SCHOOL OF ELECTRONIC ENGINEERING
DUBLIN CITY UNIVERSITY (DCU)

Domain Adaptation of Neural Networks for Medical Imaging under Limited Data Constraints

---

Author
Sidra Aleem, B.S, M.S.

Supervisors
Prof. Suzanne Little,
Dr. Kevin McGuinness
Co-Supervisor
Dr. Julia Dietlmeier

A Dissertation submitted in fulfillment of the requirements for the award of Doctor of Philosophy (Ph.D.)

July 2025

Declaration

I hereby certify that this material, which I now submit for assessment on the program of study leading to the award of Doctor of Philosophy is entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Sidra Aleem **Student Number:** 20213367

Date: 2025–07–24

Dedication

Dedicated to my beloved parents

My extraordinary mother, **Robina Yasmeen**, is the cornerstone of my life. Her unwavering love, support, and guidance have shaped me into the person I am today. I hold her in the highest regard and will be forever grateful for everything you have done.

My dear father, **Mirza Abdul Aleem**, whose support and presence have been a constant source of strength in my life. He has always made things seem effortless, bearing the weight of challenges so that we never realized their true magnitude. I will be eternally grateful to you.

Acknowledgments

First and foremost, I would like to express my sincere and deepest gratitude to Dr. Kevin McGuinness. The successful completion of this PhD journey would not have been possible without his invaluable support and guidance. His exceptional mentorship has been instrumental in shaping not only the direction of this thesis but also my overall development as a researcher. Coming from a non-AI background, I was initially daunted by the technical depth and breadth of the field. However, Dr. Kevin's willingness to answer even the basic questions and his continuous support helped me to think critically and refine both my work and skills. I extend my sincere gratitude to my co-supervisors, Suzanne Little and Julia Dietlmeier, for their constant support throughout my PhD journey. Suzanne's confidence in my abilities and continuous encouragement, whether during the weekly meetings or simply by being available to listen, made the process more manageable and less overwhelming. Dr. Julia's proactive support and career guidance have been helpful.

I sincerely thank my collaborator, Mayug Manipramabil, for being an essential part of this research journey. His thoughtful problem-solving and technical advice have greatly helped me grow as a researcher. I appreciate your support and our collaborative work over the years. I am also grateful to Eric Arazo for his readiness to help, his willingness to engage in discussions about research ideas, and the support he has provided me.

I am thankful to my friends, especially Fatima and Hamza, for being my family away from home. Your constant support, whether it was listening, comforting me, or helping me stay optimistic about my journey, even while facing your own challenges,

gave me strength when I needed it most. Thanks to Faithful for always lifting my spirits and taking care of me like your own sister, I have truly found a brother in you here in Dublin. A sincere thanks to Boi, your presence and support have helped me a lot. I would also thank Amina for always being there for me through both the difficult moments and the good times, for patiently listening to my stories over and over without complaint, and for consistently helping me navigate through my struggles. Thanks to Ali for his prayers for the successful completion of my PhD, and for instantly shifting into a serious mode whenever I needed motivation. I appreciate Halima for listening to my problems and somehow calming me in unexpected ways with your blunt and straightforward opinions. And a big thank you to Mehreen for bringing the coffee machine into the lab, it genuinely helped me stay focused and get through my work.

Last but not least, I am immensely grateful to my wonderful sisters, Seerat and Arooj, for always being my anchor. You have supported me through everything, understood the challenges of my PhD journey, encouraged me to pursue my life with a positive outlook, and offered comfort in every possible way. Words cannot fully express how truly blessed I am to have both of you in my life. Thank you to my brother, Awais, for laying the foundation that allowed me to begin my journey abroad and for your support. And to my brother, Usman— thank you for always welcoming me with an open heart and making me feel completely at home every time I visited you in the UK.

Publications

1. **Sidra Aleem**, Fangyijie Wang, Mayug Maniparambil, Eric Arazo, Julia Dietlmeier, Kathleen Curran, Noel E O' Connor, Suzanne Little. "Test-Time Adaptation with SaLIP: A Cascade of SAM and CLIP for Zero-shot Medical Image Segmentation", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (CVPR) 2024.

2. **Sidra Aleem**, Julia Dietlmeier, Eric Arazo, Suzanne Little. "ConvLoRA and AdaBN based Domain Adaptation via Self-Training", IEEE 21st International Symposium on Biomedical Imaging (ISBI), 2024.

3. **Sidra Aleem**, Mayug Maniparambil, Suzanne Little, Noel O'Connor, Kevin McGuinness. "An Ensemble Deep Learning Approach for COVID-19 Severity Prediction Using Chest CT Scans", IEEE 25th Irish Machine Vision and Image Processing Conference (IMVIP), Belfast, August 2023.

4. **Sidra Aleem**, Teerath Kumar, Suzanne Little, Malika Bendechache, Kevin McGuinness. "Random data augmentation based enhancement: a generalized enhancement approach for medical datasets", IEEE 24th Irish Machine Vision and Image Processing Conference (IMVIP), Belfast, August 2022.

Contents

1 Introduction **2**
- 1.1 Motivation . . . 3
  - 1.1.1 Domain Shift . . . 3
  - 1.1.2 Domain Shift between Natural and Medical Imaging . . . 5
  - 1.1.3 Domain Shift within Medical Imaging . . . 6
  - 1.1.4 Medical Data Scarcity . . . 8
  - 1.1.5 Domain Adaptation in Medical Imaging under Limited Data Constraint . . . 10
- 1.2 Hypothesis and Research Questions . . . 11
- 1.3 Contributions . . . 13
- 1.4 Thesis Structure . . . 15

2 Background **17**
- 2.1 Deep Learning . . . 17
  - 2.1.1 Fundamentals of Deep Learning . . . 18
  - 2.1.2 Convolutional Neural Network . . . 20
  - 2.1.3 Training a Neural Network . . . 22
- 2.2 Deep Learning in Medical Imaging Analysis . . . 25
- 2.3 Approaches to Overcome Domain Shift Challenges . . . 26
  - 2.3.1 Transfer Learning . . . 26
  - 2.3.2 Domain Adaptation . . . 28
  - 2.3.3 Comparison of Adaptation Techniques . . . 29
- 2.4 Literature Review . . . 31
  - 2.4.1 Supervised Domain Adaptation . . . 31
  - 2.4.2 Unsupervised Domain Adaptation . . . 33
  - 2.4.3 Semi-Supervised Domain Adaptation . . . 35
  - 2.4.4 Adaptation of Foundation Models . . . 38
- 2.5 Summary . . . 42

3 Domain Shift in Medical Imaging: Need for Domain Adaptation **44**
- 3.1 Introduction . . . 45
  - 3.1.1 Motivation . . . 47
  - 3.1.2 STOIC 2021- COVID-19 AI Challenge: Overview . . . 47
- 3.2 Related Work . . . 48
- 3.3 Methodology . . . 50
  - 3.3.1 Pre-Processing . . . 50
  - 3.3.2 Ensemble of Neural Networks with Test Time Augmentations 52
- 3.4 Experimental Framework . . . 55

3.4.1 Dataset . . . 55
3.4.2 Implementation details . . . 55
3.5 Results and Analysis . . . 57
3.5.1 Sampling function . . . 57
3.5.2 Model Evaluation and Selection for Qualification Phase . . . 59
3.5.3 Final Phase Submission: Ensemble Approach with TTA . . . 67
3.6 Summary . . . 69
3.6.1 Insights . . . 70

4 Unsupervised Parameter Efficient Domain Adaptation for Multi-Target Medical Applications 72
4.1 Introduction . . . 73
4.2 Related Work . . . 76
4.2.1 Parameter Efficient Adaptation . . . 76
4.2.2 Batch Normalization based Adaptation . . . 77
4.2.3 Unsupervised Domain Adaptation (UDA) . . . 78
4.3 Methodology . . . 79
4.3.1 Preliminaries . . . 79
4.3.2 Proposed Approach: Unsupervised Parameter-Efficient Adaptation using Convolutional Low-Rank Adaptation and Adaptive Batch Normalization . . . 81
4.4 Experimental Framework . . . 87
4.4.1 Datasets . . . 87
4.4.2 Training setup . . . 88
4.5 Results and Analysis . . . 89
4.5.1 Source Model . . . 89
4.5.2 Early Segmentation Head Refinement . . . 92
4.5.3 ConvLoRA based Parameter Efficient Domain Adaptation . . . 93
4.5.4 Ablations . . . 97
4.5.5 Evaluation on M&M Dataset . . . 102
4.5.6 Analysis and Limitation . . . 104
4.6 Summary . . . 105
4.6.1 Insights . . . 106

5 Test Time Domain Adaptation of Foundation Models for Medical Image Segmentation 108
5.1 Introduction . . . 109
5.2 Related Work . . . 116
5.2.1 Segment Anything Model . . . 116
5.2.2 Adapting SAM for Medical Image Segmentation . . . 116
5.2.3 Contrastive Learning Image Pre-training . . . 119
5.3 Methodology . . . 120
5.3.1 Preliminaries . . . 120
5.3.2 Proposed Approach: SaLIP . . . 125
5.4 Experimental Framework . . . 130
5.4.1 Datasets and Metrics . . . 130
5.4.2 Implementation Details . . . 130
5.5 Results and Analysis . . . 131
5.5.1 Comparative Analysis of SaLIP and Other Methods . . . 131

5.5.2 Hyper-parameter Optimization . . . 138
5.5.3 ROI Mask Retrieval . . . 139
5.5.4 Area-based Mask Filtering . . . 142
5.5.5 Limitations and Potential Solutions . . . 143
5.5.6 Ablations . . . 150
5.6 Summary . . . 152
5.6.1 Insights . . . 153

6 Adaptation of Foundation Models for Fine-Grained Medical Imaging Analysis 155
6.1 Introduction . . . 156
6.2 Related Work . . . 160
6.3 Methodology . . . 162
6.3.1 Foundation Models Adaptation . . . 162
6.3.2 Fine-Grained Segmentation . . . 165
6.4 Experimental Framework . . . 167
6.4.1 Datasets and Metrics . . . 167
6.4.2 Implementation Details . . . 167
6.5 Results and Analysis . . . 168
6.5.1 Zero-shot Performance of CLIP on Fine-Grained Medical Tasks 168
6.5.2 Few-Shot Adaptation of CLIP using Adapters . . . 169
6.5.3 Evaluation of the Proposed SaLIP-V Framework on Fine-Grained Medical Tasks . . . 176
6.6 Summary . . . 187
6.6.1 Insights . . . 188

7 Conclusion 190
7.1 Hypothesis and Research Questions . . . 192
7.2 Research Contributions and Proposed Solutions . . . 195
7.3 Recommendations and Future Work . . . 197
7.4 Closing Remarks . . . 199

A Appendix Title 202
A.1 Textual Prompt Engineering for SaLIP using GPT-3.5 . . . 202
A.1.1 Lungs . . . 202
A.1.2 Prompt Engineering based on Spatial Location of Lungs . . . 205
A.1.3 Fetal Head . . . 207
A.2 Textual Prompt Engineering for Fine-grained Medical Tasks . . . 210
A.2.1 Polyps . . . 210

List of Figures

1.1 Domain shift caused by variations in weather and the day/night cycle [1]. 4

1.2 Impact of domain shift on the model's generalizability. Figure adapted from [2]. 5

1.3 Domain shift in medical imaging caused by variations in imaging modalities. 7

1.4 Top row: image slices, bottom row: corresponding intensity distribution of normalized T1-weighted (a, b) and T2-weighted (c, d) MRIs from different scanners [3]. 8

1.5 Intensity distribution of MRI axial-slice pixels for gray matter segmentation collected from four different data sets [4]. 8

2.1 Representation of a single neuron: The input features x_1, x_2 and x_3 are linearly combined and weighted with the corresponding weights w_1, w_2 and w_3. The weight w_0 corresponds to the bias of the neuron and is added to the linear combination. The non-linear function (g) is then applied to the output. 18

2.2 Graphical representation of different activation functions. 19

2.3 Multi-layer structure of a neural network: an input layer with three elements, hidden layer 1 and layer 2 with four neurons each, and an output layer. Figure adapted from [5]. 20

2.4 Convolution operation applied to one channel image. Figure adapted from [6]. 21

2.5 A CNN for handwritten digits classification. Figure adapted from [7]. 22

2.6 Schematic of backpropagation: x represents the input vector, y is the ground truth label, $h_w(\cdot)$ is the neural network with w parameters, $h_w(x)$ is the prediction of the network for the input x, and $L(\cdot)$ is the loss function that computes the error value $L(h_w(x), y)$ used for the parameter update. 25

2.7 Overview of transfer learning [8]. 27

2.8 Adaptation from the synthetic source domain to the real target domain [9]. 29

2.9 Overview of domain adaptation. Figure adapted from [2]. 30

3.1 Chest CT scans illustrating lung abnormalities associated with various stages of COVID-19 progression. 46

3.2 Comparative Analysis: **a)** Scaling **b)** Windowing. 51

3.3 The schematic overview of our proposed ensemble approach for COVID-19 severity prediction. 53

3.4 Centered Sampling with One Window (CS-1W): Example of slices retained with CS-1W. 58

3.5 Centered sampling with three windows (CS-3W): Example of slices retained with CS-3W. 58

4.1 Comparison of regular fine-tuning and LoRA's re-parameterization. In LoRA, the pre-trained weight matrix W remains frozen, while only the low-rank decomposition matrices A and B receive gradient updates [10]. 80

4.2 Source model (Φ_{src}) pre-trained on the source domain (src). 83

4.3 ESH initialization with the source domain. The source model is frozen and the gradient is backpropagated only to ESH. 84

4.4 2D U-Net with Early Segmentation Head (ESH): ConvLoRA adapters facilitate parameter efficient adaptation in the encoder, along with AdaBN throughout the network. 85
4.5 CC359 dataset: Different domains [11]. 87
4.6 M&M dataset [12]: Different domains. (Yellow: RV, Blue: LV, Green: MYO). 88
4.7 Performance of source model without adaptation on the target domain: GE 1.5 [11] 90
4.8 Performance of source model without adaptation on the target domain: Philips 1.5 [11]. 90
4.9 Performance of source model without adaptation on the target domain: Philips 3 [11]. 91
4.10 Performance of source model without adaptation on the target domain: Siemens 1.5 [11]. 91
4.11 Performance of source model without adaptation on the target domain: Siemens 3 [11]. 91
4.12 ESH initialization using the source domain: Loss and surface dice score curves. 93
4.13 Qualitative Results for CC359 [11] 97
4.14 Qualitative comparison: source model vs proposed adaptation for target domain: GE 1.5 [11] 97
4.15 Qualitative comparison: source model vs proposed adaptation for target domain: Philips 1.5 [11] 98
4.16 Qualitative comparison: source model vs proposed adaptation for target domain: Philips 3 [11]. 98
4.17 Qualitative comparison: source model vs proposed adaptation for target domain: Siemens 1.5 [11]. 99
4.18 Qualitative comparison: source model vs proposed adaptation for target domain: Siemens 3 [11]. 99

4.19 ConvLoRA integrated to the entire network (ConvLoRA). 101
4.20 Qualitative results target domain: GE [12]. 103
4.21 Qualitative results target domain: Siemens [12]. 103
4.22 Qualitative results target domain: Canon [12]. 104

5.1 Various segmentation modes of the Segment Anything Model. 112
5.2 The proposed SaLIP framework: The input image is processed through SAM's "everything mode", generating a set of masks for potential regions in the image. The image is then cropped based on the mask coordinates and passed to CLIP's image encoder. GPT-3.5 is used to generate visually descriptive sentences (VDTs) for target ROI. The retrieved ROI crop from CLIP is used to generate a bounding box prompt based on the coordinates of the ROI. This prompt and the input image are then passed to SAM's probabilistic segmentation for final segmentation masks. 115
5.3 Components of the Segment Anything Model [13]. 117
5.4 Architecture of Segment Anything Model [13]. 121
5.5 Architecture of Masked Auto-encoder [14]. 121
5.6 Architecture of the CLIP model [15]. 124
5.7 Architecture of our proposed SaLIP framework. 125
5.8 Pool of region proposal masks predicted by SAM_{EM} using grid-wise point prompts (red). 126
5.9 Cropped regions of the original input image based on SAM_{EM} masks. 127
5.10 Retrieval of the relevant mask using CLIP 128
5.11 Qualitative Analysis of Un-prompted SAM's Performance. 134
5.12 Qualitative Analysis for Performance Comparison: GT-SAM (Upper Bound), Un-prompted SAM, and SaLIP (Ours). 136
5.13 SaLIP Qualitative Results: X-ray labels and masks dataset [16]. . . . 137
5.14 SaLIP Qualitative Results: HC18 dataset [17]. 138

5.15 Effect of hyperparameters on region proposals generated by SAM_{EM} in an image. 139

5.16 Effect of hyperparameter optimization on mask generation by SAM_{EM}. 140

5.17 Comparison of various techniques for ROI mask retrieval. 141

5.18 Limitation of foundation models to perform domain-specific tasks in medical domain [16]. 142

5.19 Qualitative Results: a) No Filtering : All SAM-generated region proposals are fed to CLIP, leading to miss-classification of the ROI. b) Area filtering (ours): applies area filtering to SAM-generated region proposals to remove the masks encompassing ROI, thereby reducing the likelihood of miss-classification by CLIP. 144

5.20 SAM failure cases: First row: SAM_{EM} fails to generate a mask for the fetal head, resulting in miss-classification by CLIP. Second row: SAM_{EM} generates a mask for the right lung but fails to generate a mask for the left lung, eventually CLIP retrieves the wrong crop as ROI. 144

5.21 Qualitative Results: CLIP Failure cases for HC18 [17]: SAM_{EM} predicts a mask for the fetal head ("region proposal column"). However, CLIP does not retrieve the correct mask. 145

5.22 CLIP failure cases: First row: SAM_{EM} generates multiple masks for both ROIs (left and right lung). CLIP while correctly recognizing the right lung, identifies a second mask for the same lung region and fails to retrieve the crop corresponding to the left lung. Second row: CLIP did not retrieve the left lung crop. 146

5.23 Visual Prompt Engineering: Multiple visual prompts i.e., red circle (in this case) are drawn over an image and CLIP is tasked to choose the correct one given a caption. The image is taken from [18]. 148

5.24 VPT results on X-ray labels and masks dataset [16]: Although SAM_{EM} generated masks for both lungs, VPTs did not facilitate CLIP in accurately retrieving ROIs. 149

6.1 A polyp in the colon [19]. Cropping ROI in the image, results in the loss of global context necessary for fine-grained medical imaging tasks. While the visual prompting preserves the global context. . . . 158

6.2 The proposed SaLIP-V framework: a) few-shot classification of fine-grained image sub-regions: a linear classifier is trained in a few-shot setting to classify different sub-regions of the image. b) segmentation of fine-grained regions: the adapted classifier is used to identify the correct ROI from the pool of SAM-generated various sub-regions, and the selected ROI is then segmented. 159

6.3 Visual prompting and linear classifier adaptation: a) Few-shot dataset creation: SAM_{EM} generates region proposals for various regions in the image (M). These sub-regions are labeled with specific class labels by comparing (M) with the ground truth (GT). b) Few-shot training: Visual prompts (VPT) are overlaid on the input image to create I_{VPT}. This set is processed through frozen CLIP-V, and a linear layer is trained using few-shot examples to classify sub-regions as either ROI or irrelevant. 164

6.4 Architecture of our proposed SaLIP-V framework: SAM_{EM} segments image sub-regions using a grid of key-points (G), red bounding box visual prompts are overlaid on the input image I according to the resulting sub-regions (M) generating a pool of images I_{VPT}. These images are then processed through the frozen CLIP-V model and classified by the adapted linear classifier. The images categorized as ROI by the classifier are subsequently passed to SAM_{PSM} to get the final segmentation mask. 166

6.5 Impact of polyp variability on SaLIP-V performance. 183

6.6 SAM Limitation: No masks are generated for camouflaged polyp regions. Region proposals: Masks generated by SAM. 184
6.7 Few-shot dataset creation: comparison of labeling approaches. 186
6.8 SAM Limitation: Partial segmentation of the region of interest. . . . 186

List of Tables

3.1 Comparison of Sampling Functions: Impact on AUC Performance. . . 59

3.2 Evaluation of various models for COVID-19 severity prediction. . . . 62

3.3 Comparison of top-performing models on STOIC public data and qualification phase private test set. 63

3.4 Impact of metadata on COVID-19 severity prediction (AUC). 64

3.5 Impact of augmentation on COVID Severity Prediction. 66

3.6 Effect of augmentation on model generalization to the private test set. 66

3.7 Performance across various data splits (AUC: COVID-19 Severity). . 67

3.8 Final leaderboard results: Comparison with top-ranked methods [20]. 68

4.1 Evaluation of the source model on the CC359 target domains [11]. . . 92

4.2 Comparative analysis of Constrained adaptation: standard fine-tuning and proposed ConvLoRA adapters. 94

4.3 Comparative analysis of unsupervised domain adaptation approaches. 95

4.4 Comparison of Trainable Parameters for Different Adaptation Strategies. 96

4.5 Ablation Study: Placement of ConvLoRA adapters in the encoder and respective SDS, (Enc: Encoder). 100

4.6 Ablation Study: Placement of ConvLoRA adapters in the decoder and respective SDS, (Dec: Decoder). 100

4.7 Impact of ConvLoRA: Comparing Integration into the Full Model vs the Encoder. 102

4.8 ConvLoRA performance of M&M dataset [12] 102

5.1 Comparison of SaLIP with other methods. 133
5.2 Comparison of SaLIP with supervised adaptation approaches. 137
5.3 Comparative Analysis: impact of area-filtering on ROI Mask Retrieval. 143
5.4 Quantitative Analysis: CLIP retrieval performance with Separate Prompts vs Combined Prompts 147
5.5 Evaluation of visual prompting to enhance CLIP's recognition performance. 150
5.6 Ablation: Comparison of SAM's variant. 150
5.7 Ablation: Performance comparison between SAM-CLIP and SaLIP (ours). 151

6.1 Zero-Shot Classification Performance of CLIP: Crops vs. BBOX VPT. 169
6.2 Few-shot adaptation of CLIP's textual branch using various adapters. 171
6.3 Performance comparison of different SaLIP-V configurations on the chest X-ray dataset. 172
6.4 CLIP's textual branch adaptation branch using various adapters with swapped prompts. 173
6.5 CLIP's visual branch adaptation: comparison of different adapters. . 174
6.6 CLIP visual and textual features adaption in few-shot setting: comparison of different adapters. 175
6.7 Few-Shot Adaptation Comparisons: CLIP-V vs. DINOv2. 177
6.8 Evaluation of SAM-CLIP-V: A comparison of classification performance using the spatial average of patch embeddings vs CLS token embeddings. 179
6.9 Performance Improvement Analysis with Samplers 181

Abstract

"Domain Adaptation of Deep Neural Networks for Medical Imaging under Limited Data Constraints"

Sidra Aleem

Medical imaging analysis has advanced significantly due to developments in computer vision. However, deep learning models are typically trained on consistent data distributions, which hampers generalizability when evaluated on datasets with varying distributions. This issue is especially prominent in medical imaging, where heterogeneity arises from differences in acquisition sites, imaging protocols, scanner types, and patient demographics. Additionally, strong performance of neural networks is linked to the availability of large, labeled datasets. However, annotated data is scarce in medical imaging, and domain expertise is not readily available, further hindering robust model development.

