AI domains and applications, such as remote sensing [23], medical segmentation [100], robotics [59], tracking [158], and more. Considering that SAM cannot produce the specific semantics directly, subsequent studies such as semantic-SAM[85] and ground-SAM [49] tend to enhance SAM's semantic understanding by training the additional classifier. However, certain segmentation tasks, like shadow segmentation and camouflaged object detection, pose notable challenges for SAM, potentially leading to limitations or unsatisfied performance, as discussed in [28, 65]. For example, when dealing with medical images that often exhibit smooth boundaries, SAM faces difficulties in directly achieving successful segmentation of tumors and biological organs [100].

When considering collaboration with SSL, SAM's remarkable generalization ability emerges as a promising approach to refine pseudo-labels in semi- or weakly-supervised segmentation. Exploring the synergistic potential between SAM and semi-supervised semantic segmentation is an avenue for future research. This entails a mutually beneficial design where the semi-supervised algorithm can provide initial segmentation masks (followed by some specific filtering mechanisms), which can serve as highly effective visual prompts for SAM. Simultaneously, the segmentation masks generated by SAM can offer comprehensive and accurate indications of pixel-level relationships, thereby serving as valuable supplementary information for further refining the pseudo-labels. This reciprocal interaction between SAM and semi-supervised techniques holds substantial potential for enhancing the overall performance of the segmentation process.

2.4 Semi-supervised Deep Visual Learning

In this section, we first present the conventional semi-supervised learning algorithms and then introduce the most recent studies related to our work in the field of semi-supervised learning.

2.4.1 Conventional Semi-supervised Learning

Early attempts [2, 42] in semi-supervised research can be traced back to the 1970s, and since then, the goal of SSL has always been to improve the learning performance by leveraging additional unlabeled instances. Given that SSL has consistently provided an effective solution to alleviate the labeling burdens, numerous studies on this topic [185] have been conducted even prior to the era of deep learning. Some key traditional methods [185, 160] in semi-supervised learning continue to be widely adopted or have inspired recent research, including but not limited to, graph-based methods [184, 8, 14], semi-supervised support vector machines [68, 154, 155], self- and co-training [101, 15], and generative models [103]. In generative SSL studies [108], the Expectation-Maximization (EM) algorithm was widely applied. It alternates between estimating the model parameters using the labeled data and estimating the missing labels for the unlabeled data. Semi-supervised SVMs [10] extended traditional SVMs to incorporate unlabeled data and proposed a penalty term for misclassifying unlabeled data points. Graph-based SSL methods [180] represent data as nodes in a graph, where edges encode relationships or similarities between data points. Label propagation [62, 171] methods, one of the most straightforward graph-based SSL methods, treat the labeled data as anchor points and propagate labels to nearby unlabeled data points based on a similarity metric. Self-training is training the model with a small set of labeled data, using the model to make predictions on unlabeled data, and subsequently adding the most confident predictions to the labeled set. Co-training involves training two or more models simultaneously and providing supervision for each other. Despite their simplicity, such studies are still widely adopted in recent semi-supervised classification [152, 177] and semi-supervised segmentation [159, 31]. In contemporary research, deep neural networks have emerged as a dominant force across various research fields. Given the advantages and challenges associated with deep models, it becomes crucial to embrace the conventional SSL methods and explore novel SSL methods tailored for deep learning scenarios [160].

2.4.2 Semi-supervised Classification

Semi-supervised learning has been researched for decades, and the essential idea is to learn from the unlabeled data to enhance the training process. Therefore, many SSL methods aim to dig guidance information for the unlabeled data and cooperate with less labeled data to train models. Current dominant methods tend to propose pseudo-labels on unlabeled data [111, 114], either for self-training-based or consistency-based SSL approaches (which take the prediction of one augmented crop as the pseudo-label for the other).

Self-training-based approaches [83, 4, 101, 156] first train on the small amount of labeled data and then make predictions on unlabeled data in a form of probability distributions over the classes. Next, the unlabeled data and their corresponding pseudo-labels will be added to the labeled data if the maximal probability of the predicted pseudo-labels is higher than a predefined threshold (i.e. high confidence). After that, these approaches train on the augmented labeled data and infer on the remaining unlabeled data, repeating the process until the model is able to make confident predictions. Some works in [16, 122, 39] extended the self-training from single model and single view to multiple models and multiple views, aiming to propose more confident pseudo-labels. The main weakness of such approaches is that the model cannot effectively handle wrong pseudo-labels, and the errors may quickly be accumulated, resulting in performance degradation.

Based on the clustering assumption that *if points are in the same cluster, they are likely to be of the same class* [22], many consistency-regularization (CR) based SSL approaches [124, 79, 134, 13, 76, 132, 176] have been proposed recently. As shown in Figure 2.1, these approaches primarily encourage invariant predictions on two perturbed inputs derived from a

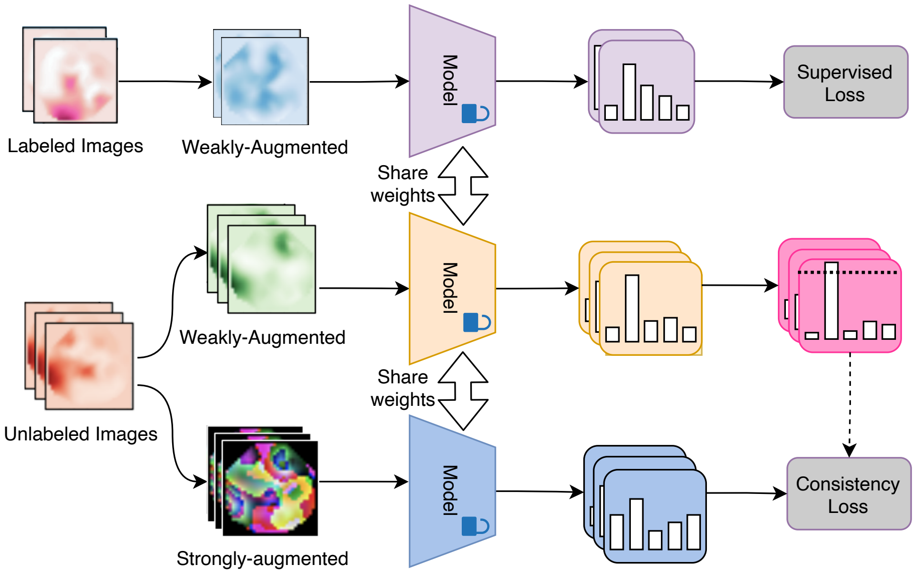


Figure 2.1: FixMatch, one of the most simple but effective SSL methods. At each iteration, a single unlabeled image will generate two random crops, a weakly-augmented one (e.g. image translates, flip-and-shift) and a strong-augmented one (e.g. RandAug [35], CTAug [12]). The predictions on the weakly-augmented instances will be regarded as the pseudo-label for their corresponding strongly-augmented variants. Note that only the unlabeled data with high-confidence predictions will be involved in the training process, which is critical to alleviating the confirmation bias [4]. Meanwhile, the labeled data is used to update the model via a supervised loss (e.g., a cross-entropy loss) at each iteration.

single image, which can also be regarded as pseudo-labelling one input for the other. Typical approaches such as Ladder Network [124] and Π model [79] applied Gaussian noise and random translation transformations to generate two different views and enforced consistency between the predictions of them. Different from self-training-based approaches, CR-based approaches simultaneously train the labeled and unlabeled data at the iteration level. In this way, the selected unlabeled data in the current iteration will not directly affect the training in the next iteration, and the potential errors will not be accumulated as before. Authors in [134] proposed a Mean Teacher model and highlighted the quality of pseudo-labels in SSL. In specific, it introduced a weight-averaged teacher model to generate more robust targets for unlabeled data. After that, many works [106, 140, 151] extensively explored various data augmentation strategies for SSL training and drew a vital conclusion that stronger and more realistic data augmentation strategies were beneficial and necessary. Holistic approaches like MixMatch [13], ReMixMatch [12] and FixMatch [132] combined these findings and integrated other useful techniques, such as MixUp [169], entropy minimization [47], distribution alignment (DA) [17] into an unified framework, resulting in better performance. However, the correlation between labeled and unlabeled data and the relationship among different unlabeled instances are ignored in these approaches.

More recent works have been proposed to further improve the SSL performance through introducing other deep learning techniques on top of the FixMatch method. Xu. et al. [153] replaced the fixed high-confidence threshold with a time-dependent threshold that is gradually increased from a low value to one. SelfMatch [70] designed a two-stage approach to combine the power of contrastive self-supervised learning and consistency-based semi-supervised learning. Authors in [1] exploited the pre-trained model from source domain and intended to improve SSL performance via transfer learning techniques. Most complicated, [87] unified the ideas of consistency regularization, entropy minimization, contrastive learning, distribution alignment, and graph-based SSL, and proposed Co-Match to jointly train two contrastive representations on unlabeled data and smooth the pseudo-labels under the help of a large memory bank. Despite their promising results, progresses along this line often involve complicated network structures, heavy computations, and more hyper-parameters.

In addition to those two main methods, there are some SSL works on the graph-based and generative-based methods. In the graph-based methods, the labeled and unlabeled samples are treated as nodes of a graph, and the SSL task can be accomplished by propagating the labels from the labeled nodes to the unlabeled ones by utilizing the instance similarity. Recent papers [62, 171] adopt this idea and integrate the label propagation techniques in deep semi-supervised learning. As for the generative-based methods, authors in [110] propose Semi-Supervised

Generative Adversarial Networks (GAN), which can make full use of GAN to learn good feature representation from the unlabeled data, and simultaneously train a generative model and a classifier. Motivated by BiGAN [38], Augmented-BiGAN [75] first uses the generator to estimate tangents of potential data manifold and then employ it to inject invariances into the model's classifier, resulting in impressive accuracy gains. However, few follow-up works have been proposed based on these approaches, simply because of their poorer performance compared to the above two main approaches.

2.4.3 Semi-supervised Semantic Segmentation

Motivated by the progress in semi-supervised classification, some studies aim to achieve dense segmentation performance with only a fraction of labels. Similar to SSC, SSS studies are also built on top of two main branches, *i.e.*, the self-training based methods [159], and consistency regularization based methods [145]. Based on our summary, as shown in Table 2.1, recent advanced methods tend to enhance the SSS performance from three different directions, including “augmentations", “more supervision," and “pseudo-rectifying". Almost all existing studies applied various strong data augmentations to perturb unlabeled data while some of them [115, 94] also perturbed the inputs at the feature level. In the branch of “more supervision", multiple training branches, training stages, or losses (MBSL) are widely adopted from the perspective of model perturbations [57, 159, 94, 78]. As the quality of pseudo-labels is critical for semi-supervised training [177, 175], ECS [102] and ELN [77] also introduced additional trainable correcting networks (ACN) to further polish the pseudo-labels. Despite of their promising performance, recent state-of-the-art methods [144, 94, 50, 77] usually come at the cost of combining increasingly complex mechanisms, *e.g.* contrastive learning [93], or multiple ensembling models. Differenly, we aim to propose a simple and clean method to boost the SSS performance. Specifically, to the best of our knowledge, all the existing studies indiscriminately perturb unlabeled samples and minimize an average consistency loss over all unlabeled samples. On the contrary, we differentiate different samples

Method	Augmentations		More Supervision			Pseudo-rectifying		
	SDA	FT	MBSL	CT	UCL	UAFS	ACN	PR
CCT [115]		✓	✓	✓				
SCN [61]				✓			✓	
ECS [102]			✓				✓	
UAMT [163]	✓					✓	-	-
SSMT [57]	✓		✓			✓		
PseudoSeg [186]	✓					✓		
CAC [78]			✓		✓	✓		
DARS [55]	✓		✓					✓
AEL [56]	✓							✓
PC^{2}Seg [179]	✓		✓		✓	✓		
C3-Semiseg [182]	✓				✓	✓		✓
SimpleBase [164]	✓		✓			✓		
ReCo [92]	✓				✓	✓		
CPS [31]	✓			✓				
ST++ [159]	✓		✓					
ELN [77]	✓		✓				✓	
USRN [50]	✓		✓			✓		✓
PSMT [94]	✓	✓	✓			✓		
U^{2}PL [144]	✓				✓	✓		✓

Table 2.1: Comparison of recent SSS algorithms in terms of “Augmentations", “More supervision", and “Pseudo-rectifying" (sorted by their publication date). We explain the abbreviations as follows. “**SDA**": Strong data augmentations, including various intensity-based and cutmix-related augmentations, “**FT**": Feature-based augmentations, “**MBSL**": multiple branches, training stages, or losses, “**CT**": Co-training, “**UCL**": unsupervised contrastive learning, “**UAFS**": uncertainty/attention filtering/sampling, “**ACN**": additional correcting networks, “**PR**": prior-based re-balancing techniques. **Note that**, branches of “more supervision" and “pseudo-rectifying" typically require more training efforts. Differently, our method enjoys the best simplicity but the highest performance.

in terms of the learning difficulty, evaluated as instance hardness. We utilize the hardness to guide the training process and achieve new state-of-the-art performance on several segmentation benchmarks.

On the other hand, instance hardness [131, 121, 130, 20] has been widely studied in hard example mining [165] and curriculum learning [181]. Their evaluation mainly depends on the instantaneous or historical training losses with respect to ground truths. Lacking accurate label information makes hardness measurements of unlabeled instances much more challenging. Some works [165, 67, 143] perform hardness analysis on unlabeled data to split all the samples into the hard and easy groups by sorting or ranking the hardness with a predefined threshold. Such methods only require **qualitative** analysis for selecting or filtering purposes. However, specific **quantitative** hardness analysis, especially on segmentation tasks, is still under-explored. In our proposed method, we need the quantitative hardness to determine the mixup between strongly and weakly augmented crops, as well as the exact unsupervised loss weight for each unlabeled instance. Thus we propose a new class-weighted symmetric metric to evaluate the hardness of unlabeled instances in segmentation tasks.

As for semi-supervised medical image segmentation (SSMIS), most of recent studies follow the same designing ideas as in natural domain [163, 147, 7, 148, 98, 173]. UA-MT [163] uses a mean-teacher (MT) framework and encourages the student model to gradually generate consistent predictions as the teacher model based on the proposed uncertainty-aware training scheme. SASSNet [89] further enforces a geometric shape constraint upon the segmentation outputs. DTC [97] designs an additional task-level constraint into a dual task-consistency framework. Recent state-of-the-art methods tend to introduce more advance techniques to further improve SSMIS performance. MC-Net [147] perturbs the predictions with multiple different decoders and encourages the prediction consistency between the perturbed decoders. Authors in [120] propose to pre-train the image encoder with meta-labels and then introduce an

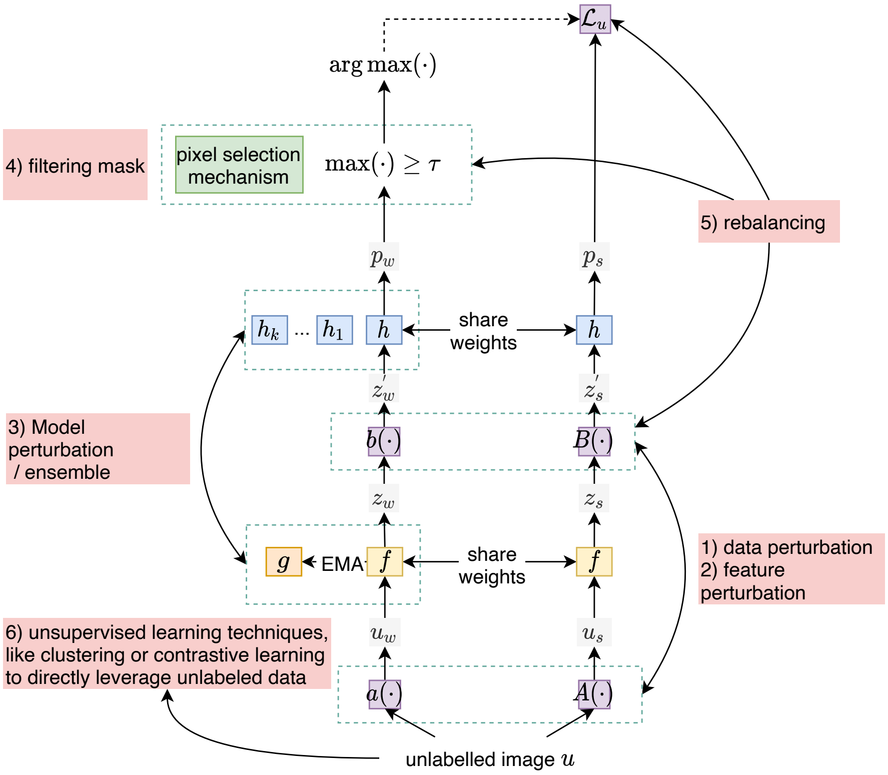


Figure 2.2: Different perspectives to improve SSL. Based on our explorations, there are main six directions: 1) design appropriate data perturbations 2) employ feature-level perturbations like VAT, dropouts 3) apply model-level perturbations or multiple model ensembling 4) propose different pseudo-label filtering strategies, like thresholds, uncertainty estimations. 5) introduce proper rebalancing techniques to the input or model levels considering the imbalance natures in practice. 6) integrate various unsupervised techniques, especially self-supervised contrastive learning, to directly leverage unlabeled data.

extra self-paced contrastive learning in semi-supervised framework. CT-CT [98] introduces an extra transformer branch and encourages prediction consistency between the CNN model and the Transformer model to enable the model to benefit from the two learning paradigms. SS-Net [148] employs the feature-level virtual adversarial training (VAT) and prototype-level contrastive losses to achieve promising performance. Despite their impressive performance, we clearly observe that SSMIS studies along this line come at the cost of introducing more complex techniques, e.g., extra network structures or additional training procedures and losses. Differently, in this work, we redirects the focus towards the semi-supervised problem itself, and highlight the cruciality of data perturbation and model stabilization to generate substantial and appropriate prediction disagreement in SSMIS.

2.4.4 Summary

Based on our understanding, in Figure 2.2, we provide a summary of recent advanced methods on pseudo-labeling based semi-supervised learning. Though the figure is built on top of the consistency regularization (CR) based methods, the critical improvement strategies in self-training based methods, essentially enjoy the same idea.

As we discussed in before, the key of SSL studies lie in the effective and comprehensive utilization of unlabeled data. The most straightforward and effective way is to generate pseudo-labels for unlabeled data, which is also currently dominant strategies in SSL studies. To train models on the labeled and unlabeled data simultaneously, CR-based methods designed a simple framework and directly encourage the prediction consistency between two differently augmented views of the same unlabeled image. The potential logic is that the different model's prediction derived from the same unlabeled instance are supposed to have the same semantic outputs. Therefore, the key to such pseudo-labeling based studies is to produce **prediction disagreement** [172, 174]. As shown in Figure 2.2, the top 3 research directions is to generate prediction disagreement from the data level, feature level and the model level. Certainly, one of the simplest approaches is to apply various label-preserving perturbations. In the model level, ensemble techniques are also widely applied, like the PI model [79], and well-known Mean-teacher [134].

In addition, one of the critical factors in SSL studies is the accuracy of pseudo-labels, *i.e.*, the filtering strategies in Figure 2.2. Without specific designs to refine the pseudo-labels, the SSL performance can be significantly degraded due to the accumulated errors (also known as confirmation bias [4]). It also becomes one of the main challenges in SSL studies, and many strategies have been proposed accordingly, like uncertainty estimations [163], sharpening [13], and high-confidence thresholds [132]. Some studies also focus on the re-balancing techniques to address the prediction collapsing [51] and long-tail issues in segmentation [55]. Techniques like the distribution alignment [12], Sinkhorn-Knopp algorithm [36], have been widely applied to encourage balancing predictions in a explicit or implicit manner.

Certainly, various unsupervised learning algorithms can be utilized to directly harness the unlabeled data, as depicted in Figure 2.2. Inspired by the remarkable achievements of self-supervised learning, numerous studies have focused on integrating standard or modified contrastive learning into the framework of SSL. By applying the contrastive loss at the feature embedding level, enhanced representation capabilities can be achieved, thereby indirectly improving the SSL performance. The primary advantage of this approach is that it maximizes the utilization of all unlabeled data, resulting in the highest possible leverage of such data.

2.5 Public Dataset

We examine the performance of our proposed methods on popular classification and segmentation benchmark datasets.

For semi-supervised classification, we conduct experiments on five public classification datasets, including CIFAR-10 [73], CIFAR-100 [73], SVHN [107], STL-10[33], and Mini-Imagenet [125].

- CIFAR-10 and CIFAR-100 are labeled subsets of the 80 million tiny images dataset with 10 and 100 classes, respectively. Both of them contain 50000 32x32 training images and 10000 32x32 testing images. In CIFAR-100, the classes are organized into 20 superclasses, each containing five fine-grained classes.
- SVHN (Street View House Numbers) consists of 10-class colorful 32x32 house numbers. It has 73257 training images and 26032 testing images, which are obtained from house numbers in Google Street View images.

- STL-10 is a popular image classification dataset commonly used in computer vision research. It is an extension of the CIFAR-10 with a larger number of classes and higher resolution images. It is composed of 5,000 labeled images of size 96x96 from 10 classes, along with 10,000 unlabeled images.
- Sampled from ImageNet ILSVRC, Mini-ImageNet consists of 50000 training images and 10000 testing images, evenly distributed across 100 classes. The images are RGB color images with a resolution of 84x84 pixels. This dataset is particularly valuable for evaluating models' ability to learn from limited labeled examples and generalize to unseen classes.

For semi-supervised semantic segmentation, we examine the superiority of our propose method on following public datasets, including two natural images and two medical images,

- Pascal VOC2012 [41]. It is a standard semantic segmentation benchmark with 21 semantic classes (including the background). The classic VOC 2012 includes 1,464 fine-labeled training images and 1,449 validating images. As a common practice, the blended training set is also involved, including additional 9118 training images from the Segmentation Boundary (SBD) dataset [52].
- Cityscapes [34]. It is a large dataset on urban street scenes with 19 segmentation classes. Cityscapes contains images captured from a car-mounted camera, simulating the viewpoint of a driver or an autonomous vehicle. The images cover a variety of urban scenes, including streets, intersections, sidewalks, buildings, vehicles, and pedestrians. Specifically, it consists of 2975 training and 500 validation images with fine annotations. It commonly serves as a benchmark for evaluating and comparing state-of-the-art algorithms in urban scene understanding and semantic segmentation.
- ACDC [11] (Automated Cardiac Diagnosis Challenge) is a medical dataset focused on cardiac image analysis, specifically targeting the

assessment of cardiac function. It consists of cardiac magnetic resonance imaging (MRI) scans acquired from patients with four different cardiac conditions. Each MRI scan is typically represented as a stack of 2D images acquired at different time points during the cardiac cycle. The dataset contains 100 MRI scans from 100 patients, divided into 3 sets: training (70 scans), validation (10 scans), and testing (20 scans).

- LA (left atrium) dataset is constructed from the Atrial Segmentation Challenge dataset [1], which consists of a collection of 100 3D gadolinium-enhanced MR imaging scans (GE-MRIs). Following UA-MT [163], we also split the 100 scans into 80 scans for training and 20 scans for evaluation.

[1] http://atriaseg2018.cardiacatlas.org/

Chapter 3

Label-guided Self-training for Semi-supervised Learning

In this chapter, we focus on the conventional semi-supervised classification (SSC) task and aim at enhancing the utilization of all available unlabeled data and harnessing the instance relationships to maximize the value of the labeled information in SSC. We propose a Label-guided Self-training approach to semi-supervised learning (SSL), doubted as LaSSL, and verity its superiority on classification benchmarks.

3.1 Introduction

The principal idea of SSL is to dig guidance information for the unlabeled data and cooperate with few labeled data to train models. Current state-of-the-art (SOTA) SSL approaches, either the classic self-training-based [83, 4, 156] or the more recent consistency-based approaches [134, 106, 13, 12, 132], largely rely on the pseudo-labelling of the unlabeled data [114]. The former approaches first train the model based on the labeled data and then use the model's predictions on unlabeled data as pseudo-labels. Differently, the latter approaches usually generate two crops from a single image via data perturbations and take the prediction of one crop as the pseudo-label for the other. Such approaches commonly adopt a high-threshold mask to alleviate the confirmation bias[4], but excluding samples with low-confidence pseudo-labels results in severe inefficiencies in exploiting unlabeled data and consumes a longer training time. More importantly, the label information in such approaches only

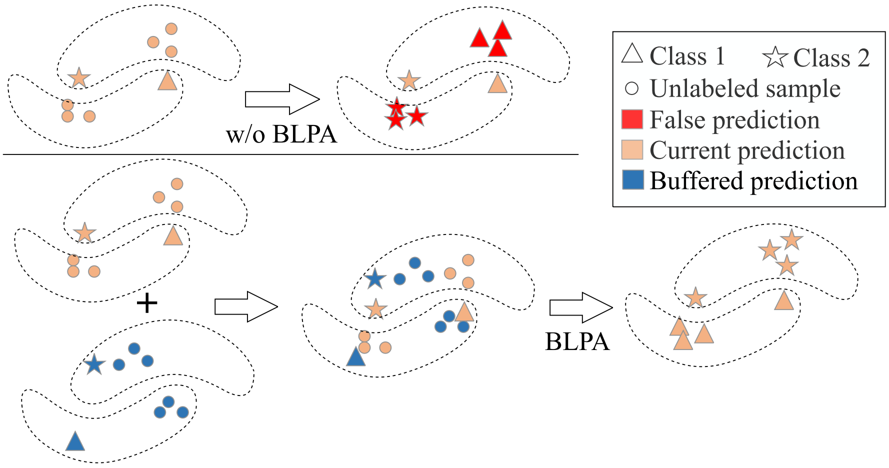


Figure 3.1: Buffer-aided label propagation algorithm (BLPA) utilizes the buffered labeled data to increase label information and the unlabeled data to enhance the potential manifold. Therefore, BLPA is more accurate compared to the standard distance-based labeling.

contributes as a supervised loss, but its direct effects on pseudo-label generations are not explicitly considered.

Inspired by the observed limitations of the existing SSL approaches as above, in this chapter, we propose LaSSL, a **La**bel-guided **S**elf-training approach to **S**emi-supervised **L**earning. The term “label-guided" emphasizes the full exploitation of label information based on sample relations, which is achieved by two intrinsically connected strategies aiming at improving the generation of pseudo-labels.

Firstly, given the potential semantic content carried by ground-truth labels and pseudo-labels, LaSSL obtains the instance relations at the prediction level and explores a better feature embedding through a proposed class-aware contrastive loss, so that the same-class samples are gathered and the different-class samples are scattered. Consequently, all the unlabeled samples are involved. At the same time, better feature representations also indirectly benefit the quality of pseudo-labels. Our approach differs from the assumption of instance discrimination in contrastive learning[63], where each image instance is treated as a distinct class of its own.