This research addresses these challenges by proposing novel domain adaptation methods to improve neural network generalization across diverse medical imaging domains. The methods achieve effective adaptation while minimizing the dependency on large labeled datasets, addressing the limited data availability in real-world medical settings.

This work has developed three alternatives to supervised domain adaptation, with several key innovations: (1) A novel, unsupervised, parameter-efficient domain adaptation framework for multi-target medical imaging domains is proposed. It overcomes the limitations of supervised training and the scarcity of labeled data. (2) A novel test-time adaptation framework to adapt natural foundation models, enabling zero-shot transferability to medical tasks without relying on labeled data. It addresses several key challenges: the need for supervised training, domain-specific fine-tuning, the unavailability of annotated data, lack of domain expertise, and computational constraints. (3) A few-shot learning framework is proposed to adapt foundation models for fine-grained medical tasks, highlighting the intrinsic limitations of foundation models when applied to complex medical tasks.

These frameworks have improved our understanding of how domain adaptation can be effectively utilized for medical imaging analysis with limited labeled data and high data variability. This thesis serves as a valuable resource for medical practitioners and tool developers in designing innovative algorithms and applications for healthcare.

Chapter 1

Introduction

Image segmentation is a critical task in medical imaging analysis. During segmentation, an image is partitioned into meaningful regions. The precise segmentation and labeling of these structures can assist clinicians with disease diagnosis, prognosis, and treatment planning. Deep learning models, particularly when trained on large, labeled homogeneous datasets, have demonstrated the potential to match or outperform the clinical experts in certain cases [21, 22, 23]. Segmentation tasks are diverse in the medical domain, ranging from identifying large anatomical structures to detecting subtle pathological changes. Importantly, the heterogeneous nature of imaging modalities, acquisition protocols, patient demographics, and anatomical differences limits the generalizability and practical usability of models trained to predict a fixed set of predetermined classes in real-world clinical settings, as additional labeled data is required to capture new visual concepts.

This thesis explores alternatives to conventional supervised learning approaches for adapting neural networks to diverse medical imaging segmentation tasks. It focuses on the study of alternatives to address several key challenges that affect the generalization of neural networks: the heterogeneity of data, imaging modalities, patient demographics, acquisition sites, and protocols , the scarcity of annotated datasets, the complexity of anatomical structures, and the limited availability of domain expertise.

In the following chapters, innovative solutions are proposed for the robust adap-

tation of neural networks with minimal reliance on additional data and computational resources. The generalization of the proposed methods is evaluated across diverse medical imaging tasks.

This chapter provides an overview of the research presented in this thesis and outlines the motivation behind it, as discussed in Section 1.1. Section 1.2 presents the hypothesis and research questions. Section 1.4 outlines the structure of this thesis.

1.1 Motivation

1.1.1 Domain Shift

Although deep learning models have shown impressive performance on supervised learning tasks, they often struggle to generalize well when the training and test sets do not share the same distribution [24, 25, 26, 27]. Traditional machine learning models are typically trained under the assumption of independent and identically distributed (i.i.d.) data, meaning that the train and test sets follow the same distribution and are independent of each other [28]. However, this assumption rarely holds true in the real world.

Domain shift is a problem that arises when the data distribution in the train set differs from the test set data distribution [24, 29, 1, 30]. The model's generalizability significantly deteriorates when the model is presented with data from a new unforeseen domain that it did not encounter during training [31, 32, 1]. Domain shift can arise from various factors such as changes in lighting, acquisition devices, and background variations, as well as differences in image collection methods, sensor types, data sources, or geographical locations.

Figure 1.1 shows examples of how weather conditions can cause significant domain shift in image data. The first row shows images from the source domain, where the model is trained, while the second row shows images from the target domain, where the model is evaluated. These examples illustrate how variations in weather

conditions and day/night cycles create substantial visual differences in the data, even though all images were acquired from the same location. Such differences lead to domain shift between the source and target domains.

Figure 1.1: Domain shift caused by variations in weather and the day/night cycle [1].

Impact of Domain Shift

Below are a few key points to understand the impact of domain shift:

1. **Poor Robustness and Generalization:** A model that is not robust to domain shifts may fail to generalize well when evaluated on an unforeseen data domain. For instance, as illustrated in Figure 1.1, even though the data is collected from the same location, the day/night cycle and weather changes introduced variations that the model may not have encountered during training. Consequently, it will impact the model's generalizability, as illustrated in Figure 1.2.

2. **Decreased Model Accuracy:** When a model is trained on data from one domain (e.g., medical images from one hospital) and evaluated on another (e.g., images from a different hospital), differences in acquisition protocols, patient populations, demographics, and data distributions (e.g., lighting, resolution) can lead to a significant drop in accuracy. Thus, the model may not generalize well to the new domain, leading to a sub-optimal performance as outlined in Chapter 3.

3. **Increased Cost of Model Adaptation:** Adapting models to address domain shifts often requires re-training or fine-tuning with data from the new domain. This process demands a substantial amount of labeled data and is resource-intensive (Section 2.4.1). This issue is particularly pronounced when multiple target domains exist for a single task, such as (e.g, MRI scans from various hospitals). Training separate models for each domain is not computationally efficient and is often impractical. These challenges are explored in detail, with novel solutions proposed in Chapter 4.

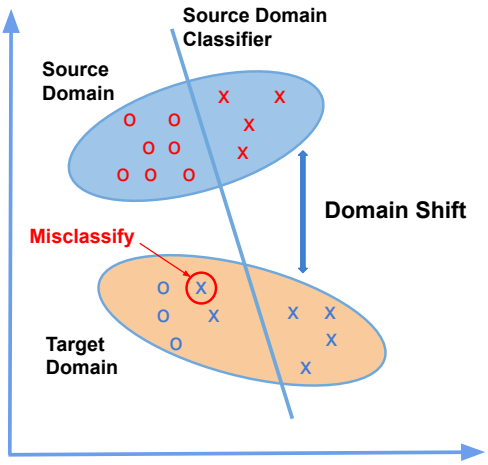


Figure 1.2: Impact of domain shift on the model's generalizability. Figure adapted from [2].

1.1.2 Domain Shift between Natural and Medical Imaging

Natural imaging datasets, such as ImageNet [33], significantly differ from medical imaging domains like X-rays, MRIs, and CT scans in terms of content, structure, and visual features [34, 35]. Models trained predominantly on natural images learn features like edges, textures, and shapes of everyday objects, which may not be suitable or effective for complex and challenging medical tasks, as experimentally demonstrated in Chapter 5. Medical images often have specific intensity patterns, contrasts, and structural elements that are not present in natural images. Their grayscale nature (e.g., in X-rays) differs from the RGB color space of natural imag-

ing [28]. Consequently, methods that perform well on natural images often fail to achieve optimal results in the medical imaging domain (Chapter 5 and Chapter 6).

Medical imaging analysis tasks often require the identification of specialized domain-specific features. The effective analysis further demands a detailed understanding of different image sub-regions to perform fine-grained tasks, such as recognizing complex pathological structures with varying morphologies and spatial patterns or detecting tumors that are subtle and small. These domain-specific features are not well represented in the natural imaging domain, leading to significant differences in the statistical distribution of features, which can result in poor performance when the model is evaluated on medical imaging tasks Section 5.5, Sections 5.5.5 and 6.5.3).

Additionally, unlike classification tasks in natural imaging domains, such as object identification, medical imaging tasks are considerably more complex. In medical imaging, pathologies are often camouflaged, resembling other anatomical structures (Figure 6.6). It is especially challenging to distinguish between various lesions due to the diverse spatial patterns and varying appearances of pathologies (as discussed in Section 6.5.3). Furthermore, classifying or recognizing anatomical structures based on their spatial location presents additional challenges (see Sections 5.5.5, 6.5.2 and 6.5.3).

All the challenges mentioned above highlight the significant differences between natural and medical imaging tasks, as well as the impact on a model's generalizability caused by domain shifts, especially due to the differences between features learned from natural images and those from medical images. These challenges are thoroughly investigated through experimentation, with novel solutions proposed in Chapter 5 and Chapter 6.

1.1.3 Domain Shift within Medical Imaging

The medical imaging domain does not only face challenges due to the domain shift between natural imaging and medical imaging (Section 1.1.2) but also from the

domain shift within medical imaging itself. The domain shift within medical imaging can arise from various factors such as different imaging modalities, scanner types, acquisition protocols, sites, patient populations, disease progression stages, age and gender differences, as well as variations in noise and artifacts [25, 28].

The diverse nature of medical imaging modalities leads to variations in how features are represented within the same region of interest. For instance, in multi-modal imaging such as MRI and CT, the same anatomical region is represented by distinctly different features and visual characteristics [23]. One such scenario is illustrated in Figure 1.3 (Cross Modality), where chest CT and MRI scan highlighted different features of the same region. Similarly, different types of microscopy produce varying representations of pathological structures. Even within a single imaging modality, where the imaging modality is the same, however, the variations in scanner manufacturers, models, and acquisition parameters can cause significant differences in the acquired data as illustrated in Figure 1.3 (Single Modality).

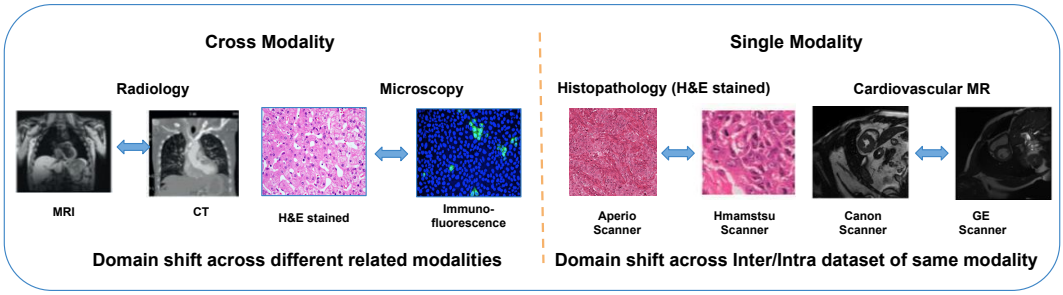


Figure 1.3: Domain shift in medical imaging caused by variations in imaging modalities.

Similarly, different imaging protocols, such as T1-weighted versus T2-weighted MRI sequences, highlight distinct anatomical features and pathologies, leading to divergent data characteristics [36]. Figure 1.4 shows the image-level distribution heterogeneity caused by different scanners. Figure 1.5 illustrates the problem of inter-center domain shift in terms of the intensity distribution of structural magnetic resonance imaging (MRI) at four independent sites (UCL, Montreal, Zurich, Vanderbilt) in Gray Matter Segmentation [4].

These intra- and inter-modality variations pose significant challenges to the gen-

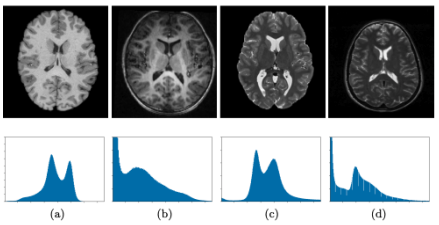


Figure 1.4: Top row: image slices, bottom row: corresponding intensity distribution of normalized T1-weighted (a, b) and T2-weighted (c, d) MRIs from different scanners [3].

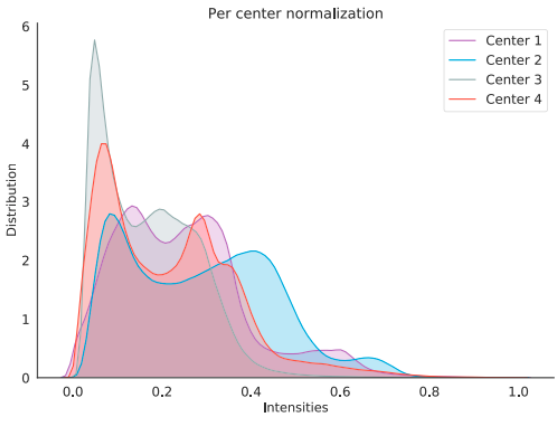


Figure 1.5: Intensity distribution of MRI axial-slice pixels for gray matter segmentation collected from four different data sets [4].

eralizability of machine learning models, particularly when applied to diverse imaging tasks. Models trained on data from a specific scanner or protocol fail to generalize when evaluated on the data acquired under different conditions. It emphasizes the need for robust domain adaptation techniques to mitigate the challenges of domain shift in medical imaging. These challenges are thoroughly explored through experiments, with novel solutions presented in the following chapters of this thesis.

1.1.4 Medical Data Scarcity

Deep learning has fundamentally transformed computer vision, offering unprecedented capabilities to solve complex visual perception tasks. The availability of large-scale, diverse, and well-annotated datasets (e.g., ImageNet [33]) has contributed substantially to the success of deep learning models in computer vision tasks [37, 1, 38]. Training on vast amounts of data enables neural networks to learn complex representations that are robust and generalize effectively to unseen samples [39, 40].

However, the data presented to the model has to be exhaustively annotated: the presence of mislabeled samples or samples that do not belong to the expected

distribution of classes can lead to degradation of the representations learned by the model [41, 42, 43].

The acquisition of medical datasets poses unique challenges compared to natural imaging datasets. While the latter benefits from the relative ease of collecting images of everyday objects, medical datasets are typically much smaller, often comprising only hundreds to thousands of images [44]. Although gathering data in the medical imaging field might seem straightforward, the process is far more complex than collecting images of common objects. The primary obstacle in acquiring large datasets within the medical or clinical domain arises from the complex nature of the data acquisition itself. This process is heavily influenced by a variety of factors such as the type of disease, geographic location, the type and settings of imaging equipment, time constraints, patient privacy concerns, and institutional copyright restrictions [45].

While datasets such as RadImageNet [46], MIMIC-CXR [47], CheXpert [48], and ARCH [49] are available, they are skewed toward radiology and X-ray images. Furthermore, medical imaging datasets are often small, and training neural networks on such limited data can lead to overfitting, particularly when using deep neural networks [23, 39].

Moreover, annotating medical images requires specialized domain knowledge, to ensure both accuracy and clinical relevance, which necessitates the involvement of trained healthcare professionals [23, 50]. This expertise is often not available readily and the process is both time-consuming and labor-intensive.

Additionally, medical imaging tasks are inherently more complex than those in natural imaging. Medical imaging analysis tasks often require fine-grained region-level annotations, in addition to global image-level labels. Acquiring such precise labels in the medical domain is extremely challenging (Chapter 6).

Consequently, developing effective methods for training deep learning models with limited or no labeled data is crucial for medical imaging applications.

1.1.5 Domain Adaptation in Medical Imaging under Limited Data Constraint

Solutions to address the generalizability challenges of neural networks arising from domain shifts in medical imaging (Sections 1.1.2 and 1.1.3) and the limited availability of extensively annotated medical data (Section 1.1.4) focus on developing robust domain adaptation methods that require minimal or no annotated data.

- Labeled data is available only from the source domain (the domain on which the model is trained), while the target domain (the domain to which the model must be adapted) contains only unlabeled data.
- The source domain data is unavailable, and only unlabeled data from the target domain is accessible.

In the first scenario, the model is initially trained on labeled data from the source domain. The goal is to adapt the knowledge learned from the labeled source domain to an unlabeled target domain. This approach is commonly known as "unsupervised domain adaptation". Our proposed approach for addressing this scenario, particularly within the realm of multi-target domain adaptation in the medical field, is detailed in Chapter 4.

The second scenario, known as "test-time domain adaptation", does not involve a source domain. Instead, it adapts the model to a new, unseen target domain during the testing phase, without the need to re-train on the entire dataset. This adaptation relies solely on unlabeled data from the target domain. Solutions for this scenario, particularly in the context of adapting natural foundation models to medical organ segmentation, are discussed in Chapter 5.

A significant body of research has explored the adaptation of foundation models, predominantly trained on natural imaging data, to medical imaging tasks. However, most of these efforts have focused on global image-level tasks or tasks where clean, labeled data is readily available. The application of these models to complex fine-grained medical tasks (which have wider real-world applications) remains largely

unexplored. Solutions for adapting foundation models to fine-grained medical tasks, along with the associated challenges and limitations, are discussed in Chapter 6.

1.2 Hypothesis and Research Questions

The proposed hypotheses guided the development of this thesis toward addressing several core challenges in medical imaging analysis: **(a)** Assessing the impact of domain shifts (Section 1.1.2, 1.1.3), especially regarding the generalizability of neural networks and the computational constraints involved when adapting these models across multiple medical domains. **(b)** Developing effective strategies to address domain shifts and adapt neural networks to medical imaging scenarios with limited or no annotated data. **(c)** Exploring the adaptation of foundation models, predominantly trained on natural images, to diverse medical imaging tasks aims to address several challenges in the medical domain. The evaluation is conducted on both global image-level tasks and fine-grained medical analysis tasks.

Computer vision has relied heavily on models pre-trained on ImageNet [33] using supervised learning. Recent advancements have introduced alternative "foundation models" in computer vision, which benefit from increasingly large datasets [51, 13, 52, 53, 37].

The research presented in this thesis initially focused on the adaptation of convolution neural networks and transformer-based approaches. However, with the evolution of the field, foundation models have gained significant attention, and their potential benefits have been widely discussed in the literature. The research presented in this thesis aligns with this paradigm shift. Thus two hypotheses are proposed: Hypothesis 1 (H1) focuses on the adaptation of pre-foundation model approaches, such as CNNs, while Hypothesis 2 (H2) explores the adaptation of foundation models for the medical imaging domain.

H1. In medical imaging scenarios with multiple target domains, low-rank adapters can facilitate parameter-efficient adaptation of convolutional neural networks. It

provides an alternative to training separate dedicated networks for each domain and achieves performance similar to full model adaptation while reducing computational overhead.

- Initial domain adaptation methods typically involve pre-training a model with a labeled source domain, with the eventual adaptation on labeled/unlabeled the target domain (Section 2.4.1). In medical imaging, there are diverse target domains (Section 1.1.3), and creating separate models for each target domain, with the same trainable parameters as the base model, is impractical and computationally expensive. Parameter-efficient fine-tuning (PEFT) with low-rank adaptation has proven effective in adapting Large Language Models (LLMs) to various downstream tasks [10]. While PEFT-based adaptation has been widely explored in LLMs and transformer-based architectures [24, 54, 55], its application to convolution neural networks (CNNs), particularly for multi-target domain adaptation in medical imaging, remains unexplored.

The following research questions have guided and shaped the exploration of Hypothesis 1:

- **RQ 1:** What are the key challenges and limitations of supervised adaptation approaches when applied to diverse medical imaging datasets? Specifically, how do domain shifts and data scarcity affect the generalization of neural networks for medical imaging tasks?

- **RQ 2:** How could the parameter-efficient adaptation approach be enforced in the unsupervised adaptation of convolutional neural networks? Could convolutional neural networks benefit from the features learned through self-supervised training when using parameter-efficient adaptation?

H2. In the absence of annotated data or domain expertise, the test-time adaptation of foundation models (FMs) can enable efficient adaptation to diverse medical imaging tasks. For fine-grained analysis, FM-extracted features can be easily adapted without requiring large datasets.

- The use of Vision-Language Foundation Models (VLFMs) in medical imaging faces several significant challenges, including domain shift caused by features learned from pre-training on natural images and adaptation to downstream medical imaging tasks; overfitting due to limited medical data, and high computational overhead [56, 57, 58]. VLFMs are typically adapted to downstream medical imaging tasks through additional training, fine-tuning, or parameter-efficient adaptation (Section 2.4.4). However, the adaptation of VLFMs in medical imaging without additional training, domain-specific prompt engineering, or annotated data is not explored widely. Additionally, existing research has primarily focused on the adaptation of VLFMs for coarse image-level medical imaging tasks (Section 5.2). In contrast, medical imaging analysis tasks often require fine-grained region understanding and precise labeling. This research gap emphasizes the need for further exploration of VLFMs for fine-grained medical imaging analysis tasks.

The following research questions have guided and shaped the exploration of Hypothesis 2:

- **RQ 3:** Can test-time adaptation of foundation models provide a more robust alternative to unsupervised or semi-supervised domain adaptation approaches? Can foundation models be effectively adapted to diverse medical imaging tasks without relying on annotated data, additional training, or specialized domain expertise?

- **RQ 4:** Can foundation models be effectively adapted to challenging fine-grained medical imaging tasks?

1.3 Contributions

This section presents a description of the main contributions of this thesis as a result of the work described in subsequent chapters.

- **C1:** This thesis introduces novel solutions to enhance the generalizability of neural networks and addresses challenges posed by domain shift in medical imaging (Section 1.1.2 and 1.1.5) and the scarcity of annotated data and domain expertise in the medical sector (Section 1.1.4). To overcome the limitations of traditional supervised domain adaptation, this thesis specifically proposes multiple novel alternatives, including unsupervised (Chapter 4), test-time (Chapter 5), and few-shot (Chapter 6) domain adaptation strategies.

- **C2:** To overcome the limitations of traditional supervised domain adaptation methods, such as the creating dedicated separate fine-tuned models for each new domain and the risk of overfitting due to limited medical data, this research proposes a novel unsupervised, parameter-efficient domain adaptation approach tailored for multi-target medical imaging (Chapter 4). The key contribution of this method is to offer an unsupervised alternative that not only achieves high accuracy but also provides a computationally efficient solution for adapting CNNs across multi-target medical applications.

- **C3:** This research presents a novel test-time adaptation method to adapt foundation models, primarily trained on natural imaging, to medical imaging in a zero-shot manner (Chapter 5). It overcomes several critical challenges in adapting foundation models to the medical domain: it requires no annotated data, eliminates the need for supervised or task-specific training, bypasses the need for specialized domain knowledge in prompt engineering, and, as the proposed approach is adapted fully at test time, it alleviates the computational constraints typically associated with foundation models.

- **C4:** This research also experimentally evaluates several key challenges in adapting foundation models specifically for fine-grained medical imaging tasks. Unlike existing techniques that focus on global image-level tasks, the proposed few-shot adaptation method focuses on adapting foundation models for fine-grained medical imaging tasks (Chapter 6). The proposed method is evaluated

on tasks such as recognizing organs based on spatial location and identifying pathological structures that lack pre-defined shapes, spatial locations, or morphology.

1.4 Thesis Structure

The remainder of the thesis is structured as follows:

In Chapter 2 the technical background of the thesis, covering the principles of domain adaptation is provided. It also includes an overview of the fundamentals of neural networks and deep learning techniques for medical image segmentation. It presents a comprehensive literature review of domain adaptation research conducted in the field of medical imaging.

Chapter 3 presents our work done for the STOIC 2021 COVID-19 AI Challenge. It provides comprehensive details of all experiments conducted throughout the competition. A comparative analysis of the proposed approach with other participating teams is provided. Additionally, it highlights the insights gained from this challenge, particularly regarding domain shift issues in the medical imaging domain, which helped shape the research presented in this thesis.

Chapter 4, presents our work on unsupervised parameter-efficient adaptation of convolution neural networks for multiple medical target domains. It provides the architectural details of our proposed parameter-efficient adaptation framework. It also discusses current methods, highlighting their main limitation and suggesting potential solutions based on experimental observations. This chapter offers an in-depth comparative analysis of various domain adaptation approaches and insights gained from experimental evaluations, on the impact of domain shift on neural network generalizability across diverse medical imaging tasks.

Chapter 5 is focused on the adaptation of natural foundation models to the medical imaging domain. It presents our proposed framework for test-time adaptation of visual and language foundation models, including its architectural design and implementation details. It also includes a literature review of relevant work, highlighting

limitations in the direct applicability of foundation models to medical imaging. Additionally, it presents an extensive experimental evaluation of the proposed approach to show its effectiveness across diverse imaging modalities.

Chapter 6 presents our work on adaptation of natural foundation models for complex, fine-grained medical imaging analysis tasks. This chapter covers the architectural design and implementation details of the proposed approach. It provides a literature review of existing approaches, highlighting their key limitations, followed by potential solutions to address these issues. The extensive experimental evaluation aimed at designing an effective framework to handle fine-grained pathological structures in the medical domain is presented in detail. Additionally, the challenges and limitations of language models for fine-grained medical imaging tasks are demonstrated experimentally.

Lastly, Chapter 7 provides a comprehensive overview of the research presented in this thesis. It discusses the research objectives, highlights key contributions, and outlines the limitations of the work. Additionally, it proposes potential future directions for advancing domain adaptation research using neural networks and foundation models to address complex medical imaging tasks.

Chapter 2

Background

This chapter presents the theoretical background and a comprehensive literature review related to domain adaptation in medical imaging segmentation, which is the primary focus of this thesis. Section 2.1 provides the fundamentals of deep learning including convolutional neural networks and how neural networks are trained. Section 2.2 provides a brief overview of medical imaging segmentation and the challenges deep learning methods encounter in generalizing within the medical domain due to domain shift. Section 2.3 provides an overview of techniques to overcome domain shift challenges. Finally, section 2.4 presents a detailed literature review of relevant approaches to the challenges introduced in Chapter 1.

2.1 Deep Learning

Deep learning is a subfield of artificial intelligence and machine learning. In recent years, it has gained significant attention due to its remarkable success in a variety of tasks. Deep learning can be applied to various tasks, such as image classification, natural language processing, speech recognition, and semantic segmentation. The core idea of deep learning is to develop algorithms that are capable of mapping the input data to successive levels of abstraction until reaching the output space corresponding to the given task.

2.1.1 Fundamentals of Deep Learning

A neuron is the basic processing unit within a neural network. It receives input either directly from the raw input data or from the outputs of preceding neurons. It applies a linear transformation to the input, followed by a non-linear activation function, and provides the output to the subsequent neurons. The weights associated with the linear transformation are adjusted during the training process by minimizing a specified loss function (Section 2.1.3).

As depicted in Figure 2.1, inputs on the left (x_1, x_2, x_3) are linearly combined with the weights (w_1, w_2, w_3). The non-linear function (g) , known as an activation function, is then applied to the result. Additionally, each neuron has a bias (b) that is added to the matrix multiplication. Hence, the output of a single neuron becomes $\hat{y} = g(w^T x)$, where $w^T = (w_0, w_1, w_2, w_3)$ are the weights (and bias) of the neuron and $x^T = (x_1, x_2, x_3)$ are the features of the input vector (with $x_0 = 1$).

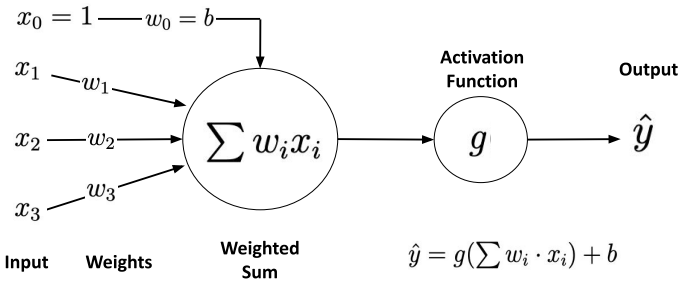


Figure 2.1: Representation of a single neuron: The input features x_1, x_2 and x_3 are linearly combined and weighted with the corresponding weights w_1, w_2 and w_3. The weight w_0 corresponds to the bias of the neuron and is added to the linear combination. The non-linear function (g) is then applied to the output.

Given an input vector of n features $x^T = (x_0, x_1, ..., x_n)$, each layer of the neural network performs a matrix multiplication between this input and the weight matrix associated with that layer. The weight matrix $W = [w_0, w_1, ..., w_d]$, has dimensions $n \times d$, where each column vector $w_i^T = (w_0, w_1, ..., w_n)$ represents the weights corresponding to neuron i in the layer. This multiplication linearly projects the input to a d dimensional space, that corresponds to the number of neurons in a layer. After the linear transformation, a non-linear activation function (g) is applied to

the resulting vector, yielding the output $g(W^T x)$.

Activation functions

Activation functions are essential components of artificial neurons, as they introduce non-linearity into the neural network. This non-linearity is crucial because it allows the network to learn and approximate complex, non-linear patterns in the data. Without it, the network would be limited to modeling only linear relationships, regardless of its depth. Commonly used activation functions include the sigmoid, hyperbolic tangent (tanh), and rectified linear unit (ReLU), along with variations like leaky ReLU and parametric ReLU, as shown in Figure 2.2.

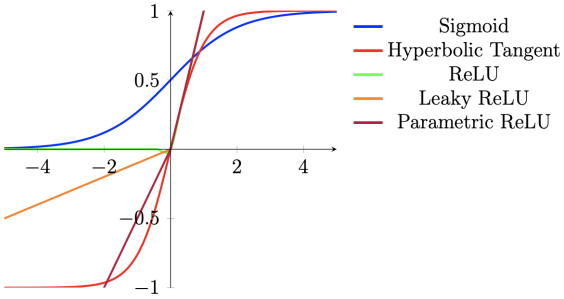


Figure 2.2: Graphical representation of different activation functions.

Figure 2.3 shows a common representation of a neural network. The leftmost layer represents the input layer, which consists of the features derived from the input samples. The subsequent layers, known as hidden layers, consist of multiple interconnected neurons which extract the features. The rightmost layer denotes the output layer, which generates predictions from the neural network for a given input.

While applying the traditional neural network structure to images, two main challenges are encountered. Firstly, depending on the number of features from the samples, the number of connections between the different layers would vary. For instance, an image of 200×200 with 3 color channels (RGB), would lead to an input layer with $200 \times 200 \times 3 = 120,000$ neurons and therefore 120,000 weights for each neuron in the subsequent layer. Furthermore, it is highly desirable to incorporate multiple neurons and layers, thereby resulting in a substantial increase in the number of parameters. However, the utilization of complete connectivity is

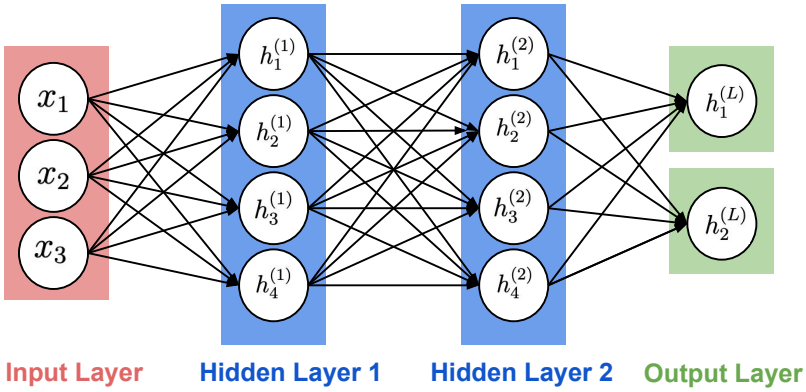


Figure 2.3: Multi-layer structure of a neural network: an input layer with three elements, hidden layer 1 and layer 2 with four neurons each, and an output layer. Figure adapted from [5].

inefficient and leads to an excessive number of parameters, ultimately giving rise to the issue of overfitting. Thus, scalability is the significant issue that is faced with such an architecture. Secondly, as each of the pixels is treated as an individual feature, the potential local spatial information within the data gets disregarded. The convolutional neural networks discussed in the following section successfully addressed these challenges.

2.1.2 Convolutional Neural Network

The primary feature of Convolutional Neural Networks (CNNs) is that each neuron processes a localized subset of adjacent input features, often organized as a square grid of pixels, particularly in visual tasks. Instead of processing every individual pixel of the input, CNNs apply a set of shared weights across the image, allowing for more efficient feature extraction and learning. This operation is called convolution [59] (illustrated in Figure 2.4), and formally can be expressed as:

$$S(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \tag{2.1}$$

where I is the input and K is the kernel (or filter) that corresponds to a set of weights associated with a neuron. Then m and n correspond to the dimensions of the input. The convolution operation is equivalent to the traditional image filtering

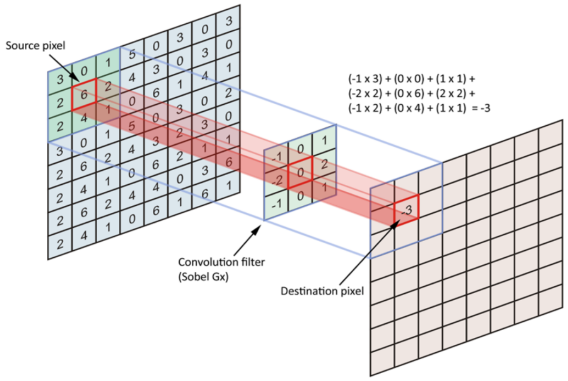


Figure 2.4: Convolution operation applied to one channel image. Figure adapted from [6].

operation, where a filter, defined by its weights, is multiplied with every position in the image. When the filter pattern aligns with a patch in the image, it produces a high value and activates the corresponding neuron. It significantly reduces the number of weights in the model.

Structure of CNN

A CNN consists of three main components: a convolutional layer, a non-linearity function, and a pooling layer. The convolutional layer applies kernels, which perform linear transformations on the input data illustrated in Figure 2.4. In the image domain, convolutions can be thought of as combinations of filtering operations including edge detectors and Gaussian blurs among others. Following the convolutional layer, a non-linearity is often introduced to allow the network to capture complex, non-linear relationships between features. Pooling layers are then used to progressively reduce the spatial dimensions of the feature maps by summarizing local regions. It is typically achieved by taking the maximum or average value of neighboring elements. This reduction helps the network become more invariant to small translations in the input images. Pooling also serves to adapt CNNs to the fully connected layers in the later stages of the model, where the input feature size needs to remain consistent regardless of the original image dimensions. In some cases, pooling layers can be replaced by larger strides in the convolutional process, which reduces spatial dimensions by skipping input features during convolution, achieving

a similar effect. The architecture of CNN is illustrated in Figure 2.5.

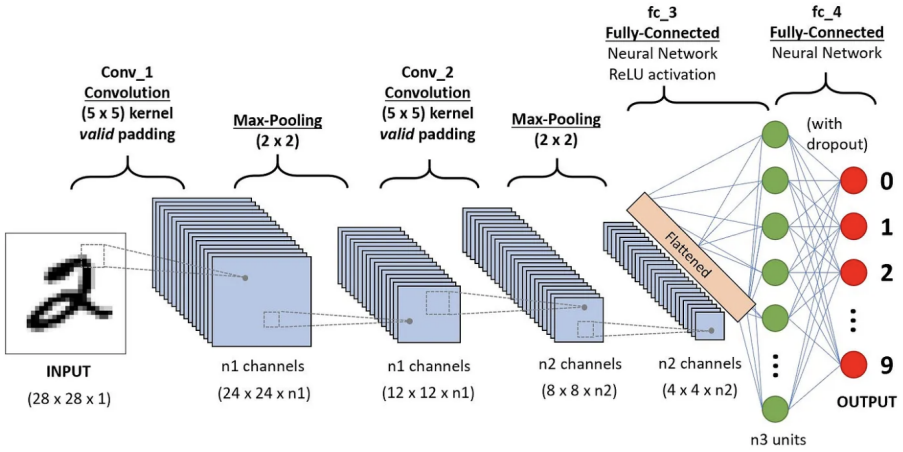


Figure 2.5: A CNN for handwritten digits classification. Figure adapted from [7].

Other elements often found embedded into CNNs are normalization layers that aim at stabilizing the training process by keeping the statistics of certain parts of the network normalized (zero mean and unit standard deviation): the statistics of the layer input [60], or the layer weights themselves [61].

2.1.3 Training a Neural Network

Training a neural network is a crucial step in deep learning. The parameters of the model are updated during training by an iterative refinement process. At each iteration, the input samples are fed to the model and the generated outputs are compared with the corresponding ground labels using the loss function.

Loss Function

The loss function, also referred to as the objective function or cost function, quantifies the difference between the predicted output and the target label for a given input. The goal of training a neural network is to minimize the loss function across the entire dataset. Various loss functions can be employed depending on the problem type and the desired properties of the model. Common loss functions include mean squared error (MSE) for regression tasks, cross-entropy for classification tasks, and

more complex task-specific loss functions. One of the widely used cost functions is cross entropy loss:

$$CE(\theta) = -\frac{1}{N}\sum_{i=1}^{N} y_i^T \log(h_\theta(x_i)) \tag{2.2}$$

where θ refers to the model parameters, x_i is the input, y_i refers to one hot encoded label of input x_i, N corresponds to the size of training data, and $h_\theta(x_i)$ is the model's predicted output.

Gradient Descent

A variety of algorithms are available to minimize the loss function. For developing basic intuition, gradient descent (GD) is the most suitable one. GD starts by randomly initializing the weights (θ), it is considered the initial point within the loss landscape. GD computes the best direction along which the weight vector should be updated (which is mathematically guaranteed to be the direction of the steepest descent). This direction is computed by calculating the gradient of the loss function. Once the weights are updated, GD takes another step following the same strategy. This process is repeated until convergence. The following equation shows the parameter update process:

$$\theta_{t+1} := \theta_t - \alpha \nabla_\theta l(\theta_t) \tag{2.3}$$

where θ_{t+1} are the weights of the model after the GD update and θ_t correspond to the current weights of the model. $l(\theta_t)$ corresponds to the loss of the model with the current set of weights and ∇_θ denotes the gradient operation for θ. The parameter α is called the learning rate. It defines the step size that the optimization algorithm will take.

Choosing an excessively high learning rate causes the optimization algorithm to oscillate between high-loss points and potentially diverge towards regions with higher loss values, skipping local optima. On the other hand, a very low learning rate leads to slow convergence towards a local minimum and the risk of getting trapped

in a sub-optimal minimum. A common strategy is to start with a moderately high learning rate to enable exploration of the loss landscape, gradually reducing it during training.

Backpropagation

In neural networks, backpropagation provides an efficient method for calculating the gradient of the loss. It employs the chain rule to compute the gradients of errors in different layers with respect to their weights. This gradient indicates how sensitive the error in each layer is to changes in its weights. By utilizing the chain rule, the loss with respect to the weights is derived as:

$$\frac{\partial l(\theta)}{\partial \theta} = \frac{\partial l(\theta)}{\partial_o} \times \frac{\partial_o}{\partial \theta} \tag{2.4}$$

where $\frac{\partial l(\theta)}{\partial_o}$ is the derivative of $l(\theta)$ with respect to previous layer (o), $\frac{\partial_o}{\partial \theta}$ is the derivative of the output of the previous layer with respect to the weights of that layer θ. For the previous layer, the gradients are computed similarly using the loss associated with that layer.

As the name suggests, backpropagation involves propagating the error computed in the last layer backward through the network to calculate the gradients for each layer. The schematic overflow of the backpropagation is illustrated in Figure 2.6. In the forward pass, the input data is propagated through the network to compute the output predictions and the loss value. During the backward pass, gradients of the loss function are calculated for each model parameter, starting from the output layer and progressing toward the input layer. This calculation is accomplished by applying the chain rule of calculus to determine the partial derivatives utilizing the chain rule outlined in Eq. 2.4.