Secondly, on top of the sample relations improved by the revised

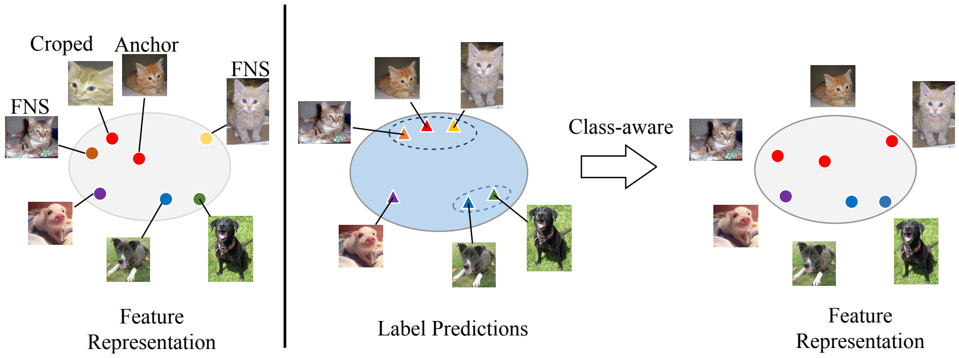


Figure 3.2: Typical contrastive learning (the left part) is based on the instance discrimination. Only the anchor and its augmented crop are considered similar, while all the other instances are treated to be distinct classes. Obviously, there may exist many false negative samples (FNS). Differently, the class-aware contrastive loss (CACL) makes full use of the label information to explore the instance relationships and make contrastive learning more reasonable.

contrastive learning, we propagate the labels from the labeled samples to the unlabeled ones across the underlying data manifold via the label propagation algorithm (LPA) at the feature-embedding level. In this way, we could take advantage of the correlation between the labeled and unlabeled samples to improve pseudo-label generation. Since performing LPA on all unlabeled data (i.e., at the epoch level) is computationally inefficient and even infeasible on large datasets, in LaSSL, we perform label propagation at each mini-batch (i.e., at the iteration level), with the aid from the buffered outputs of the last iteration. The buffered data with high confidence are treated as labeled data in the current LPA prediction, providing more label information, while the buffered data with low confidence are treated as unlabeled data, helping explore the potential manifold. In addition, we perform the bagging technique on the buffered data to further reduce the impact of potential noise pseudo-labels. Figures 3.1 and 3.2 shows graphic explanations of these two strategies accordingly.

In summary, better pseudo-labels make the class-aware contrastive loss more reasonable and accurate; simultaneously, the class-aware contrastive training leads to more discriminative feature representations,

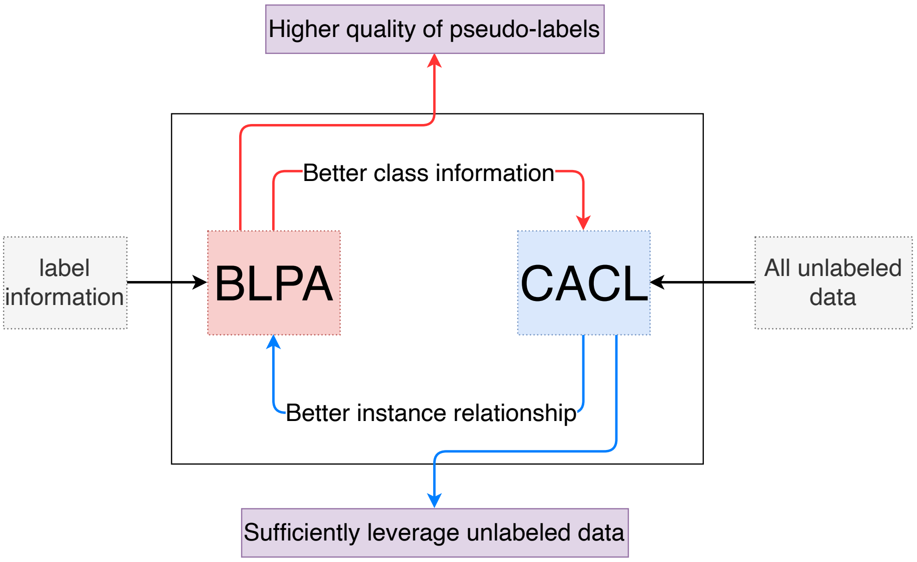

Higher quality of pseudo-labels
Better class information
label information
BLPA
CACL
All unlabeled data
Better instance relationship
Sufficiently leverage unlabeled data

Figure 3.3: Key to LaSSL: mutually-boosted designs.

which in turn can be used to polish pseudo-labels via LPA at the feature-embedding level. Therefore, unlike previous works [87, 62], our proposed two strategies are tightly coupled and mutually promoted across the whole training process. This mutually boosted design, as illustrated in Figure 3.3, is the core of LaSSL's success.

3.2 Method

In this section, we first introduce our proposed LaSSL at a high level and then present its components in detail. The full algorithm is shown in Algorithm 1.

3.2.1 Overview

Unlike typical SSL approaches, in addition to the encoder $h(\cdot)$ and predictor $f(\cdot)$, LaSSL also integrates a projector $g(\cdot)$ to learn feature representations. For simplicity, we use $F(\cdot) = f \circ h(\cdot)$ for the final prediction output and $G(\cdot) = g \circ h(\cdot)$ for the final projection output. Following the standard framework of self-training, LaSSL consists of two phases, the inference phase and training phase at each iteration, as illustrated in Figure 3.4 and 3.5.

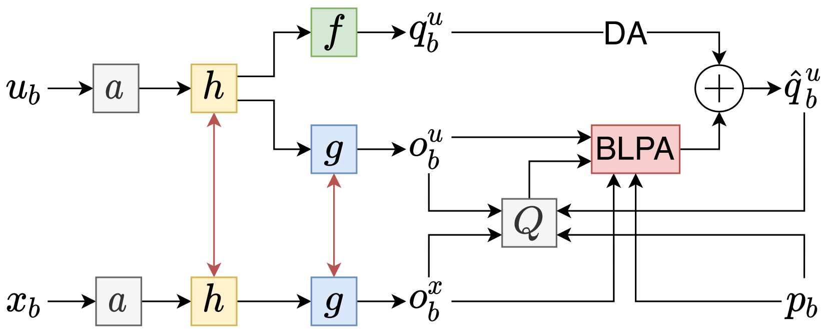


Figure 3.4: Infer on unlabeled samples and polish the pseudo-labels by BLPA under the help of labeled samples. The red two-way arrows represent “sharing weights". (x_b, p_b) denote a batch of labeled samples. $h(\cdot)$, $f(\cdot)$, and $g(\cdot)$ represent the encoder, the predictor, and the projector used in LaSSL, respectively. A and a represent the strong and weak augmentation, respectively. o_b^x and o_b^u represent the labeled and unlabeled buffered feature embedding, respectively. q_b^u and $\hat{q}_b^u$ represent the initial pseudo-labels and ultimate revised pseudo-labels for unlabeled data u_b.

Labeled data $\mathcal{X}$ and unlabeled data $\mathcal{U}$ are given in an N-class classification task. Let (x_b, p_b) be a batch of B labeled samples and u_b be a batch of μB unlabeled samples where μ denotes the size ratio of x_b to u_b. Referring to [132], we also introduce the weak and strong augmentations in LaSSL, denoted as $a(\cdot)$ and $A(\cdot)$, respectively.

Inference Phase

In the inference phase, as shown in Fig. 3.4, the main task is to generate pseudo-labels on unlabeled data and the model is not updated. Different from the standard self-training, we also infer on the labeled data. Given the unlabeled u_b and labeled x_b, we can have the projection outputs $o_b^u = G(a(u_b))$ and $o_b^x = G(a(x_b))$, respectively, and the prediction output $q_b^u = F(a(u_b))$, i.e. the pseudo-label. In addition, we maintain a First-in-First-out queue, denoted by Q, which only stores the outputs from the last iteration. This is simply because the most recent predictions are more convincing during the training. To be specific, at the i-th iteration, we have $Q_i = \{(o_b, q_b)\}$ where $o_b \in \{o_b^u\} \cup \{o_b^x\}$, $q_b \in \{\hat{q}_b^u\} \cup \{p_b\}$. Correspondingly, the dequeue data at the i-th iteration will be Q_{i-1}.

Algorithm 1: LaSSL algorithm at each iteration

```
Input: labeled data $(x_b, p_b)$, unlabeled data $u_b$, weight $\lambda_c$
Parameter: pseudo-label threshold $\tau$, similarity threshold $\varepsilon$,
  prediction ratio $\eta$, sampling $K$ times, weight $\lambda_u$.
Output: updated $h, f, g$.
  // I. Inference Phase
  obtain predictions (pseudo-labels) $q_b^u$ for $a(u_b)$
  obtain smoothed predictions $\bar{q}_b^u$ via DA
  obtain projections $o_b^x$ for $a(x_b)$ and $o_b^u$ for $a(u_b)$
  obtain the other pseudo-labels $\widetilde{q}_b^u$ via BLPA
  obtain final pseudo-labels $\hat{q}_b^u$ using Eqn. (3.10)
  // II. Training Phase
  obtain prediction $y_b^u$ and projection $z_b^u$ for $A(u_b)$
  obtain prediction $y_b^x$ and projection $z_b^x$ for $a(x_b)$
  calculate three losses using Eqns.( 3.1), (3.2), (3.12)
  combine three losses with $\lambda_u$ and $\lambda_c$
  back-propagate the loss and update $h, g, f$
  update the EMA model
```

At the projection head, we perform the proposed buffer-aided label propagation algorithm to jointly utilize the buffered information (Q_{i-1}), current outputs (o_b^u and o_b^x), and ground-truth labels (p_b), to generate another prediction $\widetilde{q}_b^u$, which is detailed at the following section. At the prediction head, referring to [12], we perform distribution alignment (DA) on the predictions of unlabeled data, $\bar{q}_b^u = \mathrm{DA}(q_b^u)$. In the operation of DA, we simply replace the uniformly moving-averaging by the exponentially moving-averaging with a decay factor of 0.99 over the historical predictions. In this way, we can not only prevent q_b^u from collapsing to certain classes but also prioritize the most current predictions. Consequently, the well-polished pseudo-labels $\hat{q}_b^u$ is obtained for the unlabeled u_b.

Training Phase

The training phase is the core to update the model with three losses, a supervised CE loss $\mathcal{L}_b^x$, an unsupervised CE loss $\mathcal{L}_b^u$, and a class-aware contrastive loss (CACL) $\mathcal{L}_b^c$. As shown in Fig. 3.5, similar to the inference phase, we can obtain the prediction output y_b^x and projection output z_b^x for labeled samples, y_b^u and z_b^u for unlabeled samples. The ground-truth

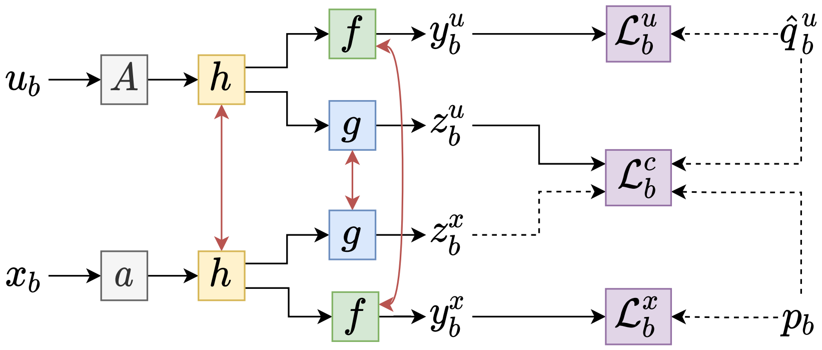


Figure 3.5: Train model on both labeled and unlabeled data by minimizing three losses. The dash line indicates “stop gradient". The definitions of the used symbols are the same as in Figure 3.4.

labels p_b and generated pseudo-labels $\hat{q}_b^u$ are used to calculated the loss $\mathcal{L}_b^x$ and $\mathcal{L}_b^u$, respectively.

$$\mathcal{L}_b^x = H(p_b, y_b^x) \tag{3.1}$$

$$\mathcal{L}_b^u = \mathbf{1}(\max(\hat{q}_b^u) \geq \tau)\, H(\hat{q}_b^u, y_b^u) \tag{3.2}$$

where $\mathbf{1}(\cdot)$ retains the pseudo-labels whose maximum probability is higher than a predefined threshold τ, i.e. high-confidence threshold. and $H(p,q)$ represents the cross-entropy (CE) between two distributions p and q. As to CACL, we first explore the instance relationship $\omega_{i,j}$ by computing the cosine similarity between their corresponding labels y_i and y_j. Specifically, we regard the different image instances as the same class if they have a high-confidence similarity, as distinct class otherwise. After that, we can minimize a class-aware contrastive loss to obtain better feature representations, so that same-class samples are gathered and the different-class samples are scattered.

Though CACL in LaSSL can help the model to make better feature representations, it has no direct effect on downstream tasks. Thus we reweight the CACL with a ramp-down function, starting from λ_c^0 along a decreasing exponential curve. i.e., as the training progresses, we will pay more attention to classification tasks, and less attention to contrastive representation learning.

3.2.2 Buffer-aided Label Propagation Algorithm

At the i-th iteration, the dequeue data Q_{i-1} contains the feature embedding, o_{b-1}, and corresponding labels, q_{b-1}, from the last iteration. To exploit these most recent historical outputs, we regard the dequeue samples with high confidence as labeled data in the current iteration, providing more label information, while treat the dequeue samples with low confidence as unlabeled data, effectively helping explore the potential manifold. However, the samples with high-confidence labels can inevitably include errors. In order to decrease the noise, we do K random sampling with replacement on the dequeue data (i.e. bagging), and denote each sampling result as $o_{b-1}(k)$ and $q_{b-1}(k)$, where $k = 1, 2, ...K$. After that, we can split the sampling data with a predefined confidence threshold τ into two groups, the high-confidence portion $(o_{b-1}^{high}(k), q_{b-1}^{high}(k))$ and the low-confidence one $(o_{b-1}^{low}(k))$. i.e., we have

$$q_{b-1}^{high}(k) = \mathbf{1}(\max(q_{b-1}(k)) \geq \tau)\, q_{b-1}(k), \tag{3.3}$$

$$o_{b-1}^{high}(k) = \mathbf{1}(\max(q_{b-1}(k)) \geq \tau)\, o_{b-1}(k), \tag{3.4}$$

$$o_{b-1}^{low}(k) = \mathbf{1}(\max(q_{b-1}(k)) < \tau)\, o_{b-1}(k). \tag{3.5}$$

Combining the dequeue data with current outputs o_b^u, o_b^x and ground truth labels p_b, we can have the compound labeled features, $o_s(k) = [o_b^x, o_{b-1}^{high}(k)]$, unlabeled features, $o_t(k) = [o_b^u, o_{b-1}^{low}(k)]$, and compound label information, $q_s(k) = [p_b, q_{b-1}^{high}(k)]$.

Subsequently, a standard LPA can be applied. First, a symmetric adjacency matrix $\Omega(k)$ with zero diagonal can be constructed by calculating the similarities of $o_s(k)$ and $o_t(k)$. Then the symmetrically normalized counterpart of $\Omega(k)$ is obtained by,

$$\widetilde{\Omega}(k) = D^{-1/2}\Omega(k)D^{1/2} \tag{3.6}$$

where D is the degree matrix of $\Omega(k)$. After that, the label information can be iteratively propagated to the unlabeled samples. A recursive equation is,

$$\Phi_{j+1}(k) = \alpha\widetilde{\Omega}(k)\Phi_j(k) + (1-\alpha)q_s(k) \tag{3.7}$$

where $\Phi_j(k)$ denotes the predicted labels on compound unlabeled samples at the j-th iteration. $\alpha \in (0,1)$ controls the amount of propagated information. In LaSSL, we use the closed-form solution [62] to obtain the optimal result directly,

$$\Phi^*(k) = (I - \alpha\widetilde{\Omega}(k))^{-1}q_s(k). \tag{3.8}$$

Since we perform LPA at the iteration-level, the computation cost is relatively small, so that BLPA can be easily scaled up to large datasets. As a result, the prediction on current unlabeled samples with the k-th sampling result can be obtained, $\phi_b(k)$, where $\phi_b(k) = \Phi^*(k)[: \mu B]$. Averaging the K results, we can have another prediction for unlabeled u_b directly from the feature-embedding level,

$$\widetilde{q}_b^u = \frac{1}{K}\sum_{k=1}^{K}\phi_b(k). \tag{3.9}$$

To conclude the inference phase, we eventually have the pseudo label $\hat{q}_b$ for u_b,

$$\hat{q}_b^u = \eta\widetilde{q}_b^u + (1-\eta)\bar{q}_b^u, \tag{3.10}$$

where η is a weight parameter to combine two predictions.

3.2.3 Class-aware Contrastive Loss

In the training phase, we have the projection outputs $z_b^x = G(a(x_b))$ and $z_b^u = G(A(u_b))$ for labeled and unlabeled data, respectively. Meanwhile, we have the complete label information for all the samples, i.e., the ground-truth labels p_b and the pseudo-labels $\hat{q}_b^u$. Through concatenating them together $\hat{y} = [p_b, \hat{q}_b^u]$, we can explore all the instance relationships at the prediction level,

$$\omega_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \text{ and } \hat{y}_i \cdot \hat{y}_j < \varepsilon \\ \hat{y}_i \cdot \hat{y}_j, & \text{if } i \neq j \text{ and } \hat{y}_i \cdot \hat{y}_j \geq \varepsilon \end{cases} \tag{3.11}$$

where ε is a similarity threshold to determine whether two distinct instances belongs to the same class. In addition to involving the labeled samples, we can have more sense about the instance classes compared to standard contrastive learning. Therefore, with the explored instance relations at the prediction level, we design a class-aware contrastive loss,

$$\mathcal{L}_b^c = -\sum_{i=1}^{|\hat{y}|} \log \frac{\sum_{j=1}^{|\hat{y}|} \omega_{i,j} \exp(z_i \cdot z_j / T)}{\sum_{j=1, j\neq i}^{|\hat{y}|} \exp(z_i \cdot z_j / T)}. \tag{3.12}$$

where T is a temperature parameter [29].

3.2.4 Putting It All Together

In summary, the total loss at each mini-batch is,

$$\mathcal{L}_b = \mathcal{L}_b^x + \lambda_u \mathcal{L}_b^u + \lambda_c \mathcal{L}_b^c, \tag{3.13}$$

where λ_u and λ_c are two weight parameter for the unsupervised consistency loss and the class-aware constrastive loss, respectively. Similar to [132], we commonly set $\lambda_u = 1.0$. However, we set λ_c as a time-variant scaling parameter to wisely control the weight of CACL. It is worth noting that, CACL aims to obtain better representations but has no direct relationship with our downstream tasks. Therefore, we emphasize CACL to improve the model at the early stages of training, togather with BLPA to enhance the accuracy of pseudo-labels. As the training progresses, we gradually focus more on downstream tasks, i.e., more on $\mathcal{L}_b^u$. To achieve this goal, we adjust λ_c in an exponentially ramping-down manner. Besides, we stop performing BLPA when the weight λ_c becomes small. It is simply because BLPA relies upon the better representations derived from CACL. Mathematically, referring to [79], given the total training epochs T_t and the ramp-down length $(T_t - T_r)$, the weight λ_c at the t-th epoch can be calculated as,

$$\lambda_c = \begin{cases} \lambda_c^0, & \text{if } t \leq T_r, \\ \lambda_c^0 \exp\left(-\frac{(t - T_r)^2}{2(T_t - T_r)}\right), & \text{otherwise.} \end{cases} \tag{3.14}$$

where λ_c^0 is set as the maximum value of λ_c. As a result, the whole training process of LaSSL can be treated as two different periods: it first exploits CACL and BLPA to update the model quickly, and then improve the model further by emphasizing downstream tasks. To further simplify the training, we stop applying CACL and BLPA when $\lambda_c \leq \hat{\lambda}_c$. These two parameters T_r and $\hat{\lambda}_c$, can affect how long the CACL and BLPA will be involved across the training process. Besides, following FixMatch and ReMixMatch, an exponential moving average (EMA) of model parameters with decay of 0.999 is utilized to produce more stable predictions.

3.3 Experiment

In this section, we conduct experiments on four classification datasets to test the effectiveness of LaSSL, including CIFAR-10 [73], CIFAR-100 [73], SVHN [107] and Mini-Imagenet [125]. Following the standard protocol in SSL, we randomly select certain number of labeled data from the training set and treat the remaining training data as unlabeled data. The mean and standard deviation of five runs on testing set with different random seeds are reported. By default, we use a Wide ResNet-28-2 as the encoder $h(\cdot)$, one linear layer as the predictor $f(\cdot)$, and a 2-layer MLP as the projector $g(\cdot)$. The default settings for hyper-parameters in LaSSL is $B = 64, \mu = 7, K = 7, \alpha = 0.8, \eta = 0.2, \tau = 0.95, \varepsilon = 0.7, T_t = 512, \lambda_c^0 = 1.0, \hat{\lambda}_c = 0.1$. Besides, we adopt a SGD optimizer with a momentum of 0.9 and a weight decay of 5e-4, and use a learning rate scheduler with cosine decay to train the model. Unless otherwise noted, we use same

	CIFAR-10		CIFAR-100		SVHN	
Methods	40 labels	250 labels	400 labels	2500 labels	40 labels	250 labels
Pseudo-label*	-	50.22±0.43	-	42.62±0.46	-	79.79±1.09
Mean-Teacher*	-	67.68±2.30	-	46.09±0.57	-	96.43±0.11
MixMatch*	52.46±11.50	88.95±0.86	33.39±1.32	60.06±0.37	57.45±14.53	96.02±0.23
UDA*	70.95±5.93	91.18±1.08	40.72±0.88	66.87±0.22	47.37±20.51	94.31±2.76
ReMixMatch*	80.90±9.64	94.56±0.05	55.72±2.06	72.57±0.31	96.64±0.30	97.08±0.48
FixMatch*	86.19±3.37	94.93±0.65	51.15±1.75	71.71±0.11	96.04±2.17	97.52±0.38
ACR†	92.38	95.01	-	-	-	-
SelfMatch†	93.19±1.08	95.13±0.26	-	-	96.58±1.02	97.37±0.43
CoMatch†	93.09±1.39	95.09±0.33			-	-
Dash†	86.78±3.75	95.44±0.13	55.24±0.96	72.82±0.21	**96.97**±**1.59**	97.83±0.10
LaSSL	**95.07**± **0.78**	**95.71** ±**0.46**	**62.33**±**2.69**,	**74.67**± **0.65**	96.91±0.52	**97.85**± **0.13**

Table 3.1: Top-1 test accuracy (%) for CIFAR-10, CIFAR-100 and SVHN on 5 different folds. All the related works are sorted by their publication date. Results with * was reported in FixMatch [132], while results with † comes from the most recent papers [70, 87, 153, 1], respectively.

codebase and parameter settings to run experiments.

3.3.1 CIFAR-10, CIFAR-100, and SVHN

CIFAR-10 and CIFAR-100 are labeled subsets of the 80 million tiny images dataset with 10 and 100 classes, respectively. Both of them contain 50000 32x32 training images and 10000 32x32 testing images. For fair comparisons, we use Wide ResNet-28-2 as the backbone for CIFAR-10 and Wide ResNet-28-8 for CIFAR-100. In Table 3.1, we compare the testing accuracy of LaSSL against recent SOTA SSL approaches with a varying number of labeled samples. We can obviously see that our LaSSL consistently outperforms other SOTA approaches on CIFAR-10 and CIFAR-100 under all settings. Especially when considering situations with very few labeled data, LaSSL improves over other SSL approaches by a large margin, e.g. achieving an average testing accuracy of 95.07% on CIFAR-10 with only 40 labels. When the number of classes is large like CIFAR-100, LaSSL can still perform well and achieve a accuracy gain of around 7% over the SOTA approach given four labels per class. Checking more details, we find that, achieving the accuracy of around 95% on CIFAR-10, LaSSL needs only four labels per class while other SSL approaches requires 25 or more labels per class. Obviously, LaSSL is more sample efficient and shows its great potential for label-scarce scenarios.

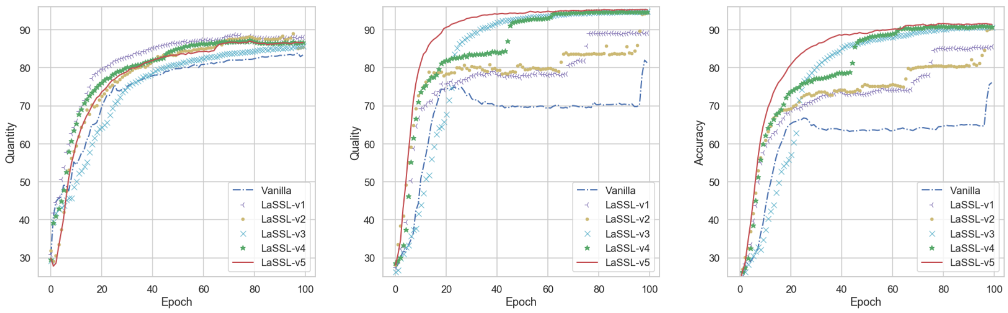
SVHN consists of 10-class colorful 32x32 house numbers. It has 73257 training images and 26032 testing images. The testing accuracy on SVHN in Table 3.1 also shows comparable results to recent state-of-the-art results achieved by Remixmatch and Dash. We can see that the results of all of recent SSL approaches on SVHN are close to the fully supervised baseline (97.3% [58]) with less than 1% difference. Though its superior is not apparent in such simple dataset, LaSSL can achieve the SOTA performance on SVHN with 250 labeled samples. Compared to Dash [1] on SVHN with 40 labeled samples, LaSSL performs slightly worse in terms of the average accuracy but can achieve a lower variance.

3.3.2 Mini-ImageNet

Following the SIMPLE [58], we test LaSSL on more complicated dataset, Mini-ImageNet[125]. Sampled from ImageNet ILSVRC, it consists of 50000 training images and 10000 testing images, evenly distributed across 100 classes. We compare the performance of LaSSL against the SOTA SSL approach, SIMPLE, on Mini-ImageNet with 4000 labeled samples. For a fair comparison, ResNet-18 is set as the backbone, and each sample is center-cropped and resized to 84x84. Apart from default parameter configurations, we set $\lambda_c^0 = 5.0, \tau = 0.8$ in this experiment. SIMPLE can achieve an average testing accuracy of **49.39**%, while LaSSL obtains a result of **60.14** ± 0.26 %. LaSSL can obviously outperform SIMPLE with a better accuracy by a significant average gain of 10.75%.

3.3.3 Ablation Study

Effectiveness of different components. To investigate the impact of three different components in LaSSL (i.e., CACL, BLPA and DA), we test LaSSL with different combinations of these components on CIFAR-10 with four labels per class. For fair comparisons, we compare their performance with the same random seed during the first 100 epochs. To better analyze the performance, we introduce two intuitive concepts, quantity and quality of pseudo-labels. “**Quantity**" refers to the amount of high-confidence pseudo-labels, calculated by the ratio of the number of high-confidence predictions to the total number of unlabeled samples.



(a) Quantity

(b) Quality

(c) Accuracy

Figure 3.6: (a), (b), (c) represent curves of the quantity, quality, and EMA test accuracy of different combinations of CACL, BLPA, and DA (better view on screen). Numerical results are listed in Table 3.2.

Method	CACL	BLPA	DA	Quant	Qual	Acc
Vanilla	✗	✗	✗	83.91	81.98	75.54
LaSSL-v1	✓	✗	✗	88.66	89.38	85.50
LaSSL-v2	✓	✓	✗	89.08	94.31	90.24
LaSSL-v3	✗	✗	✓	85.73	94.90	90.42
LaSSL-v4	✓	✗	✓	87.46	94.89	91.11
LaSSL-v5	✓	✓	✓	87.03	**95.33**	**91.65**

Table 3.2: Ablation studies on CIFAR-10 with 40 labeled data after training 100 epochs (random seed is fixed to 1.)

“**Quality**" measures how many high-confidence predictions are consistent to ground-truth labels, which can be obtained by using real labels from CIFAR-10.

It can be seen from Table 3.2 that each component matters compared to the vanilla version. Integrating all three components can achieve the highest accuracy and quality while maintaining a considerably high quantity. In Figure 3.6, we show the detailed dynamics of the quantity, quality and accuracy w.r.t the training epochs. We can observe from Figure 3.6(a) that LaSSL-v1 can consistently achieve the highest quantity in the 100 epochs, indicating that the CACL is very effective in quickly improving the number of high-confidence pseudo-labels. By comparing LaSSL-v1 to LaSSL-v2 and the vanilla to LaSSL-v3, we can find that BLPA and DA

ε	0.6	0.7	0.8	0.9	1.0
Accuracy(%)	87.64	**89.39**	87.70	87.36	85.17

Table 3.3: Effects with different similarity thresholds. The similarity is equal to 1 only when comparing the image instance with itself. Therefore, we use $\varepsilon = 1.0$ to investigate the effect of excluding the "class-aware" technique.

K	0	1	3	5	7
Accuracy(%)	92.71	92.10	94.64	93.43	**94.87**

Table 3.4: Effects with different number of samplings. In specific, $K = 0$ means the plain LPA without "buffer-aided"; $K = 1$ means exploiting the buffered data directly without sampling; while $K > 1$ investigates the complete BLPA.

are two powerful strategies to improve the quality. Besides, the dynamics of Figure 3.6(b) and 3.6(c) are closely related, suggesting that the quality of pseudo-labels is the most crucial factor affecting the final performance. The increasing tendency also indicates that LaSSL-v5 (i.e., standard LaSSL) is the most stable and accurate one with the consistently highest testing accuracy.

Impact of different similarity threshold. In Table 3.3, we compare the effect of CACL with different values of similarity threshold in terms of the testing accuracy. For fair comparisons, BLPA and DA are not involved. Since the similarity can never exceed 1.0, $\varepsilon = 1.0$ simply denotes that every instance belongs to a distinct class, i.e., without class-aware senses. We can observe that the "class-aware" strategy is indeed beneficial in SSL. Besides, there intuitively exists a trade-off, i.e. lower values of ε can involve more similarities among samples but inevitably introduce more errors. In contrast, large ε may fail exploiting the instance relations.

Impact of different sampling times. We investigate the impact of BLPA with different values of K in Table 3.4. For fair comparisons, we adopt default settings for CACL and DA. To reduce effects of wrong pseudo-labels, we sample K-times on the buffered data and average the results in BLPA. $K = 0$ means no buffer-aided, while $K = 1$ uses all the buffer data without sampling. As a result, $K = 7$ achieve the highest testing accuracy. We can also find that directly involving all the buffered data (i.e. $K = 1$) will degrade the performance due to introducing more wrong high-confidence pseudo-labels. On the other hand, though large K may introduce more computational efforts, it can generally lead to more robust predictions and higher accuracy.

3.4 Summary

In this work, we propose LaSSL, a novel SSL approach that exploits the label information to integrate a class-aware contrastive loss and buffer-aided label propagation algorithm into a self-training paradigm. Two strategies are tightly coupled and mutually boosted across the training process. Meanwhile, the label information is extensively utilized: to provide a supervised loss, to generate instance relations for CACL, and to be propagated on unlabeled samples in BLPA. Through extensive experiments, we demonstrate that LaSSL can propose better pseudo-labels with higher quality and quantity. In specific, the class-aware contrastive loss (CACL) can quickly increase the quantity of high-confidence pseudo-labels, while the buffer-aided label propagation algorithm (BLPA) can improve the quality of pseudo-labels effectively. Experiment results show that LaSSL can outperform the SOTA SSL methods on four benchmark classification datasets with different amounts of labeled data, including CIFAR10, CIFAR100, SVHN, and Mini-ImageNet. Especially for few-label settings, LaSSL can achieve very promising accuracy, e.g., given four labels per class, LaSSL achieves an average accuracy of 95.07% on CIFAR-10 and 62.33% on CIFAR-100.

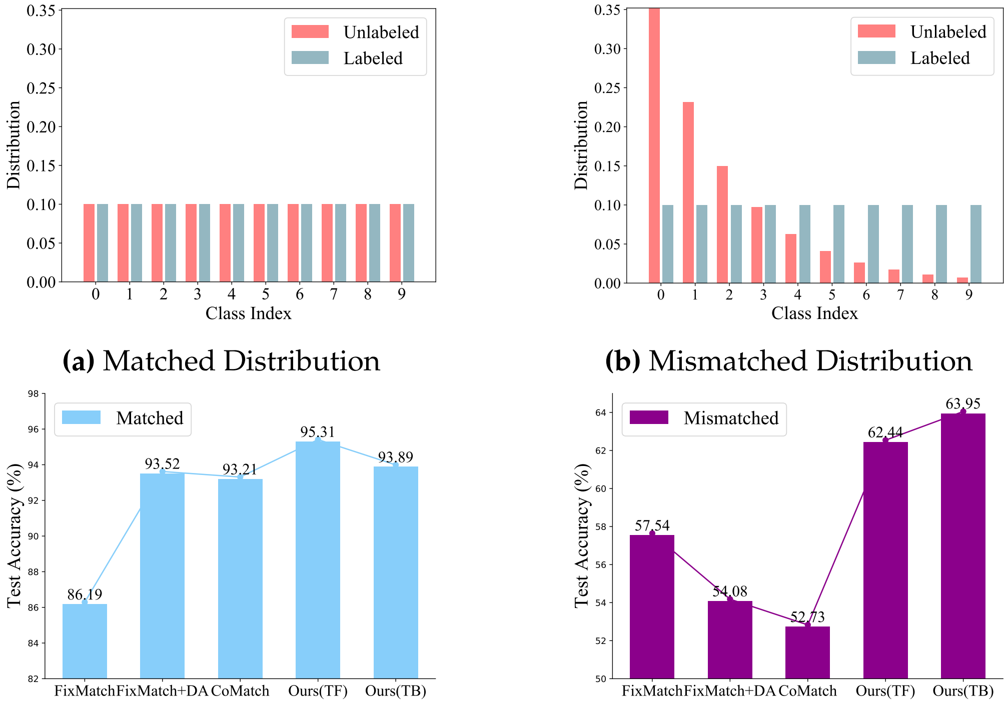
Chapter 4

DC-SSL: Addressing Mismatched Class Distribution in Semi-supervised Learning

In this chapter, we investigate a more challenging semi-supervised scenario where the labeled and unlabeled data enjoy different class distribution. We find that such mismatched issues can bring significant performance degradation to current state-of-the-art SSL methods. To this end, we propose a new SSL learning framework, named Distribution Consistency SSL (DC-SSL), which aims to improve the pseudo-labels from a distribution perspective. Extensive experiments and ablation studies are conducted to demonstrate the effectiveness of our method on popular classification datasets.

4.1 Introduction

Recent consistency-based semi-supervised learning (SSL) methods have seen fast progress and shown competitive performance to supervised learning [111, 114]. These methods commonly utilize the model trained on labeled samples to generate pseudo-labels on unlabeled samples, and then enforce prediction consistency against their corresponding perturbed variants. An implicit assumption in such methods is that the labeled and unlabeled data share the same class distribution. However, such a strong assumption cannot hold in real practice. The scarcity of labeled samples or the sampling errors can inevitably lead to a distribution mismatch



(a) Matched Distribution

(b) Mismatched Distribution

(c) Test accuracy for (a)

(d) Test accuracy for (b)

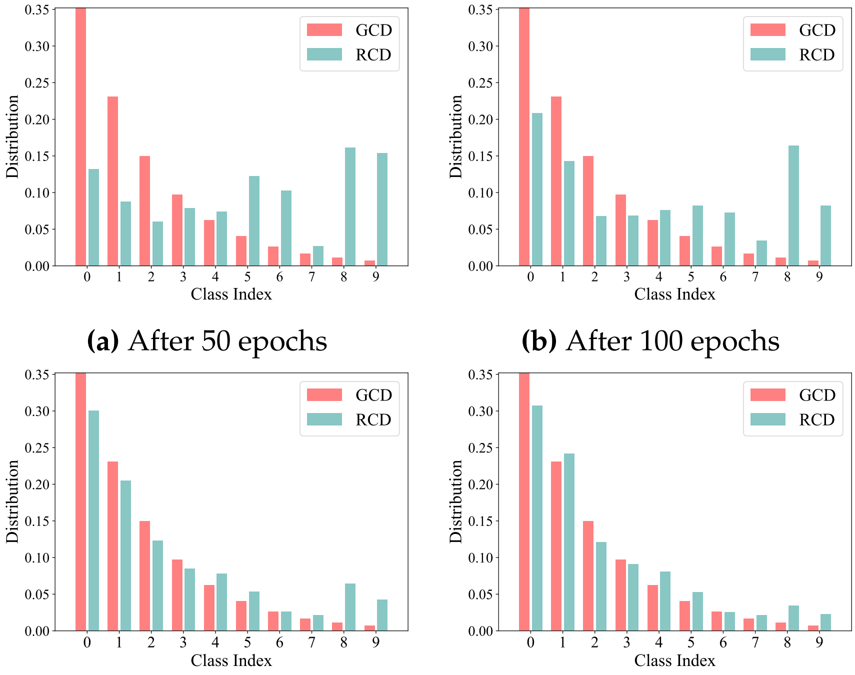
Figure 4.1: (a) and (b) show the class distributions on CIFAR10 in the matched and mismatched distributions settings, respectively. (c) and (d) show the corresponding test performance on the recent SOTA SSL methods and our proposed DC-SSL with training-free (TF) and training-based (TB) strategies.

between the labeled and unlabeled data. This could, unfortunately, invalidate most of the advanced SSL methods.

To illustrate this problem, we conducted a performance comparison under matched and mismatched distribution scenarios. As shown in Figure 4.1(c), two state-of-the-art (SOTA) SSL methods, FixMatch [132] and CoMatch [87], can achieve promising results on CIFAR-10 with only 40 labeled samples when the labeled and unlabeled class distributions are matched, *e.g.*, a high test accuracy of 93.21% of CoMatch. However, when there exists a distribution mismatch as shown in Figure 4.1(b), the test accuracy can drop sharply by around 30% on FixMatch and severely more than 40% on CoMatch. It is because the pseudo-labels on the unlabeled set are severely biased and unreliable in a mismatched distribution setting, resulting in a significant performance degradation.

Inspired by distribution alignment (DA) [12], we aim to improve the biased pseudo-labels from a distribution perspective. The basic logic is to modify the pseudo-labels by encouraging the predicted class distribution (PCD) of the unlabeled data to be close to the underlying ground-truth class distribution (GCD) across the training. However, the existing works using DA [12, 45, 87, 146] widely assumed that the labeled and unlabeled data fall in the same class distribution, and therefore took the provided labeled class distribution (LCD) as the GCD on the unlabeled set to rectify pseudo-labels. As shown in Figure 4.1(c), built into FixMatch, although DA significantly improves the performance in the matched distribution setting (*i.e.* LCD=GCD), it causes severe negative impact under the mismatched distribution scenario (*i.e.* LCD≠GCD) with a sharp accuracy drop as shown in Figure 4.1(d). A key rescue and challenge is to employ an accurate distribution to guide PCD on the unlabeled data, whereas the unlabeled GCD is commonly unknown and the known LCD is biased and unreliable.

To address the above limitations, we propose a simple but effective method, named Distribution Consistency SSL (DC-SSL), which can effectively rectify the pseudo-labels from a distribution perspective. The design of DC-SSL is based on two main components. **First**, instead of using LCD, DC-SSL directly estimates a reference class distribution (RCD) from the unlabeled data, which is regarded as a surrogate of the unknown GCD. To this end, we revisit the exponentially moving averaged (EMA) model in SSL and carefully study i) why the EMA model is employed merely for the testing instead of the training process in recent SOTA SSL methods [132, 58, 70, 87, 1], and ii) how the EMA model can benefit the distribution estimation on unlabeled samples. Based on this investigation, we design our framework to involve EMA to estimate a robust RCD by a momentum-updated scheme over historical label predictions. As shown in Figure 4.2, the estimated RCD gradually approaches GCD with the progression of the training procedure. **Second**, on top of the estimated distributions, two direct and indirect updating strategies are proposed, respectively, to modify the pseudo-labels, corresponding to the training-free and the training-based strategies. The training-free (TF) strategy directly modifies the pseudo-labels by scaling them with a



(a) After 50 epochs

(b) After 100 epochs

(c) After 200 epochs

(d) After 400 epochs

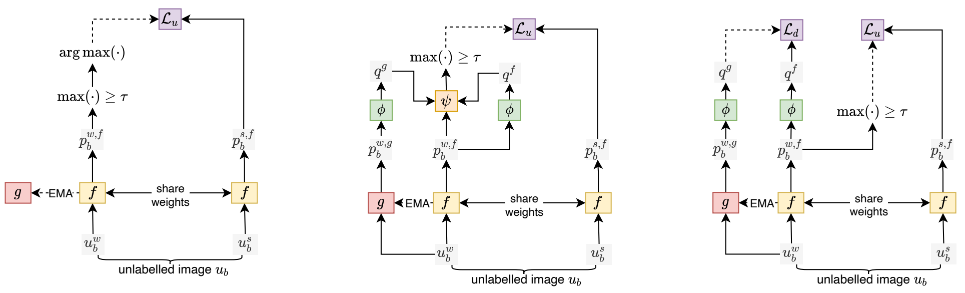
Figure 4.2: (a)-(d) compares the RCD in DC-SSL (TB) and GCD at different training stages with the mismatched setting in Figure 4.1(b).

ratio of RCD to PCD, while the training-based (TB) strategy minimizes a distribution consistency loss between PCD and RCD to indirectly enhance the SSL performance. Both strategies are orthogonal to existing consistency-based SSL methods and can be easily applied with minimal change of implementation.

4.2 Method

In an N-class classification task, the labeled data D_x and unlabeled data D_u are given to train a model with the embedding function $f(\cdot)$. In a mini-batch, suppose we have B labeled samples, $\mathcal{X} = \{(x_b, y_b)|(x_b, y_b) \in D_x\}_{b=1}^{B}$, and μB unlabeled samples, $\mathcal{U} = \{u_b|u_b \in D_u\}_{b=1}^{\mu B}$, where μ represents the size ratio of $\mathcal{U}$ to $\mathcal{X}$. In most SSL studies, the total loss can be formulated as:

$$\mathcal{L} = \mathcal{L}_x(\mathcal{X}) + \lambda_u \mathcal{L}_u(\mathcal{U}), \tag{4.1}$$



(a) Consistency-based **(b)** Training-free (ours) **(c)** Training-based (ours)

Figure 4.3: (a) shows the diagram of FixMatch, a widely adopted consistency-based SSL method. (b) and (c) are our proposed two strategies to enforce distribution consistency on top of FixMatch. Sepcifically, u_b^w and u_b^s are the weakly and strongly augmented variants of an unlabeled image u_b, respectively. f denotes the network model and g is the EMA of f. p is the network's probability prediction and τ is a high-confidence threshold. q represents the class distribution derived from historical predictions by the scheme ϕ. Without introducing new network components, our models estimate class distributions on unlabeled data, and enforce distribution consistency by either the training-free update denoted as ψ in (b) or the training-based consistency loss denoted by $\mathcal{L}_d$ in (c). Dash lines indicate “stop gradient".

where $\mathcal{L}_x$ is a supervised loss and $\mathcal{L}_u$ is an unsupervised loss within a mini-batch, measured on $\mathcal{X}$ and $\mathcal{U}$ respectively. λ_u is a weighting parameter to balance the relative importance between the labeled and the unlabeled data. Commonly, $\mathcal{L}_x$ can be obtained by

$$\mathcal{L}_x = \frac{1}{B}\sum_{b=1}^{B} H(y_b, f(x_b)), \tag{4.2}$$

where H denotes the cross entropy loss. Whereas, the form of $\mathcal{L}_u$ depends on specific SSL methods. In this section, we first review how $\mathcal{L}_u$ is formulated in the backbone consistency-based SSL learner, FixMatch. After that, we introduce the crucial components in our method on top of the backbone: RCD estimation and two updating strategies.

4.2.1 Backbone SSL Learner

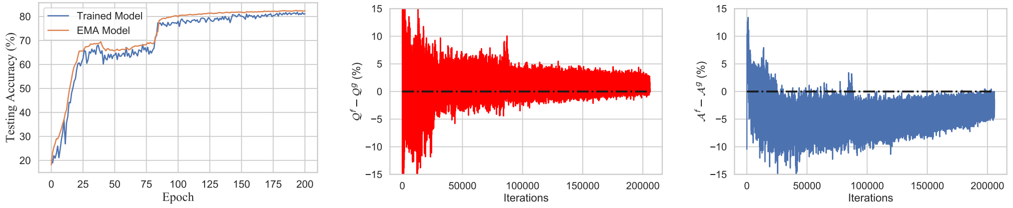
Recent consistency-based SSL methods typically use weakly-augmented unlabeled images to generate pseudo-labels and enforce consistency against their corresponding strongly-augmented variants. As shown in Figure 4.3(a), u_b^w and u_b^s are obtained through weakly and strongly augmented operations on an unlabeled instance u_b. The weakly augmented operations consists of standard flip-and-shift augmentation strategies, while the strongly augmented operations usually refer to RandAugment [35] or CTAugment [12]. Subsequently, the model f outputs probability predictions $p_b^{w,f}$ and $p_b^{s,f}$ for u_b^w and u_b^s, respectively. As the most simplified but effective consistency-based SSL method, FixMatch [132] adopted a fixed high-confidence threshold to alleviate the confirmation bias [4] of pseudo-labels. Given a predefined high-confidence threshold τ, the unsupervised loss in FixMatch can be calculated as,

$$\mathcal{L}_u = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbf{1}(\max(p_b^{w,f}) \geq \tau) H(\hat{p}_b^{w,f}, p_b^{s,f}), \tag{4.3}$$

where $\hat{p}_b^{w,f} = \arg\max(p_b^{w,f})$ denotes the hard pseudo-labels (*i.e.*, in a one-hot form) for unlabeled samples, and the operation $\mathbf{1}(\cdot)$ retains the pseudo-labels whose maximum probability is higher than the threshold τ. Besides, an exponential-moving-averaging model g is maintained along with the model f. However, in FixMatch, g is only used for the testing process and independent from the training process, as in many recent SSL methods.

4.2.2 Distribution Estimations

Properly estimating the class distribution (*i.e.*, frequency of each class on unlabeled data) is the most important problem in our design. Inspired by distribution alignment [12], our primary idea is to encourage the predicted class distribution (PCD) on unlabeled data to be close to the ground-truth class distribution (GCD). However, the lack of label information makes this GCD unknown and challenging to obtain. Almost all existing works, either in balanced SSL [87] or imbalanced SSL



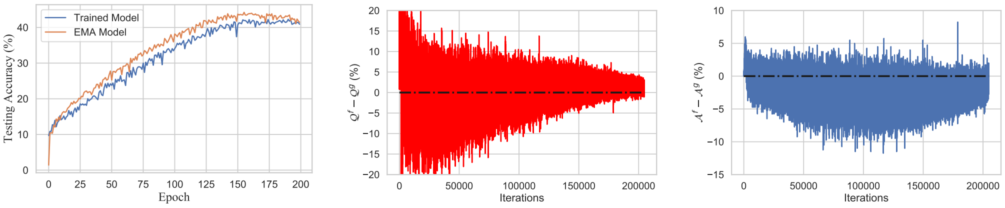
(a) **(b)** **(c)**

Figure 4.4: (a) Comparison of testing accuracy between the trained model f and its corresponding EMA model g. (b) Accuracy difference ($Q^f - Q^g$) of the high-confidence pseudo-labels in a mini-batch between f and g at each iteration. Statistically, g obtains a lower accuracy than f at about 70% iterations. (c) Accurate difference ($\mathcal{A}^f - \mathcal{A}^g$) of all pseudo-labels between f and g at each iteration. The model g can generate more accurate pseudo-labels in 96% iterations.

tasks [126], adopt the marginal distribution of the provided labeled data as the GCD of the unlabeled data, which will inevitably produce severely biased pseudo-labels, and largely degrade the SSL performance in mismatched distribution settings. Differently, in our work, instead of relying on labeled data, we purely work on unlabeled data to propose a referenced class distribution (RCD) as a surrogate of GCD. Specifically, we carefully involve the EMA model during the training period to estimate the RCD on unlabeled data. As shown in Figure 4.2, the iteratively-improved RCD can be gradually approaching the GCD across the training process. In this section, we first revisit the EMA model in SSL and then describe the momentum-updated scheme to estimate the distribution from the model's predictions.

Revisiting the EMA Model

In the literature, an EMA model with a typical decay of 0.999 is widely adopted in SSL methods for performance enhancement. To investigate its effectiveness, based on FixMatch and using CIFAR-10 with 40 labeled samples, we compare the test accuracies of the trained model f and the EMA model g across different training epochs. As shown in Figure 4.4(a), unsurprisingly, the EMA model g can consistently outperform the trained model f. Based on this, we revisit the EMA model in details by answering two questions the the following.

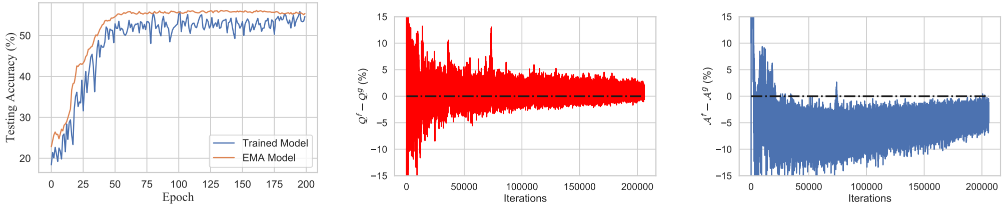


(a) **(b)** **(c)**

Figure 4.5: Following the explorations in Figure 4.4, we observe same findings on MiniImageNet with 1000 labels. (b) In terms of the accuracy difference ($Q^f - Q^g$) of the high-confidence pseudo-labels in a mini-batch, g obtains a lower accuracy than f at about 61% iterations. (c) However, the accurate difference ($\mathcal{A}^f - \mathcal{A}^g$) of all pseudo-labels between f and g shows that the model g can generate more accurate pseudo-labels in 88% iterations.

Question 1: Since the EMA model can achieve a higher test accuracy, will it be beneficial to directly exploit the predictions of the EMA model as pseudo-labels for training? Surprisingly, the answer is NO. In recent SSL studies [12, 132, 58, 87], the EMA model is only used for testing rather than proposing pseudo-labels. However, the potential reasons are not clearly explained in the literature. Thus we perform another experiment to directly use the EMA model's predictions as pseudo-labels. However, this method significantly degrades the SSL performance, achieving a testing accuracy of 45.31% compared to 82.50% of the original FixMatch. We then explore the reasons in term of the accuracy of high-confidence pseudo-labels throughout the training, denoted by Q. As shown in Figure 4.4(b), we measure the accuracy difference of the high-confidence pseudo-labels from f and g throughout a same training process, i.e. $Q^f - Q^g$. As seen, Q^f is higher than Q^g for above 70% of the training period. Therefore, directly using the EMA model's predictions leads to poor quality of the high-confidence pseudo-labels, which explains why recent SSL methods exclude the EMA model in the training process.

Question 2: How can our method use the EMA model to estimate a better class distribution on unlabeled data? By further analyzing the above experimental results, we find that, compared with f, although EMA model g obtains a lower accuracy on high-confidence predictions,



(a) **(b)** **(c)**

Figure 4.6: Following the explorations in Figure 4.4, we investigate the EMA model's performance on CIFAR10 in a mismatched distribution setting as in Figure 3.6(b). (b) In terms of the accuracy difference ($Q^f - Q^g$) of the high-confidence pseudo-labels in a mini-batch, g obtains a lower accuracy than f at about 67% iterations. (c) However, the accurate difference ($\mathcal{A}^f - \mathcal{A}^g$) of all pseudo-labels between f and g shows that the model g can generate more accurate pseudo-labels in 97% iterations.

it can produce a higher accuracy on all unlabeled data (with both high-confidence and low-confidence predictions), *i.e.*, obtaining larger amounts of accurate predictions. Let $\mathcal{A}$ be the pseudo-label accuracy on all unlabeled data in a mini-batch instead of just high-confidence ones. We investigate $\mathcal{A}^f - \mathcal{A}^g$ across the training process in Figure 4.4(c). It is observed that in most iterations, g can achieve a higher value of $\mathcal{A}$ (see the negative values of $\mathcal{A}^f - \mathcal{A}^g$), *i.e.* more accurate predictions. That is indeed what we need for better distribution estimation, since the class distribution ought to be estimated on the whole unlabeled data rather than just the high-confidence ones. Therefore, we can rely on the EMA model's predictions to make a better class distribution estimation on unlabeled data. As shown in Figure 4.5, we further find the same observations on MiniImageNet with 1000 labels. In addition, we investigate the performance of the EMA model in a mismatched distribution setting on CIFAR-10 with $|D_x| = 40$ and $\gamma_u = 50$. It can been seen from Figure 4.6 that g can outperform f throughout the whole training process in terms of the accuracy on all pseudo-labels, yet with lower accuracy on high-confidence ones.

Then can we directly use all predictions from the EMA model as pseudo-labels of the unlabeled data to train models? No, it will also largely decrease the test accuracy due to the well-known issue in SSL, i.e., the confirmation bias [4]. Combining the entropy minimization [47], it is claimed in [132] and [4] that retaining only the pseudo-labels with high-confidence predictions can effectively alleviate the bias. In the following section, we provide our solution to estimate the class distribution by the predictions of EMA model.

In summary, we observe that the EMA model can achieve a higher accuracy of pseudo-labels on all unlabeled data but a lower accuracy on high-confidence ones.

Estimating Distribution from Predictions

The next problem is how we derive the class distribution from EMA's predictions on unlabeled data. Since the class distribution between different mini-batches can vary considerably, a natural way to improve the estimation is to involve multiple mini-batches. As proposed in ReMixMatch [12], a direct way to estimate the class distribution is to average over historical predictions. However, such a method requires maintaining a memory bank to store the model's predictions from the most recent K mini-batches. More importantly, it ignores temporal differences among historical predictions, i.e., the more recent predictions are more accurate throughout the training. Therefore, we adopt a momentum-updated strategy, denoted by ϕ in Figure 4.3(b) and Figure 4.3(c), to estimate the class distribution, requiring calculations only on the current mini-batch. ϕ is essentially a weighted averaging scheme and will assign higher weights on more recent predictions. Given the prediction results $\{p_b^{w,f}\}_{b=1}^{\mu B}$ on the trained model f within a mini-batch, its corresponding class distribution q^f can be estimated as

$$q^f := \alpha\, q^f + \frac{(1-\alpha)}{\mu B}\sum_{b=1}^{\mu B} p_b^{w,f}, \tag{4.4}$$

where α is a momentum coefficient. In such ways, we cannot only decrease the memory cost but also prioritize the most recent predictions. Likewise, given the EMA model's prediction $\{p_b^{w,f}\}_{b=1}^{\mu B}$, we can obtain another distribution estimation, q^g,

$$q^g := \alpha\, q^g + \frac{(1-\alpha)}{\mu B} \sum_{b=1}^{\mu B} p_b^{w,g}. \tag{4.5}$$

4.2.3 Updating Strategies

At each mini-batch, we produce two distribution estimations from the unlabeled samples: 1) the predicted class distribution (PCD), q^f, estimated by the trained model via Equation (4.4), and 2) the reference class distribution (RCD), q^g, derived by the EMA model via Equation (4.5). Based on q^f and q^g, we design two alternative training strategies to improve pseudo-labels either directly or indirectly.