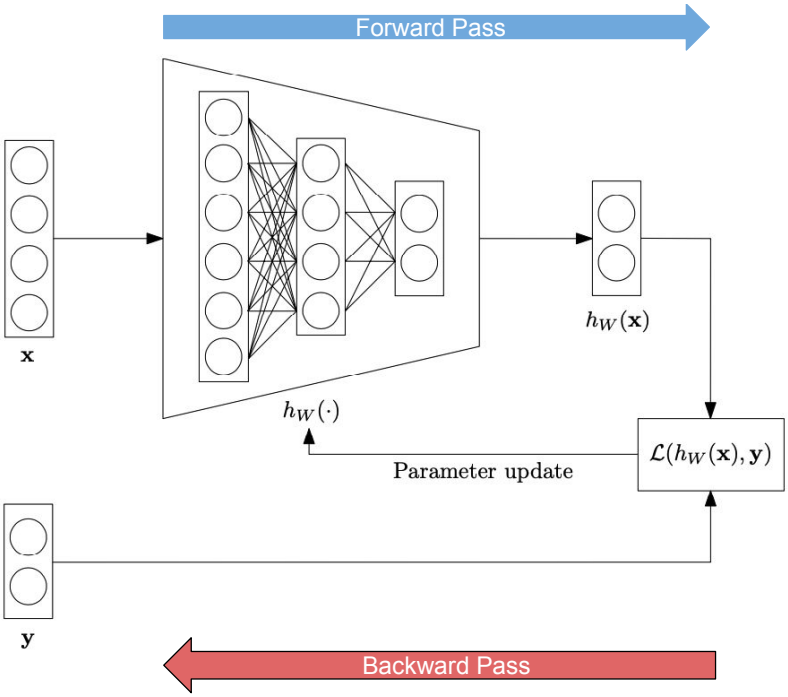


Figure 2.6: Schematic of backpropagation: x represents the input vector, y is the ground truth label, $h_w(\cdot)$ is the neural network with w parameters, $h_w(x)$ is the prediction of the network for the input x, and $L(\cdot)$ is the loss function that computes the error value $L(h_w(x), y)$ used for the parameter update.

2.2 Deep Learning in Medical Imaging Analysis

Medical image segmentation is a critical task in computer-aided diagnosis. It involves identifying and isolating specific regions of interest (ROIs) within medical images. These ROIs can represent organs, lesions, or other anatomical structures [62]. Accurate segmentation is crucial for various clinical applications, including disease diagnosis, treatment planning, and disease progression monitoring.

Deep learning-based models have widely been used for image segmentation. Their ability to learn complex image features has significantly improved segmentation accuracy across various tasks. In particular, U-shaped network variations have remained the go-to architectures during these past years. To train these models, the basic and most used loss functions are the cross-entropy loss and the dice score loss for segmentation [63].

However, the challenges associated with the generalization of deep learning models to various medical imaging tasks have been thoroughly discussed in Chapter 1.

"Additional challenges include the scarcity of medical data..."

These challenges include: the task-specific nature of the models (Section 1.1.1), impact on model generalization due to domain shifts between the feature distributions of natural imaging and medical imaging tasks (Section 1.1.2), domain shift with the medical domain itself which arise from various factors such as differences in scanners, imaging modalities, and institutions (Section 1.1.3), lack of medical data availability (Section 1.1.4) and computational costs related to the adaptation of pre-trained models to the downstream tasks. These challenges are multifaceted and will be explored in depth in the subsequent chapters of this thesis.

2.3 Approaches to Overcome Domain Shift Challenges

2.3.1 Transfer Learning

Transfer learning is the most common approach to reduce the impact of domain shift (Section 1.1.1) [64]. Transfer learning is a machine learning technique in which knowledge learned through one task or dataset is used to improve model performance on another related task/ a different dataset.

In a typical transfer learning setting, there are two concepts: "domain" and "task" [8, 65, 66]. A domain refers to the feature space of a specific dataset and the marginal probability distribution of features. A task refers to the main objective function of the model. The goal of transfer learning is to transfer the knowledge learned from the task T_a on domain A to the task T_b on domain B [8]. Note that either the domain or the task may change during the transfer learning process.

Initially, a large model is trained on a domain with abundant data and annotations. Subsequently, the model is fine-tuned on a different domain, where only a smaller dataset is available as illustrated in Figure 2.7. It offers several notable benefits. It can significantly enhance performance, particularly when the downstream

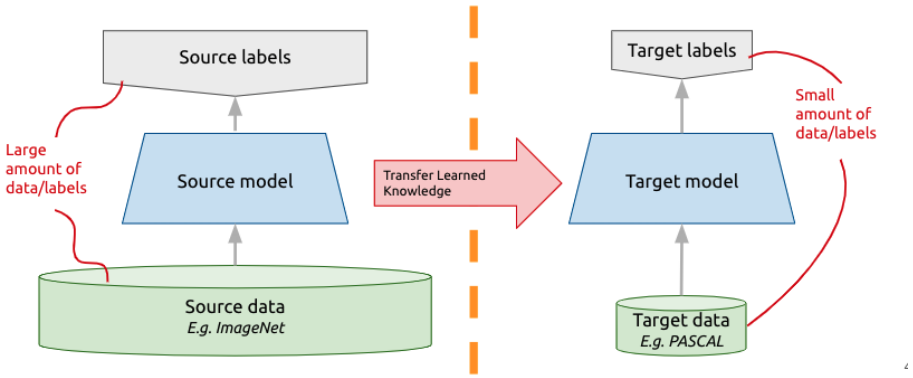


Figure 2.7: Overview of transfer learning [8].

task has limited labeled data [67, 68]. It also speeds up the training process because the model is initialized with weights that have already been trained on a large dataset. As a result, it can converge more quickly towards the new downstream task [69].

One major limitation of transfer learning is its reliance on a large, annotated dataset in the source domain, which is often not readily available. The medical datasets are often small due to the challenges of acquiring labeled data like data privacy regulations, the high costs of expert annotation, and the variability in imaging practices across healthcare institutions (Section 1.1.4). Fine-tuning a deep learning model (with millions of parameters) on small datasets can lead to overfitting. It also does not yield optimal results when there is a significant domain shift between the source domain on which the model is pre-trained and the target domain used on which the model is fine-tuned (Sections 3.5.2, 6.5.2). Furthermore, fine-tuning the large models is expensive and requires substantial resources.

The pre-trained models can also be used as fixed feature extractors. However, the pre-trained models are often trained on large, generic natural imaging datasets (e.g., ImageNet [33]). On the other hand, medical images (e.g., MRI, CT scans, X-rays) have vastly different features such as grayscale, high noise, and structural variability. The medical imaging analysis tasks often require identifying minute, domain-specific details like small tumors or subtle lesions. Thus using models as fixed feature extractors may not transfer well in such high-precision domains, re-

sulting in sub-optimal performance (as demonstrated experimentally in Chapter 6, Section 6.5.1, 6.5.2). This discrepancy arises from domain shift, where the characteristics of the source domain (natural images) differ from the target domain i.e. medical images (Sections 1.1.1, 1.1.2, 1.1.3).

2.3.2 Domain Adaptation

Deep learning models do not generalize well if the test set has a different distribution from the training set due to domain shift between the two distributions (Section 1.1.1). For instance, in medical image segmentation, the MRI and CT scans of the same region of interest look very different. If a model trained on MRI scans is applied to CT scans, it will likely perform poorly due to domain shift which arises from differences in imaging modalities. However, obtaining annotated datasets for each new imaging modality or task is not practically possible (Section 1.1.4).

Domain adaptation is a specific type of transfer learning that aims to adapt a model trained on one domain to perform well on another domain. In this context, the task remains the same but the data distribution changes [70, 71]. It involves two domains: **source domain:** the domain on which the model is initially trained using labeled examples. **target domain:** the domain whose data distribution differs from the source domain. The target domain is either unlabeled or contains only a small amount of labeled data. The model pre-trained on the source domain is evaluated to perform a similar task in the target domain.

The source and target domains may differ in input feature distribution, output labels, or both. One of the DA scenarios is illustrated in Figure 2.8, where a model trained on the synthetic dataset is adapted to the real target domain for the application of semantic segmentation.

The goal of domain adaptations (DA) is to address the differences in data distribution between the source and target domains so that the model generalizes well across both as shown in Figure 2.9. DA approaches enable the model to identify underlying patterns in the data that are relevant to the task at hand, while ignoring

domain-specific discrepancies. In other words, the model must distinguish between domain-specific features and task-relevant features, to generalize well on the latter.

Notation

Let $X \times Y$ represent the joint feature space and the corresponding label space respectively. A source domain S and a target domain T are defined on $X \times Y$, with different distribution P_s and P_t i.e. $D^S \neq D^T$.

In the source domain, there are n_s labeled samples represented as $D^S = \{(x^s, y^s)\}_{i=1}^{n_s}$. In the target domain, there are n_t samples, which may or may not include labels, represented as $D^T = \{(x^t)\}_{j=1}^{n_t}$.

The primary goal of domain adaptation is to adapt the model trained on source domain (S) so it can generalize effectively on a related but different target domain (T), despite significant differences between the two domains.

2.3.3 Comparison of Adaptation Techniques

Domain adaptation, domain randomization, and domain generalization are commonly used for adaptation in machine learning. It is important to highlight how domain adaptation is better suited for medical imaging, particularly under limited data constraints.

Domain randomization introduces diverse variations during training to improve generalization; however, it has limitations in medical imaging compared to domain

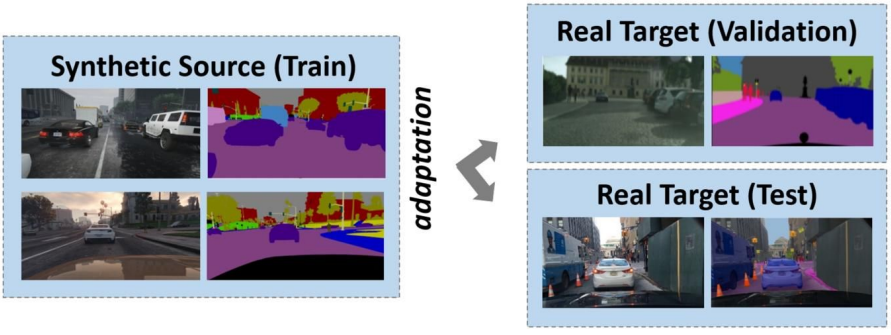


Figure 2.8: Adaptation from the synthetic source domain to the real target domain [9].

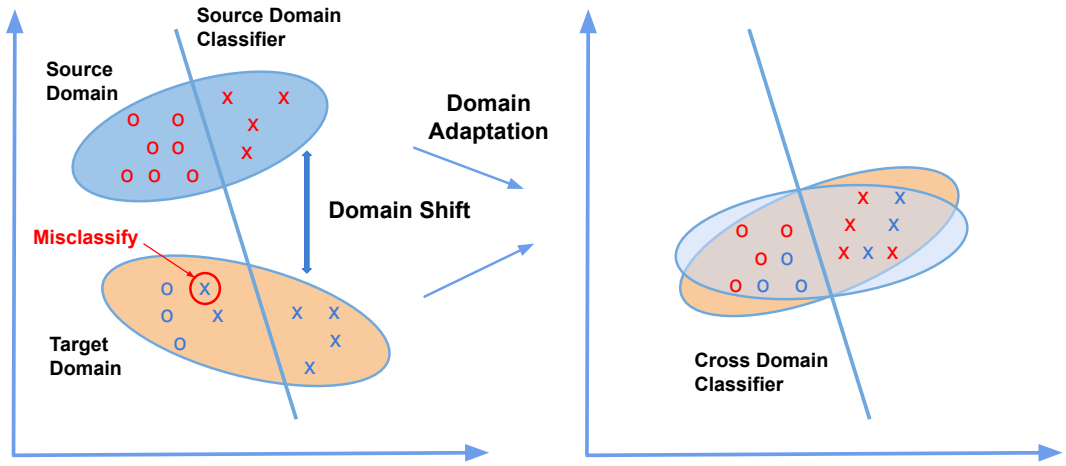


Figure 2.9: Overview of domain adaptation. Figure adapted from [2].

adaptation. Randomizing intensity, contrast, noise, and anatomical deformations may create unrealistic variations that do not reflect real pathological changes, leading to misleading feature learning [72]. Moreover, excessive randomization can distort critical diagnostic details, reducing model reliability. Additionally, in medical imaging, clinical validation and trust are crucial; artificial variations introduced by randomization may not accurately represent patient populations, limiting real-world applicability [73]. Therefore, while domain randomization enhances robustness, domain adaptation remains a more reliable approach for adaptation in medical imaging.

Domain generalization aims to train a model that generalizes well across unseen domains, without access to data from the target domain during training [74]. The model is trained on multiple source domains, and the goal is to make the model robust to domain shift, such that it can perform well when exposed to a completely new, unseen domain at test time. For domain generalization, the need for access to multiple source domains presents challenges in the medical field, where labeled data from multiple domains are often unavailable, limiting its applicability.

Therefore, compared to the aforementioned adaptation approaches, domain adaptation is better suited for medical imaging scenarios with limited data availability.

2.4 Literature Review

This section gives a general overview of the domain adaptation landscape for the medical imaging domain. It introduces the main work done within the topic and its evolution over time. The critical literature review relevant to each piece of research is presented in the corresponding chapters.

As outlined in Chapter 1, the research presented in this thesis aligns with the research paradigm shift in computer vision. It first focuses on the adaptation of convolutional neural networks and transformer-based approaches and evolves toward foundation model-based domain adaptation methods. The literature review is structured in a similar manner to reflect this progression.

2.4.1 Supervised Domain Adaptation

Supervised Domain Adaptation (SDA) is a subfield of domain adaptation in machine learning that leverages labeled data from a source domain to improve model performance on a target domain, where a limited amount of labeled data is available [75].

One of the widely used SDA approaches involves adapting the model trained on the source domain by fine-tuning the entire model for the target domain. The effects of fine-tuning have been assessed in the context of brain lesion segmentation, utilizing CNN models pre-trained on brain MRI scans [76]. The size of the target domain dataset and the selection of different network architectures have been shown to significantly influence adaptation performance. Inspired by this, several approaches have employed CNNs pre-trained on ImageNet [33] for various medical imaging analysis downstream tasks. Samala et al. [77] proposed a two-step approach that first pre-trains an AlexNet using ImageNet [33] and then fine-tunes it with the target domain. The target domain in this case is mass lesions for breast cancer classification. Following this [78] pre-trained a VGG network on ImageNet and then fine-tuned it using labeled MRI data for Alzheimer's disease (AD) classification. For chest X-ray image classification, ImageNet has been used to pre-train CNNs with

evaluation adaptation to chest X-ray target domain [79].

However, in various medical imaging modalities, the target domain datasets are often small. Fine-tuning the entire model with small datasets is prone to overfitting. Furthermore, in the medical imaging domain, multiple target domains exist for a single imaging modality (Section 1.1.3). Thus creating fine-tuned versions for each downstream task is challenging and not practical. Moreover, adapting the entire model imposes substantial computational constraints.

To address these challenges, a novel approach for parameter-efficient adaptation of CNNs is proposed for multi-target domain adaptation in medical imaging that is both accurate and computationally efficient which is outlined in detail in Chapter 4.

Another research direction for SDA involves employing deep learning models as fixed feature extractors, followed by the adaptation of these extracted features using shallower models. For instance, ResNet [80] has been effectively utilized as a fixed feature extractor for mammographic images [81]. Using the extracted features, three alignment-based for adaptation are used: Transfer Component Analysis (TCA) [82], Correlation Alignment (CORAL) [83, 84], and Balanced Distribution Adaptation (BDA) [85]. In another application, LeNet-5 is used as a fixed feature extractor for histological images across different domains, for the task of classification of epithelium and stroma [86]. To facilitate the alignment of extracted features for adaptation, the extracted features were projected into a lower-dimensional space using principal component analysis [87].

While leveraging deep learning models as fixed feature extractors is effective for classification tasks where annotated and pre-defined class labels are available, this approach does not offer the same benefits for complex pixel-level medical image segmentation tasks. It is further exacerbated due to domain shift (Section 1.1.1), especially when adapting models pre-trained on natural images for complex medical imaging tasks, as experimentally demonstrated in Section 3.5.2 and Chapter 5. Furthermore, medical imaging analysis often requires a fine-grained understanding of different regions within the image to accurately classify each of them. A few-shot

adaption approach is proposed for this purpose, which is discussed in Chapter 6. However, achieving such semantic labels in medical images is challenging and often impractical as outlined in Chapter 1.

2.4.2 Unsupervised Domain Adaptation

Unsupervised domain adaptation (UDA) aims to generalize large-scale models, pre-trained on the source domain to an unlabeled target domain, eliminating the need for costly data annotation [26, 88].

UDA techniques often focus on aligning the feature distributions of the source and target domains. It involves strategies such as minimizing the distance between the two distributions or utilizing adversarial training to make the features from both domains indistinguishable. Some commonly used distance metrics for measuring domain differences include maximum mean discrepancy (MMD) [89], correlation alignment (CORAL) [83, 84], contrastive domain discrepancy (CDD) [90], and Wasserstein distance [91].

However, medical images often contain complex, high-dimensional features with subtle distinctions between classes, such as differentiating between visually similar lesions. Furthermore, pathological structures lack a consistent anatomical shape or pattern, and in some cases, they may closely resemble normal structures, as demonstrated in Chapter 6 (Section 6.5.3). Additionally, medical images vary significantly due to differences in imaging equipment, protocols, and settings (e.g., MRI machines from different manufacturers (Figure 1.4)). In medical imaging, the domain shift goes beyond simple distributional changes as discussed in Chapter 1.

As a result, feature-level alignment methods often fail to capture fine-grained and localized distinctive features essential for accurate diagnosis, as they tend to focus primarily on matching global feature distributions [4]. This limitation can lead to reduced model performance in clinical applications, where high-level precision and sensitivity to subtle variations are essential for reliable decision-making.

Image-level alignment is also used for UDA in the medical imaging domain. Pri-

marily, deep generative models, such as generative adversarial network [92] are used for image-level alignment. Zhu et al. proposed a cycle-consistent GAN (CycleGAN) model that can translate one image domain into another without the demand for paired training samples [93]. A cycle consistency loss is used to measure the difference between the input image and the reconstructed image. In [94] CycleGAN-based UDA method is proposed for de-noising images. It learns a mapping between the source (i.e., high noise) and the target domain (i.e., low noise) on unpaired OCT images. CycleGAN's application has also been evaluated for brain tumor segmentation [95]. First, synthetic MRI images of tumor-bearing tissue are generated using a private simulation model. These images are then transformed into realistic MRIs using CycleGAN to augment the training dataset.

While models like CycleGAN enable translation between source and target domain (e.g., MRI to CT), they struggle to preserve fine-grained anatomical details essential for accurate diagnosis [96, 97]. The generative process can introduce subtle artifacts or alter critical features, leading to potential misinterpretations [98]. CycleGAN also encounters training instability challenges, such as model collapse, where the generator produces limited output diversity [99, 100]. In medical imaging, this can be particularly detrimental, as it may result in translated images that fail to capture the necessary variability to accurately represent diverse patient anatomies and pathological conditions.

Diffusion models are increasingly used in medical imaging for generating high-quality synthetic data and augmenting datasets. However, their effectiveness in domain adaptation is limited. They struggle to transfer knowledge across different imaging domains (e.g., MRI vs. CT) due to variations in acquisition settings, scanners, and patient demographics [101]. Models trained on one dataset may fail to generalize to another, leading to poor adaptation. Additionally, diffusion models can introduce artifacts or hallucinations, generating medically implausible structures that degrade model reliability, particularly in sensitive tasks like retinal imaging [102].

Disentangled representation involves training the model to separate, or "disentangle", the underlying features that potentially contribute to variation in data [103]. The model is trained to learn representations where each latent variable (or dimension of the learned representation) corresponds to a distinct, interpretable feature, such as shape, color, lighting, or even more abstract properties like facial expression or object pose. Disentangled representation learning embeds images from the source and target domains into two distinct spaces: a shared domain-invariant content space and a domain-specific style space. A cross-modality UDA method between MRI and CT images based on disentangled representation learning for liver segmentation is proposed in [104]. Zhao et al. [105] utilized disentangled features with a shared decoder to reconstruct the original data. In the new target domain, a private encoder trained is employed on the reconstruction objective. To ensure minimal mutual information between class-invariant and class-shared features, a mutual information minimization approach is applied [106].

In contrast to natural imaging, where domain-specific attributes (e.g., weather, color intensity) facilitate clearer disentanglement, disentangling features in the medical domain (especially for pathological structures) is challenging, as discussed in Chapter 6. Moreover, the features disentangled by the model may not correspond to clinically meaningful aspects of the disease, limiting their utility for diagnostic or prognostic purposes [107].

2.4.3 Semi-Supervised Domain Adaptation

Semi-supervised Domain Adaptation (SSDA) aims to enhance the performance of a model pre-trained on a labeled source domain when applied to a related but different target domain with only a few available labels. The target domain consists of a mix of labeled and unlabeled data, while the source domain contains a substantial amount of labeled data [108].

Adversarial learning frameworks have been widely adopted for SSDA in the medical imaging domain. These methods utilize generative adversarial networks (GANs)

to align the feature distributions of the source and target domains [109]. For instance, Chen et al. [110] proposed a semi-supervised GAN framework that effectively reduces domain shift by generating synthetic samples for the target domain, improving the segmentation performance of brain tumors in MRI scans. Madani et al. [111] proposed a semi-supervised GAN-based SSDA framework for chest X-ray image classification. The discriminator performs three-category classification (i.e., normal, disease, or generated image). During training, unlabeled target data can be classified as any of those three classes but can contribute to loss computation when they are classified as generated images. Through this way, both labeled and unlabeled data can be incorporated into a semi-supervised manner. An SSDA framework is proposed for electron microscopy image segmentation [112]. Specifically, a "YNet" with one feature encoder and two decoders is proposed. One decoder is used for segmentation, while a reconstruction decoder is designed to reconstruct images from both the source and target domains. The network is initially trained in an unsupervised manner. Then, the reconstruction decoder is discarded, and the whole network is fine-tuned with labeled target samples to make the model adapt to the target domain.