Training-free Strategy

Inspired by ReMixMatch [12], we design a training-free strategy to enhance the quality of pseudo-labels from a distribution perspective. We measure the distribution dissimilarity between RCD and PCD by a ratio q^g/q^f. Then, the training-free strategy, denoted by ψ in Figure 4.3(b), can be performed via two steps: 1) revise the pseudo-label by the distribution dissimilarity ratio, and 2) normalize the revised pseudo-label in a valid probability form. Consequently, the ultimate pseudo-label $\bar{p}_b^{w,f}$ can be calculated as

$$\bar{p}_b^{w,f} = \mathrm{Normalize}(\frac{q^g}{q^f} p_b^{w,f}), \tag{4.6}$$

where $\mathrm{Normalize}(x_i) = x_i / \sum x_i$. Then the unsupervised loss $\mathcal{L}_u^{tf}$ in this strategy is,

$$\mathcal{L}_u^{tf} = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbf{1}(\max(p_b^{w,f}) \geq \tau) H(\bar{p}_b^{w,f}, p_b^{s,f}). \tag{4.7}$$

To the end, the total loss for this strategy is $\mathcal{L}_x + \lambda_u \mathcal{L}_u^{tf}$. No additional training efforts are introduced by this strategy.

Training-based Strategy

As shown in Figure 4.3(c), we also propose a training-based strategy to encourage PCD to gradually approach RCD. Specifically, given RCD and PCD, we can minimize a distribution consistency loss $\mathcal{L}_d$:

$$\mathcal{L}_d = H(p^g, p^f), \tag{4.8}$$

where we use the cross entropy loss $H(\cdot, \cdot)$ to measure the discrepancy between the two distributions. Besides, we also reserve the consistency loss at the instance level,

$$\mathcal{L}_u^{tb} = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbf{1}(\max(p_b^{w,f}) \geq \tau) H(p_b^{w,f}, p_b^{s,f}), \tag{4.9}$$

where we use the soft pseudo-labels $p_b^{w,f}$ for calculations compared to the hard labels $\hat{p}_b^{w,f}$ used in Equation (4.3). In summary, the total loss is,

$$\mathcal{L} = \mathcal{L}_x + \lambda_u \mathcal{L}_u^{tb} + \lambda_d \mathcal{L}_d, \tag{4.10}$$

where λ_u and λ_d are two weights of the consistency loss at the instance level and at the distribution level, respectively.

Remarks: Our proposed DC-SSL is conceptually analogous to an Expectation-Maximization (EM) procedure. In the E-step, DC-SSL produces distribution estimations p^g and p^f by taking f and g as available models with fixed parameters. In the M-step, DC-SSL updates the models f and g by minimizing the total loss in Equation (4.1) or Equation (4.10) on top of the two distributions estimated in the E-step. The algorithm can alternately improve the distribution estimations and the trained models.

4.3 Experiment

This section presents our experimental setup and implementation details, followed by extensive evaluations of our methods with mismatched and matched class distributions.

Method	CIFAR10,$\|D_x\|$=40			CIFAR10,$\|D_x\|$=250			CIFAR100, $\|D_x\|$=2500		MiniImageNet, $\|D_x\|$=1000	
	$\gamma_u = 50$	100	200	$\gamma_u = 50$	100	200	$\gamma_u = 100$	200	$\gamma_u = 100$	200
FixMatch	57.54	54.82	50.67	76.54	73.51	70.89	52.46	50.24	25.52	21.65
FixMatch+DA	54.08	46.71	41.37	70.78	66.25	61.69	48.96	46.59	22.92	19.82
CoMatch	52.73	46.20	38.85	69.36	64.47	60.05	47.03	43.89	20.37	19.03
Ours (TF)	62.44	56.47	52.32	79.25	76.10	72.01	56.43	52.01	27.44	23.53
Ours (TB)	63.95	57.16	53.27	81.82	77.26	73.34	59.02	52.70	29.12	24.41

Table 4.1: Mean test accuracy (%) with mismatched class distribution: balanced labeled data and imbalanced unlabeled data. $|D_x|$ is the number of labeled samples. The higher the γ_u, the more the imbalance, and the more severe the distribution mismatch.

4.3.1 Experimental Setup

Dataset and Backbone. We evaluate our methods on four SSL image classification benchmarks, CIFAR-10 [73], CIFAR-100 [73], STL-10[33], and Mini-Imagenet [125]. Of these, CIFAR-10 and CIFAR-100 contain 50,000 32x32 training images and 10000 32x32 testing images, with 10 and 100 classes, respectively. STL-10 is composed of 5,000 labeled images of size 96x96 from 10 classes, along with 10,000 unlabeled images. Mini-Imagenet consists of 50000 training images and 10000 testing images, evenly distributed across 100 classes. For fair comparison[132, 87], we use Wide ResNet-28-2 for CIFAR-10, Wide ResNet-28-8 for CIFAR-100, ResNet-18 for Mini-Imagenet and STL-10, respectively. We use Fixmatch as our backbone (the fundamental consistency-based SSL method) and compare our methods with multiple SSL baselines.

Mismatched Settings. Since the original datasets are all class-balanced, we sample the training images to investigate two mismatched cases: 1) balanced labeled samples with imbalanced unlabeled samples, and 2) balanced unlabeled samples with imbalanced labeled samples. Inspired by CIFAR-LT [18], we utilize an exponential function to mimic the imbalanced distribution. For imbalanced labeled samples, we use $\Gamma_i = \Gamma_0 \gamma_x^{-\frac{i}{N-1}}, i \in [0, N-1]$ to generate the labeled number for the i_{th} class. We use different Γ_0 to investigate different scale of imbalance, while the γ_x is calculated by the constraint $\sum_i \Gamma_i = |D_x|$. On the other hand, we refer to CIFAR-LT[18] to generate imbalanced unlabeled samples, with $M_i = M_{max} \gamma_u^{-\frac{i}{N-1}}$, where M_{max} is set as the image number of the i_{th} class in the original datasets. By adjusting the value of γ_u for difference scales of imbalance, we control the degree of distribution mismatch between the labeled and unlabeled samples, i.e., the larger the γ_u , the higher the severity of distribution mismatch.

Parameters. In our proposed methods, we introduce two new hyper-parameters: the momentum coefficient α for both strategies and the loss weight λ_d for the training-based (TS) strategy. By default, we simply set $\alpha = 0.99$, and $\lambda_d = 1.0$. Ablation studies on these parameters are provided in the next section. The default values of other training hyper-parameters are $B = 64, \mu = 7, \lambda_u = 1, \tau = 0.9$. We train our methods for 512 epochs and utilize a SGD optimizer with a momentum of 0.9 and a weight decay of 5e-4 to train the model. A learning rate scheduler with a cosine decay is used to decrease the learning rate from an initial value of 0.03. In addition, we train the model for 20 epochs to warm up before applying our proposed distribution consistency.

4.3.2 Results for Mismatched Distribution

Imbalanced unlabeled samples. In Table 4.1, we test the performance in a mismatched distribution setting where we have balanced labeled data but imbalanced unlabeled data. It can be clearly seen that, as the γ_u gets larger , *i.e.*, the mismatch issue is more severe, the test accuracy decreases considerably on all SSL benchmarks across different amounts of labeled samples. The mismatched distribution in SSL is a very challenging problem indeed. Compared to other SOTA SSL methods, our methods with either TF or TB strategies can achieve a remarkable performance improvement. In all our tested cases, our TB strategy can boost the mean accuracy of FixMatch by around 3%, and the accuracy of CoMatch by around 11% on average. Interestingly, we find that CoMatch obtains the worst results in all the tests among different baselines. This is because CoMatch extensively exploits the label information carried on the labeled samples to modify the pseudo-labels of unlabeled samples. In addition to the standard DA technique, it maintains a large memory bank to smooth the pseudo-labels by aggregating information from nearby labeled samples in the embedding space. However, relying heavily on labeled samples can only be helpful when the labeled and

Method	STL-10
	$\lvert D_x \rvert$=1000
FixMatch	65.38
FixMatch+DA	66.53
CoMatch	79.80
Ours (TF)	84.61
Ours (TB)	82.47

Table 4.2: Mean test accuracy (%) for STL-10 averaged on 5 different folds. All the related works are reported in CoMatch [87].

Method	CIFAR10, $\lvert D_x \rvert$=250		CIFAR100, $\lvert D_x \rvert$=2500		MiniImageNet, $\lvert D_x \rvert$=1000	
	$\Gamma_0 = 100$	200	$\Gamma_0 = 100$	200	$\Gamma_0 = 40$	80
FixMatch	69.76	46.53	61.31	41.38	36.20	28.33
FixMatch+DA	61.80	27.61	50.94	31.82	33.87	23.53
CoMatch	57.87	26.77	48.02	30.08	30.24	21.47
Ours (TF)	72.21	52.59	64.63	41.23	39.07	31.75
Ours (TB)	73.04	48.49	65.24	42.09	40.13	32.82

Table 4.3: Mean test accuracy (%) with mismatched class distribution: imbalanced labeled data and balanced unlabeled data. $|D_x|$ is the number of labeled samples. The higher the Γ_0, the more the imbalance, and thus the more severe the distribution mismatch.

unlabeled distributions are identical. In the mismatched distribution setting, closely dependending on label information can cause severe negative effects, as can be seen from the test results. Although our methods share a similar idea of the DA to improve pseudo-labels from a distribution perspective, our methods significantly outperform other DA-based baselines (*i.e.*, Fixmqtch+DA and Comatch) due to our proposed better RCD estimated directly on unlabeled samples.

Imbalanced labeled samples. We also investigate another mismatch setting in Table 4.3: imbalanced labeled data but balanced unlabeled data. It can be seen that our methods can effectively improve the performance by rectifying the pseudo-labels from a distribution perspective. The overall results further demonstrate the superiority of our methods, *e.g.*, TB strategy can obtain a mean accuracy of 40.13% on MiniImageNet with imbalanced 1000 labeled samples, against 36.20% of FixMatch and

30.24% of CoMatch.

Observing the results from Tables 4.1 and 4.3, we can also find that, our TB strategy can mostly achieve better performance than our TF strategy at different degrees of distribution mismatch. This stems from their different levels of influence on the pseudo-labels. The TF strategy can pose strong effects on the pseudo-labels by directly modifying them with a ratio of RCD to PCD. Differently, the TB strategy does not directly adjust the pseudo-labels but indirectly improves the pseudo-labels by enforcing their aggregated distribution to gradually approach the RCD. That is, the TB strategy can take effects in a more moderate manner. In the mismatched case, as shown in Figure 4.2, our estimated RCD may not be very accurate at the early stages of the training process, but can be gradually improved to approach the ground-truth distribution across the training process. Therefore, our TB strategy is more suitable for the mismatched cases and can gradually enhance the SSL performance along with the iteratively-improved RCD.

STL10. This dataset contains out-of-distribution images in the unlabeled set, where the distribution mismatch between labeled and unlabeled sets inherently exists. Following [87], we evaluate on the five pre-defined folds and Table 4.2 shows that DC-SSL with both strategies can consistently outperform the SOTA methods, with more than 15% average accuracy improvements against FixMatch and more than 3% improvements against CoMatch.

4.3.3 Results for Matched Distribution

In Table 4.4 we also compare our strategies with recent SOTA SSL methods on conventional SSL settings. Following AlphaMatch and CoMatch, we also exploit the pre-known GCD as RCD to test our proposed two strategies. Surprisingly, without introducing more advanced techniques like alpha-divergence or contrastive learning techniques, our two strategies can consistently achieve a higher test accuracy than these SOTA

Method	CIFAR10		CIFAR100		MiniImageNet
	$\lvert D_x \rvert$=40	250	400	2500	1000
MixMatch[13]	52.46	88.95	33.39	60.06	33.74*
FixMatch[132]	86.19	94.93	51.15	71.71	39.03*
AlphaMatch[45]	91.35	95.03	61.26	74.98	-
CoMatch[87]	93.21*	95.14*	60.71*	74.36*	43.72*
Ours (TF)	95.31	95.87	62.47	75.10	45.19
Ours (TB)	93.89	95.24	61.33	74.62	44.23

Table 4.4: Mean test accuracy (%) in conventional SSL settings with balanced and matched distributions, *i.e.*, $\Gamma_i = \frac{|D_x|}{N}$ and $\gamma_u = 1$. Results with * in baselines are provided by our own testings.

α	0.8	0.9	0.99	0.999
Accuracy (%)	93.14	94.82	95.38	94.64

Table 4.5: Effect of the EMA ratio in our TF strategy

methods, especially when the labeled data is severely scarce. On CIFAR10 with only 40 labels, our TB strategy can obtain a high average accuracy of 95.31%, which is significantly better than 86.19% of FixMatch. It can also be seen from the table that AlphaMatch and CoMatch (both integrating the DA technique) can also achieve remarkable performance gains over FixMatch, demonstrating that modifying the pseudo-labels from a distribution perspective can effectively enhance the SSL performance. Comparing the results in Table 4.1, we further verify our claim that an accurate distribution of unlabeled samples is the key. Unsurprisingly, since we have the accurate distribution information in conventional SSL settings, directly modifying the pseudo-labels in our TB strategy can be more effective than our TF strategy that indirectly improves the pseudo-labels in a more moderate way.

4.3.4 Effects of Hyper-parameters

We first examine the effects of two hyper-parameters introduced in our proposed strategies using CIFAR10 with 40 labels in the conventional

λ_d	1.0	3.0	5.0	7.0
TB (matched)	93.92	94.33	94.67	94.81
TB (mismatched)	64.01	62.79	59.03	61.55

Table 4.6: Effect of the loss weight of $\mathcal{L}_d$ in our TB strategy.

SSL setting. The momentum coefficient α affects how the class distribution is estimated from historical predictions. A larger value of α can involve more historical predictions and relatively weaken the importance of the current predictions, therefore leading to more stable results as shown in Table 4.5. Meanwhile, the effect of the loss weight λ_d can be seen from Table 4.6: different values of λ_d can slightly affect the accuracy in the matched case while a smaller λ_d can better favor the mismatched case (following the same mismatched distribution setting as in Figure 3.6(b)). It is simply because a lower weight can better fit the iteratively-improved RCD and improve the pseudo-labels smoothly. By default, we set $\lambda_d = 1$ in all tests.

4.4 Summary

In this chapter, we carefully study how to improve SSL especially when there is a class distribution mismatch between the labeled and unlabeled sets. To address the mismatched issue, we propose DC-SSL, which can rectify the pseudo-labels from a distribution perspective and achieves the state-of-the-art performance across many SSL benchmarks under matched and mismatched class distribution scenarios. For example, in conventional matched distribution settings, DC-SSL (TF) can achieve a higher average accuracy of 95.31% on CIFAR10 (40 labels) compared to the previous SOTA of 93.21% and the baseline FixMatch of 86.19%. In the mismatched settings, our methods consistently outperform other SSL methods, *e.g.*, DC-SSL (TB) can obtain an average accuracy of 63.95% on CIFAR10 in a mismatched setting as in Figure 3.6(b), compared to Fixmatch of 57.54% and CoMatch of 52.73%. Our main contributions are summarized as follows,

- We revisit the EMA model in SSL and observe that it can be helpful in estimating unlabeled class distributions, although it may not produce more accurate high-confidence pseudo-labels directly.
- We propose a new method, DC-SSL, to enhance SSL performance from a distribution perspective. Two effective strategies are designed to improve the pseudo-labels by encouraging PCD of unlabeled data to approach an iteratively-improved RCD gradually.
- Our method can obtain new SOTA performance across different amounts of labeled data on standard SSL image classification benchmarks under both matched and mismatched distribution scenarios.

Chapter 5

Instance-specific and Model-adaptive Supervision for Semi-supervised Semantic Segmentation

In this chapter, we focus on the semi-supervised semantic segmentation (SSS) tasks. We find that most existing SSS studies treat all unlabeled data equally and barely consider the differences and training difficulties among unlabeled instances. We argue that differentiating unlabeled instances can promote instance-specific supervision to adapt to the model's evolution dynamically. To this end, we propose an instance-specific and model-adaptive supervision, dubbed as **iMAS**, for semi-supervised semantic segmentation. We thoroughly examine the efficacy of our iMAS on popular segmentation benchmarks, namely Pascal VOC and Cityscapes datasets. Our investigations demonstrate that iMas surpasses all existing methods and achieves new SOTA SSS performance.

5.1 Introduction

Though semantic segmentation studies [96, 27] have achieved significant progress, their enormous success relies on large datasets with high-quality pixel-level annotations. Semi-supervised semantic segmentation [60, 105] has been proposed as a powerful solution to mitigate the requirement for labeled data. Recent research on SSS has two main branches, including the self-training (ST) [83] and consistency regularization (CR) [134] based approaches. [159] follows a self-training paradigm and performs a selective re-training scheme to train on labeled and unlabeled data alternatively. Differently, CR-based works [115, 94] tend to apply data or model perturbations and enforce the prediction consistency between two differently-perturbed views for unlabeled data. In both branches, recent research [43, 164, 56] demonstrates that strong data perturbations like CutMix can significantly benefit the SSS training. To further improve the SSS performance, current state-of-the-art approaches [3, 144] integrate the advanced contrastive learning techniques into the CR-based approaches to exploit the unlabeled data more efficiently. Works in [61, 77] also aim to rectify the pseudo-labels through training an additional correcting network.

Despite their promising performance, SSS studies along this line come at the **cost** of introducing extra network components or additional training procedures. In addition, majorities of them treat unlabeled data equally and completely ignore the differences and learning difficulties among unlabeled samples. For instance, **randomly and indiscriminately** perturbing unlabeled data can inevitably over-perturb some difficult-to-train instances. Such over-perturbations exceed the generalization capability of the model and hinder effective learning from unlabeled data. As discussed in [164], it may also hurt the data distribution. Moreover, in most SSS studies, final consistency losses on different unlabeled instances are minimized in an **average** manner. However, blindly averaging can implicitly emphasize some difficult-to-train instances and result in model overfitting to noisy supervision.

In this chapter, we emphasize the cruciality of instance differences and aim to provide instance-specific supervision on unlabeled data in a model-adaptive way. There naturally exists two main questions. First,

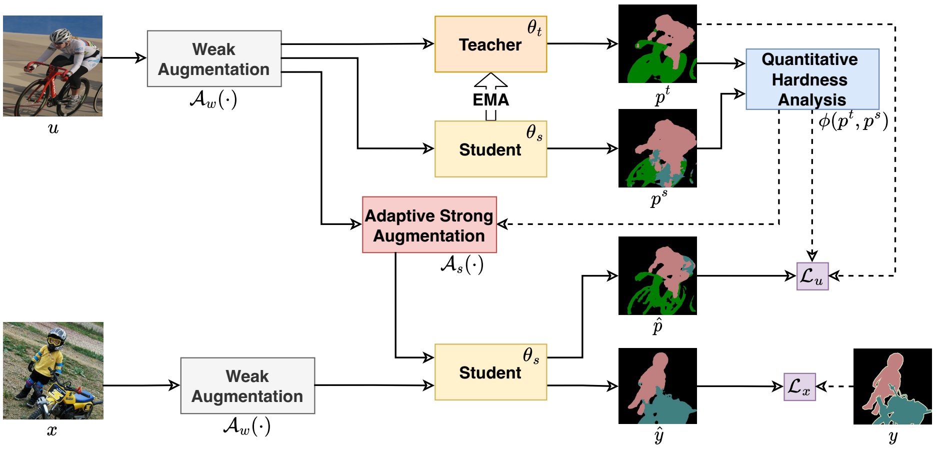


Figure 5.1: Diagram of our proposed iMAS. In a teacher-student framework, labeled data (x, y) is used to train the student model, parameterized by θ_s, by minimizing the supervised loss $\mathcal{L}_x$. Unlabeled data u, weakly augmented by $\mathcal{A}_w(\cdot)$, is first fed into both the student and teacher models to obtain predictions p^s and p^t, respectively. Then we perform quantitative hardness evaluation on each unlabeled instance by strategy $\phi(p^t, p^s)$. Such hardness information can be subsequently utilized: 1) to apply an adaptive augmentation, denoted by $\mathcal{A}_s(\cdot)$, on unlabeled data to obtain the student model's prediction $\hat{p}$; 2) to weigh the unsupervised loss $\mathcal{L}_u$ in a instance-specific manner. The teacher model's weight, θ_t, is updated by the exponential moving average (EMA) of θ_s across the training course.

how can we differentiate unlabeled samples? We design an instantaneous instance "hardness," to estimate 1) the current generalization ability of the model and 2) the current training difficulties of distinct unlabeled samples. Its evaluation is closely related to the training status of the model, *e.g.*, a difficult-to-train sample can become easier with the evolution of the model. Second, how can we inject such discriminative information into the SSS procedure? Since the hardness is assessed based on the model's performance, we can leverage such information to adjust the two critical operations in SSS, *i.e.*, data perturbations and unsupervised loss evaluations, to adapt to the training state of the model dynamically.

Motivated by all these observations, we propose an instance-specific and model-adaptive supervision, named **iMAS**, for semi-supervised semantic segmentation. As shown in Figure 5.1, following a standard consistency regularization framework, iMAS jointly trains the student and teacher models in a mutually-beneficial manner. The teacher model is an ensemble of historical student models and generates stable pseudo-labels for unlabeled data. Inspired by empirical and mathematical analysis in [48, 135], difficult-to-train instances may undergo considerable disagreement between predictions of the EMA teacher and the current student. Thus in iMAS, we first evaluate the instance hardness of each unlabeled sample by calculating the class-weighted symmetric intersection-over-union (IoU) between the segmentation predictions of the teacher (the historical) and student (the most recent) models. Then based on the evaluation, we perform model-adaptive data perturbations on each unlabeled instance and minimize an instance-specific weighted consistency loss to train models in a curriculum-like manner. In this way, different unlabeled instances are perturbed and weighted in a dynamic fashion, which can better adapt to the model's generalization capability throughout the training processes.

5.2 Method

The goal of semi-supervised semantic segmentation is to generalize a segmentation model by effectively leveraging a labeled training set $D_x = \{(x_i, y_i)\}_{i=1}^{|D_x|}$ and a large unlabeled training set $D_u = \{u_i\}_{i=1}^{|D_u|}$, with typically $|D_x| \ll |D_u|$. In our method, following the consistency regularization (CR) based semi-supervised classification approaches [132, 151], we aim to train the segmentation encoder and decoder on both labeled and unlabeled data simultaneously. In each iteration, given a batch of labeled samples $\mathcal{B}_x = \{(x_i, y_i)\}_{i=1}^{|\mathcal{B}_x|}$ and unlabeled samples $\mathcal{B}_u = \{u_i\}_{i=1}^{|\mathcal{B}_u|}$, the overall training loss is formulated as,

$$\mathcal{L} = \mathcal{L}_x + \lambda_u \mathcal{L}_u, \tag{5.1}$$

where λ_u is a scalar hyper-parameter to adjust the relative importance between the supervised loss $\mathcal{L}_x$ on $\mathcal{B}_x$ and the unsupervised loss $\mathcal{L}_u$ on

Algorithm 2: iMAS algorithm in a mini-batch.

```
Input: Labeled batch $\mathcal{B}_x = \{(x_i, y_i)\}_{i=1}^{|\mathcal{B}_x|}$, unlabeled batch
  $\mathcal{B}_u = \{u_i\}_{i=1}^{|\mathcal{B}_u|}$ ($|\mathcal{B}_x| = |\mathcal{B}_u|$), hardness evaluation strategy $\phi$, weak
  augmentation $\mathcal{A}_w(\cdot)$, adaptive strong augmentation $\mathcal{A}_s(\cdot)$
Parameter: confidence threshold $\tau$, unsupervised loss weight $\lambda_u$

$\mathcal{L}_x = \frac{1}{|\mathcal{B}_x|}\sum_{i=1}^{|\mathcal{B}_x|}\frac{1}{H\times W}\sum_{j=1}^{H\times W}\mathrm{H}(\hat{y}_i(j), y_i(j))$    // calculate the supervised loss.
for $u_i \in \mathcal{B}_u$ do
    $p_i^s = f_{\theta_s}(\mathcal{A}_w(u_i))$    // obtain segmentation predictions on weakly-augmented instances.
    $p_i^t = f_{\theta_t}(\mathcal{A}_w(u_i))$    // obtain pseudo-labels from the teacher model.
    $\gamma_i = \phi(p_i^t, p_i^s)$    // evaluate the hardness of each instance.
end for
$\mathcal{L}_u = \frac{1}{|\mathcal{B}_u|}\sum_{i=1}^{|\mathcal{B}_u|}\frac{\gamma_i}{2H\times W}\sum_{j=1}^{H\times W}[\mathbb{1}(\max(p_i^t(j)) \geq \tau)\mathrm{H}(f_{\theta_s}(\mathcal{A}_s^I(u_i)), p_i^t(j)) + \mathbb{1}(\max(p_i^{t'}(j)) \geq \tau)\mathrm{H}(f_{\theta_s}(\mathcal{A}_s^C(u_i)), p_i^{t'}(j))]$    // calculate model-adaptive consistency loss
return $\mathcal{L} = \mathcal{L}_x + \lambda_u \mathcal{L}_u$
```

$\mathcal{B}_u$. Without introducing extra losses or network components, iMAS can perform effectively quantitative hardness analysis for each unlabeled instance and then supervise the training on unlabeled data in a model-adaptive fashion across the training course. In this section, we first introduce our proposed iMAS at a high level in Sec. 5.2.1 and then present the detailed designs in terms of the quantitative hardness analysis in Sec. 5.2.2 and the model-adaptive guidance in Sec. 5.2.3.

5.2.1 Overview

Built on top of the CR-based semi-supervised framework, iMAS jointly trains a student model with the learnable weights θ_s and a teacher model with the learnable weights θ_t in a mutually-beneficial manner. The complete algorithm is shown in algorithm 2. On the one hand, the teacher model is updated by the exponential moving averaging of the student weights, *i.e.*,

$$\theta_t \leftarrow \alpha\theta_t + (1-\alpha)\theta_s, \tag{5.2}$$

where α is a common momentum parameter, set as 0.996 by default. On the other hand, the student model relies on the pseudo-labels generated by the teacher model to be trained on the unlabeled data. Specifically, the student model is trained via minimizing the total loss $\mathcal{L}$ in Equation 5.1, which consists of two cross-entropy loss terms, $\mathcal{L}_u$ and $\mathcal{L}_x$, applied on labeled and unlabeled data, respectively. Let $\mathrm{H}(z_1, z_2)$ denote the cross-entropy loss between prediction distributions z_1 and z_2. The supervised loss $\mathcal{L}_x$ is calculated as,

$$\mathcal{L}_x = \frac{1}{|\mathcal{B}_x|}\sum_{i=1}^{|\mathcal{B}_x|}\frac{1}{H\times W}\sum_{j=1}^{H\times W}\mathrm{H}(\hat{y}_i(j), y_i(j)), \tag{5.3}$$

where $\hat{y}_i = f_{\theta_s}(\mathcal{A}_w(x_i))$, represents the segmentation result of the student model on the i-th weakly-augmented labeled instance. j represents the j-th pixel on the image or the corresponding segmentation mask with a resolution of $H \times W$. The weak augmentation $\mathcal{A}_w$ includes standard resizing, cropping, and flipping operations. Importantly, the way to leverage the unlabeled data is the key to semi-supervised learning and also the crucial part differentiating our method from others. In most CR-based studies, the standard (*std*) unsupervised loss $\mathcal{L}_u^{std}$ is simply,

$$\mathcal{L}_u^{std} = \frac{1}{|\mathcal{B}_u|}\sum_{i=1}^{|\mathcal{B}_u|}\frac{1}{H\times W}\sum_{j=1}^{H\times W}\mathbb{1}(\max(p_i^t(j)) \geq \tau)\mathrm{H}(\hat{p}_i(j), p_i^t(j)), \tag{5.4}$$

where $\hat{p}_i = f_{\theta_s}(\mathcal{A}_s^{std}(u_i))$ represents the segmentation output of the student model on the i-th unlabeled instance augmented by $\mathcal{A}_s^{std}$, while $p_i^t = f_{\theta_t}(\mathcal{A}_w(u_i))$ represents the segmentation outputs of the teacher model on the i-th weakly-augmented unlabeled instance. τ is a predefined confidence threshold to select high-confidence predictions. $\mathcal{A}_s^{std}$ represents standard **instance-agnostic** strong augmentations, including intensity-based data augmentations [35] and CutMix [166] as shown in

Table 5.1. However, such operations are limited in ignoring the differences and learning difficulties among unlabeled samples.