However, the application of GANs in the medical imaging domain is associated with several challenges, which are discussed in detail in the previous Section 2.4.2

Self-training is another form of SSDA. Pseudo-labeling is one of the techniques for self-training, where a model iteratively re-trains itself using its own predictions on unlabeled data [113, 43]. In the context of brain tumor segmentation, Zhou et al. utilized pseudo-labels to enhance segmentation performance by iteratively refining model predictions based on high-confidence outputs [114]. Lee et al. introduced the concept of pseudo-labeling in a semi-supervised setup, demonstrating its effectiveness in various domains, including medical imaging [115]. A method called self-training with noisy a student is proposed, which enhances model accuracy in various medical imaging tasks by generating pseudo-labels for unlabeled data, demonstrating superior performance in lung disease classification tasks [113]. The

term "noisy student" refers to a specific semi-supervised learning approach where a student model is trained using pseudo-labels generated by a teacher model, often with added noise to improve generalization.

Pseudo-labeling relies on the model predictions to generate labels for unlabeled data. If the model is initially inaccurate, it may assign incorrect labels, propagating errors, and degrading overall performance. The performance of pseudo-labeling can diminish when the model encounters a target domain that is significantly different from the source domain, especially if there is high variability and noise in medical images.

Consistency regularization (CR) is also a form of self-training [43], it encourages the model to produce similar predictions for perturbed versions of the same input data. This technique has been effectively used in medical imaging tasks. CR is used in lung nodule classification. The perturbed version of the image is created using augmentation techniques to create variations of the input data [116]. The model was trained to maintain consistent predictions across these variations, leading to improved robustness against domain shift.

The effectiveness of CR heavily depends on the perturbed images. For instance, if augmentations are used for perturbation, the poorly chosen augmentations can lead to misleading results, as the model might learn to be consistent over irrelevant transformations rather than meaningful variations. If not carefully managed, the model may overfit to the augmented versions of the training data rather than generalizing to real-world data, limiting its practical applicability.

Ensemble learning leverages multiple models to generate more stable pseudo labels. It has been adapted in SSDA, where a baseline CNN, referred to as the student network is utilized, which processes labeled samples from the source domain and makes predictions after being trained with a segmentation loss [4]. A second network, known as the teacher network, generates predictions based solely on unlabeled samples from the target domain. The teacher network is updated using an exponential moving average of the student network's weights, implementing a tem-

poral ensemble strategy. During training, the domain shift is minimized through a consistency loss that compares the predictions from both networks. This method is evaluated on the SCGM challenge dataset [117], which consists of multi-center, multi-vendor spinal cord anatomical MR images from healthy subjects. A similar structure is employed based on SSDA for brain tumor segmentation. In addition to the consistency loss that measures the discrepancy between the teacher and student network predictions, they incorporate an adversarial loss to enhance adaptation performance [118]. Their approach is validated through experiments on the BraTS dataset [119].

However, ensemble approaches such as mean teacher, require careful management of model and weights, which can complicate the training process and require additional resources. Moreover, averaging weights over time can lead to slower convergence rates, prolonging training times.

2.4.4 Adaptation of Foundation Models

Foundation models have revolutionized the field of computer vision by introducing large-scale, pre-trained models capable of performing a wide range of tasks. Unlike traditional task-specific models, foundation models are trained on vast, diverse datasets, enabling them to learn general representations that can be adapted to new tasks and domains with significantly reduced data and effort [15, 120, 121].

Since the research work presented in this thesis focuses on medical imaging analysis, this section introduces the foundation models that have been widely used in this domain. Primarily, the foundation model- segment anything model (SAM), has been used for two of the proposed approaches (Chapter 5 and Chapter 6). The detailed related work on SAM adaptation for medical imaging is outlined in Chapter 5 (Section 5.2). Here the related work has been presented from the perspective of different foundation models.

Despite the significant advancements in foundation models within NLP and computer vision, their impact on medical imaging has been limited. One of the primary

reasons for this is the difficulty in accessing and aggregating large-scale datasets comparable to those used in other domains. Medical imaging data are often fragmented across different institutions and subject to stringent privacy regulations, making it challenging to amass the extensive datasets required for training large foundation models. Since its release, the segment anything model (SAM) has gained significant attention [13]. It incorporates a ViT-based encoder and a lightweight transformer-based decoder, showcasing remarkable zero-shot segmentation capabilities. The architecture details are discussed in preliminaries Section 5.3.1. Several studies have conducted assessments of SAM's performance in medical imaging tasks using default architectural design [122, 56, 123, 124, 125, 126]. Furthermore, researchers have explored slight modifications of the SAM model to tailor it specifically for medical imaging applications [127, 128] or fine-tuned it for specific datasets [129, 130, 55].

While these architecture-based modifications and dataset specific fine-tuning methods mentioned above have proven effective. Since SAM is a foundation model, adapting it first to a source domain and then to downstream target domains introduces significant computational overhead. To address this challenge, several methods have employed parameter-efficient fine-tuning to adapt SAM, utilizing lightweight adapters for enhanced efficiency. However, it is experimentally demonstrated in Chapter 5, that the effectiveness of PEFT is highly dataset specific (Section 4.5.6). Furthermore, most studies have evaluated SAM's effectiveness in segmenting every region within an image. While this appears to be enticing, in the medical domain it has limited practical application in real-world scenarios (Section 5.1). Additionally, as a foundation model, SAM introduces significant computational overhead, which cannot be overlooked.

To address these challenges, a training-free/fine-tuning free framework-SaLIP is proposed to adapt SAM specifically for the medical imaging domain, as detailed in Chapter 5. SaLIP does not require source domain access, instead, it is fully adapted to the target domain completely at test time, in a training-free manner, allowing SAM to perform effectively in new medical imaging tasks with minimal

computational overhead.

Segment everything everywhere all at once model (SEEM) [131] is another foundational model for segmentation, which has shown excellent performance across numerous benchmarks. However, there is currently limited research on its application in the medical domain. Despite an extensive search, we found no relevant references on its use in medical contexts, making it a compelling candidate for the medical imaging domain.

The contrastive learning image pre-training family (CLIP) of models is pre-trained on image-text pairs [132]. MedCLIP is the adaptation of CLIP for medical imaging. It is trained on a massive dataset of unpaired medical images and text descriptions. It has been evaluated on image classification, retrieval, and zero-shot learning [133]. MedCLIP is predominantly pre-trained on X-ray datasets, including MIMIC-CXR [47], CheXpert [48], COVID [134], and RSNA Pneumonia [135]. Its evaluation is primarily conducted on chest X-ray datasets, with the learned knowledge transferred through transfer learning for specific downstream applications. In particular, MedCLIP's transferability is assessed on downstream supervised tasks and image-level classification tasks. In contrast, methods proposed in the subsequent chapters address a more challenging domain adaptation problem by bridging the gap between general-purpose models and specialized medical tasks. Specifically, the broader applicability is enabled across diverse medical imaging domains through unsupervised adaptation for multi-target domains (Chapter 4), test-time adaptation of foundation models (Chapter 5), and few-shot adaptation for fine-grained medical imaging (Chapter 6).

A self-supervised contrastive learning method to detect multiple pathologies in chest X-rays [136] by leveraging unlabeled data to learn discriminative feature representations, thereby enhancing pathology classification accuracy. This approach leveraged image-text pairings and zero-shot classification techniques to achieve radiologist level performance, without explicit training on the target pathologies. BLIP is trained on large datasets consisting of biomedical images (such as radiology, histo-

pathology, and microscopy images) paired with relevant textual annotations (like clinical notes, image captions, and descriptions) [137]. It employs contrastive learning techniques to maximize the similarity between matching image-text pairs while minimizing it for non-matching pairs. A multi-modal global-local representation learning framework for medical images by leveraging radiology reports in [138]. Specifically, an attention based framework for learning global and local representations by contrasting image sub-regions and words in the paired report.

BioMedCLIP [139], while a specialized model trained on biomedical data, it is designed for tasks where the training and testing data come from similar biomedical domains. Additionally, BioMedCLIP [139] is pre-trained on radiology reports, making it skewed toward this type of data and is limited to certain medical imaging types. BioMedCLIP's [139] textual encoder is trained on existing biomedical literature and captions. While this is useful for some medical tasks, it may not be flexible in handling new terminologies, emerging diseases, or institution-specific vocabularies. In contrast, the approaches proposed in the subsequent chapters of this thesis are designed to be domain-agnostic.(Chapter 4, 5 and 6), which is key when trying to address the robustness of medical image recognition in varied scenarios, such as imaging from different hospitals, devices, or imaging modalities (e.g., MRI, CT, X-ray).

The effectiveness of contrastive learning in the medical imaging domain has primarily been evaluated on image-level/global tasks. To adapt CLIP for medical applications, the approaches mentioned above, have used medical image-text pairs to train CLIP. However, these datasets are largely skewed toward chest X-rays and the radiology domain [140]. More importantly, in medical imaging analysis, particularly for disease diagnosis and prognosis, a fine-grained, region-level understanding is essential.

Furthermore, acquiring biomedical text and images from web sources for fine-grained, subtle anatomical structures and pathologies presents considerable challenges. When we opted for the recent research trends of leveraging large language

models like GPT [52] to generate textual prompts, it worked well for medical organ segmentation (Chapter 5). However, it showed limited effectiveness for fine-grained recognition of anatomical structures based on spatial location and more complex pathologies, as experimentally validated in Chapter 6.

DINOv2 (Distillation with No Labels v2) is a vision-based foundation model developed by Meta [53]. It benefits from training on a large-scale curated dataset, resulting in representations that capture the semantic meaning of images remarkably well. DINOv2 is particularly notable for its ability to generate high-quality, dense feature maps that improve downstream performance on tasks requiring detailed spatial understanding. DINOv2 has been used for classification in several medical tasks i.e. diabetic retinopathy, detecting lesions, and skin lesions and abnormalities in lungs [141].

The effectiveness of DINOv2 has been primarily evaluated for image-level classification, whereas medical imaging analysis often requires pixel-level classification. Obtaining region-level labels required for fine-grained tasks in medical tasks remains extremely challenging and impractical (Section 6.1). When we evaluated DINOv2 for more complex fine-grained medical tasks, specifically the classification of lung regions based on spatial localization and region-based pathological structures did not yield favorable results for these fine-grained tasks (Chapter 6). This outcome suggests limitations in DINOv2's ability to handle the detailed feature differentiation necessary for medical imaging applications.

2.5 Summary

This chapter provides an introduction to the fundamental principles of deep learning and their applications to computer vision applications in Section 2.1. A brief overview of deep learning-based approaches for medical imaging analysis is outlined in Section 2.2. Section 2.3 presents the widely used approach to overcome domain shift and the challenges in the medical imaging domain. Section 2.4 presents a comprehensive literature review for domain adaptation and its evolution over time

with an emphasis on medical imaging segmentation.

In subsequent chapters, the results achieved through the course of this research are described and discussed. Furthermore, each chapter includes an extended literature review focused on specific sub-topics, accompanied by comprehensive descriptions of the employed methodologies and experiments conducted.

Chapter 3

Domain Shift in Medical Imaging: Need for Domain Adaptation

This chapter provides a comprehensive overview of findings from our participation in the STOIC 2021 COVID-19 AI Challenge [20]. The objective of this challenge was to automatically predict COVID-19 severity from Computed Tomography (CT) scans of COVID-19 suspects and patients. The experiments in this chapter are conducted on the STOIC dataset from STOIC 2021 COVID-19 AI Challenge [20]. It addresses Research Question 1 (RQ1): "What are the key challenges and limitations of supervised adaptation approaches when applied to diverse medical imaging datasets? Specifically, how do domain shifts and data scarcity affect the generalization of neural networks for medical imaging tasks?"

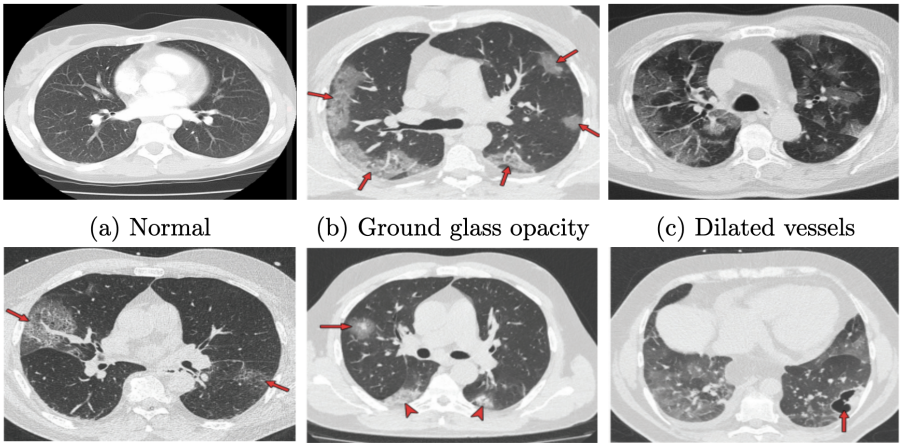
Importantly, participation in this challenge introduced us to the critical issue of domain shift in the medical imaging domain (discussed in Section 1.1.1). Participation in the STOIC challenge helped us to establish a clear research direction for this thesis, i.e., the need for robust domain adaptation techniques to address the challenges posed by domain shift and limited data availability in the medical domain. Domain adaptation is a specific type of transfer learning that aims to adapt a model trained on one domain to perform well on another domain. In this context, the task remains the same but the data distribution changes [70, 71] (Section 2.3.2).

Our proposed approach achieved the 4^{th} position in the challenge. The work discussed in this chapter has been published in the 25^{th} IEEE Conference on Irish Machine Vision and Image Processing Conference (IMVIP), 2023 [142]. The code to replicate experiments and results is publicly available at:
https://github.com/aleemsidra/STOIC2021-COVID-19-AI-Challenge.

Section 3.1 outlines our motivation behind the research on COVID-19 severity prediction and gives an overview of the STOIC 2021 – COVID-19 AI Challenge [20]. Section 3.2 reviews the existing methods for COVID-19 prediction. Section 3.3 outlines the proposed ensemble approach and its architectural design. Section 3.4 provides an overview of the STOIC dataset and the experimental setup. Section 3.5 presents a comprehensive in-depth analysis of all the experiments and a comparative analysis of the proposed approach with other leading methods. Finally, Section 3.6, summarizes the findings from the challenge and highlights how participation in the STOIC challenge contributed to help shape the research direction for this thesis.

3.1 Introduction

The automated medical imaging analysis research domain has undergone a transformative evolution, largely driven by advancements in deep learning [143]. This progress not only stems from technical and algorithmic innovations but also from the availability of high-quality labeled datasets [40, 144], which have facilitated the exploration of new architectures. However, medical imaging analysis poses unique challenges to the generalization of deep learning models due to several factors such as diverse imaging modalities that can result in domain shift [145, 146, 147] (Section 1.1.3). For instance, magnetic resonance imaging (MRI) provides excellent contrast for soft tissue, while CT scans offer superior spatial resolution [148, 149], as shown in Figure 1.3 (Cross Modality). Therefore, the choice of modality is critical in achieving accurate results for specific clinical applications. The choice of an inappropriate imaging modality can result in diagnostic inaccuracies, hinder model generalization, and degrade the performance of deep learning models trained on such



(a) Normal (b) Ground glass opacity (c) Dilated vessels

(d) Paving pattern (e) Consolidation (f) Cavitating lesions

Figure 3.1: Chest CT scans illustrating lung abnormalities associated with various stages of COVID-19 progression.

data, ultimately leading to suboptimal outcomes.

The COVID-19 pandemic highlighted the critical role of medical imaging in diagnosis and emphasized the need for efficient automated diagnostic systems. Chest X-rays have been more widely used in the prognosis and monitoring of COVID-19 patients due to their cost-effectiveness and availability [150, 151]. However, chest CT scans are more accurate in detecting lung abnormalities associated with COVID-19 [152, 153, 154]. CT has proven to be an invaluable tool for evaluating lung conditions, aiding in the prediction and severity assessment of COVID-19, and detecting related complications.

Several chest CT findings, such as ground-glass opacities, dilated vessels, consolidations, paving patterns, and cavitating lesions [154], have been directly associated with COVID-19. In severe cases of COVID-19, chest CT scans often show ground glass opacities in the peripheral lung regions and dilated pulmonary vessels. As COVID-19 progresses, consolidations and cavitating lesions tend to increase [154]. These abnormalities are shown in Figure 3.1. Furthermore, previous studies have focused primarily on COVID-19 positivity prediction or relevant features extraction [155, 156, 157, 158, 159, 160].

3.1.1 Motivation

Given the advanced diagnostic capabilities and comprehensive clinical insights provided by CT imaging, this research utilizes the CT modality to analyze COVID-19 severity. While COVID-19 positivity prediction and feature extraction methods contribute to diagnosing the disease, it is essential, to emphasize that these approaches alone are insufficient to aid treatment decisions or accurately predicting disease severity. Proper assessment of COVID-19 severity is vital for effective clinical management, as it directly affects treatment strategies and patient outcomes. Additionally, the reliance on private datasets in COVID-19 research poses a significant challenge, restricting data accessibility and impeding the reproducibility of proposed methods. These challenges motivated our participation in the STOIC 2021 COVID-19 AI challenge [20].

3.1.2 STOIC 2021- COVID-19 AI Challenge: Overview

The STOIC 2021 COVID-19 AI Challenge focused on developing fully automated methods to distinguish between severe and non-severe COVID-19 cases, with "severe" defined as death or intubation within one month (AUC computed with COVID-19 positive patients only, primary metric). The challenge was organized utilizing the data from the STOIC dataset [161] (Section 3.4.1).

The STOIC 2021 challenge consisted of the following phases:

1. **Qualification Phase:** participating teams developed algorithms using the STOIC public dataset, with evaluation conducted by submitting docker containers [1] of the proposed methods to the "Qualification" leaderboard. This leaderboard ranked submissions based on performance on a subset of approximately 200 test scans. The participating teams were limited to one submission per week on the leaderboard.

2. **Qualification (Last Submission):** determined which teams were eligible

[1] A docker container is a lightweight, standalone, and executable software package that includes everything needed to run an application [162], Accessed: [08.02.2025].

for the final phase. Submissions for this leaderboard were evaluated on a separate test set of approximately 800 scans, distinct from the set used for the "Qualification" leaderboard.

3. **Final Phase:** selected finalists could submit a docker container [2] to train an improved model using the full train set of over 9,000 CT scans. This set included 2,000 scans from the STOIC public dataset, 200 scans from the "Qualification" leaderboard, approximately 800 scans from the "Qualification (Last Submission)" leaderboard, and an additional 6,000+ scans from the STOIC database [161].

3.2 Related Work

Several deep learning techniques have proven effective in medical diagnosis and analysis while leveraging CT scan data. As discussed in Section 3.1, the majority of the existing research focused on COVID-19 prediction and feature extraction. This section provides an overview of the work conducted in these areas.

A fully automated method for detecting COVID-19 from chest CT scans is proposed in [163]. The method first applies an image processing algorithm to exclude CT images where the lung interior is not clearly visible. It then utilizes a novel architecture based on a feature pyramid network, designed for classification tasks [164]. A new dataset containing 48,260 CT scan images from 282 healthy subjects and 15,589 images from 95 COVID-19 infected subjects is introduced and evaluated using a 10-fold cross-validation approach. However, the proposed method has not been clinically validated on a large, diverse population. Since the dataset was collected from a single center, its generalizability to other populations is uncertain. The impact of domain shift, arising from variations in acquisition sites and protocols, has not been validated (Section 1.1.3). COVIDCTNet, a multi-step deep learning model, is proposed for diagnosing COVID-19 using a small cohort of CT images [165]. The

[2] A docker container is a lightweight, standalone, and executable software package that includes everything needed to run an application [162], Accessed: [08.02.2025].

model detects COVID-19 from CT scans, achieving a sensitivity of 98.7% and specificity of 96.1% on a dataset of 349 CT scans from 105 COVID-19 positive patients and 68 healthy individuals. However, the dataset used in this study is not publicly available, limiting the model's reproducibility. Moreover, the evaluation was conducted on a small dataset, and further validation on larger, more diverse datasets is necessary.

A 3D version of the regularized network (RegNet) has been employed for diagnosing COVID-19 using chest CT images [166]. RegNet is based on the regularized network architecture [167] and is trained on a large dataset consisting of COVID-19 positive and negative cases. While the authors have implemented sample-efficient techniques, there remains a risk of overfitting with deep learning models. Moreover, the proposed method is limited by the dataset, which only includes COVID-19 positive and negative cases and lacks data for other lung diseases that may coexist. As a result, the model's performance could decline in the presence of such conditions. Additionally, the model was trained and evaluated on a limited dataset from a specific geographic region, potentially restricting its generalization to other populations (Section 1.1.4). Hossein et al. [168] propose a deep learning model leveraging self-attention and multi-scale encoder-decoder networks, demonstrating strong performance in medical imaging tasks like lung nodule detection, retinal vessel segmentation, and brain tumor segmentation. However, its effectiveness on 3D CT scans, a modality more suitable for detailed lung analysis (as discussed in Section 3.1) remains unexplored.