Differently, in our iMAS, we treat each instance discriminatively and provide instance-specific supervision on the training of unlabeled data. As shown in Figure 5.1, we first evaluate the hardness of each weakly-augmented unlabeled instance via strategy ϕ, and then employ the **instance-specific and model-adaptive** supervision on the strong augmentations $\mathcal{A}_s$ as well as the calculations of unsupervised loss $\mathcal{L}_u$, which are elaborated in following sections.

5.2.2 Quantitative Hardness Analysis

In iMAS, we perform quantitative hardness analysis to differentiate distinct unlabeled samples. In most hardness-related studies, the instantaneous or historical training losses [181, 131] to the ground truth are used to assess the instance hardness. However, in semi-supervised segmentation, evaluating the hardness of unlabeled data is challenging at 1) lacking accurate ground-truth labels and 2) dynamic changes closely related to the model performance. A “hard" sample can become easier with the evolution of the model, but such dynamics cannot be easily identified without accurate label information. Inspired by [48, 143], it is more difficult for the teacher and student models to achieve consensus on a hard instance. Hence we design a symmetric class-weighted IoU between the segmentation results of the student and teacher models to evaluate the instantaneous hardness. The class-weighted design is used to alleviate the class-imbalanced issue in segmentation tasks.

Such evaluation, denoted by ϕ, can be regarded as a function of the model performance and dynamically estimate the training difficulties of unlabeled crops throughout the training process. Specifically, as shown in Figure 5.1, we first obtain the segmentation predictions p_i^s and p_i^t on the i-th weakly-augmented unlabeled instance, from the student and teacher models, respectively,

$$p_i^s = f_{\theta_s}(\mathcal{A}_w(u_i)), \rho_i^s = \frac{1}{H \times W} \sum_{j=1}^{H \times W} \mathbb{1}(\max(p_i^s(j)) \geq \tau) \tag{5.5}$$

$$p_i^t = f_{\theta_t}(\mathcal{A}_w(u_i)), \rho_i^t = \frac{1}{H \times W} \sum_{j=1}^{H \times W} \mathbb{1}(\max(p_i^t(j)) \geq \tau) \tag{5.6}$$

where ρ_i^s and ρ_i^t represent the high-confidence ratios on p_i^s and p_i^t, respectively. Let $\mathrm{wIOU}(z_1, z_2)$ denote the class-weighted IoU between segmentation predictions z_1 and z_2. Note that, this evaluation is not commutative, *i.e.*, $\mathrm{wIOU}(z_1, z_2) \neq \mathrm{wIOU}(z_2, z_1)$. To make wIoU valid for hardness evaluation at each iteration, the symmetric hardness γ_i for i-th unlabeled instance is calculated as,

$$\gamma_i = \phi(p_i^t, p_i^s) = 1 - [\frac{\rho_i^s}{2} \mathrm{wIOU}(p_i^s, p_i^t) + \frac{\rho_i^t}{2} \mathrm{wIOU}(p_i^t, p_i^s)] \tag{5.7}$$

where $1/2$ ensures the hardness is in $[0, 1]$. In this way, the harder instance that requires better generalization ability holds a larger value of γ while the easier one will be identified by a smaller γ.

5.2.3 Model-adaptive Supervision

With the quantitative hardness evaluation for each unlabeled instance, we carefully inject such information into the training process by performing instance-specific and model-adaptive strong perturbations and loss modifications. Specifically, we first leverage the instance hardness for adaptive augmentations both individually and mutually. By “individually", we adjust the intensity-based augmentation applied on each instance according to its absolute hardness value; by “mutually", we replace random pairs of unlabeled data in CutMix with specific **hard-easy pairs** assigned by sorting the corresponding hardness. Moreover, instead of indiscriminately averaging the losses, we **weigh** the losses of different unlabeled instances by multiplying their corresponding hardness. We present these details below.

Weak Augmentations	
Random scale	Randomly resizes the image by $[0.5, 2.0]$.
Random flip	Horizontally flip the image with a probability of 0.5.
Random crop	Randomly crops an region from the image (513 × 513, 769 × 769).
Strong intensity-based Augmentations	
Identity	Returns the original image.
Invert	Inverts the pixels of the image.
Autocontrast	Maximizes (normalize) the image contrast.
Equalize	Equalize the image histogram.
Gaussian blur	Blurs the image with a Gaussian kernel.
Contrast	Adjusts the contrast of the image by [0.05, 0.95].
Sharpness	Adjusts the sharpness of the image by [0.05, 0.95].
Color	Enhances the color balance of the image by [0.05, 0.95]
Brightness	Adjusts the brightness of the image by [0.05, 0.95]
Hue	Jitters the hue of the image by [0.0, 0.5]
Posterize	Reduces each pixel to [4,8] bits.
Solarize	Inverts all pixels of the image above a threshold value from [1,256).
CutMix augmentation	
CutMix	Copy and paste random size regions among different unlabeled images.

Table 5.1: List of various image transformations in iMAS.

Model-adaptive Strong Augmentations

The popular strong augmentations in recent semi-supervised segmentation studies mainly consist of two different types: intensity-based augmentation and CutMix, as shown in Table 5.1. In iMAS, we apply instance-specific adjustments to both types of augmentations.

Intensity-based augmentations. Standard intensity-based data augmentations randomly select two kinds of image operations from an augmentation pool and apply them to the weakly-augmented instances. However, as discussed by [164], strong augmentations may hurt the data distribution and degrade the segmentation performance, especially during the early training phase. Unlike distribution-specific designs [164], we simply adjust the augmentation degree for an unlabeled instance by mixing its strongly-augmented and weakly-augmented outputs. Formally, the ultimate augmented output of the i-th unlabeled instance, $\mathcal{A}_s^I(u_i)$, can be obtained by,

$$\mathcal{A}_s^I(u_i) \leftarrow \gamma_i \mathcal{A}_s^I(u_i) + (1-\gamma_i)\mathcal{A}_w(u_i), \tag{5.8}$$

where the distortion caused by the intensity-based strong augmentation is proportionally weakened by the corresponding weakly-augmented output. In this way, harder instances with larger hardness are not perturbed significantly so that the model will not be challenged on potentially out-of-distribution cases. On the other hand, easier instances with lower values of γ, which have been well fitted by the model, can be further learned from their strongly-augmented variants. Such model-adaptive augmentations can better adjust to the model's generalization ability.

CutMix-based augmentations. CutMix [166] is a widely adopted technique to boost semi-supervised semantic segmentation. It is applied between unlabeled instances with a predefined probability. It can randomly copy a region from one instance to another, and so do their corresponding segmentation results. The augmentation pairs are generated randomly. Differently, in iMAS, we improve the standard CutMix by a model-adaptive design, which is distinct in two ways: **1)** the mean hardness determines the trigger probability of CutMix augmentation over the mini-batch instead of using a predefined hyper-parameter; **2)** the copy-and-paste pairs are assigned specifically between the hard and easy samples. According to the instance hardness, we obtain two sequences by sorting unlabeled samples of a mini-batch in the ascending and descending orders, respectively. We then aggregate two sequences element-by-element to generate the hard-easy pairs. Formally, given a specific hard-easy pair, (u_m, u_n), the model-adaptive CutMix can be expressed as,

$$\left.\begin{aligned} \mathcal{A}_s^C(u_m) &\leftarrow M_m \odot u_n + (\mathbf{1} - M_m) \odot u_m \\ p_m^{t'} &\leftarrow M_m \odot p_n^t + (\mathbf{1} - M_m) \odot p^t, \\ \mathcal{A}_s^C(u_n) &\leftarrow M_n \odot u_m + (\mathbf{1} - M_n) \odot u_n \\ p_n^{t'} &\leftarrow M_n \odot p_m^t + (\mathbf{1} - M_n) \odot p_n^t \end{aligned}\right\}, \tag{5.9}$$

$$\text{by a triggering probability of } \overline{\gamma} = \frac{1}{|\mathcal{B}_u|} \sum_{n=1}^{|\mathcal{B}_u|} \gamma_n, \tag{5.10}$$

where M_m and M_n denote the randomly generated region masks for u_m

Method	ResNet-50		
	1/16 (662)	1/8 (1323)	1/4 (2646)
Supervised*	63.8	69.0	72.5
MT [134]	66.8	70.8	73.2
CCT [115]	65.2	70.9	73.4
CutMix-Seg [43]	68.9	70.7	72.5
GCT [69]	64.1	70.5	73.5
CAC [78]	70.1	72.4	74.0
CPS [31]	72.0	73.7	74.9
PSMT† [94]	72.8	75.7	76.4
ELN [77]	70.5	73.2	74.6
ST++ [159]	72.6	74.4	75.4
iMAS (ours)	74.8	76.5	77.0
U^2PL‡ [144]	72.0	75.2	76.2
iMAS (ours)‡	**75.9**	**76.7**	**77.1**

Table 5.2: Comparison with SOTA methods on **PASCAL VOC 2012** `val` set under different partition protocols, using R50 as the backbone. Labeled images are sampled from the *blender* training set (augmented by SBD dataset), including $10,583$ samples in total. ‡ means the results are obtained by setting the output_stride as 8 in DeepLabV3+ (16 for others). * denotes our reproduced results.

and u_n, respectively. Besides, the pseudo-labels need to be revised accordingly after applying CutMix data augmentations, obtaining $p_m^{t'}$ and $p_n^{t'}$. This mutual augmentation is applied following a Bernoulli process, *i.e.*, triggered only when a uniformly random probability is higher than the average hardness $\overline{\gamma}$.

Model-adaptive Unsupervised Loss

Considering the learning difficulty of each instance, we design a model-adaptive unsupervised loss to learn from unlabeled data differentially. Inspired by curriculum learning [9], we prioritize the training on easy samples over hard ones. Precisely, we weigh the unsupervised losses for each instance by multiplying their corresponding easiness, evaluated by $1-\gamma$. Combined with model-adaptive augmentations, we can calculate

Method	ResNet-101		
	1/16 (662)	1/8 (1323)	1/4 (2646)
Supervised*	67.4	72.1	74.7
MT [134]	70.6	73.2	76.6
CCT [115]	68.0	73.0	76.2
CutMix-Seg [43]	72.6	72.7	74.3
GCT [69]	69.8	73.3	75.3
CAC [78]	72.4	74.6	76.3
CPS [31]	74.5	76.4	77.7
PSMT† [94]	75.5	78.2	78.7
ELN [77]	72.5	75.1	76.6
ST++ [159]	74.5	76.3	76.6
iMAS (ours)	76.5	77.9	78.1
U^{2}PL‡ [144]	74.4	77.6	78.7
iMAS (ours)‡	**77.2**	**78.4**	**79.3**

Table 5.3: Comparison with SOTA methods on **PASCAL VOC 2012** `val` set under different partition protocols, using R101 as the backbone. All notations are the same as in Table 5.2.

the unsupervised loss by,

$$\mathcal{L}_u = \frac{1}{|\mathcal{B}_u|}\sum_{i=1}^{|\mathcal{B}_u|}\frac{1-\gamma_i}{2H\times W}\sum_{j=1}^{H\times W}[\mathbb{1}(\max(p_i^t(j)) \geq \tau)\mathrm{H}(f_{\theta_s}(\mathcal{A}_s^I(u_i)), p_i^t(j)) + \mathbb{1}(\max(p_i^{t'}(j)) \geq \tau)\mathrm{H}(f_{\theta_s}(\mathcal{A}_s^C(u_i)), p_i^{t'}(j))]. \tag{5.11}$$

Since the hardness is evaluated upon each (weakly augmented) image instance, under its guidance, the two strong augmentations are performed separately rather than in a cascading manner. In this way, the model will not be trained on over-distorted variants, and our model-adaptive designs can be effectively utilized.

5.3 Experiment

In this section, we examine the efficacy of our method on standard semi-supervised semantic segmentation benchmarks and conduct extensive ablation studies to further verify the superiority and stability.

5.3.1 Experimental Setup

Dataset and backbone. Following recent SOTAs [31, 159] in semi-supervised segmentation, we adopt DeepLabv3+ [27] based on Resnet [54] as our segmentation backbone and investigate the test performance on Pascal VOC2012 [41] and Cityscapes [34], in terms of the mean intersection-over-union (mIOU). The classical VOC2012 consists of 21 classes with 1464 training and 1449 validation images. As a common practice, the blended training set is also involved, including additional 9118 training images from the Segmentation Boundary (SBD) dataset [52]. Cityscapes is a large dataset on urban street scenes with 19 segmentation classes. It consists of 2975 training and 500 validation images with fine annotations.

Implementation details. For both the student and the teacher models, we load the ResNet weights pre-trained on ImageNet [37] for the encoder and randomly initialize the decoder. An SGD optimizer with a momentum of 0.9 and a polynomial learning-rate decay with an initial value of 0.01 are adopted to train the student model. The total training epoch is 80 for VOC2012 and 240 for Cityscapes. Following [144], training images are randomly cropped into 513×513 and 769×769 for Pascal VOC2012 and Cityscapes, respectively. On Cityscapes, we also use the sliding evaluation to examine the performance on validation images with a resolution of 1024×2048. We set $\mathcal{B}_u = \mathcal{B}_x = 16$ and adopt the sync-BN for all runs.

5.3.2 Comparison with State-of-the-Art Methods

In this section, we demonstrate the superior performance of our iMAS on both classic and blended VOC 2012 and Cityscapes under different semi-supervised partition protocols. It is noteworthy that, on blended VOC, U^{2}PL [144] prioritizes selecting high-quality labels from classic VOCs. Instead, we randomly sample labels from the entire dataset and adopt the same partitions as specified in [31, 94]. Therefore, we reproduce corresponding results on U^{2}PL and evaluate iMAS with different output_strides, 8 and 16, respectively, for fair comparisons.

Method	1/16 (92)	1/8 (183)	1/4 (366)	1/2 (732)	Full (1464)
Supervised *	45.5	57.5	66.6	70.4	72.9
CutMix-Seg [43]	52.2	63.5	69.5	73.7	76.5
PseudoSeg [186]	57.6	65.5	69.1	72.4	73.2
PC^{2}Seg [179]	57.0	66.3	69.8	73.1	74.2
CPS [31]	64.1	67.4	71.7	75.9	-
PSMT [94]	65.8	69.6	76.6	78.4	80.0
ST++ [159]	65.2	71.0	74.6	77.3	79.1
iMAS (ours)	68.8	74.4	78.5	79.5	81.2
U^{2}PL‡ [144]	68.0	69.2	73.7	76.2	79.5
iMAS‡(ours)	**70.0**	**75.3**	**79.1**	**80.2**	**82.0**

Table 5.4: Comparison with SOTA methods on *classic* **PASCAL VOC 2012** `val` set under different partition protocols. Labeled images are sampled from the official VOC `train` set, including 1,464 samples in total. Results are reported using Resnet-101. All notations are the same as in Table 5.2.

PASCAL VOC 2012. In Tables 5.2 and 5.4, we compare our iMAS with recent SOTA methods on blended and classic VOC, respectively. We can clearly see from Table 5.2 that iMAS can consistently outperform others regardless of using ResNet-50 or ResNet-101 as the segmentation encoder. The performance gain becomes more noticeable and clear as fewer labels are available. *e.g.*, in the 1/16 partition, iMAS can improve the supervised baseline by 11% and 9.1% when using ResNet-50 and ResNet-101 as the encoders, respectively, and improve the ST++ [159] by 2.2% and 2.0%, accordingly. Checking the results among different partitions, we can also observe that iMAS can even obtain better performance while using fewer labels compared to other SOTAs. For example, iMAS can obtain a high mIOU of 75.9% using only 662 labels, while U^{2}PL requires 1323 labels to obtain a comparable performance of 75.2% mIOU on blended VOC. It suggests our method is more label efficient and potentially a good solution for label-scarce scenarios. In classic VOC with high-quality labels, our methods can outperform SOTA methods by a notable margin, as shown in Table 5.4. We attribute this improvement

Method	1/16 (186)	1/8 (372)	1/4 (744)	1/2 (1488)
Supervised *	64.0	69.2	73.0	76.4
MT [134]	66.1	72.0	74.5	77.4
CCT [115]	66.4	72.5	75.7	76.8
GCT [69]	65.8	71.3	75.3	77.1
CPS [31]	74.4	76.6	77.8	78.8
CPS† [144]	69.8	74.3	74.6	76.8
PSMT [94]	-	75.8	76.9	77.6
ELN [77]	-	70.3	73.5	75.3
ST++ [159]	-	72.7	73.8	-
U^{2}PL * [144]	67.8	72.5	74.8	77.1
iMAS (ours)	74.3	77.4	78.1	79.3
U^{2}PL‡* [144]	69.0	73.0	76.3	78.6
iMAS (ours)‡	**75.2**	**78.0**	**78.2**	**80.2**

Table 5.5: Comparison with SOTA methods on **Cityscapes** val set under different partition protocols. Labeled images are sampled from the Cityscapes train set, including 2,975 samples in total. Results are reported using Resnet-50. * and † represent reproduced results in iMAS and U^{2}PL, respectively. Results with ‡ are obtained by setting the output_stride as 8 in DeepLabV3+.

to the model-adaptive guidance that treats each unlabeled instance differently and effectively leverages them by instance-specific strategies in HegSeg. Generally, in both classic and blended cases, reserving a large feature map (*i.e.*, set output_stride=8) can slightly improve the test performance.

Cityscapes. In table 5.5, we evaluate our method on more challenging Cityscapes with ResNet-50 as the segmentation encoder. iMAS with output_stride= 8 can achieve high mIOUs of 75.2%, 78.1%, 78.2%, 80.2%, in four different splits (1/16, 1/8, 1/4, 1/2), respectively. When output_stride= 16, given only 186 labeled images, iMAS can obtain a notable performance gain of 10.3% against the supervised baseline and 6.5% against the previous best, U^{2}PL. Not relying on any pseudo-rectifying networks [77] or extra self-supervised supervisions [144], iMAS achieves substantially better performance than the previous SOTAs, especially

iMAS on			mIOU (%)
Loss $\mathcal{L}_u$	Augs of $\mathcal{A}_s^I$	Augs of $\mathcal{A}_s^C$	
			72.1 (supervised)
✓			75.5 (3.4↑)
	✓		76.5 (4.4↑)
		✓	76.9 (4.8↑)
✓	✓	✓	**77.9 (5.8↑)**

Table 5.6: Ablation studies on the effectiveness of the instance-specific model-adaptive supervision on the unsupervised loss, intensity-based and CutMix augmentations, respectively. Results are reported on **PASCAL VOC 2012** under the 1/8 (1323) partition using Resnet-101 as the backbone. Improvements over the baseline are marked in blue.

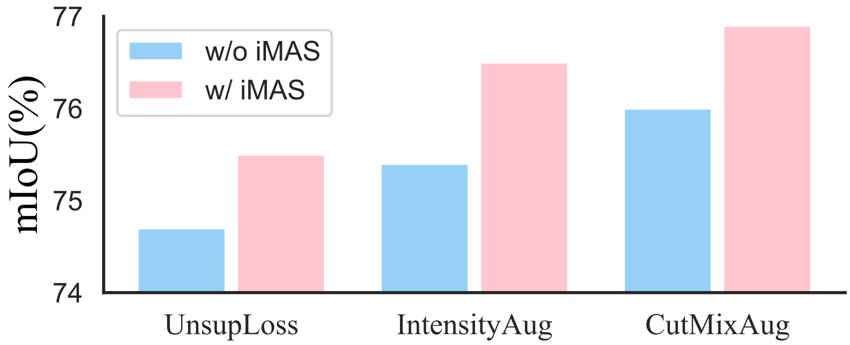
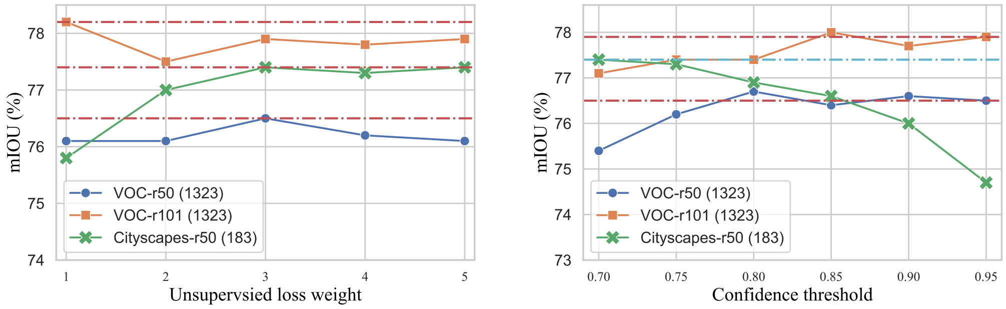


Figure 5.2: Effectiveness of iMAS on the unsupervised loss, intensity-based and CutMix augmentations, respectively.

with fewer labels. Despite the simplicity of iMAS, the impressive performance further demonstrates the effectiveness and importance of our instance-specific and model-adaptive guidance. Surely, regardless of different semi-supervised approaches, we can see from Tables 5.5 that providing more labeled samples can easily improve the semi-supervised performance.

5.3.3 Ablations Study

We conduct ablation studies in the 1/8 partitions of blended VOC and Cityscapes, and examine the impact of the model-adaptive guidance and approach-related hyper-parameters.



(a) λ_u **(b)** τ

Figure 5.3: We examine the effect of the loss weight and confidence threshold on VOC and Cityscapes under the 1/8 protocol in Figure (a) and (b), respectively. Best viewed on screen.

Effectiveness of model-adaptive guidance. The key of iMAS lies in the instance-specific and model-adaptive guidance. In Table 5.6, we conduct a series of experiments on VOC2012 dataset to demonstrate its effectiveness on three components, the unsupervised loss, intensity-based and CutMix augmentations, respectively. It can been seen from Figure 5.2 that performing model-adaptive guidance can consistently improve the standard operations, yielding around 1% improvements on all standard counterparts. The powerfulness of strong augmentations can also be witnessed, as discussed in [159]. As a whole, iMAS can bring an improvement of 5.8% against the supervised baseline.

Impact of hyper-parameters. In Figure 5.3, we investigate the influence of different λ_u and τ on both datasets. It can be seen from Figure 5.3(a) that iMAS is not very sensitive to the loss weight on VOC while a large λ_u is beneficial for Cityscapes. By default, we set $\lambda_u = 3$ for all runs. According to Figure 5.3(b), we set $\tau = 0.95$ for VOC and $\tau = 0.7$ for Cityscapes as default settings. This is simply because Cityscapes is a more challenging dataset requiring better discriminating ability and using a high-threshold will prevent models effectively learning from unlabeled samples.

Hardness Aanalysis The hardness evaluation closely depends on

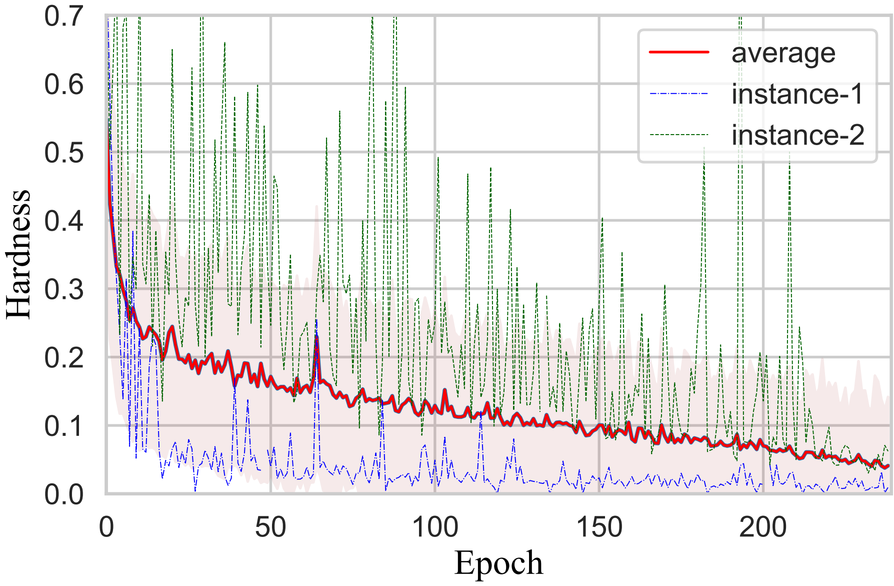
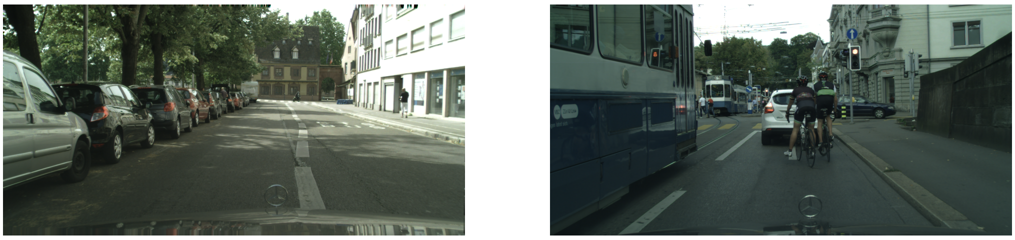


Figure 5.4: We examine how the mean instance hardness varies across the training course on Cityscapes under the 1/4 partition.



(a) Instance-1 (easy one)

(b) Instance-2 (hard one)

Figure 5.5: RGB images of Instance-1 and Instance-2 in Fig. 5.4

distinct instances and the model's training status. We can see from Figure 5.4 that both the mean and standard deviation of hardness evaluations on unlabeled data decrease as training processes and the model performance improves. Specifically, easy instances (*e.g.*, Instance-1) can hold a low hardness from the very beginning, while the hardness of hard instances (*e.g.*, Instance-2) fluctuates along the training process but eventually decreases. To verify the correctness, in Figure 5.5, we show the corresponding images of the two instances in Figure 5.4. Compared with Instance-2 (**I2**), the hardness of Instance-1 (**I1**) drops rapidly to below 0.1 after 20 epochs, while Instance-2's hardness is higher and decreases

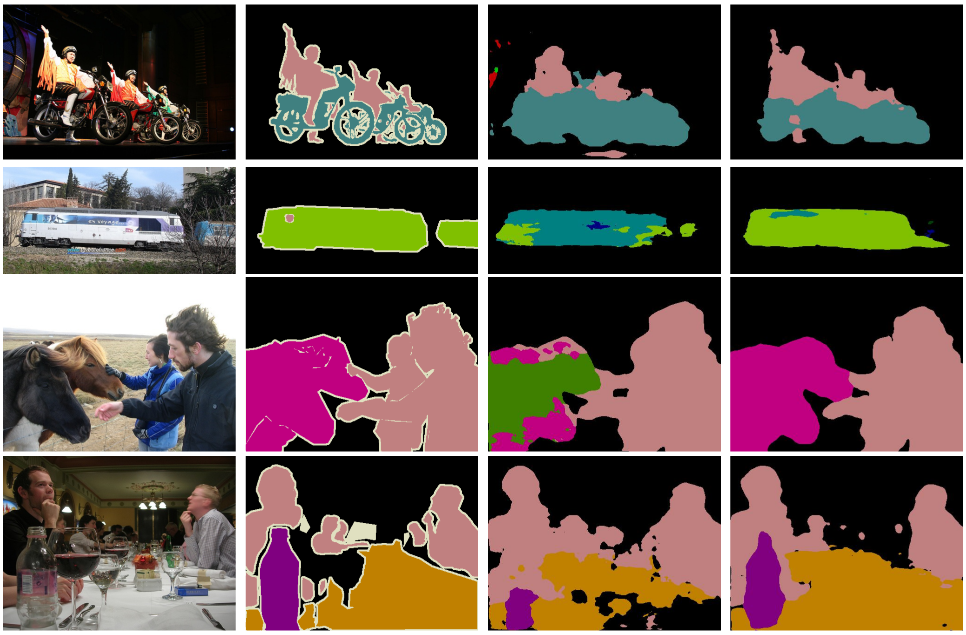


Figure 5.6: Qualitative results on Pascal VOC2012 using 183 fine labels. Columns from left to right denote the original images, the ground-truth, the supervised segmentation results, and the iMAS segmentation results, respectively.

slowly, *i.e.*, I1 is much easier than I2. This is consistent with our human perception that the bicycle-car overlap in I2 is harder to segment. Moreover, from the algorithm side, I2 contains more minority classes like *wall, bicycle, train, traffic light and sign*, while I1 mainly includes majority classes like *road, building, cars*. Such observation is also supported by the ultimate test performance on different categories, *e.g.*, *road: 98.05, cars:95.62, wall:51.39, train:44.49*. Hardness fluctuations result from randomly scaled and cropped operations, a common practice in segmentation. I2 fluctuates more frequently than I1, which also reflects the training difficulty of I2.

Qualitative Results. We also present some segmentation results on Pascal VOC 2012 in Figure 5.6 under the 183 partition protocol, using the Resnet-101 as the encoder. We can see that many mis-classified pixels and ignored segmentation details like arms in the supervised-only results are corrected in iMAS.

5.4 Summary

In this chapter, we highlight the instance uniqueness and propose iMAS, an instance-specific and model-adaptive supervision for semi-supervised semantic segmentation. Relying on our class-weighted symmetric hardness-evaluating strategies, iMAS treats each unlabeled instance discriminatively and employ model-adaptive augmentation and loss weighting strategies on each instance. Without introducing additional networks or losses, iMAS obtains new SOTA performance on Pascal VOC 2012 and Cityscapes datasets under different partition protocols. For example, our method obtains a high mIOU of 75.3% with only 183 labeled data on VOC 2012, which is 17.8% higher than the supervised baseline and 4.3% higher than the previous SOTA. Our main contributions are summarized as follows,

- iMAS can boost the SSS performance by highlighting the instance differences, without introducing extra network components or training losses.
- We perform a quantitative hardness-evaluating analysis for unlabeled instances in segmentation tasks, based on the class-weighted teacher-student symmetric IoU.
- We propose an instance-specific and model-adaptive SSS framework that injects instance hardness into loss evaluation and data perturbation to dynamically adapt to the model's evolution.

Chapter 6

Boosting Semi-supervised Medical Image Segmentation with Data Perturbation and Model Stabilization

In this chapter, we study the Semi-supervised Medical Image Segmentation (SSMIS), where the labeled data is even scarce. Instead of integrating any recent advanced techniques like contrastive learning losses or multiple learning branches, we emphasize the crutiality of data perturbation and model stabilization in SSMIS, and propose a simple yet effective approach to boost SSMIS performance significantly, dubbed as DPMS. Despite its simplicity, DPMS can obtain new state-of-the-art (SOTA) performance on the public 2D ACDC and 3D LA datasets across various semi-supervised settings.

6.1 Introduction

Medical image segmentation is an urgent vision task that contributes to medical image reasoning, which is vital in the development of the computer-aided diagnosis (CAD) system [137, 147, 66, 90]. Conventional supervised medical image segmentation methods rely heavily on extensive pixel-level annotated data for the model training. In practical medical applications, obtaining large amounts of fine-grained annotated

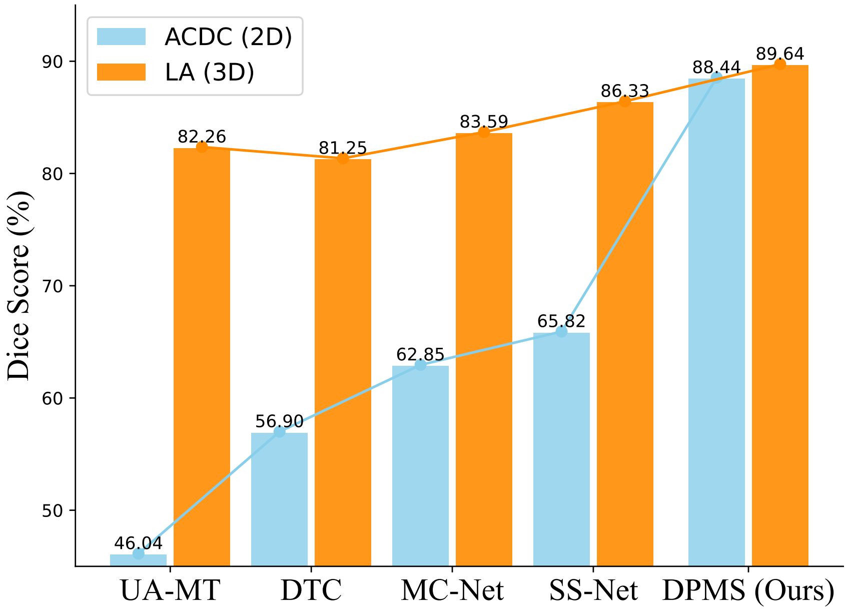


Figure 6.1: We compare our DPMS with recent SSIMS methods in terms of the Dice score (%) on 2D ACDC and 3d LA datasets with only 5% labeled data. Remarkable performance gains can be observed.

data is costly and even infeasible, greatly hindering their wide applications [161, 162]. To this end, many studies have been focused on semi-supervised medical image segmentation (SSMIS), which aims at learning a deep segmentation model by using a limited number of annotated medical images and abundant unlabeled medical images to achieve satisfied segmentation performance [163, 147, 7, 148, 98].

Following the research line of semi-supervised semantic segmentation on natural images [159, 94], SSMIS studies have evolved from earlier self-training based methods [6, 142] into recent dominant consistency regularization (CR) based approaches [163, 148]. CR-based approached methods, like Mean-Teacher [134] and FixMatch [132], leverage the label-preserving data or model perturbations to encourage prediction consistency on differently perturbed views from the same input. The key to such methods is to generate prediction disagreements on unlabeled data [174]. To further improve SSMIS performance, recent studies tends to introduce more advanced and complicated techniques, such as additional transformer-based branch [98], extra feature-level perturbations and constraints [148], additional self-supervised contrastive losses [120]. Despite their impressive performance, these methods usually come at the cost of increasingly complex designs and benefit SSMIS in an indirect manner. Differently, we focus on the semi-supervised problem itself to produce appropriate prediction disagreement, and strive to propose a simple yet highly effective method to boost SSMIS directly.

In an effort to simplify SSMIS studies, in this chapter, we diverges from complex designs and redirects the focus towards the intrinsic nature of the semi-supervised problem, *i.e.,* to produce appropriate prediction disagreement. As shown in Figure 6.2, we first conduct a thorough analysis by revisiting SSMIS from three distinct perspectives: the data, the model, and the loss supervisions. Through a comprehensive study of the corresponding strategies, we rigorously evaluate their effectiveness and implications. Specifically, employing data augmentations is the most straightforward and effective way to generate label-preserving disagreement in SSMIS. However, most of previous works focus on investigating the effectiveness of data perturbations in the natural image domain, while few works studied the effectiveness on medical images. Hence, we first revisit the contributions of different data perturbations to the SSMIS problem. Based on our findings, we figure out that data perturbations can yield sufficient prediction disagreements and boost the SSMIS performance. However, as discussed in [164, 174], too strong data perturbations will inevitably hurt the distribution of the original data and consequently degrade the performance. To tackle the issue, we further study the model stabilization in SSMIS, where we come up with two simple yet effective model stabilization strategies, *i.e.,* the EMA-BN and Extra-Weak, to prevent the model statistics from being severely disturbed. As a result, we highlight the significance of data perturbation and model stabilization in SSMIS.

Motivated by our revisiting, we propose DPMS, that adopts a simple teacher-student framework to employ the effective **D**ata **P**erturbation and **M**odel **S**tabilization strategies to boost SSMIS. On the one hand, DPMS poisons the unlabeled data via various strong augmentations, including geometrical transformantions, intensity-based perturbation and

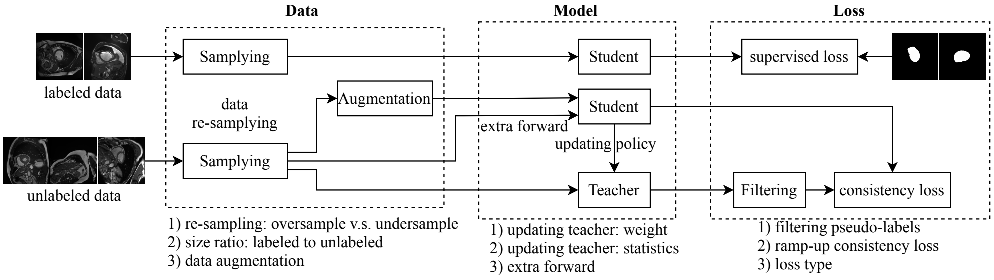


Figure 6.2: Revisiting SSMIS in terms of the data, the model and the loss supervision.

copy-paste [44], to enlarge prediction disagreements considerably. On the other hand, DPMS utilizes an extra forward (forwarding unlabeled data into the student model) and momentum updating strategies for normalization statistics to stabilize the training on unlabeled data effectively. Without bells and whistles, our simple DPMS can achieve remarkable performance improvement compared to current state-of-the-art (SOTA) SSMIS methods. As shown in Figure 6.1, DPMS consistently outperform other methods by a large margin, **e.g.** obtaining a remarkable 22.62% Dice improvement compared to previous SOTA SS-Net on ACDC with 3 labels.

6.2 Method

In this section, we first introduce the formulation of SSMIS in Sec. 6.2.1 and then provide a comprehensive revisiting on core elements of the data, the model and the loss supervision in SSMIS in Sec. 6.2.2. Finally, Sec. 6.2.3 describe our proposed DPMS in detail.

6.2.1 Problem Formulation

In a common SSMIS task, labeled data $\mathcal{X}$ and unlabeled data $\mathcal{U}$ are provided, with typically $|X| \ll |U|$. In terms of the training process, let $\mathcal{B}_x = \{(x_i, y_i)\}_{i=1}^{B}$ be a batch of labeled samples and $\mathcal{B}_u = \{u_i\}_{i=1}^{\mu B}$ be a batch of unlabeled samples, where μ denotes the size ratio of $|\mathcal{B}_u|$ to $|\mathcal{B}_x|$. Then the goal of SSMIS is to train a deep segmentation model on both labeled an unlabeled data.

In a common teacher-student framework, the student model, parameterized by θ_s, is first trained on the labeled data via a standard supervised loss $\mathcal{L}_x$,

$$\mathcal{L}_x = \frac{1}{|\mathcal{B}_x|} \sum_{i=1}^{B} \frac{1}{H \times W} \sum_{j=1}^{H \times W} \ell(\hat{y}_i(j), y_i(j)), \tag{6.1}$$

where $\hat{y}_i$ denotes the student model's prediction output on the weakly augmented input x_i, *i.e.*, $\hat{y}_i = f(a(x_i); \theta_s)$, and j represents the j-th pixel on the image or the corresponding segmentation mask with a resolution of $H \times W$. ℓ represents the loss function used to supervise the training, which can be dice loss, cross-entropy loss, or a compound loss of both. Weak augmentations, denoted by $a(\cdot)$, include random geometrical transformations, like random cropping and flipping operations. The teacher model, parameterized by θ_t, is typically not trained directly on the labeled or unlabeled data, but updated by the weights and statistics from the student model. We discuss more in Sec 6.2.2.

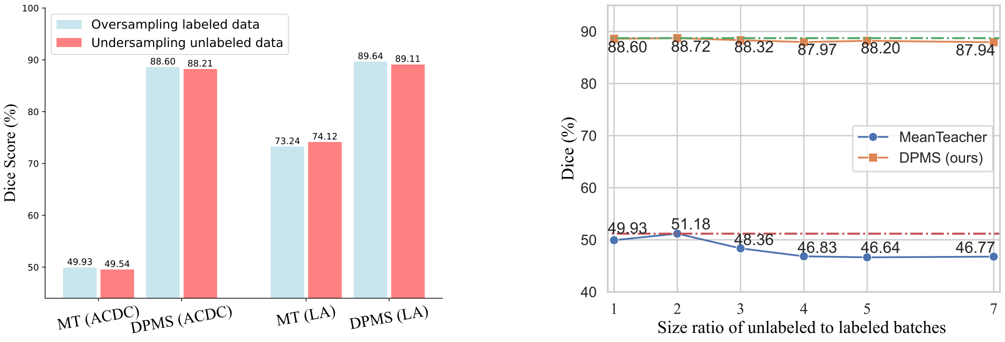
On the other hand, the unsupervised consistency loss on unlabeled data, denoted by $\mathcal{L}_{\sqcap}$, can differ from method to method. Following a standard CR-based method, the unlabeled data can be leveraged via enforcing prediction consistency on differently augmented views of the same input. Let $A(\cdot)$ and $a(\cdot)$ represent two different augmentation strategies, $\mathcal{L}_u$ can be formulated as,

$$\mathcal{L}_u = \frac{1}{|\mathcal{B}_u|} \sum_{i=1}^{\mu B} \frac{1}{H \times W} \sum_{j=1}^{H \times W} \psi(u_i)\, \ell(f(A(u_i); \theta_s), f(a(u_i); \theta_t)) \tag{6.2}$$

where $\psi(u_i)$ represent the selection strategy to filter out unlabeled data with less confident predictions. In summary, the total training loss is,

$$\mathcal{L} = \mathcal{L}_x + \lambda_t \mathcal{L}_u \tag{6.3}$$

where λ_t denote the loss weight to adjust the importance of consistency loss $\mathcal{L}_u$ and can also be a function of the iteration index t, *i.e.*, iteration dependent.



(a) Re-sampling

(b) Size ratios

Figure 6.3: Effect of different re-sampling strategies and size ratios of unlabeled to labeled batches. By default, we apply the “oversampling labeled data" and set the size ratio $\mu = 1$ for fair comparisons.

6.2.2 Revisiting SSMIS

As shown in Figure 6.2, we then revisit the core elements of the data, the model and the loss supervisions in SSMIS studies. Extensive exploration and examination are conducted.

Data: Sampling and Augmentation

Regarding the data problem in SSMIS, we aim to address three primary questions. First, data re-sampling. Since the amount of labeled and unlabeled data differs significantly, *e.g.*, $|X| \ll |U|$, employing sampling strategies is necessary to facilitate training on both labeled and unlabeled sets. Naturally, there are two different ways: 1) oversampling the labeled data and 2) under-sampling the unlabeled data. We examine the two different sampling strategies for 2D ACDC and 3D LA semi-supervised segmentations on the plain Mean-teacher and our proposed DPMS methods. As depicted in Figure 6.3(a), both re-sampling strategies demonstrate comparable and closely aligned segmentation performance across different datasets and SSMIS methods. This is simply because all the labeled and unlabeled data can be sufficiently traversed as long as the total training iterations are significant. Therefore, different re-sampling approaches will not large affect the SSMIS performance, and we oversample the labeled data by default in our study.

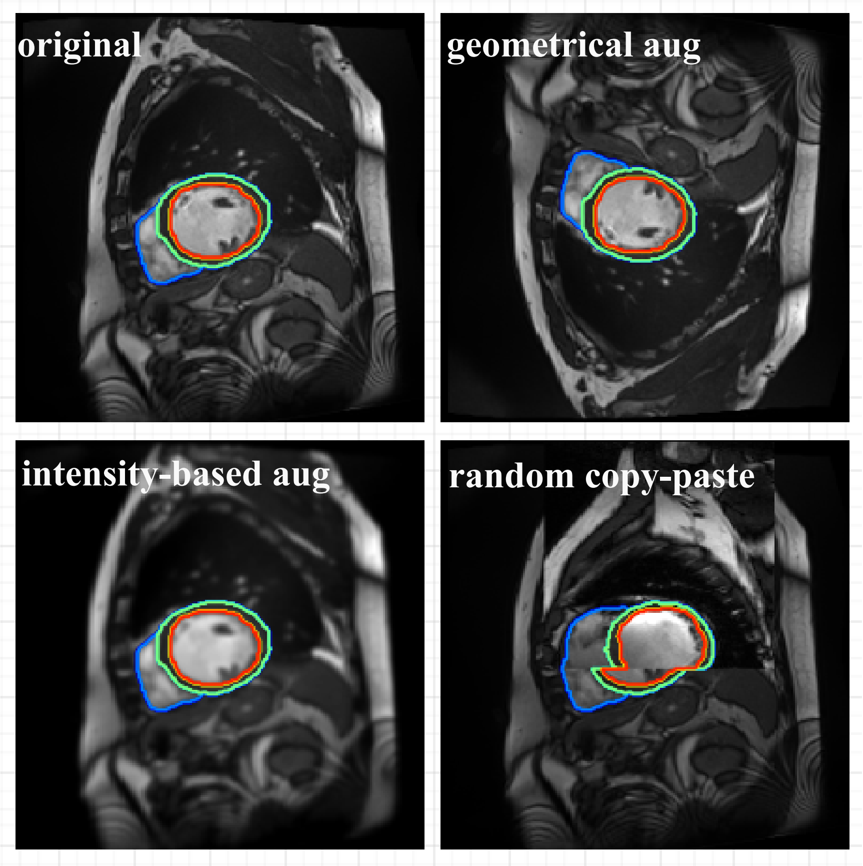


Figure 6.4: Visual examples of different data augmentations on cardiac images.

Second, the size ratio μ. The effect of different μ is extensively discussed in semi-supervised classification [132], but rarely explored in SSMIS. Similar to the loss weight of consistency loss $\mathcal{L}_u$, larger μ will prioritize the importance of unlabeled training. We investigate its effect on ACDC datasets with different SSMIS methods. As shown in Figure 6.3(b), we can clearly see that $\mu = 2$ can achieve the best performance. However, different values of μ have less influences on our proposed DPMS. Thus we select $\mu = 1$ by default for fair comparison with other SSMIS methods.

Third, the critical data augmentations. The key to SSMIS lies in producing appropriate prediction disagreement, and applying data augmentations can be the most straightforward and effective way to generate such label-preserving disagreement. As shown in Figure 6.4, we

Augmentations $A(\cdot)$			Dice(%)	
Geometrical	Intensity	Copy-paste	ACDC (3)	LA (4)
✓			44.87	72.09
✓	✓		64.61	82.07
✓		✓	59.82	74.31
✓	✓	✓	73.33	86.17

Table 6.1: Effect of different data augmentations on SSMIS. All the results are examined without any model stabilization strategies and with loss weight $\lambda_u = 1.0$, thresholds $\tau = 0.95$ and $\tau = 0.8$, loss type “Dice" and “CE" for datasets ACDC and LA, respectively.

investigate three popular kinds of data augmentations, *i.e.*, geometrical transformations (random cropping and flipping), intensity-based augmentations (randomly adjusting brightness and contrast), and Copy-and-paste [44] (widely applied in semi-supervised semantic segmentation [94, 31, 174]). In Table 6.1, we examine the effective of each type of augmentations and their combinations on 2D ACDC and 3D LA datasets. Particularly, since the geometrical transformations will alter the image and corresponding segmentation mask, we first need to apply the same geometrical transformation when examining the intensity-based or copy-and-paste augmentations. As we expected, these strong augmentations can significant boost the SSMIS performance on both 2D and 3D datasets. Especially, the intensity-based augmentation has proven to be the most effective way to perturb the unlabeled instance in SSMIS.

Model: Updating and Stabilization

When considering the model design in SSMIS, two main questions arise: 1) how the pseudo-labels are generated, and 2) how the model are stabilized to prevent collapsing caused by strong data augmentations. In the literature, there are two basic ways to produce pseudo-labels. First, utilize the ensemble model, *i.e.*, the teacher model, to generate pseudo-labels for unlabeled instances, like the widely applied Mean-teacher framework [134]. It is worth noting that the consistency-based methods are essentially the same as pseudo-labeling, where the predictions from one

Model			Augs	Dice(%)	
Ema-teacher	Ema-BN	Extra-Weak		ACDC (3)	LA (4)
				45.80	69.59
✓				44.87	72.09
✓	✓			46.62	76.05
✓		✓		48.75	79.53
✓	✓	✓		49.24	79.90
			✓	80.03	84.46
✓			✓	73.33	86.17
✓	✓		✓	84.09	87.83
✓		✓	✓	87.11	88.02
✓	✓	✓	✓	88.44	89.33

Table 6.2: Effect of different model stabilization strategies on SSMIS. The “Augs" denotes all kinds of augmentations in Table 6.1 are applied. Loss weights and thresholds are the same as in Table 6.1.

view serve as the pseudo-labels for another view. Second, utilize the student model itself to generate pseudo-labels, like the FixMatch [132]. As we can see from Table 6.2, compared to using the student model, adopting the teacher model to generate pseudo-labels performs better on 3D LA dataset but worse on 2D ACDC dataset when applying strong data augmentation discussed in Sec. 6.2.2. As a result, it remains inconclusive as to which strategy is ultimately superior. In our study, we adopt the standard mean-teacher framework as our baseline, and design more stabilization strategies to further improve the performance.

As elucidated in [164, 174], applying strong data augmentations carries the potential risk of over-perturbations, which can hurt the data distribution and consequently degrade the SSMIS performance. Hence, ensuring the stabilization of the model becomes crucial when employing strong data perturbations in the context of semi-supervised learning. Unlike existing studies, we did not revise the augmentation strategies [174], nor did we consider rectifying strategies like distribution-specific BN [21]. Differently, we design two simple yet effective strategies to stabilize the training.

First, in addition to updating the weights of the teacher model, we also updating the batch normalization (BN) statistics via the exponential

Loss			Dice(%)
loss type	Threshold	Ramp-up	ACDC (3)
"dice+ce"	✓	✓	88.07
"ce"	✓	✓	87.05
"dice"	✓	✓	**88.44**
"dice"			86.53
"dice"	✓		88.22
"dice"		✓	87.25

Table 6.3: Effect of different consistency loss supervisions on 2d ACDC dataset with 3 labels.

moving average (EMA) of BN of the student model.

$$\theta_t \leftarrow \alpha\theta_t + (1-\alpha)\theta_s, \tag{6.4}$$

$$\nu_t \leftarrow \alpha\nu_t + (1-\alpha)\nu_s, \tag{6.5}$$

where ν_t and ν_s represent the BN statistics of the teacher and the student model, respectively. α is a momentum parameter, set as 0.99 by default. We can see from Table 6.2 that applying EMA-BN can effective improve the SSMIS performance on both 2D ACDC and 3D LA datasets. Improvements become even more pronounced when applying strong data augmentation. For instance, using EMA-BN yields an improvement of 10.76% compared to the augmentation baseline, which further emphasizes the importance of model stabilization.

Second, to further stabilize the BN statistics, we forward the weakly augmented inputs to the student model, not only the teacher model, dubbed as “Extra-weak". Despite its embarrassing simplicity, we can see from Table 6.2 that “Extra-weak" can effectively boost the SSMIS performance, especially when the strong data augmentations are applied. Indeed, applying all these stabilization strategies can successfully improve the augmentation baseline by a large margin on both 2D and 3D SSMIS tasks.

	Loss		Dice(%)
loss type	Threshold	Ramp-up	LA (4)
"dice+ce"	✓	✓	89.02
"ce"	✓	✓	**89.33**
"dice"	✓	✓	88.82
"ce"			83.68
"ce"	✓		87.78
"ce"		✓	88.30

Table 6.4: Effect of different consistency loss supervisions on 3d LA dataset with 4 labels.

Loss Supervision: Loss Type and Filtering

In terms of the loss supervision in SSMIS, we explore following three questions: 1) the appropriate loss type, 2) pseudo-label selections, and 3) the ramp-up policy of unlabeled consistency loss. In the literature, there are three widely adopted losses, the Dice loss, the cross-entropy (CE) loss, and the compound loss of both. As we can see from Tables 6.3 and 6.4, different loss types can achieve close and comparable performance. In specific, the "dice loss" has demonstrated superior performance on 2D ACDC, while the "ce loss" has shown optimal results for 3D LA. Considering that the ACDC dataset comprises four distinct classes while the LA dataset contains only two classes, the Dice loss emerges as the more appropriate option for ACDC due to its capability to alleviate class imbalance issues.

Employing pseudo-label selection process has been widely studied in SSMIS. It is regarded as an effective and necessary procedure to address the confirmation bias or accumulated errors in semi-supervised learning [4]. Following the FixMatch [132], we simply adopt a pre-defined threshold, denoted as τ, to filter out the unlabeled data with less confident pseudo-labels. As we can see from Tables 6.3 and 6.4, a high-confidence threshold can effectively improve the SSMIS performance, *e.g.*, yielding 4.1% and 1.69% Dice improvements on LA and ACDC, respectively. More detailed ablations studies on the threshold τ is provided in Sec. 6.3.4.

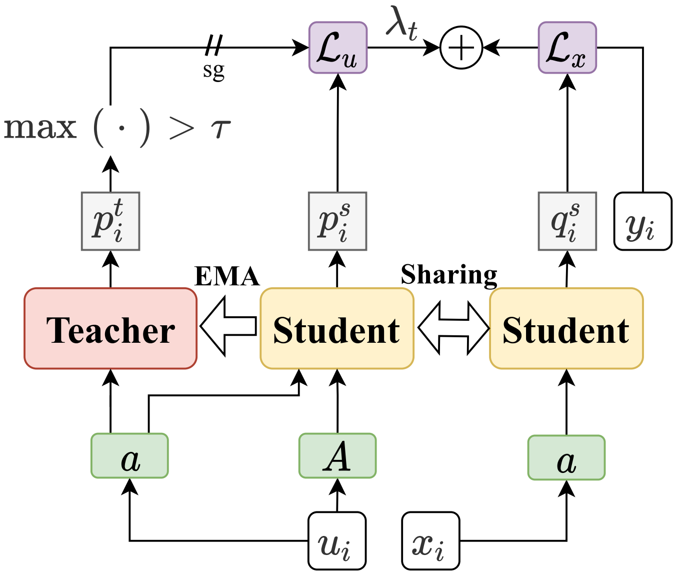


Figure 6.5: DPMS employs weak and strong augmentations (denoted by a and A, respectively) to perturb unlabeled inputs u_i. In a standard teacher-student framework, the student model is trained on the provided labeled data (x_i, y_i) via a standard supervised loss $\mathcal{L}_x$, as well as on the unlabeled data u_i via a consistency loss $\mathcal{L}_u$ supervised by pseudo-labels generated from the teacher model. The weights and buffer statistics of the teacher model are updated using the exponential moving average of the corresponding values from the student model. Pseudo-labels p_i^t are further filtered by a high-confidence threshold τ. A ramp-up term λ_t is adopted to leverage the unlabeled data gradually.

Following Mean-Teacher [134] and UA-MT [163], we further investigate the effect of the ramp-up policy of the unsupervised loss in SSMIS. As discussed in PI-model [79], we adopt a simple iteration-dependent loss weight ramp-up function λ_t to release the impact of consistency loss gradually. Specifically, λ_t is starting from zero and ramping up along a Gaussian curve during the first 150 epochs to the ultimate value of λ_u. The ablation study of the hyper-parameter λ_u is provided in Sec. 6.3.4. As shown Table 6.4, applying ramp-up strategy can bring a remarkable performance improvement of 4.62% on LA dataset with 4 labels.