A contrastive cross-site learning framework has been proposed to address the domain shift issue by leveraging information from multiple sources. This approach uses a contrastive loss function to help the model learn shared representations across different datasets, facilitating better generalization across domains [169]. The method was evaluated on two publicly available COVID-19 CT datasets: COVID-CT and SARS-COV-2 CT-Scan [170], achieving an impressive accuracy of 97.3% on the combined dataset. However, the dataset was imbalanced, with a relatively small number

of COVID-19 positive cases compared to negative cases. This imbalance could introduce bias in the model, hindering its ability to effectively learn the distinguishing features of COVID-19 from CT scans.

3.3 Methodology

This section presents our proposed approach for COVID-19 severity prediction. It begins with a detailed description of the pre-processing steps our method applied to the STOIC dataset, as outlined in Section 3.3.1. Following this, Section 3.3.2 provides an in-depth discussion of the architectural design of our proposed ensemble method.

3.3.1 Pre-Processing

The CT scans from the STOIC dataset have a spatial resolution of 512×512 with a depth ranging from 128 to 600 slices. The radiodensity within this volume is measured using Hounsfield Units (HU), ranging from -1024 HU to 3071 HU, and stored as 12-bit numbers [171]. However, directly scaling HU values to the range of 0 to 1 can lead to low-contrast CT images, making it challenging to identify and extract relevant features associated with COVID-19.

To overcome this challenge, a windowing function is applied to adjust the brightness and contrast of CT images to enhance the visibility of specific structures or tissues [172]. Windowing plays a critical role in accurate CT scan interpretation by optimizing the visibility of specific anatomical features and pathological details as shown in Figure 3.2. This refinement ultimately helps the models to make more precise predictions, detect subtle patterns and increase their robustness in distinguishing features relevant to COVID-19 severity. Windowing has two components: window width and window level. The window width controls the range of signal intensities displayed in the CT image. The window level sets the center of the window width range and adjusts the midpoint of the grayscale display [173]. By changing

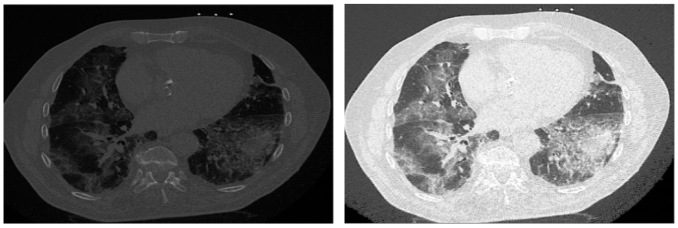


Figure 3.2: Comparative Analysis: a) Scaling b) Windowing.

the window level, specific tissues or structures can be highlighted, making them more distinguishable from the background to obtain high-contrast images from CT volumes. The upper and lower gray level values are calculated using window width and window level as:

$$\begin{aligned} x &= WL + \frac{WW}{2} \\ y &= WL - \frac{WW}{2} \end{aligned} \tag{3.1}$$

where x is the lower gray level value, y is the upper gray level value, WW is window width and WL is window level. To enhance the visibility of lung structures in CT scans, a lung window with a window width (WW) of 1500 HU and a window level (WL) of -600 HU is employed. This configuration optimizes the contrast and brightness of the images, facilitating the detection of abnormalities such as ground-glass opacities and consolidations.

Figure 3.2 demonstrates the effectiveness of windowing as compared to direct scaling, highlighting its superior ability to enhance contrast differences, which can be particularly useful for COVID-19 severe prediction. The windowing clearly enhanced the visibility of key CT features, which could significantly improve the diagnostic accuracy of models in detecting lung pathologies related to COVID-19.

A CT volume consists of numerous slices, but not all contain meaningful diagnostic features. Utilizing all slices can increase training time and computational load while potentially reducing accuracy due to the inclusion of irrelevant slices, such as those that are completely black or lack features relevant to COVID-19 severity prediction.

To address this issue, a sampling function is employed to retain only those slices from the CT volume where the lung region is clearly visible. It is essential as these

slices have essential features for COVID-19 severity assessment. Retaining only the most relevant slices improves analysis accuracy while minimizing the impact of irrelevant data or artifacts. Section 3.5.1 provides a comparative analysis and evaluation of different sampling functions applied to standardize the STOIC dataset, highlighting their overall impact on improving model performance.

Among the evaluated sampling strategies (Section 3.5.1), the centered sampling– which selects 32 slices from the middle range of the CT scan after removing the first 12% and the last 6% of slices, outperformed the others. The exclusion range was determined via data exploration of a few CT scans in the STOIC dataset [161].

The public STOIC dataset consists of only 2,000 CT scans. When trained on such small data, the neural networks are prone to overfitting (Section 1.1.4). To address this issue, CT volumes were converted into 2D slices. This transformation significantly increases the dataset size. As a result, the training set became more diverse, which enabled the model to better capture underlying patterns, generalize well, and make more reliable predictions despite the data scarcity (experimentally demonstrated in Section 3.5.2).

3.3.2 Ensemble of Neural Networks with Test Time Augmentations

Ensemble models are machine learning methods that combine the predictions of multiple base models to produce more accurate predictions. The idea behind ensemble methods is that when combined effectively, a group of diverse models can outperform individual models. These are particularly effective when dealing with limited data as these models can capture different data features and reduce overfitting, a common challenge with small datasets [174].

To overcome the risk of overfitting and improve the generalization, the training data is first split into five random folds. Each fold is then processed through an ensemble of ResNet18 [80] and MobileNetV3 [175] for COVID-19 severity prediction. These models were selected to create an ensemble is based on experimental

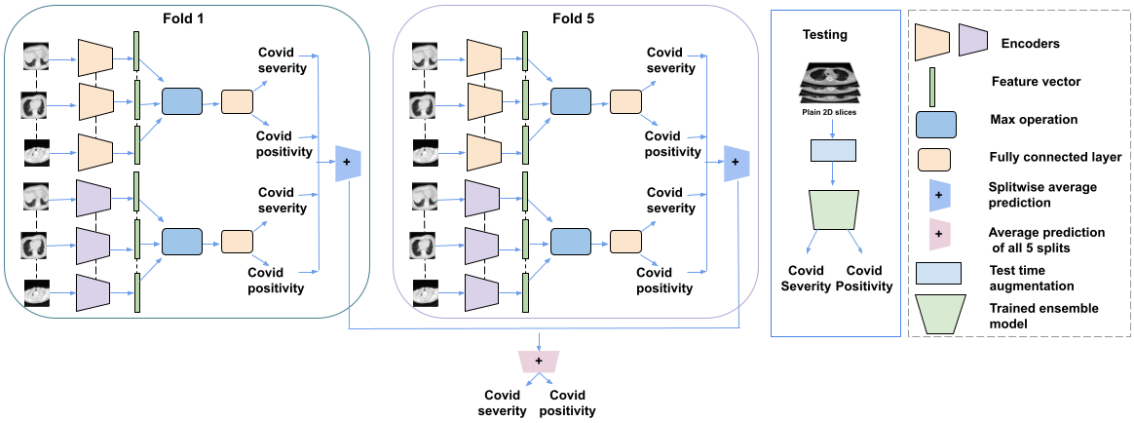


Figure 3.3: The schematic overview of our proposed ensemble approach for COVID-19 severity prediction.

evaluation, as detailed in Section 3.5.2. Figure 3.3 illustrates the architecture of the proposed ensemble approach for COVID-19 severity prediction.

In each data fold, every pre-processed slice (as detailed in Section 3.3.1) first undergoes augmentation (Section 3.4.2) and is then processed by the ensemble of ResNet18 [80] and MobileNetV3 [175] to generate feature vectors as:

$$\begin{aligned} z_i^{\text{ResNet}} &= f^{\text{ResNet}}(S_i) \\ z_i^{\text{MobileNet}} &= f^{\text{MobileNet}}(S_i) \end{aligned} \tag{3.2}$$

where S_i is the pre-processed slice, $f(S_i)$ is the feature extraction function, z_i^{ResNet} and $z_i^{\text{MobileNet}}$ are the features extracted from each network respectively.

As detailed in Section 3.3.1, the proposed approach processes CT volumes as individual 2D slices. However, our ultimate goal is to assess the overall COVID-19 severity based on the complete CT volume (evaluated using AUC). To achieve this, the technique of selecting the maximally activated features across all slices of a 3D volume is employed. This approach captures the most prominent and informative features from the entire volume, while also reducing data complexity [176]. This process involves evaluating the feature vectors generated for each slice (Eq. 3.2: z_i^{ResNet}, $z_i^{\text{MobileNet}}$) and identifying the features with the highest activation values across slice dimension. These features typically correspond to the most significant patterns or anomalies related to the condition under study—in this case, COVID-19

severity. The maximally activated features along the slice dimension of the feature vectors are calculated as:

$$\begin{aligned} z^{\max}_{j\text{ResNet18}} &= \max_{i=1}^{32} \{z_{i,\text{ResNet18}}\} \\ z^{\max}_{j\text{MobileNet}} &= \max_{i=1}^{32} \{z_{i,\text{MobileNet}}\} \end{aligned} \tag{3.3}$$

where max refers to the function used to evaluate the feature vectors of all slices within a CT volume. It selects the maximally activated feature across the slice dimension. $z^{\max}_{j\text{ResNet18}}$ and $z^{\max}_{j\text{MobileNet}}$ represent the maximally activated feature maps extracted from the ResNet18 and MobileNet models, respectively.

These features are then passed through a fully connected layer to predict the COVID severity $\widehat{x_s}$ and COVID positivity $\widehat{x_p}$ labels as shown in Figure 3.3. For model ensembling, at each fold, the predicted probability from both the ResNet18 and MobileNet models are averaged to generate a combined prediction as shown in Figure 3.3. This procedure is repeated across all five data folds, and the resulting averaged probabilities from each ensemble are further averaged to obtain the final predictions for COVID severity $\widehat{y_s}$ and $\widehat{y_p}$.

During inference, our proposed approach leverages test-time augmentations (TTA). The primary goal of TTA is to improve model performance by helping the model generalize better to unseen variations of the test data. By augmenting the input image in different ways (e.g., rotating, flipping, cropping– Section 3.4.2), the model becomes less sensitive to the exact position, orientation, or scale of the objects in the image and it makes the model robust to data variability [177]. The augmented images generated through TTA are then fed into our ensemble model. The predictions of the augmented versions are initially combined using simple averaging. Finally, the outputs from the ensemble model are aggregated to produce the final labels for COVID-19 severity and positivity, as illustrated in Figure 3.3.

3.4 Experimental Framework

3.4.1 Dataset

The STOIC 2021 dataset [161] comprises of chest CT scans from 10,735 subjects. It was acquired during the first wave of the COVID-19 pandemic from multiple hospitals in France, from March to April 2020. For STOIC 2021 COVID-19 AI challenge, it has been divided randomly into a publicly available training set (2,000 subjects) released under the CC BY-NC 4.0 license. The final evaluation of the algorithm is conducted by training on a private training set of over 7,000 subjects, followed by testing on a private test set of approximately 1,000 subjects (challenge phases are discussed in Section 3.1.2).

The CT scans are in .MHA format and have a resolution of 512×512. The metadata includes each subject's age (ranging from 35 to 85 years) and gender information (distribution: 57.4% male, 42.6% female). RT-PCR results serve as the ground truth for COVID-19 infection status, while one-month outcomes (death or intubation) indicate severity. The primary goal of this challenge is to predict the severity of COVID-19 from chest CT scans. The STOIC public dataset is highly imbalanced, with only 301 out of 2000 subjects with severe cases.

3.4.2 Implementation details

To develop an effective approach for COVID-19 severity prediction, a variety of convolutional neural networks (CNNs) were evaluated, including ResNet-18 [80], MobileNetV3 [175], 3D U-Net, ConvNeXT Tiny [178], and a custom CNN model with various configurations (Section 3.5.2). The STOIC public dataset was divided into splits of 80:10:10 for training, validation, and testing, respectively. For all the experiments we trained the models for 100 epochs using a batch size of 16 with a learning rate of 5×10^{-4}, a learning rate decay of 0.5 every 40 epochs with StepLR. To prevent model bias toward the majority class, a weighted random sampler was used.

The models were optimized using the binary cross-entropy loss with logits criterion. The primary evaluation metric was the area under the curve (AUC) for COVID-19 severity, while the AUC for COVID-19 positivity was used as a secondary evaluation metric, as defined by the STOIC challenge organizers.

Our findings indicate that data augmentations played a significant role in enhancing severity prediction during both training and testing. The values for different hyperparameters are determined based on experimental evaluation. The following augmentations have been used:

- **Flip:** randomly flips the image horizontally with a probability of 50%.
- **Random Crop:** the original image is randomly cropped, which makes the model more robust to different object positions and sizes within the image. The image is cropped from the original dimension of 512×512 to 224×224.
- **Random Gamma:** the original image brightness is adjusted by applying random gamma values. It helps in simulating diverse environmental lighting conditions. The gamma value is randomly selected between 0.8 and 1.2.
- **Median Blur:** each pixel's value is replaced with the median value of the neighboring pixels within a specified kernel size, effectively reducing noise while preserving edges. The kernel size is set to 5.
- **Color Jitter:** applies random changes to the brightness, contrast, saturation, and hue of an image. Brightness and contrast are adjusted by up to 50%. Saturation is modified by up to 40% and hue is kept unchanged.
- **Safe Rotate:** the original image is rotated within a specified range while ensuring that the content remains within the frame and is not distorted. The rotation limit is set to 30 i.e. the image is randomly rotated within a range of -30 to +30 degrees. The rotation probability is set to 70%.
- **mixup:** create new training examples by mixing or interpolating pairs of images (or other types of data) and their corresponding labels [179]. It uses a

random mixing coefficient derived from a beta distribution with a parameter α. For all the models, α was set to 0.8, but for the ConvNeXT Tiny model, an α value of 0.3 was found to be more optimal. *mixup* helps to improve the model's generalization by making it less sensitive to specific examples and encouraging it to focus on broader patterns in the data.

- **Test time Augmentation (TTA)**: For test-time data augmentation (TTA) center cropping, corner cropping, and safe rotation are also used.

All experiments were conducted using Python 3.8.17 and the open-source library PyTorch 2.0.1. The experiments were conducted on a desktop system with the Ubuntu 20.04.6 LTS operating system, CUDA 11.6, an NVIDIA GeForce RTX 3090 GPU, and 64 GB of RAM.

3.5 Results and Analysis

3.5.1 Sampling function

The proposed approach employs sampling to retain slices containing lung structures, to preserve meaningful features for COVID-19 severity prediction (Section 3.3.1). To select the optimal sampling approach, various sampling methods were tested, as detailed below:

- **Uniform sampling with one window (US-1W):** it uniformly samples 32 slices from the CT volume with a WW of 1500 HU and a WL of -600 HU.

- **Centered sampling (CS-1W):** it samples 32 slices from the middle of the CT volume, excluding the first 12% and the last 6% of slices. The window width (WW) is set to 1500 HU and the window level (WL) to -600 HU. The exclusion is based on our data analysis of several CT scans from the STOIC dataset, aimed at removing regions corresponding to the abdomen and lower jaw. Figure 3.4 illustrates a few of the slices retained using CS-1W.

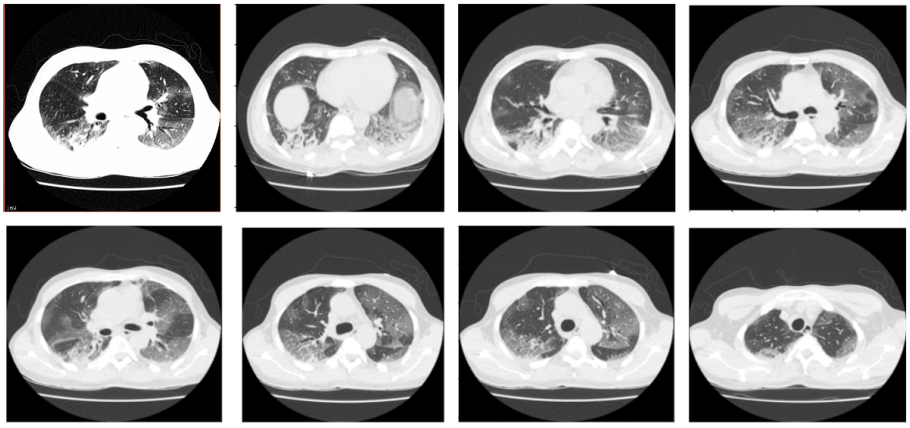


Figure 3.4: Centered Sampling with One Window (CS-1W): Example of slices retained with CS-1W.

- **Centered sampling with three windows (CS-3W):** Different windows highlight distinct anatomical structures/features. To leverage the strengths of different windows, CS-3W uses multiple windows and then combines these into a single, more informative image as shown in Figure 3.5. The three windows used are: Mediastinum windows (WW:-380, WL:1200), and custom windows (WW:900, WL:-112), and used as three channels of a slice.

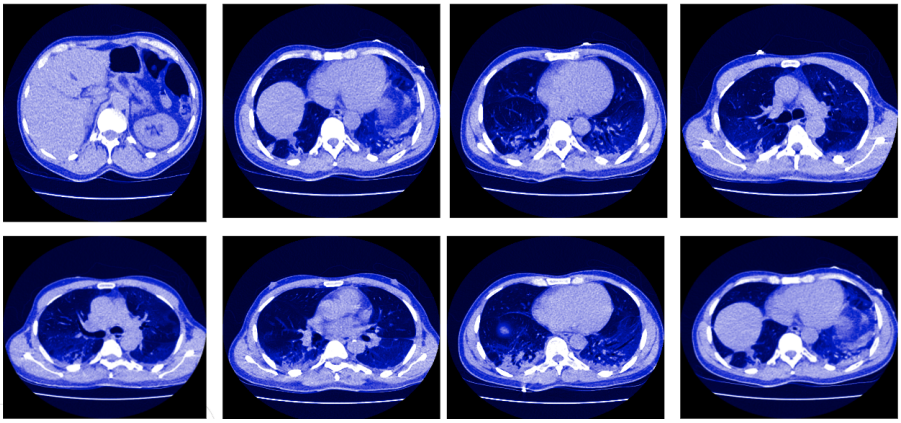


Figure 3.5: Centered sampling with three windows (CS-3W): Example of slices retained with CS-3W.

The effect of all three sampling functions on COVID-19 severity prediction using ConvNext model [178] is presented in Table 3.1. Notably, CS-1W outperformed the other sampling methods, achieving the highest AUC for severity prediction. This evaluation is performed on the STOIC public dataset (Section 3.4.1).

CS-1W retains only the central slices of a CT volume, where lung structures are

Table 3.1: Comparison of Sampling Functions: Impact on AUC Performance.

Sampling	STOIC Public Data	
	AUC Severity	AUC COVID
US-1W	0.687	0.780
CS-1W	**0.845**	0.748
CS-3W	0.775	0.784

most prominent. This slice range was found to contain features most relevant to COVID-19 severity prediction, as shown in Figure 3.4. In contrast, slices at the beginning and end of the volume contain insufficient lung anatomy and were not effective for this task.

In the case of CS-3W, utilizing additional windows alongside the lung window highlighted other anatomical structures as well, which obscured the critical features related to COVID-19 as illustrated in Figure 3.5. As a result, using additional windows in this case proved detrimental, as reported in Table 3.1.

CS-1W outperformed the rest of the sampling approaches as reported in Table 3.1. Based on these findings, CS-1W was selected for all the subsequent experiments.

3.5.2 Model Evaluation and Selection for Qualification Phase

Stage 1: Model Evaluation on STOIC Public Data

Throughout the different stages of the STOIC challenge, several deep learning models were systematically tested and evaluated. Given the critical importance of model selection, a quantitative comparative analysis was performed to identify the optimal model. The model that achieved the highest AUC for the COVID-19 severity label on the STOIC public dataset was submitted to the qualification leaderboard to assess its generalization on the private test set (challenges phases are discussed in Section 3.1.2). This approach was consistently followed throughout the competition with multiple submissions made to the challenge leaderboard. This section outlines the various experiments conducted on the STOIC public dataset for model

evaluation and selection for qualification phase submission.

The experimentation was initiated with a simple baseline model: a custom four-layer convolutional neural network (CNN). Each layer included a convolution operation, followed by batch normalization, a ReLU activation function, and max pooling. After these layers, an adaptive max pooling layer was incorporated, followed by a fully connected layer. This model achieved AUC scores of 0.687 and 0.780, for COVID-19 severity and positivity respectively.