6.2.3 Our Method: DPMS

Based on the above comprehensive revisiting, we can clearly observe the significance of the data perturbation and model stabilization in SSMIS. To this end, instead of integrating other complicated designs like contrastive losses [120], additional transformer branches [98], we simply follow a plain mean-teacher framework, and propose our method DPMS by integrating our explored effective perturbing and stabilizing strategies. As shown in Figure 6.5, we first employ the weak and strong data augmentations, $a(\cdot)$ and $A(\cdot)$ on data inputs, and feed the augmented data into the student and teacher model to obtain predictions accordingly,

$$p_i^s = f(A(u_i);\theta_s), \tag{6.6}$$

$$q_i^s = f(a(x_i);\theta_s), \tag{6.7}$$

$$p_i^t = f(a(u_i));\theta_t). \tag{6.8}$$

Then the student model can be trained on both labeled and unlabeled data by the total loss,

$$\mathcal{L} = \frac{1}{B}\sum_{i=1}^{B}\frac{1}{H\times W}\sum_{j=1}^{H\times W}\ell(q_i^s(j), y_i(j)) + \lambda_t\,\mathbb{1}(\max(p_i^t(j))\geq\tau)\ell(p_i^s(j), p_i^t(j)), \tag{6.9}$$

where $\mathbb{1}(\max(p_i^t(j)) \geq \tau)$ represent to retain the pseudo-labels whose maximum probability is higher than the pre-defined high-confidence threshold τ. On the other hand, the teacher model is trained by using Equations 6.4 and 6.5 to update its weights and BN statistics, respectively.

6.3 Experiment

6.3.1 Datasets

We examine the effectiveness of our proposed DPMS on two public SSMIS benchmarks, *i.e.*, the Automated Cardiac Diagnosis Challenge (ACDC) and Left Atrium (LA) datasets. The ACDC dataset is a 2D benchmark medical dataset focusing on cardiac image analysis, targeting the assessment of cardiac function. It contains 100 Magnetic Resonance Imaging (MRI) scans from 100 patients, which can be divided into a training set containing 70 MRI scans, a validation set containing 10 MRI scans and a testing set containing 20 MRI scans. Following SS-Net [148] and CT-CT [98], we resize all the slices of 2D ACDC dataset into 256×256 pixels and normalized the intensity into [0, 1].

The LA dataset is a 3D benchmark medical dataset constructed from the Atrial Segmentation Challenge dataset [1], which consists of a collection of 100 fully annotated 3D gadolinium-enhanced MRI scans. In the pre-processing stage, all scans of the 3D LA dataset were center-cropped on the heart region for fair comparisons among different methods and normalized to zero mean and unit variance. During the train stage, the scans are randomly cropped into 112x112x80 and randomly flipped. Following UA-MT [163], it is divided into a training set containing 80 MRI scans and a validation set containing 20 MRI scans.

6.3.2 Implementation Details

Follow the previous works [163], we utilize the UNet [127] and VNet [104] as our backbones on the ACDC and LA datasets, respectively. We use an SGD optimizer with a momentum of 0.9 and a polynomial learning-rate decay with an initial value of 0.01 to train the student model. We train the segmentation model on the ACDC dataset with a batch size of 24 (12 labeled and 12 unlabeled instances) for 30,000 iterations. On LA, following existing studies, we adopt a batch size of 4 (2 labeled and 2 unlabeled instances) for training 15, 000 iterations. By default, we oversampling labeled data and set the size ratio $\mu = 1$, the momentum parameter $\alpha = 0.99$, the maximum loss weight $\lambda_u = 2.0$ for all runs.

6.3.3 Comparison with SOTAs

We compare our DPMS method with recent SSMIS methods, including UA-MT [163], SASSNet [89], DTC [97], URPC [99], MC-Net [147], SS-Net

[1] http://atriaseg2018.cardiacatlas.org/

Method	# Scans used		Metrics				Complexity	
	Labeled	Unlabeled	Dice(%)↑	Jaccard(%)↑	95HD(voxel)↓	ASD(voxel)↓	Para.(M)	MACs(G)
U-Net	3 (5%)	0	47.83	37.01	31.16	12.62	1.81	2.99
U-Net	7 (10%)	0	79.41	68.11	9.35	2.70	1.81	2.99
U-Net	14 (20%)	0	85.15	75.48	6.20	2.12	1.81	2.99
U-Net	70 (All)	0	91.44	84.59	4.30	0.99	1.81	2.99
UA-MT [163]	3 (5%)	67 (95%)	46.04	35.97	20.08	7.75	1.81	2.99
SASSNet [89]			57.77	46.14	20.05	6.06	1.81	3.02
DTC [97]			56.90	45.67	23.36	7.39	1.81	3.02
URPC [99]			55.87	44.64	13.60	3.74	1.83	3.02
MC-Net [147]			62.85	52.29	7.62	2.33	2.58	5.39
SS-Net [148]			65.82	55.38	6.67	2.28	1.83	2.99
CT-CT [98]			65.50	-	16.2	-	28.93	2.99
DPMS (Ours)			**88.44** ±0.15	**80.00** ±0.20	**2.03** ±0.56	**0.59** ±0.09	1.81	2.99
UA-MT [163]	7 (10%)	63 (90%)	81.65	70.64	6.88	2.02	1.81	2.99
SASSNet [89]			84.50	74.34	5.42	1.86	1.81	3.02
DTC [97]			84.29	73.92	12.81	4.01	1.81	3.02
URPC [99]			83.10	72.41	4.84	1.53	1.83	3.02
MC-Net [147]			86.44	77.04	5.50	1.84	2.58	5.39
SS-Net [148]			86.78	77.67	6.07	1.40	1.83	2.99
CT-CT [98]			86.40	-	8.60	-	28.93	2.99
DPMS (Ours)			**89.82** ±0.34	**82.06** ±0.51	**1.72** ±0.52	**0.52** ±0.06	1.81	2.99
UA-MT [163]	14 (20%)	56 (80%)	85.87	76.78	5.06	1.54	1.81	2.99
SASSNet [89]			87.04	78.13	7.84	2.15	1.81	3.02
DTC [97]			86.28	77.03	6.14	2.11	1.81	3.02
URPC [99]			85.07	75.61	6.26	1.77	1.83	3.02
MC-Net [147]			87.83	79.14	4.94	1.52	2.58	5.39
DPMS (Ours)			**91.06** ±0.14	**84.03** ±0.23	**1.27** ±0.11	**0.36** ±0.03	1.81	2.99

Table 6.5: Comparisons with recent SSMIS methods on the ACDC dataset with 3 (5%), 7 (10%), 14 (20%) labels in terms of Dice, Jaccard, 95HD, ASD and model complexities. Results of our proposed DPMS are average over 3 runs, where the mean and standard deviation are reported.

[148] and CT-CT [98]. We follow the same data partition protocols as UA-MT [163] and SS-Net [148] to carry on our experiments and report the mean performance averaged over three runs together with the standard error of the mean (SEM). Following previous works [163, 148], we adopt Dice Score (%), Jaccard Score (%), 95% Hausdorff Distance (95HD) in voxel and Average Surface Distance (ASD) in voxel to evaluate the performance of different methods.

We first investigate the effectiveness of our DPMS on the **2D ACDC dataset** in Table 6.5. We observe that our DPMS method achieves new state-of-the-art performance under all protocols without introducing any additional parameters or multiply-accumulate operations (MACs) complexity, surpassing the baseline method by a large margin. It should also be noted that when there are only 5%, 10% or 20% labeled data available

Method	# Scans used		Metrics				Complexity	
	Labeled	Unlabeled	Dice(%)↑	Jaccard(%)↑	95HD(voxel)↓	ASD(voxel)↓	Para.(M)	MACs(G)
V-Net	4(5%)	0	52.55	39.60	47.05	9.87	9.44	47.02
V-Net	8(10%)	0	82.74	71.72	13.35	3.26	9.44	47.02
V-Net	16(20%)	0	86.96	77.31	11.85	3.22	9.44	47.02
V-Net	80(All)	0	91.47	84.36	5.48	1.51	9.44	47.02
UA-MT [163]	4 (5%)	76 (95%)	82.26	70.98	13.71	3.82	9.44	47.02
SASSNet [89]			81.60	69.63	16.16	3.58	9.44	47.05
DTC [97]			81.25	69.33	14.90	3.99	9.44	47.05
URPC [99]			82.48	71.35	14.65	3.65	5.88	69.43
MC-Net [147]			83.59	72.36	14.07	2.70	12.35	95.15
SS-Net [148]			86.33	76.15	9.97	2.31	9.46	47.17
DPMS (Ours)			**89.64** ±0.22	**81.29** ±0.35	**5.99** ±0.40	**1.77** ±0.05	9.44	47.02
UA-MT [163]	8 (10%)	72 (90%)	86.28	76.11	18.71	4.63	9.44	47.02
SASSNet [89]			85.22	75.09	11.18	2.89	9.44	47.05
DTC [97]			87.51	78.17	8.23	2.36	9.44	47.05
URPC [99]			85.01	74.36	15.37	3.96	5.88	69.43
MC-Net [147]			87.50	77.98	11.28	2.30	12.35	95.15
SS-Net [148]			88.55	79.62	7.49	1.90	9.46	47.17
DPMS (Ours)			**90.49** ±0.21	**82.69** ±0.35	**6.39** ±0.20	**1.53** ±0.05	9.44	47.02
UA-MT [163]	16 (20%)	64 (80%)	88.74	79.94	8.39	2.32	9.44	47.02
SASSNet [89]			89.16	80.60	8.95	2.26	9.44	47.05
DTC [97]			89.52	81.22	7.07	1.96	9.44	47.05
URPC [99]			88.74	79.93	12.73	3.66	5.88	69.43
MC-Net [147]			90.12	82.12	8.07	1.99	12.35	95.15
DPMS (Ours)			**91.64** ±0.26	**84.62** ±0.43	**5.21** ±0.28	**1.44** ±0.07	9.44	47.02

Table 6.6: Comparisons with recent SSMIS methods on the 3D LA dataset with 4 (5%), 8 (10%), 16 (20%) labels in terms of Dice, Jaccard, 95HD, ASD and model complexities.

with the remaining data unlabeled, our DPMS method exceeds the baseline method by over 40%, 20% or 3% in terms of the Dice Score, respectively. Especially when there are only 3 labeled data available, our DPMS method can surpass the previous SOTA SS-Net by over 20% Dice score. We can also see that when using 20% labeled data, our method achieves comparable performance with the fully supervised upper bound (using all labeled data). Surprisingly, DPMS can even achieve better 95HD and ASD results than the fully supervised method, indicating the great potential of our proposed method.

We further compare our method with recent SSMIS methods on the **3D LA dataset** and the results are reported in Table 6.6. It can be clearly seen from the table that our DPMS can consistently outperform other methods under all partition protocols. When there are 5%, 10% and 20% labeled data available, our method can surpass the previous SOTA method by around 3%, 2% and 1% in terms of the Dice Score, respectively. It should be specially noted that when there are 20% labeled data available with the rest data unlabeled, our DPMS method can even outperform the fully supervised method in terms of all evaluation metrics,

DPMS		**RV**		**Myo**		**LV**		**Mean**	
Perturbation	Stablization	Dice(%)	95HD	Dice(%)	95HD	Dice(%)	95HD	Dice(%)↑	95HD↓
		28.54	34.20	49.51	19.97	58.59	28.87	45.55	27.68
✓		55.69	14.09	44.96	18.96	55.51	16.74	52.05	16.60
	✓	64.34	11.59	68.92	9.82	82.41	12.43	71.89	11.28
✓	✓	86.25	1.80	87.01	1.11	92.55	1.72	88.60	1.54

Table 6.7: Ablations on data perturbation and model stabilization when using 3 cases as labeled data on ACDC. RV, Myo, LV represent the right ventricle, myocardium and left ventricle, respectively.

λ_u	0.5	1.0	1.5	2.0	3.0
ACDC (3)	87.99	88.44	88.28	**88.60**	88.26
LA (4)	88.27	89.33	**89.72**	89.64	89.35

Table 6.8: Ablations on the loss weight λ_u, set as 2.0 by default.

which further indicates the robustness of our method against the potentially incorrect annotations. Thus our method can possibly be a potential candidate to address noisy segmentation problems.

6.3.4 Ablation Study

In this section, we analyze the isolated effects of the data perturbation and model stabilization in DPMS, as well as the sensitiveness of the hyper-parameters used in our method, *i.e.*, the maximum loss weight of the unlabeled data λ_u and the pre-defined threshold to select high-confident samples τ.

We first verify the isolated contributions of the data perturbation and the model stabilization in Table 6.7. Here we report the category-wise performance and the mean performance on the ACDC dataset using 3 labeled data (5% labels) with the remaining data unlabeled, and we evaluate the performance in terms of the Dice Score and 95HD. It can be inferred from the table that either the data perturbation or the model stabilization can contribute to the ultimate segmentation performance, where the data perturbation can mainly contribute to the recognition of the hard-to-distinguish classes like the right ventricle (RV), while it shows only a limited contribution to the model on recognizing those

τ	0.7	0.75	0.8	0.85	0.9	0.95
ACDC (3)	88.21	87.78	87.97	88.07	88.42	**88.44**
LA (4)	88.95	89.27	**89.33**	89.14	89.03	88.87

Table 6.9: Ablations on threshold τ, set as 0.95 and 0.8 for dataset ACDC and LA, respectively.

easy-to-distinguish classes like the myocardium (Myo) or the left ventricle (LV). The main reason is that data augmentation can provide effective disagreements on unlabeled data, thus contributing to the robustness of the model, while the too-strong data perturbation may potentially destroy the original data distribution, leading to performance degradation. On the contrary, model stabilization can greatly contribute to the recognition of the model on any class. The main reason is that a stable teacher model, especially the stable BN statistics, can generate more accurate predictions (*i.e.*, pseudo labels) for unlabeled data to train the student model. In this way, the student model could learn useful semantics from the supervision, thus contributing to the model learning. Equipping the data perturbation with the model stabilization can further contribute to the recognition performance. The main reason is that the stable teacher model can provide accurate supervision for the student model, while the data perturbation provides the prediction disagreement between the student model and the teacher model. Learning from stable predictions generated by the teacher model to mitigate the prediction disagreements will greatly enable the student model to learn useful semantics, and improve the robustness of the student model, which can further contribute to the stability of the teacher model. Therefore, the recognition performance of the model when combining the data perturbation with the model stabilization can greatly improve the SSMIS performance.

We further examine the sensitivity of the hyper-parameters used in our DPMS method in Table 6.8 and Table6.9. Here we conduct the experiments on both the ACDC and the LA datasets, with only 5% labeled data available. It can be seen from the tables that our DPMS method is robust to different values of the hyper-parameters, where the variation of the performance using different hyper-parameters is less than 1%

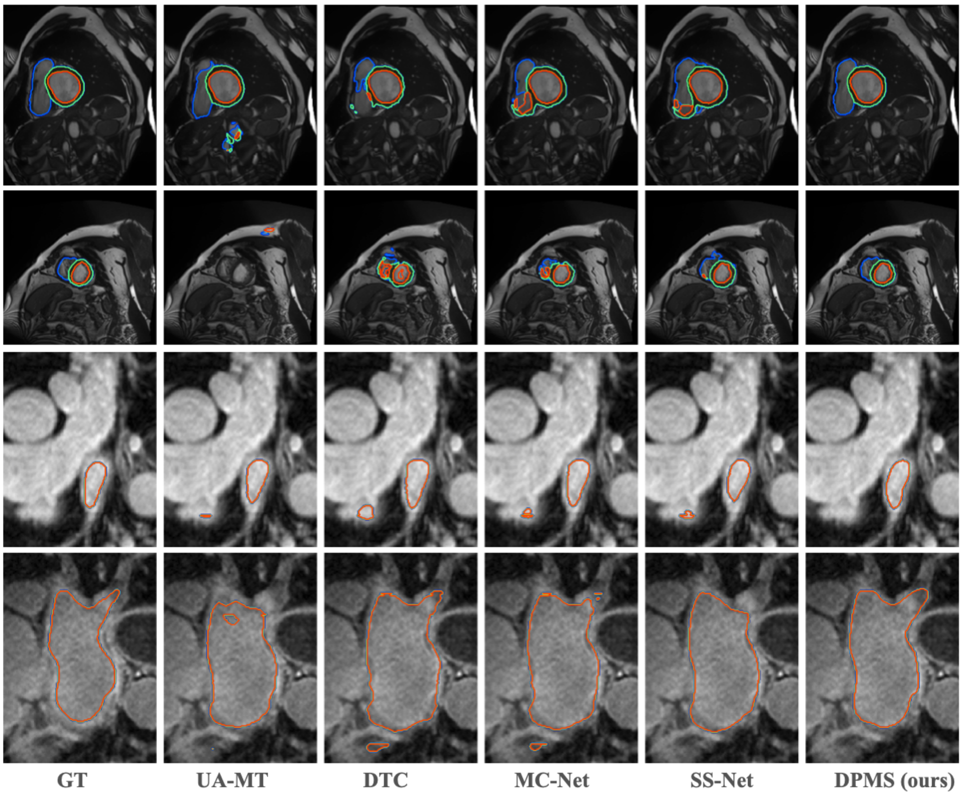


Figure 6.6: Qualitative results on ACDC (top 2 rows) and LA (bottom 2 rows) using only 5% labeled data. Columns from left to right denote the segmentation results of the ground-truth, UA-MT [163], DTC [97], MC-Net [147], SS-Net [148], and our proposed DPMS, respectively.

Dice, indicating the robustness of our method. Especially, it can be inferred from Table 6.8 that a higher weight λ_u for the unlabeled data can improve the performance of the model, suggesting that the model will learn more semantics from unlabeled data. In this work, we adopt the λ_u as 2.0 on both the ACDC and LA datasets, and we set the threshold τ as 0.95 and 0.8 for the ACDC and LA, respectively, to obtain the best performance.

6.3.5 Qualitative Visualization

Figure 6.6 shows some representative qualitative results on ACDC and LA dataset with 5% labeled data. The UA-MT obtains the worse segmentation results, *e.g.*, not capable of segmenting the myocardium and left ventricle in the second row. Though DTC and MC-Net can obtain better results on ACDC than UA-MT, both methods mis-classify the foreground in the third row on the LA dataset. In contrast, we can observe that many mis-classified regions and ignored segmentation details in the segmentation results of other SSMIS methods can be successfully corrected and segmented by our proposed DPMS, which further demonstrates the effectiveness of DPMS.

6.4 Summary

In this chapter, we challenge the prevailing trend observed in recent SSMIS studies, where the focus has shifted towards increasingly complex designs. Instead, we propose a simple yet highly effective method that emphasizes the significance of data perturbation and model stabilization to boost SSMIS performance. Specifically, we undertake a thorough examination of the essential components of the data, model, and loss supervision in SSMIS. Through in-depth analysis, we find that perturbing and stabilizing strategies play a critical role in achieving promising segmentation performance. Our main contributions are summarized as follows,

- We revisit the key elements of data, model, and loss supervision in semi-supervised medical image segmentation. Through in-depth analysis, we conduct comprehensive studies on various strategies associated with each element.
- We break the trend of recent SSMIS studies that tend to introduce increasingly complicated designs and propose a simple yet effective DPMS that emphasize the significance of data perturbation and model stabilization to boost SSMIS.
- Benefiting from the perturbing and stabilizing designs, our simple DPMS can readily achieve new state-of-the-art performance on public 2D and 3D SSMIS benchmarks, especially effective in label-scarce scenarios.

We hope our DPMS can serve as a strong baseline and inspire more simple yet effective methods in future SSMIS studies.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we first introduce the problem of semi-supervised learning (SSL) and provide a comprehensive literature review about deep visual learning and various label-efficient SSL methods. We also discuss that the main challenges in SSL lie in the effective and comprehensive utilization of unlabeled data. Afterwards, we have proposed four new SSL methods for different downstream tasks.

In Chapter 3, we focus on the conventional semi-supervised classification (SSC) and propose a novel SSL approach that can effectively utilize the label information to integrate a class-aware contrastive loss (CACL) and buffer-aided label propagation algorithm (BLPA) into a self-training paradigm. CACL and BLPA are seamlessly integrated and mutually promoted across the whole training process. In Chapter 4, we further discuss the more practical setting in SSC, *i.e.*, the distribution mismatch between the labeled and unlabeled sets. We first revisit the EMA model in SSL and observe that it can be helpful in estimating unlabeled class distributions, although it may not produce more accurate high-confidence pseudo-labels directly. A new method, DC-SSL, is then proposed to enhance SSL performance from a distribution perspective. Proposed methods are evaluated on multiple SSL benchmarks.

In Chapter 5, we highlight the instance uniqueness and argue that differentiating unlabeled instances can promote instance-specific supervision to adapt to the model's evolution dynamically. We first perform a quantitative hardness-evaluating analysis for unlabeled instances in segmentation tasks, based on the class-weighted teacher-student symmetric IoU. We then propose an instance-specific and model-adaptive semi-supervised semantic segmentation (SSS) framework that injects instance hardness into loss evaluation and data perturbation to dynamically adapt to the model's evolution. In Chapter 6, we break the trend of semi-supervised medical image segmentation (SSMIS) studies that integrate increasingly complex designs. We highlight the significance of data perturbation and model stabilization in SSMIS, and propose DPMS, a simple and clean two-branch teacher-student framework that can achieve readily better performance than existing methods.

We have conducted extensive experiments and ablation studies on popular SSC and SSS benchmarks to evaluate the effectiveness of our proposed LaSSL, DC-SSL, iMas and DPMS methods.

7.2 Future work

Considering the recent development of SSL, we discuss the following three potential research directions in future work.

- More flexible pseudo-polishing strategies. As we discussed in Chapters 1, 3 and 4, the accuracy of generated pseudo-labels is of significant importance to the SSL performance. Thus, improving the quality of pseudo-labels becomes the most crucial factor in SSL studies. However, existing studies tend to adopt stringent filtering strategies, like the high-confidence threshold or fixed uncertainty constraint, to select a portion of unlabeled data. Considering the learning difficulties of different classes and the long-tail natures of semantic segmentation, such fixed strategies cannot be an optimal solution. Thus more flexible and advanced strategies can be designed to polish the pseudo-labels further, and better performance should be achieved.
- Equipping semi-supervised segmentation with weak labels. As

we can see from both semi-supervised classification and segmentation tasks, adding more labeled samples is the most effective way to boost the semi-supervised performance, while few labels commonly cannot achieve satisfactory performance. In practice, it may not be easy to acquire sufficient label data, especially for segmentation tasks that require dedicated pixel-level annotations. To address this issue, we can equip semi-supervised segmentation with weak labels, like scribble annotations, bounding boxes, and image-level semantics. These weak labels are commonly easier to obtain than accurate per-pixel labels. The goal is to obtain satisfactory segmentation performance using a few accurately-annotated labeled data and certain amounts of weakly-labeled data (e.g., scribble annotations) and large amounts of unlabeled data.

- Semi-supervised Learning in multi-modal tasks. Though purely visual tasks have attracted much attention, there are few semi-supervised studies on various multi-modal tasks, like visual grounding with natural language and open-vocabulary segmentation and detection. For multi-modal tasks, we need not only visual or language labels but also multi-modal corresponding label information, which commonly requires more labelling efforts. Therefore, it can be interesting and necessary to study label-efficient multi-modal tasks.

In addition, recent big foundation models have achieved very impressive performance on different downstream tasks. If our focus remains solely on enhancing SSL performance on various predefined settings, the importance of SSL studies may significantly diminish. It can possibly become a waste if SSL methods are merely applied as supplementary tuning strategies. In the current era of large models, the question of how to effectively explore further research in semi-supervised learning remains an open challenge.

Bibliography

[1] A. Abuduweili, X. Li, H. Shi, C.-Z. Xu, and D. Dou. "Adaptive Consistency Regularization for Semi-Supervised Transfer Learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 6923–6932.

[2] A. Agrawala. "Learning with a probabilistic teacher". In: *IEEE Transactions on Information Theory* 16.4 (1970), pp. 373–379.

[3] I. Alonso, A. Sabater, D. Ferstl, L. Montesano, and A. C. Murillo. "Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 8219–8228.

[4] E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, and K. McGuinness. "Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning". In: *2020 International Joint Conference on Neural Networks*. IEEE. 2020.

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation". In: *IEEE transactions on pattern analysis and machine intelligence* 39.12 (2017), pp. 2481–2495.

[6] W. Bai, O. Oktay, M. Sinclair, H. Suzuki, M. Rajchl, G. Tarroni, B. Glocker, A. King, P. M. Matthews, and D. Rueckert. "Semi-supervised learning for network-based cardiac MR image segmentation". In: *Medical Image Computing and Computer-Assisted Intervention*. Springer. 2017, pp. 253–260.

[7] Y. Bai, D. Chen, Q. Li, W. Shen, and Y. Wang. "Bidirectional Copy-Paste for Semi-Supervised Medical Image Segmentation". In: *arXiv preprint arXiv:2305.00673* (2023).

[8] M. Belkin, P. Niyogi, and V. Sindhwani. "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples." In: *Journal of machine learning research* 7.11 (2006).

[9] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. "Curriculum learning". In: *Proceedings of the 26th annual international conference on machine learning*. 2009, pp. 41–48.

[10] K. Bennett and A. Demiriz. "Semi-supervised support vector machines". In: *Advances in Neural Information processing systems* 11 (1998).

[11] O. Bernard, A. Lalande, C. Zotti, F. Cervenansky, X. Yang, P.-A. Heng, I. Cetin, K. Lekadir, O. Camara, M. A. G. Ballester, et al. "Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: is the problem solved?" In: *IEEE transactions on medical imaging* 37.11 (2018), pp. 2514–2525.

[12] D. Berthelot, N. Carlini, E. D. Cubuk, A. Kurakin, K. Sohn, H. Zhang, and C. Raffel. "Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring". In: *8th International conference on learning representations*. 2020.

[13] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel. "MixMatch: A Holistic Approach to Semi-Supervised Learning". In: *Advances in neural information processing systems*. 2019.

[14] A. Blum and S. Chawla. "Learning from Labeled and Unlabeled Data using Graph Mincuts". In: *Proceedings of the Eighteenth International Conference on Machine Learning*. 2001, pp. 19–26.

[15] A. Blum and T. Mitchell. "Combining labeled and unlabeled data with co-training". In: *Proceedings of the eleventh annual conference on Computational learning theory*. 1998, pp. 92–100.

[16] A. Blum and T. Mitchell. "Combining labeled and unlabeled data with co-training". In: *Proceedings of the eleventh annual conference on Computational learning theory*. 1998, pp. 92–100.

[17] J. S. Bridle, A. J. Heading, and D. J. MacKay. "Unsupervised Classifiers, Mutual Information and 'Phantom Targets'". In: *Advances in neural information processing systems*. 1992.

[18] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma. "Learning imbalanced datasets with label-distribution-aware margin loss". In: *arXiv preprint arXiv:1906.07413* (2019).

[19] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. "End-to-end object detection with transformers". In: *Proceedings of the european conference on computer vision*. Springer. 2020, pp. 213–229.

[20] H.-S. Chang, E. Learned-Miller, and A. McCallum. "Active bias: Training more accurate neural networks by emphasizing high variance samples". In: *Advances in Neural Information Processing Systems* 30 (2017).

[21] W.-G. Chang, T. You, S. Seo, S. Kwak, and B. Han. "Domain-specific batch normalization for unsupervised domain adaptation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 7354–7362.

[22] O. Chapelle, B. Scholkopf, and A. Zien. "Semi-supervised learning [book reviews]". In: *IEEE Transactions on Neural Networks* 20.3 (2009), pp. 542–542.

[23] K. Chen, C. Liu, H. Chen, H. Zhang, W. Li, Z. Zou, and Z. Shi. "RSPrompter: Learning to prompt for remote sensing instance segmentation based on visual foundation model". In: *arXiv preprint arXiv:2306.16269* (2023).

[24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. 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* (2017).

[25] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. "Semantic image segmentation with deep convolutional nets and fully connected crfs". In: *arXiv preprint arXiv:1412.7062* (2014).

[26] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. "Rethinking atrous convolution for semantic image segmentation". In: *arXiv preprint arXiv:1706.05587* (2017).

[27] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. "Encoder-decoder with atrous separable convolution for semantic image

segmentation". In: *Proceedings of the european conference on computer vision*. 2018.

[28] T. Chen, L. Zhu, C. Ding, R. Cao, S. Zhang, Y. Wang, Z. Li, L. Sun, P. Mao, and Y. Zang. "SAM Fails to Segment Anything?– SAM-Adapter: Adapting SAM in Underperformed Scenes: Camouflage, Shadow, and More". In: *arXiv preprint arXiv:2304.09148* (2023).

[29] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. "A simple framework for contrastive learning of visual representations". In: *International conference on machine learning*. 2020, pp. 1597–1607.

[30] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton. "Big self-supervised models are strong semi-supervised learners". In: *arXiv preprint arXiv:2006.10029* (2020).

[31] X. Chen, Y. Yuan, G. Zeng, and J. Wang. "Semi-Supervised Semantic Segmentation with Cross Pseudo Supervision". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021.

[32] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, et al. "Palm: Scaling language modeling with pathways". In: *arXiv preprint arXiv:2204.02311* (2022).

[33] A. Coates, A. Ng, and H. Lee. "An analysis of single-layer networks in unsupervised feature learning". In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. 2011, pp. 215–223.

[34] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. "The cityscapes dataset for semantic urban scene understanding". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2016.

[35] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le. "Randaugment: Practical automated data augmentation with a reduced search space". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition Workshop*. 2020. DOI: 10.1109/CVPRW50498.2020.00359.

[36] M. Cuturi. “Sinkhorn distances: Lightspeed computation of optimal transport”. In: *Advances in neural information processing systems* 26 (2013), pp. 2292–2300.

[37] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. “Imagenet: A large-scale hierarchical image database”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2009.

[38] J. Donahue, P. Krähenbühl, and T. Darrell. “Adversarial feature learning”. In: *arXiv preprint arXiv:1605.09782* (2016).

[39] W. Dong-DongChen and Z.-H. WeiGao. “Tri-net for semi-supervised deep learning”. In: *International Joint Conferences on Artificial Intelligence*. 2018.

[40] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *arXiv preprint arXiv:2010.11929* (2020).

[41] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. “The pascal visual object classes challenge: A retrospective”. In: *International journal of computer vision* 111 (2015), pp. 98–136.

[42] S. Fralick. “Learning to recognize patterns without a teacher”. In: *IEEE Transactions on Information Theory* 13.1 (1967), pp. 57–64.

[43] G. French, T. Aila, S. Laine, M. Mackiewicz, and G. Finlayson. “Semi-supervised semantic segmentation needs strong, high-dimensional perturbations”. In: *British Machine Vision Conference*. 2020.

[44] G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T.-Y. Lin, E. D. Cubuk, Q. V. Le, and B. Zoph. “Simple copy-paste is a strong data augmentation method for instance segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021.

[45] C. Gong, D. Wang, and Q. Liu. “AlphaMatch: Improving Consistency for Semi-supervised Learning with Alpha-divergence”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 13683–13692.

[46] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*. Vol. 1. MIT Press, 2016.

[47] Y. Grandvalet and Y. Bengio. "Semi-supervised learning by entropy minimization". In: *Advances in Neural Information Processing Systems*. 2005.

[48] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al. "Bootstrap your own latent-a new approach to self-supervised learning". In: *Advances in neural information processing systems* 33 (2020), pp. 21271–21284.

[49] Grounded-SAM Contributors. *Grounded-Segment-Anything*. https://github.com/IDEA-Research/Grounded-Segment-Anything. 2023.

[50] D. Guan, J. Huang, A. Xiao, and S. Lu. "Unbiased Subclass Regularization for Semi-Supervised Semantic Segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[51] G. Gui, Z. Zhao, L. Qi, L. Zhou, L. Wang, and Y. Shi. "Improving Barely Supervised Learning by Discriminating Unlabeled Samples with Super-Class". In: *Advances in Neural Information Processing Systems*. 2022.

[52] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik. "Semantic contours from inverse detectors". In: *2011 international conference on computer vision*. IEEE. 2011, pp. 991–998.

[53] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. "Momentum contrast for unsupervised visual representation learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 9729–9738.

[54] K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2016.

[55] R. He, J. Yang, and X. Qi. "Re-distributing Biased Pseudo Labels for Semi-supervised Semantic Segmentation: A Baseline Investigation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021.

[56] H. Hu, F. Wei, H. Hu, Q. Ye, J. Cui, and L. Wang. “Semi-supervised semantic segmentation via adaptive equalization learning”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 22106–22118.

[57] L. Hu, J. Li, X. Peng, J. Xiao, B. Zhan, C. Zu, X. Wu, J. Zhou, and Y. Wang. “Semi-supervised NPC segmentation with uncertainty and attention guided consistency”. In: *Knowledge-Based Systems* 239 (2022), p. 108021.

[58] Z. Hu, Z. Yang, X. Hu, and R. Nevatia. “SimPLE: Similar Pseudo Label Exploitation for Semi-Supervised Classification”. In: *arXiv preprint arXiv:2103.16725* (2021).

[59] S. Huang, Z. Jiang, H. Dong, Y. Qiao, P. Gao, and H. Li. “Instruct2Act: Mapping Multi-modality Instructions to Robotic Actions with Large Language Model”. In: *arXiv preprint arXiv:2305.11176* (2023).

[60] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang. “Adversarial learning for semi-supervised semantic segmentation”. In: *British Machine Vision Conference*. 2018.

[61] M. S. Ibrahim, A. Vahdat, M. Ranjbar, and W. G. Macready. “Semi-supervised semantic image segmentation with self-correcting networks”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020.

[62] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum. “Label propagation for deep semi-supervised learning”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 5070–5079.

[63] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon. “A survey on contrastive self-supervised learning”. In: *Technologies* 9.1 (2021), p. 2.

[64] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio. “The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2017, pp. 11–19.

[65] W. Ji, J. Li, Q. Bi, W. Li, and L. Cheng. “Segment anything is not always perfect: An investigation of sam on different real-world applications”. In: *arXiv preprint arXiv:2304.05750* (2023).

[66] Y. Ji, H. Bai, C. Ge, J. Yang, Y. Zhu, R. Zhang, Z. Li, L. Zhanng, W. Ma, X. Wan, et al. “Amos: A large-scale abdominal multi-organ benchmark for versatile medical image segmentation”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 36722–36732.

[67] S. Jin, A. RoyChowdhury, H. Jiang, A. Singh, A. Prasad, D. Chakraborty, and E. Learned-Miller. “Unsupervised hard example mining from videos for improved object detection”. In: *Proceedings of the European Conference on Computer Vision*. 2018, pp. 307–324.

[68] T. Joachims et al. “Transductive inference for text classification using support vector machines”. In: *International conference on machine learning*. Vol. 99. 1999, pp. 200–209.

[69] Z. Ke, K. L. Di Qiu, Q. Yan, and R. W. Lau. “Guided collaborative training for pixel-wise semi-supervised learning”. In: *Proceedings of the european conference on computer vision*. 2020.

[70] B. Kim, J. Choo, Y.-D. Kwon, S. Joe, S. Min, and Y. Gwon. “Self-Match: Combining Contrastive Self-Supervision and Consistency for Semi-Supervised Learning”. In: *arXiv preprint arXiv:2101.06480* (2021).

[71] D. Kim, D. Cho, D. Yoo, and I. S. Kweon. “Learning image representations by completing damaged jigsaw puzzles”. In: *IEEE Winter Conference on Applications of Computer Vision*. IEEE. 2018, pp. 793–802.

[72] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al. “Segment anything”. In: *arXiv preprint arXiv:2304.02643* (2023).

[73] A. Krizhevsky and G. Hinton. “Learning multiple layers of features from tiny images”. In: *Handbook of Systemic Autoimmune Diseases* 1.4 (2009).

[74] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Communications of the ACM* 60.6 (2017), pp. 84–90.

[75] A. Kumar, P. Sattigeri, and T. Fletcher. "Semi-supervised learning with gans: Manifold invariance with improved inference". In: *Advances in Neural Information Processing Systems* 30 (2017).

[76] C.-W. Kuo, C.-Y. Ma, J.-B. Huang, and Z. Kira. "Featmatch: Feature-based augmentation for semi-supervised learning". In: *European Conference on Computer Vision*. Springer. 2020, pp. 479–495.

[77] D. Kwon and S. Kwak. "Semi-supervised Semantic Segmentation with Error Localization Network". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[78] X. Lai, Z. Tian, L. Jiang, S. Liu, H. Zhao, L. Wang, and J. Jia. "Semi-supervised Semantic Segmentation with Directional Context-aware Consistency". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021.

[79] S. Laine and T. Aila. "Temporal ensembling for semi-supervised learning". In: *International conference on learning representations*. 2017.

[80] G. Larsson, M. Maire, and G. Shakhnarovich. "Colorization as a proxy task for visual understanding". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 6874–6883.

[81] G. Larsson, M. Maire, and G. Shakhnarovich. "Learning representations for automatic colorization". In: *Proceedings of the european conference on computer vision*. Springer. 2016, pp. 577–593.

[82] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[83] D.-H. Lee et al. "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks". In: *Workshop on challenges in representation learning, the international conference on machine learning*. 2013.

[84] B. Li, Y. Zhang, L. Chen, J. Wang, J. Yang, and Z. Liu. "Otter: A multi-modal model with in-context instruction tuning". In: *arXiv preprint arXiv:2305.03726* (2023).

[85] F. Li, H. Zhang, P. Sun, X. Zou, S. Liu, J. Yang, C. Li, L. Zhang, and J. Gao. "Semantic-sam: Segment and recognize anything at any granularity". In: *arXiv preprint arXiv:2307.04767* (2023).

[86] J. Li, D. Li, S. Savarese, and S. Hoi. "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models". In: *arXiv preprint arXiv:2301.12597* (2023).

[87] J. Li, C. Xiong, and S. Hoi. "CoMatch: Semi-supervised Learning with Contrastive Graph Regularization". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021.

[88] K. Li, Y. He, Y. Wang, Y. Li, W. Wang, P. Luo, Y. Wang, L. Wang, and Y. Qiao. "Videochat: Chat-centric video understanding". In: *arXiv preprint arXiv:2305.06355* (2023).

[89] S. Li, C. Zhang, and X. He. "Shape-aware semi-supervised 3D semantic segmentation for medical images". In: *Medical Image Computing and Computer Assisted Intervention*. Springer. 2020.

[90] A. Lin, B. Chen, J. Xu, Z. Zhang, G. Lu, and D. Zhang. "Ds-transunet: Dual swin transformer u-net for medical image segmentation". In: *IEEE Transactions on Instrumentation and Measurement* 71 (2022), pp. 1–15.

[91] H. Liu, C. Li, Q. Wu, and Y. J. Lee. "Visual instruction tuning". In: *arXiv preprint arXiv:2304.08485* (2023).

[92] S. Liu, S. Zhi, E. Johns, and A. J. Davison. "Bootstrapping semantic segmentation with regional contrast". In: *International conference on learning representations*. 2022.

[93] X. Liu, F. Zhang, Z. Hou, Z. Wang, L. Mian, J. Zhang, and J. Tang. "Self-supervised learning: Generative or contrastive". In: *arXiv preprint arXiv:2006.08218* 1.2 (2020).

[94] Y. Liu, Y. Tian, Y. Chen, F. Liu, V. Belagiannis, and G. Carneiro. "Perturbed and strict mean teachers for semi-supervised semantic segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[95] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. "Swin transformer: Hierarchical vision transformer using shifted windows". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 10012–10022.

[96] J. Long, E. Shelhamer, and T. Darrell. “Fully convolutional networks for semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.

[97] X. Luo, J. Chen, T. Song, and G. Wang. “Semi-supervised medical image segmentation through dual-task consistency”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 10. 2021, pp. 8801–8809.

[98] X. Luo, M. Hu, T. Song, G. Wang, and S. Zhang. “Semi-supervised medical image segmentation via cross teaching between cnn and transformer”. In: *International Conference on Medical Imaging with Deep Learning*. 2022, pp. 820–833.

[99] X. Luo, W. Liao, J. Chen, T. Song, Y. Chen, S. Zhang, N. Chen, G. Wang, and S. Zhang. “Efficient semi-supervised gross target volume of nasopharyngeal carcinoma segmentation via uncertainty rectified pyramid consistency”. In: *Medical Image Computing and Computer Assisted Intervention*. Springer. 2021.

[100] J. Ma and B. Wang. “Segment anything in medical images”. In: *arXiv preprint arXiv:2304.12306* (2023).

[101] G. J. McLachlan. “Iterative reclassification procedure for constructing an asymptotically optimal rule of allocation in discriminant analysis”. In: *Journal of the American Statistical Association* 70.350 (1975), pp. 365–369.

[102] R. Mendel, L. A. de Souza, D. Rauber, J. P. Papa, and C. Palm. “Semi-supervised Segmentation Based on Error-Correcting Supervision”. In: *Proceedings of the european conference on computer vision*. 2020.

[103] D. J. Miller and H. Uyar. “A mixture of experts classifier with learning based on both labelled and unlabelled data”. In: *Advances in neural information processing systems* 9 (1996).

[104] F. Milletari, N. Navab, and S.-A. Ahmadi. “V-net: Fully convolutional neural networks for volumetric medical image segmentation”. In: *2016 fourth international conference on 3D vision (3DV)*. Ieee. 2016, pp. 565–571.

[105] S. Mittal, M. Tatarchenko, and T. Brox. "Semi-supervised semantic segmentation with high-and low-level consistency". In: *IEEE transactions on pattern analysis and machine intelligence* (2019).

[106] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii. "Virtual adversarial training: a regularization method for supervised and semi-supervised learning". In: *IEEE transactions on pattern analysis and machine intelligence* 41.8 (2018), pp. 1979–1993.

[107] Y. Netzer and T. Wang. "Reading Digits in Natural Images with Unsupervised Feature Learning". In: *Advances in neural information processing systems workshop on deep learning and unsupervised feature learning* (2011).

[108] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. "Text classification from labeled and unlabeled documents using EM". In: *Machine learning* 39 (2000), pp. 103–134.

[109] M. Noroozi and P. Favaro. "Unsupervised learning of visual representations by solving jigsaw puzzles". In: *Proceedings of the european conference on computer vision*. Springer. 2016, pp. 69–84.

[110] A. Odena. "Semi-supervised learning with generative adversarial networks". In: *arXiv preprint arXiv:1606.01583* (2016).

[111] A. Oliver, A. Odena, C. Raffel, E. D. Cubuk, and I. J. Goodfellow. "Realistic evaluation of deep semi-supervised learning algorithms". In: *arXiv preprint arXiv:1804.09170* (2018).

[112] OpenAI. ChatGPT.https://openai.com/blog/chatgpt/. 2023.

[113] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. "Dinov2: Learning robust visual features without supervision". In: *arXiv preprint arXiv:2304.07193* (2023).

[114] Y. Ouali, C. Hudelot, and M. Tami. "An Overview of Deep Semi-Supervised Learning". In: *arXiv preprint arXiv:2006.05278* (2020).

[115] Y. Ouali, C. Hudelot, and M. Tami. "Semi-supervised semantic segmentation with cross-consistency training". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020.

[116] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. "Training language

models to follow instructions with human feedback". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 27730–27744.

[117] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. "Training language models to follow instructions with human feedback". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 27730–27744.

[118] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. "Enet: A deep neural network architecture for real-time semantic segmentation". In: *arXiv preprint arXiv:1606.02147* (2016).

[119] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. "Context encoders: Feature learning by inpainting". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2536–2544.

[120] J. Peng, P. Wang, C. Desrosiers, and M. Pedersoli. "Self-paced contrastive learning for semi-supervised medical image segmentation with meta-labels". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 16686–16699.

[121] R. Prudêncio, J. Hernández-Orallo, and A. Martınez-Usó. "Analysis of instance hardness in machine learning using item response theory". In: *Second International Workshop on Learning over Multiple Contexts in ECML*. 2015.

[122] S. Qiao, W. Shen, Z. Zhang, B. Wang, and A. Yuille. "Deep co-training for semi-supervised image recognition". In: *Proceedings of the european conference on computer vision*. 2018, pp. 135–152.

[123] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. "Learning transferable visual models from natural language supervision". In: *International conference on machine learning*. PMLR. 2021, pp. 8748–8763.

[124] A. Rasmus, H. Valpola, M. Honkala, M. Berglund, and T. Raiko. "Semi-supervised learning with ladder networks". In: *arXiv preprint arXiv:1507.02672* (2015).

[125] S. Ravi and H. Larochelle. "Optimization as a model for few-shot learning". In: *International conference on learning representations*. 2016.

[126] M. N. Rizve, K. Duarte, Y. S. Rawat, and M. Shah. “In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning”. In: *arXiv preprint arXiv:2101.06329* (2021).

[127] O. Ronneberger, P. Fischer, and T. Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical Image Computing and Computer Assisted Intervention*. 2015.

[128] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé, et al. “Bloom: A 176b-parameter open-access multilingual language model”. In: *arXiv preprint arXiv:2211.05100* (2022).

[129] K. Simonyan and A. Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[130] M. R. Smith and T. Martinez. “A comparative evaluation of curriculum learning with filtering and boosting in supervised classification problems”. In: *Computational Intelligence* 32.2 (2016), pp. 167–195.

[131] M. R. Smith, T. Martinez, and C. Giraud-Carrier. “An instance level analysis of data complexity”. In: *Machine learning* 95.2 (2014), pp. 225–256.

[132] K. Sohn, D. Berthelot, C.-L. Li, Z. Zhang, N. Carlini, E. D. Cubuk, A. Kurakin, H. Zhang, and C. Raffel. “Fixmatch: Simplifying semi-supervised learning with consistency and confidence”. In: *arXiv preprint arXiv:2001.07685* (2020).

[133] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. “Going deeper with convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[134] A. Tarvainen and H. Valpola. “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results”. In: *Advances in Neural Information Processing Systems*. 2017.

[135] Y. Tian, X. Chen, and S. Ganguli. "Understanding self-supervised learning dynamics without contrastive pairs". In: *International Conference on Machine Learning*. 2021, pp. 10268–10278.

[136] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. "Llama: Open and efficient foundation language models". In: *arXiv preprint arXiv:2302.13971* (2023).

[137] J. M. J. Valanarasu and V. M. Patel. "Unext: Mlp-based rapid medical image segmentation network". In: *Medical Image Computing and Computer Assisted Intervention*. Springer. 2022.

[138] J. E. Van Engelen and H. H. Hoos. "A survey on semi-supervised learning". In: *Machine Learning* 109.2 (2020), pp. 373–440.

[139] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[140] V. Verma, K. Kawaguchi, A. Lamb, J. Kannala, Y. Bengio, and D. Lopez-Paz. "Interpolation consistency training for semi-supervised learning". In: *arXiv preprint arXiv:1903.03825* (2019).

[141] G. Wang, X. Liu, C. Li, Z. Xu, J. Ruan, H. Zhu, T. Meng, K. Li, N. Huang, and S. Zhang. "A noise-robust framework for automatic segmentation of COVID-19 pneumonia lesions from CT images". In: *IEEE Transactions on Medical Imaging* 39.8 (2020), pp. 2653–2663.

[142] X. Wang, H. Chen, H. Xiang, H. Lin, X. Lin, and P.-A. Heng. "Deep virtual adversarial self-training with consistency regularization for semi-supervised medical image classification". In: *Medical image analysis* 70 (2021), p. 102010.

[143] Y. Wang, C. Xu, C. Liu, L. Zhang, and Y. Fu. "Instance credibility inference for few-shot learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 12836–12845.

[144] Y. Wang, H. Wang, Y. Shen, J. Fei, W. Li, G. Jin, L. Wu, R. Zhao, and X. Le. "Semi-Supervised Semantic Segmentation Using Unreliable Pseudo-Labels". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[145] Z. Wang, Z. Zhao, X. Xing, D. Xu, X. Kong, and L. Zhou. "Conflict-Based Cross-View Consistency for Semi-Supervised Semantic Segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 19585–19595.

[146] C. Wei, K. Sohn, C. Mellina, A. Yuille, and F. Yang. "CReST: A Class-Rebalancing Self-Training Framework for Imbalanced Semi-Supervised Learning". In: *arXiv preprint arXiv:2102.09559* (2021).

[147] Y. Wu, Z. Ge, D. Zhang, M. Xu, L. Zhang, Y. Xia, and J. Cai. "Mutual consistency learning for semi-supervised medical image segmentation". In: *Medical Image Analysis* 81 (2022), p. 102530.

[148] Y. Wu, Z. Wu, Q. Wu, Z. Ge, and J. Cai. "Exploring smoothness and class-separation for semi-supervised medical image segmentation". In: *Medical Image Computing and Computer Assisted Intervention*. Springer. 2022.

[149] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo. "SegFormer: Simple and efficient design for semantic segmentation with transformers". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 12077–12090.

[150] J. Xie, R. Girshick, and A. Farhadi. "Unsupervised deep embedding for clustering analysis". In: *International conference on machine learning*. 2016, pp. 478–487.

[151] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, and Q. V. Le. "Unsupervised data augmentation for consistency training". In: *Advances in Neural Information Processing Systems*. 2020.

[152] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le. "Self-training with noisy student improves imagenet classification". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020.

[153] Y. Xu, L. Shang, J. Ye, Q. Qian, Y.-F. Li, B. Sun, H. Li, and R. Jin. "Dash: Semi-Supervised Learning with Dynamic Thresholding". In: *International Conference on Machine Learning*. 2021, pp. 11525–11536.

[154] Z. Xu, R. Jin, J. Zhu, I. King, and M. Lyu. "Efficient convex relaxation for transductive support vector machine". In: *Advances in neural information processing systems* 20 (2007).

[155] Z. Xu, R. Jin, J. Zhu, I. King, M. Lyu, and Z. Yang. "Adaptive regularization for transductive support vector machine". In: *Advances in Neural Information Processing Systems* 22 (2009).

[156] I. Z. Yalniz, H. Jégou, K. Chen, M. Paluri, and D. Mahajan. "Billion-scale semi-supervised learning for image classification". In: *arXiv preprint arXiv:1905.00546* (2019).

[157] J. Yang, D. Parikh, and D. Batra. "Joint unsupervised learning of deep representations and image clusters". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 5147–5156.

[158] J. Yang, M. Gao, Z. Li, S. Gao, F. Wang, and F. Zheng. "Track anything: Segment anything meets videos". In: *arXiv preprint arXiv:2304.11968* (2023).

[159] L. Yang, W. Zhuo, L. Qi, Y. Shi, and Y. Gao. "St++: Make self-training work better for semi-supervised semantic segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[160] X. Yang, Z. Song, I. King, and Z. Xu. "A survey on deep semi-supervised learning". In: *IEEE Transactions on Knowledge and Data Engineering* (2022).

[161] C. You, R. Zhao, L. H. Staib, and J. S. Duncan. "Momentum contrastive voxel-wise representation learning for semi-supervised volumetric medical image segmentation". In: *Medical Image Computing and Computer Assisted Intervention*. Springer. 2022.

[162] C. You, Y. Zhou, R. Zhao, L. Staib, and J. S. Duncan. "Simcvd: Simple contrastive voxel-wise representation distillation for semi-supervised medical image segmentation". In: *IEEE Transactions on Medical Imaging* 41.9 (2022), pp. 2228–2237.

[163] L. Yu, S. Wang, X. Li, C.-W. Fu, and P.-A. Heng. "Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2019, pp. 605–613.

[164] J. Yuan, Y. Liu, C. Shen, Z. Wang, and H. Li. “A Simple Baseline for Semi-supervised Semantic Segmentation with Strong Data Augmentation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021.

[165] Y. Yuan, K. Yang, and C. Zhang. “Hard-aware deeply cascaded embedding”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2017, pp. 814–823.

[166] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. “Cutmix: Regularization strategy to train strong classifiers with localizable features”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019.

[167] B. Zhang, Z. Tian, Q. Tang, X. Chu, X. Wei, C. Shen, et al. “Segvit: Semantic segmentation with plain vision transformers”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 4971–4982.

[168] H. Zhang, X. Li, and L. Bing. “Video-llama: An instruction-tuned audio-visual language model for video understanding”. In: *arXiv preprint arXiv:2306.02858* (2023).

[169] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. “mixup: Beyond empirical risk minimization”. In: *arXiv preprint arXiv:1710.09412* (2017).

[170] R. Zhang, P. Isola, and A. A. Efros. “Split-brain autoencoders: Unsupervised learning by cross-channel prediction”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1058–1067.

[171] Y. Zhang, B. Deng, K. Jia, and L. Zhang. “Label propagation with augmented anchors: A simple semi-supervised learning baseline for unsupervised domain adaptation”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 781–797.

[172] Z. Zhao, Y. Liu, M. Zhao, D. Yin, Y. Yuan, and L. Zhou. “Rethinking Data Perturbation and Model Stabilization for Semi-supervised Medical Image Segmentation”. In: *arXiv preprint arXiv:2308.11903* (2023).

[173] Z. Zhao, S. Long, J. Pi, J. Wang, and L. Zhou. “Instance-specific and Model-adaptive Supervision for Semi-supervised Semantic

Segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 23705–23714.

[174] Z. Zhao, L. Yang, S. Long, J. Pi, L. Zhou, and J. Wang. "Augmentation Matters: A Simple-yet-Effective Approach to Semi-supervised Semantic Segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 11350–11359.

[175] Z. Zhao, M. Zhao, Y. Liu, D. Yin, and L. Zhou. "Entropy-based Optimization on Individual and Global Predictions for Semi-Supervised Learning". In: *Proceedings of the 31st ACM International Conference on Multimedia*. 2023, pp. 8346–8355.

[176] Z. Zhao, L. Zhou, Y. Duan, L. Wang, L. Qi, and Y. Shi. "DC-SSL: Addressing Mismatched Class Distribution in Semi-Supervised Learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[177] Z. Zhao, L. Zhou, L. Wang, Y. Shi, and Y. Gao. "LaSSL: Label-guided Self-training for Semi-supervised Learning". In: *Proceedings of the AAAI conference on artificial intelligence*. 2022.

[178] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr, et al. "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 6881–6890.

[179] Y. Zhong, B. Yuan, H. Wu, Z. Yuan, J. Peng, and Y.-X. Wang. "Pixel Contrastive-Consistent Semi-Supervised Semantic Segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021.

[180] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. "Learning with local and global consistency". In: *Advances in neural information processing systems* 16 (2003).

[181] T. Zhou, S. Wang, and J. Bilmes. "Curriculum learning by dynamic instance hardness". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 8602–8613.

[182] Y. Zhou, H. Xu, W. Zhang, B. Gao, and P.-A. Heng. "C3-SemiSeg: Contrastive Semi-supervised Segmentation via Cross-set Learning and Dynamic Class-balancing". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021.

[183] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny. "Minigpt-4: Enhancing vision-language understanding with advanced large language models". In: *arXiv preprint arXiv:2304.10592* (2023).

[184] X. Zhu, Z. Ghahramani, and J. D. Lafferty. "Semi-supervised learning using gaussian fields and harmonic functions". In: *International conference on machine learning*. 2003, pp. 912–919.

[185] X. Zhu and A. Goldberg. *Introduction to Semi-Supervised Learning*. Morgan & Claypool Publishers, 2009.

[186] Y. Zou, Z. Zhang, H. Zhang, C.-L. Li, X. Bian, J.-B. Huang, and T. Pfister. "PseudoSeg: Designing Pseudo Labels for Semantic Segmentation". In: *International conference on learning representations*. 2021.