Following the baseline model, a pre-trained model was fine-tuned using the STOIC dataset from the challenge (Section 3.4.1). This approach is particularly beneficial because pre-trained models can be adapted to specific tasks with limited data, utilizing the knowledge gained from larger datasets [67]. Specifically, a 3D UNet model, pre-trained on lung abnormalities, is used to improve the performance for the downstream task i.e. COVID-19 severity prediction [180]. This model was initially trained on 534 CT scans from the LIDC-IDRI dataset [181] and 77,074 X-ray images from the ChestXray dataset [182].

However, contrary to the anticipation of improved performance, the UNet-3D model when fine-tuned on the STOIC dataset, achieved an AUC of 0.64 for COVID-19 severity, which was lower than the AUC achieved with a custom CNN baseline model trained from scratch as reported in Table 3.2.

This decline in performance can be attributed to two main factors: **domain shift within medical imaging domain**: Although both the source domain (LIDC-IDRI dataset [181]) and the target domain (STOIC dataset) contain CT scans of the same anatomical region (lungs), however the acquisition sites and protocols are different. This disparity in domain characteristics and predictive tasks led to a domain shift (Section 1.1.1). Additionally, the pre-trained 3D UNet model was also trained on chest X-ray datasets [182], further exacerbating the domain shift between the source and target dataset (STOIC data consisting of CT scans). Thus both modalities focus on lungs (a few examples of cross-modality in medical imaging are illustrated in Figure 1.3 (cross modality)). As a result, the pre-trained model failed to generalize

effectively to the downstream task of predicting COVID-19 severity, even after fine-tuning. It limited the model's ability to transfer its learned knowledge to the STOIC dataset. **b) Limited Data:** Fine-tuning with a limited STOIC dataset did not prove to be effective and resulted in sub-optimal adaptation to the downstream task. Thus in the presence of substantial domain shift and limited data transfer learning is not helpful (Section 2.3.1).

Notably, the CNN, trained from scratch on the STOIC dataset, outperformed the pre-trained 3D UNet model. This suggests that the domain shift significantly hindered the generalization ability of the 3D model, as reported in Table 3.2. Building on these findings, further experimentation was conducted using 2D models.

During our participation in the STOIC challenge, Meta introduced the ConvNeXT model [178], which modernizes traditional CNNs by integrating advanced normalization and activation functions. This update enhances performance and efficiency, bridging the gap between classic CNNs and newer architectures. The ConvNeXT tiny version was employed, leading to a substantial improvement in contrast to CNN baseline performance. It achieved an AUC score of 0.845 for severity and 0.748 for positivity as shown in Table 3.2.

To compare the performance of ConvNeXT with more commonly used CNN architectures, we trained and evaluated several models, including ResNet [80] and MobileNetV3 [175]. Various versions of ResNet were tested, including ResNet-18, ResNet-32, and ResNet-101, with ResNet-18 yielding the best performance. The lower performance of the larger ResNet models can be attributed to their size. When the large models are trained on limited data it can potentially lead to overfitting (Section 2.3.1). In addition to ResNet, the latest version of MobileNetV3 [175] was also evaluated, as the literature indicated its strong performance in online deep learning challenges. The experimental results of these models are reported in Table 3.2.

Building on the promising results from ResNet-18 and MobileNetV3, these models were further evaluated as fixed feature extractors. A logistic regression classifier

Table 3.2: Evaluation of various models for COVID-19 severity prediction.

Model	STOIC Public Data	
	AUC Severity	AUC COVID
ConvNext [178]	**0.845**	0.748
MobileNetV3 [175]	**0.817**	0.780
ResNet18 [80]	**0.775**	0.784
MobileNetV3 + LR + HPO	0.750	0.701
MobileNetV3 + LR	0.702	0.660
CNN	0.687	0.780
Resnet-18 + LR + HPO	0.664	0.651
Resnet-18 + LR	0.654	0.601
3D U-Net [180]	0.642	0.60

was trained using the extracted features [183]. However, this approach yielded lower AUC for COVID-19 severity prediction compared to training ResNet-18 and MobileNetV3 directly on the STOIC dataset, as shown in Table 3.2.

The reduced performance of using these models as fixed feature extractors can be attributed to the fact that they were pre-trained on ImageNet, which may not capture the domain-specific features required for COVID-19 severity prediction [33]. Thus these models are optimized for feature extraction from natural imaging, rather than medical imagery. As a result, when leveraged as fixed feature extractors without additional training or fine-tuning it proved to be less effective for downstream medical tasks, specifically to COVID-19 severity prediction in this case. This issue arises from the domain shift between natural and medical imaging and it impacts the model's generalization, as discussed in detail in Section 1.1.2. The results for this approach are detailed in Table 3.2, presented as ResNet-18 + LR and MobileNetV3 + LR.

To rule out the possibility that the decline in AUC when using models as fixed feature extractors was due to the hyperparameters of the logistic regression classifier, hyperparameter optimization was performed. While this optimization resulted in small improvements, it still remained significantly lower than that of training the models using STOIC datasets. The results for this approach are presented in Table 3.2, as ResNet-18 + LR + HPO and MobileNetV3 + LR + HPO.

Performance Evaluation on the Qualification Leaderboard

After evaluating the models on the STOIC public dataset (Section 3.5.2), the next objective is to assess generalization on the private test by submitting the models on the competition's qualification leaderboard (Section 3.1.2). Due to submission constraints set by the challenge rules, we only selected the best-performing models on the STOIC public dataset for submission to the qualification leaderboard. As reported in Table 3.2, ConvNext Tiny, ResNet18 [80], and MobileNetV3 [175] outperformed the other approaches and therefore we submitted them on the qualification leaderboard.

The results of the generalization of the submitted models on the qualification leaderboard's private test set are reported in Table 3.3. This table provides a performance comparison of these models on the STOIC public dataset, along with their generalization to the previously unseen private test set from the qualification phase.

Table 3.3: Comparison of top-performing models on STOIC public data and qualification phase private test set.

Model	STOIC Public Data		Qualification Leadboard	
	AUC Severity	AUC COVID	AUC Severity	AUC COVID
ConvNext [178]	0.845	0.7480	0.748	0.800
ResNet-18 [80]	0.775	0.784	**0.752**	0.784
MobileNetV3 [175]	0.817	0.780	**0.779**	0.735

Among the three submitted models, ConvNeXT [178] achieved the highest performance on the STOIC public dataset. However, its performance significantly decreased on the test set and it failed to generalize effectively as reported in Table 3.3. This decline can likely be attributed to ConvNeXT's sensitivity to variations in data distribution and noise, which were more pronounced in the qualification test set.

In contrast, well-established CNNs like ResNet-18 [80] and MobileNetV3 [175] exhibited better generalization to the private test set and proved to be more robust and adaptable to the variations, leading to more consistent performance across both the STOIC public set and private test set from qualification phase as reported in Table 3.3.

Stage 2: Evaluation of Augmentation Techniques and Utilization of Meta-data on STOIC Public Data

ResNet-18 [80] and MobileNetV3 [175] generalized well to the private dataset, demonstrating robust performance on the unseen private test set in the qualification leaderboard (Section 3.5.2). Based on these results, further experiments were conducted to improve the performance of these two models.

Metadata can complement the features extracted from deep learning models by providing valuable contextual information, which enhances feature interpretation and improves model performance. Additionally, metadata enables more precise model tuning and evaluation by offering insights into the data's origin and characteristics [184]. In the case of the STOIC dataset, metadata includes age and sex labels. We encoded metadata and combined it with the features extracted from the models to improve COVID-19 severity prediction

The results achieved by using extracted features from the models and metadata are reported in Table 3.4. Contrary to the expectation, incorporating metadata with the features extracted from the models led to a decline in AUC for COVID-19 severity prediction. Specifically, ResNet-18 [80] achieved an AUC of 0.775 using its own extracted features, but the AUC dropped to 0.742 when metadata was included. A similar pattern was observed with MobileNetV3 [175], where the AUC decreased from 0.817 to 0.795 with the addition of metadata. This decline suggests that the age and sex metadata may have introduced noise rather than providing valuable, discriminative information to enhance model predictions. As a result, we did not use metadata features in the final phase of the challenge.

Table 3.4: Impact of metadata on COVID-19 severity prediction (AUC).

Model	STOIC Public Data	
	Features	Meta-data + Features
Resnet-18	**0.775**	0.742
MobileNetV3	**0.817**	0.795

Augmentations enhance the training set by introducing diverse variations, which

improves model generalization and reduces overfitting, allowing the model to learn from a wider range of representations [185, 186]. Thus, incorporating augmentation strategies can lead to more robust and accurate models. Arange of data augmentations are used to enhance further the performance of ResNet-18 [80] and MobileNetV3 [175]. To assess the impact of augmentations on overall predictions, we experimented with four different augmentation sets. The functionality of each augmentation in these sets is outlined in Section 3.4.2.

1. **Basic:** includes horizontal flip, random crop to 224×224, random gamma adjustment, and color jitter with brightness of 0.5, contrast of 0.5, and saturation of 0.4.

2. **Advanced:** builds upon basic augmentation set with additional transformations: safe rotation up to 30 degrees and median blur.

3. **Comprehensive:** combines basic, advanced augmentation sets, and Mixup [179] using α set to 0.8.

4. **Extended:** incorporates comprehensive augmentation set along with test time augmentation (TTA), which includes center crop, crops around four corners, and safe rotations of -5, 5, and 10 degrees.

The performance improvements achieved with each of these augmentation sets are presented in Table 3.5. Notably, every set contributed to enhancing the model's generalization to unseen data as compared to the baseline (where augmentations were not used, reported in Table 3.2). Among the four sets, the extended augmentation set achieved the most significant improvements. It improved the AUC for COVID-19 severity from 0.775 to 0.863 for ResNet18 [80] and from 0.817 to 0.841 for MobileNetV3 [175].

Performance Evaluation on the Qualification Leaderboard

To identify the optimal set of augmentations, ResNet-18 [80] and MobileNetV3 [175]

Table 3.5: Impact of augmentation on COVID Severity Prediction.

Augmentation	STOIC Public Data	
	Resnet-18	MobileNetV3
Basic	0.775	0.817
Advanced	0.795	0.831
Comprehensive	0.842	0.829
Extended	**0.863**	**0.841**

were evaluated with all four augmentation methods, and their generalization performance was assessed on the qualification leaderboard test set.

Among the four augmentation sets, the *extended* set demonstrated a noticeable improvement on the test set as shown in Table 3.6. The extended set has both train and test time augmentation. This outcome highlights the importance of using augmentation not just during training but also at test time.

Table 3.6: Effect of augmentation on model generalization to the private test set.

Augmentation	Public data		Qualification LB	
	ResNet18	MobileNet	ResNet18	MobileNet
Basic	0.775	0.817	0.752	0.779
Advanced	0.795	0.831	0.781	0.793
Comprehensive	0.842	0.829	0.790	0.795
Extended	**0.863**	**0.841**	**0.815**	**0.821**

Stage 3: Evaluation of Performance Improvement with Data Splits on the STOIC Public Dataset

To further enhance the generalization of ResNet-18 [80] and MobileNetV3 [175] data splitting was employed. This technique involves dividing the dataset into multiple subsets, each emphasizing different features and patterns. By training the models on these diverse subsets, the risk of overfitting to specific data characteristics is minimized, allowing the models to learn more robust and generalized representations. Data splitting also makes efficient use of the available dataset, enabling the models to capture a broader range of data variations. Specifically, five random data splits were created, and both models were trained on each split as shown in Figure 3.3.

Table 3.7: Performance across various data splits (AUC: COVID-19 Severity).

Model	Split 1	Split 2	Split 3	Split 4	Split 5
ResNet-18	0.788	0.786	0.811	0.810	0.805
MobileNetV3	0.854	0.804	0.766	0.830	0.799

The performance metrics for each split are presented in Table 3.7. ResNet-18 [80] shows a consistent performance across the splits, suggesting it effectively captures a broad range of features. MobileNetV3 [175], while exhibiting more variation in performance, also benefits from learning different features in each split, which allows it to adapt to the specific characteristics of each subset.

3.5.3 Final Phase Submission: Ensemble Approach with TTA

ResNet18 [80] and MobileNetV3 [175] have demonstrated consistent performance on the STOIC public dataset and strong generalization to the unseen test set in the qualification phase as comprehensively discussed in Sections 3.5.2. Based on these results, an ensemble approach is proposed that combines the strengths of ResNet18 [80] and MobileNetV3 [175] is proposed (Section 3.3.2). This ensemble method is coupled with the benefits of test-time augmentation (TTA) discussed in the previous section.

The proposed ensemble model, combined with Test-Time Augmentation (TTA) for the inference stage, was submitted in the final phase. In this phase, the models were first trained on the complete STOIC dataset (Section 3.1.2), and TTA was subsequently applied during inference to further enhance COVID-19 severity prediction

Comparative Analysis

The proposed ensemble approach was ranked 4^{th} on the final leaderboard (Section 3.1.2). The first ranked team initially pre-trained ConvNext [178] on the

MosMed dataset [187] for severity classification and UperNet [188] on the TCIA dataset [189] for lesion segmentation. The models were then fine-tuned on the STOIC dataset, utilizing metadata along with the outputs of both pre-trained models as feature vectors. For evaluation, a 5-fold cross-validation approach was employed, and an ensemble model was used for final testing.

The team ranked second employed two vision encoders, pre-trained on iBot [190] using self-supervised learning, to analyze both plain slices and segmented lung regions. The extracted features were then concatenated with meta-data (age and sex labels) and used as inputs for the logistic regression classifier to make predictions.

Team 3 utilized a lung segmentation model combined with autodidactic pre-training on the segmented images. The network's output was then merged with age data and passed through a fully connected layer. Finally, an ensemble of five models, along with test-time augmentation (TTA), was employed for predictions.

A comparison of our proposed method of ensemble models with TTA as compared to these approaches is reported in Table 3.8. In contrast to the complex methodologies employed by the top-ranking teams, the proposed ensemble approach is notably simpler. Despite the reduced complexity, it achieved 4^{th} rank on the final leaderboard, highlighting its competitive performance. The simplicity of the proposed approach did not compromise effectiveness, indicating that a well-designed ensemble approach can be both effective while maintaining efficiency.

Table 3.8: Final leaderboard results: Comparison with top-ranked methods [20].

Rank	AUC Severity	AUC COVID
First	0.815	0.616
Second	0.811	0.845
Third	0.794	0.837
Fourth (Ours)	0.787	0.829

Experiments with the STOIC public dataset revealed that incorporating meta-data did not enhance performance using the proposed approach (Section 3.5.2). Consequently, we did not use metadata in the final phase. However, other teams,

despite similar findings, chose to include metadata in their final submissions, which appeared to contribute to performance improvements. Given that the results of our proposed approach were very close to those of Team 3, it is plausible that including metadata could have further improved the ranking of the proposed ensemble method. In summary, while the proposed ensemble method is simpler compared to others, it demonstrates competitive performance, and further refinement, such as incorporating metadata could improve the performance.

3.6 Summary

This chapter presents the work we conducted during our participation in the COVID-19 AI STOIC 2021 Challenge. It focused on COVID-19 severity prediction using the STOIC CT scan dataset. To develop an effective solution, a variety of deep learning models were comprehensively evaluated (Section 3.5.2). The assessment of these models for COVID-19 severity prediction was conducted in two ways: a) fine-tuning pre-trained models, and b) using models as fixed feature extractors and training a logistic regression classifier on the extracted features.

Particularly, an ensemble approach is proposed that combines the strengths of ResNet18 [80] and MobileNetV3 [175] (Section 3.3). The specific model choice is based on comparative analysis and experimental evaluation of various models as detailed in Section 3.5.2. The proposed ensemble approach was enhanced with test-time augmentation (TTA), which proved effective in improving model generalization on the unseen private test set during the qualification phase (Section 3.5.2).

The proposed ensemble approach coupled with TTA ultimately achieved 4^{rth} place in the challenge. It achieved an AUC of 0.787 and 0.829 for COVID severity and COVID positivity on the private test set of the final phase of the STOIC challenge (Section 3.5.3).

Participating in this challenge offered valuable hands-on experience, enhancing our practical skills in implementing and evaluating deep learning approaches while addressing the associated challenges.

3.6.1 Insights

Participation in this challenge proved highly valuable as it helped us to gain insights on the applications of deep learning to real-world medical applications. A few of these insights are:

1. Domain shift due to variations in imaging modalities and protocols poses a significant challenge to model generalization. For example, ConvNext [178] outperformed other evaluated models on the STOIC public dataset but it did not generalize effectively to the STOIC private test set (Table 3.3). Although both subsets of STOIC data used the same imaging modality, differences in acquisition sites and protocols introduced a domain shift that affected the model's generalization (Section 3.5.2).

2. The nature of tasks in medical and natural imaging differs significantly, and as a result, features learned from models pre-trained on natural imaging often fail to generalize effectively to real-world medical applications due to the domain shift between the two (Section 1.1.2). For instance, when we used these models as fixed feature extractors and adapted them to the COVID-19 severity prediction task using a classifier, they did not yield effective results (Section 3.5.2, Table 3.2).

3. The medical imaging domain often suffers from limited available data, making conventional approaches like fine-tuning models less effective and prone to overfitting. For example, when we applied UNET-3D, pre-trained on medical images, and fine-tuned it using the STOIC public dataset, its performance was even lower than that of a simple CNN baseline trained from scratch (Section 3.5.2, Table 3.2). This highlights the challenge of applying pre-trained models in medical imaging tasks with small datasets, where fine-tuning may not necessarily yield better results.

These insights have been pivotal in shaping the future research direction of this thesis: developing innovative and robust adaptation approaches for neural networks

to address domain shift challenges while minimizing reliance on annotated data and computational resources with the aim for more efficient and scalable approaches.

Chapter 4

Unsupervised Parameter Efficient Domain Adaptation for Multi-Target Medical Applications

This chapter outlines the key contributions of this research to unsupervised multi-target domain adaptation. It addresses Research Question 2 (RQ2): "How could the parameter-efficient adaptation approach be enforced in the unsupervised adaptation of convolutional neural networks? Could convolutional neural networks benefit from the features learned through self-supervised training when using parameter-efficient adaptation ?" In this context, this chapter presents a novel, parameter-efficient approach for unsupervised adaptation of convolutional neural networks. It is evaluated experimentally through segmentation tasks: brain segmentation from T1-weighted MRI scans and cardiac structures segmentation from cardiac MRI scans. The work presented in this chapter has been published in 21^{st} IEEE International Symposium on Biomedical Imaging, 2024 [25]. The code to replicate experiments and results is publicly available at: https://github.com/aleemsidra/ConvLoRA.

Section 4.1 gives an introduction and outlines the motivation for developing an unsupervised parameter-efficient domain adaptation approach for medical imaging, highlighting the limitations of existing methods. Section 4.2 provides a literature

review of domain adaptation approaches relevant to the proposed approach. Section 4.3 introduces the proposed approach and elaborates on its architectural design. Section 4.4 outlines the datasets used and the experimental framework. Section 4.5 presents the results obtained from the comprehensive experiments. Finally, Section 4.6 presents the conclusion and highlights how the findings of this work guided the subsequent research presented in the following chapters.

4.1 Introduction

Many computer vision applications leverage transfer learning to transfer knowledge from large-scale pre-trained models to various downstream tasks. It typically involves fine-tuning, a process that updates all parameters of the pre-trained model via back-propagation [191] (Section 2.3.1). Deep neural networks have achieved state-of-the-art performance when both train and test sets share the same distribution. However, domain shift, (i.e., change in data distribution) between train (source domain) and test (target domain) sets, significantly deteriorates the model's generalizability [24, 25, 27, 26, 88] (Sections 1.1.1, 1.1.2, and 1.1.3). This issue is particularly pronounced in multi-center medical studies, where various imaging centers employ different scanners, protocols, and subject populations [2, 88].

Furthermore, large-scale models' performance relies on large-scale labeled data [192, 193, 194, 195]. To address performance degradation caused by domain shifts, one approach is to acquire labeled data and retrain the model. However, in the medical field, annotated data is often scarce due to privacy restrictions, limited availability, and the need for expert intervention (Section 1.1.4). Consequently, large-scale models that require vast amounts of data cannot be directly applied to medical image segmentation.

Unsupervised domain adaptation (UDA) aims to generalize large-scale models, pre-trained on the source domain to an unlabeled target domain, eliminating the need for costly data annotation [26, 88]. In such a scenario, the source domain contains a vast amount of labeled data, while the target domain typically consists of

a smaller, unlabeled dataset. UDA is generally achieved through fine-tuning, where a model pre-trained on the source domain is adapted to different target domains. However, a major downside of fine-tuning is that it results in a dedicated model for each target domain with the same parameters as the original pre-trained model [10, 196]. As a result, multiple target domains would each require a dedicated model models with the same parameter count as the original pre-trained model.

Thus UDA methods can be effective for single-target domain adaptation, resulting in a single model for a specific target domain. Conversely, in multi-target DA (MT-DA) the objective is to adapt the model to multiple unlabeled target domains. MT-DA has a broader applicability to real-world scenarios. Specifically in the medical domain, there are diverse target domains each with distinct characteristics. These domains vary in imaging modalities (such as MRI, CT scans, and X-rays), imaging acquisition devices, acquisition sites, and patient populations (Section 1.1.3). Thus, training separate models for each target domain with the same trainable parameters as the source model is impractical and prohibitively expensive.

Parameter Efficient Fine-Tuning (PEFT) aims to streamline fine-tuning by optimizing resource usage, reducing computational costs, and minimizing memory requirements. Unlike conventional fine-tuning, PEFT keeps the majority of the core model parameters frozen while adapting a significantly smaller subset of parameters. [197, 198]. The idea is to adapt a pre-trained model to a specific task or dataset without modifying all of its parameters, which can be computationally expensive and require a lot of memory. PEFT has proven its effectiveness as an adaptation strategy for Large Language Models (LLMs) [199, 200, 201, 202, 197]. It enables both efficient learning and faster updates. PEFT outperforms approaches that adapt the entire model, offering superior generalization, particularly in limited data scenarios. By focusing on a smaller set of parameters, PEFT efficiently adjusts the model without the need for extensive re-training, making it more effective when labeled data is scarce [197, 203, 110]. There are various types of PEFT, which are discussed in detail in Section 4.2.1.

Existing approaches predominantly focus on applying PEFT to transformer-based architectures [54, 55]. However, PEFT adapter-based approaches have not been explored for adapting convolutional neural networks (CNNs) to diverse target domains. To the best of our knowledge, both the use of PEFT in medical imaging for unsupervised domain adaptation (UDA) and its application in CNN adaptation have not yet been explored [204].

To address this research gap, a novel parameter-efficient approach for multi-target unsupervised domain adaptation (MT-UDA) is proposed. It not only outperforms conventional supervised fine-tuning based adaptation but is also computationally efficient and has a low memory footprint. **First**, Convolutional Low-Rank Adaptation (ConvLoRA) is proposed, as an adaptation of Low-Rank Domain Adaptation (LoRA) in LLMs [10] (discussed in detail Section 4.3.1). ConvLoRA is specifically designed for application in CNNs, offering a novel approach to address domain adaptation challenges in the context of image data. Instead of creating separate fine-tuned models for each target domain—each with the same number of parameters as the base model—we integrate our proposed ConvLoRA adapters into the pre-trained base model. We adapt only the ConvLoRA parameters while keeping the rest of the base model's parameters frozen. The architectural design and functionality of ConvLoRA are discussed in detail in Section 4.3.2. It allows faster updates by adapting only a small subset of domain-specific parameters. in**Second**, we further mitigate domain shift introduced by statistical differences in mean and variance between source and target data without additional training and computational resources. It is achieved by utilizing Adaptive Batch Normalization (AdaBN) [30]. The rationale behind choosing AdaBN over traditional BN adaptation, along with a detailed explanation of its functionality, is presented in Section 4.3.1.

Contributions

- Inspired by recent advancements in LLMs, a novel multi-target unsupervised domain adaptation (MT-UDA) approach is proposed which leverages proposed parameter-efficient ConvLoRA and AdaBN. To the best of our knowledge, this

is the first work to adapt LoRA [10] to CNNs, specifically for UDA in the context of medical image segmentation.

- The experimental results show that the proposed UDA pipeline achieves a significant reduction of over 99% in trainable parameters while simultaneously achieving competitive segmentation accuracy compared to other methods.

- The proposed framework is generic, flexible, and can be seamlessly integrated with any CNN based architectures. It significantly reduces training costs and enhances adaptability and generalization to multi-target domains.

4.2 Related Work

4.2.1 Parameter Efficient Adaptation

The development of parameter-efficient adapters for transformer-based models has emerged as a critical area of research, focusing on enhancing model performance while optimizing resource utilization. There are various techniques for parameter-efficient adaptation: lightweight adapters, low-rank adaptation, prompt tuning, Bit-Fit, and prefix tuning. Among these two are most commonly used: lightweight adapter-based and prefix tuning.

The concept of adapters was initially introduced by Houlsby [205]. It involves incorporating small, trainable modules called "adapters" into pre-trained models. These adapters facilitate efficient task-specific adaptations without modifying the core model parameters. This approach maintains the original model's performance while reducing the computational cost and resource requirements for fine-tuning. This design enhances the model's adaptability while maintaining a relatively compact parameter footprint. In contrast, Lin et al. [206] proposed a more streamlined approach with a single adapter layer per block, augmented by an additional LayerNorm [207].

Despite these advancements, inherent challenges associated with adapter layers

must be critically evaluated. One such challenge is the computational overhead introduced by integrating additional adapter layers into the base model. Although adapter layers are designed with a small bottleneck dimension to minimize additional FLOPs and parameter count, often constituting less than 1% of the original model's parameters. However, there are notable issues in practice. While the parameter count remains low, the sequential processing requirement of adapter layers can lead to increased latency, particularly in online inference scenarios where batch sizes are minimal. This issue is exacerbated in environments lacking model parallelism, such as running inference on a single GPU with models like GPT-2 [37], where the latency impact of adapter layers becomes pronounced even with minimal bottleneck dimensions.

Prefix tuning is used for adapting pre-trained language models to specific tasks by learning a fixed-length prompt that is prepended to the input text [208]. This method optimizes a small set of trainable parameters that modify the model's behavior without changing the underlying pre-trained weights. One of the primary issues is the difficulty associated with optimizing prefix tuning. Empirical observations corroborate the original findings of [209], revealing that the performance of prefix tuning exhibits a non-monotonic relationship with the number of trainable parameters. This suggests that increasing the number of trainable parameters does not consistently translate into improved performance, which complicates the tuning process.

For this work, we leveraged our proposed convolutional low rank adapters for parameter-efficient adaptation to downstream tasks. It has no additional latency as detailed in Section 4.3.2

4.2.2 Batch Normalization based Adaptation

Batch normalization based domain adaptation has been implemented in various ways, such as by aligning distributions between source and target domains or by using separate normalization parameters for each domain. In domain-specific batch

normalization (DSBN), separate batch normalization parameters are maintained for each domain (source and target). It allows the model to learn domain-specific feature representations while sharing other parameters across domains [210]. However, batch normalization relies on batch statistics, which makes DSBN sensitive to small batch sizes. In domain adaptation, especially in the target domain where labeled data is limited, small batch sizes are often used, which can lead to inaccurate estimates of batch statistics and degrade performance. An adversarial learning approach incorporating batch normalization to align feature distributions between domains is proposed in [211]. In this method, a domain discriminator is trained to differentiate between source and target domain features, while the feature extractor is optimized to extract domain-invariant features. However, this approach is prone to over-reliance on the feature alignment.

Adaptation is achieved through unsupervised fine-tuning of batch normalization layers in the target domain [210]. Adaptive batch normalization (AdaBN) is proposed to compute the mean and variance for the batch normalization layer's running statistics in the target domain, thereby improving generalization [30]. Specifically, AdaBN proposes a post-processing method to re-estimate batch normalization statistics using target samples. Test-time adaptation mitigates domain shift by recalculating running statistics for the current test input [212, 213, 214].

This work incorporates AdaBN into the proposed parameter-efficient, self-training-based adaptation approach, as detailed in Section 4.3.1 and illustrated in Figure 4.4.

4.2.3 Unsupervised Domain Adaptation (UDA)

Several works employ adversarial learning, such as CycleGAN [215] and domain-invariant feature learning [216], to adapt segmentation models [217]. A method of matching layer-wise activations across domains is proposed in [218]. An adversarial network is proposed for brain lesion segmentation [219]. Kushibar et al. [220] show that fine-tuning only the last CNN layer improves performance. However, it lacks a comparison with other domain adaptation methods. The last CNN layer is fine-

tuned, but this work focuses more on the training cases selection procedure rather than on the fine-tuning method development [221]. Cross-modality domain adaptation for cardiac MR and CT image segmentation is achieved by adapting low-level layers [222]. Fine-tuning of early U-Net layers is done for skull segmentation [196].

4.3 Methodology

This section first discusses preliminaries: low rank adaptation and adaptive batch normalization in Section 4.3.1. Subsequently, our proposed adaptation approach is presented in detail in Section 4.3.2.

4.3.1 Preliminaries

Low Rank Domain Adaptation (LoRA)

A neural network consists of numerous dense layers that perform matrix multiplication. The weight matrices in these layers are typically of full rank [109]. LoRA updates these weight matrices by leveraging the concept of low intrinsic rank [223, 224]. It refers to the idea that a large pre-trained model can be adapted to a new downstream task by updating/modifying a much smaller set of task-specific parameters, rather than updating the gradients in the entire core model [10].

In the context of a pre-trained weight matrix in a neural network, LoRA constrains its updates through a low-rank decomposition of the weight matrix. In contrast to regular fine-tuning, where all the parameters of the models are updated during training, LoRA keeps the pre-trained weights frozen. Instead, only the low-rank matrices, which hold the trainable parameters receive gradient updates.

The difference between the regular fine-tuning and LoRA is shown in Figure 4.1. In the forward pass, both the pre-trained weight matrices (W) and their low-rank counterparts (A and B) are multiplied by the same input (d), and the resulting outputs are summed coordinate-wise. However, only the low-rank matrices (A and B) are updated during the backward pass.

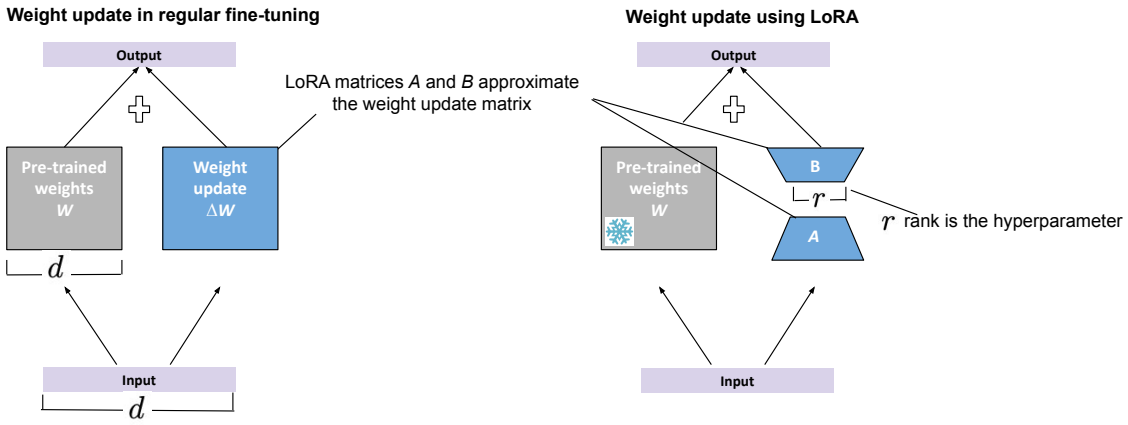


Figure 4.1: Comparison of regular fine-tuning and LoRA's re-parameterization. In LoRA, the pre-trained weight matrix W remains frozen, while only the low-rank decomposition matrices A and B receive gradient updates [10].

Thus, a single pre-trained model can be shared and used with various small LoRA modules for different downstream tasks. LoRA optimizes only a limited number of parameters, specifically the injected low-rank matrices (A and B in Figure 4.1). This approach makes training more efficient by eliminating the need to compute gradients or maintain optimizer states for most of the parameters [10]. Moreover, as illustrated in Figure 4.1, the straightforward linear design integrates the trainable low-rank matrices (A and B) with the pre-trained frozen weights (W) during deployment. This design does not introduce any additional inference latency as compared to other methods [10].

Adaptive Batch Normalization (AdaBN)

Batch Normalization (BN) was originally designed to alleviate the issue of internal covariate shifting, a common challenge encountered when training a very deep neural network [60]. It first standardizes each feature in a mini-batch and then learns a common slope and bias for each mini-batch. Formally, given the input to a BN layer, $X \in R^{n \times p}$, where n denotes the batch size, and p is the feature dimension, the BN layer transforms a feature as:

$$\hat{x} = \frac{x - E[X]}{\sqrt{Var[X] + \epsilon}}.\gamma + \beta \tag{4.1}$$

where x is input, $E[X]$ is the expected mean of the input data X, $Var[X]$ is the variance of the X, ϵ is used for numerical stability as an arbitrarily small constant, γ and β are the learnable parameters. This transformation guarantees that the input distribution of each layer remains unchanged across different mini-batches. For stochastic gradient descent optimization, a stable input distribution could greatly facilitate model convergence, leading to much faster training speed for CNN [60, 80]. During the inference phase, the global statistics of all training samples are used to normalize every mini-batch of test data.

Although BN was originally proposed to help SGD optimization, its core idea is to align training data distribution. BN normalizes activation outputs based on batch statistics. Due to the domain shift between source and target domains, applying source domain statistics to standardize the target domain can result in misclassification [225].

To overcome this issue, adaptive batch normalization (AdaBN) computes the target domain-specific batch-wise mean and variance [30]. Standardizing each layer by the respective domain's statistics ensures that each layer receives data from a similar distribution, whether it comes from the source or target domain. This approach is straightforward to implement, has zero parameters to tune as mean and variance are non-learnable parameters, and requires minimal computational resources [30]. In this work, we used AdaBN [30] instead of standard batch normalization.

4.3.2 Proposed Approach: Unsupervised Parameter-Efficient Adaptation using Convolutional Low-Rank Adaptation and Adaptive Batch Normalization

During fine-tuning, a neural network is initialized with pre-trained weights (Φ_0), which are iteratively updated to $\Phi_0 + \Delta\Phi$ by following the gradient to optimize the model's objective [109]. One of the main drawbacks of fine-tuning the entire model is that for each downstream task, a different set of parameters $\Delta\Phi$ is learned whose dimension $|\Delta\Phi|$ equals $|\Phi_0|$. The specific U-Net architecture used for this work has

approximately 24.3 million parameters ($|\Phi_0| \approx 24.3$ million) [226]. Consequently, storing and deploying numerous independently fine-tuned instances of this model can be challenging and may not be feasible, particularly in multi-target domain scenarios.

Convolutional Low Rank Adaptation (ConvLoRA)

To overcome this challenge, a new ConvLoRA adapter, an extension of LoRA [10] is proposed, for parameter-efficient unsupervised domain adaptation (UDA). In contrast to previous approaches, which focus on parameter-efficient adaptation of transformers based architecture, ConvLoRA is designed specifically for parameter-efficient adaptation of CNNs. For a pre-trained convolutional layer weight matrix ($W_{PT_{CONV}} \in R^{m \times n}$), ConvLoRA constrains the weight update by decomposing $W_{PT_{CONV}}$ using a low-rank decomposition as:

$$W_{PT_{CONV}} + \Delta W_{CONV} = W_{PT_{CONV}} + XY \tag{4.2}$$

where $W_{PT_{CONV}}$ is the pre-trained weight matrix of convolutional layer, ΔW_{CONV} is the weight update, $X \in R^{m \times r}$ and $Y \in R^{n \times r}$ are low-rank matrices and $r << min(m, n)$ is the rank. During training, $W_{PT_{CONV}}$ is frozen and does not receive gradient updates, while only X and Y contain trainable parameters. Both $W_{PT_{CONV}}$ and ΔW_{CONV} are multiplied by the input and the respective output vectors are summed coordinate-wise. Hence, the forward pass operation is as follows:

$$h = W_{PT_{CONV}} x + \Delta W_{CONV} x = W_{PT_{CONV}} x + XYx \tag{4.3}$$

where x is input, X is initialized by random Gaussian distribution and Y is zero in the beginning of training.

Step 1: Pre-training the Source Model on Source Domain

Initially, a U-Net referred to as the source model (Φ_{src}), is trained using labeled

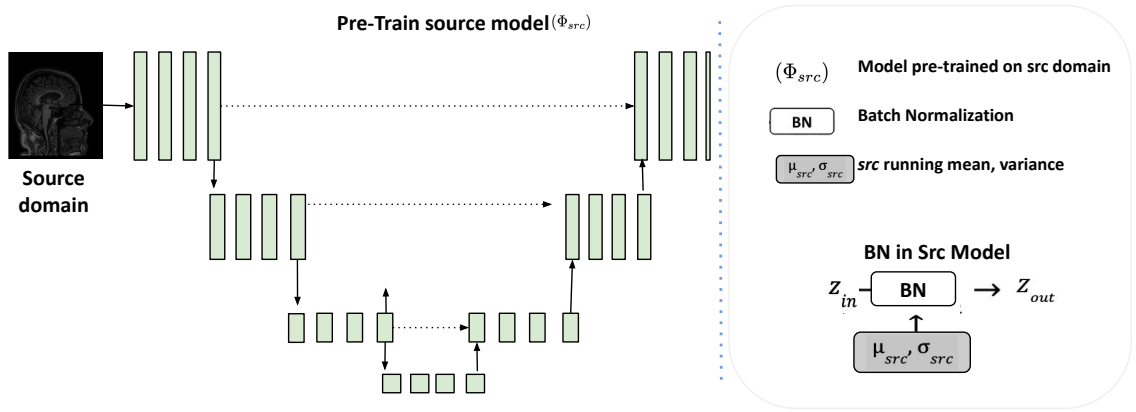


Figure 4.2: Source model (Φ_{src}) pre-trained on the source domain (src).

data from the source domain data (X_{src}) as shown in Figure 4.2. We adapted the U-Net used in [226], it has 3×3 convolution kernels, ReLU activation functions, and the skip connections are implemented as convolutions followed by a sum operation. During this stage, all model parameters are updated using source domain data, with standard batch normalization (BN) applied throughout the training process, as illustrated in Figure 4.2.

Our goal is to adapt the source model to out-of-distribution unlabeled target data (Y_{tar}) in a parameter-efficient unsupervised manner. The source model is trained using cross-entropy loss:

$$I(x,y)_{source} = -\frac{1}{N_s}\sum_{n=1}^{N_s}\sum_{n=1}^{C} y_{n,c} log p_c(x_n) \tag{4.4}$$

where N_s is the number of samples in the source domain, $p_c(x_n)$ is model's predicted probability and y_n is a one-hot-encoded vector of the true label for pixel n.

Step 2: Early Segmentation Head Refinement

To prepare for the adaptation of the source model to the target domain, we first add another segmentation head called the early segmentation head (ESH) after the encoder as shown in Figure 4.3. ESH is a small CNN containing three convolution layers each followed by a batch normalization layer. The placement of the ESH in the network was determined through ablation studies, with the encoder identified

as the optimal position (Section 4.5.4).

ESH is initialized on the source domain by pre-training with the cross-entropy loss (Eq. 4.4) between the output of ESH ($\hat{y}_{ESH(src)}$) and the source ground truth mask (GT_{src}) as shown in Figure 4.3. During this initialization process, all the weights in the source model (Φ_{src}) are kept frozen, and the gradient is back-propagated exclusively through the ESH weights, as illustrated in Figure 4.3 (loss calculation).

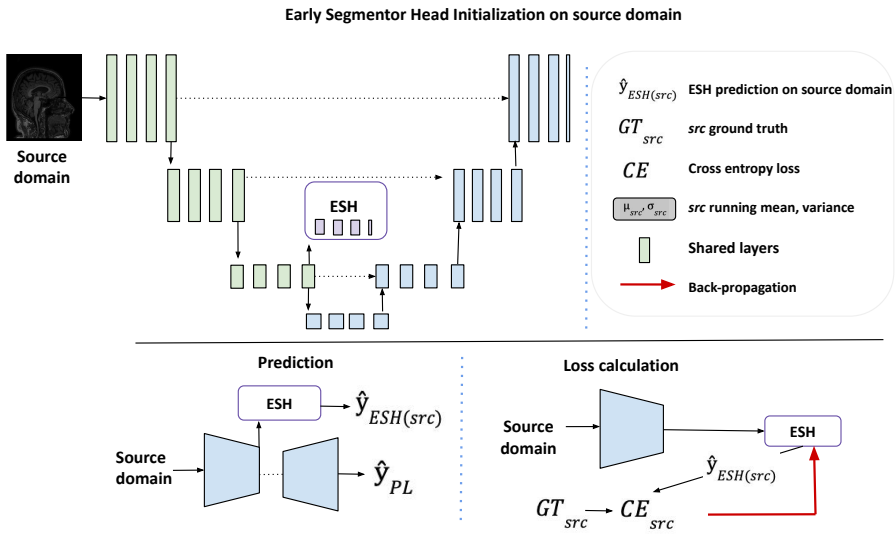


Figure 4.3: ESH initialization with the source domain. The source model is frozen and the gradient is backpropagated only to ESH.

Now the model generates two probabilistic segmentation outputs: a preliminary segmentation output from the ESH, denoted as ($\hat{y}_{ESH(src)}$), and the final segmentation output which is obtained from the source model, denoted as ($\hat{y}_{PL}$), as shown in Figure 4.3.

During the adaptation, the ESH will act as a student, which is trained by the output of the source model (Φ_{src}), which acts as a teacher. As Φ_{src} and ESH share the encoder component of the network, refining student on the target domain also benefits the teacher.

Step 3: Adaptation through ConvLoRA and AdaBN in a Self-Training Framework

For adaptation, our proposed ConvLoRA adapters are integrated into the en-

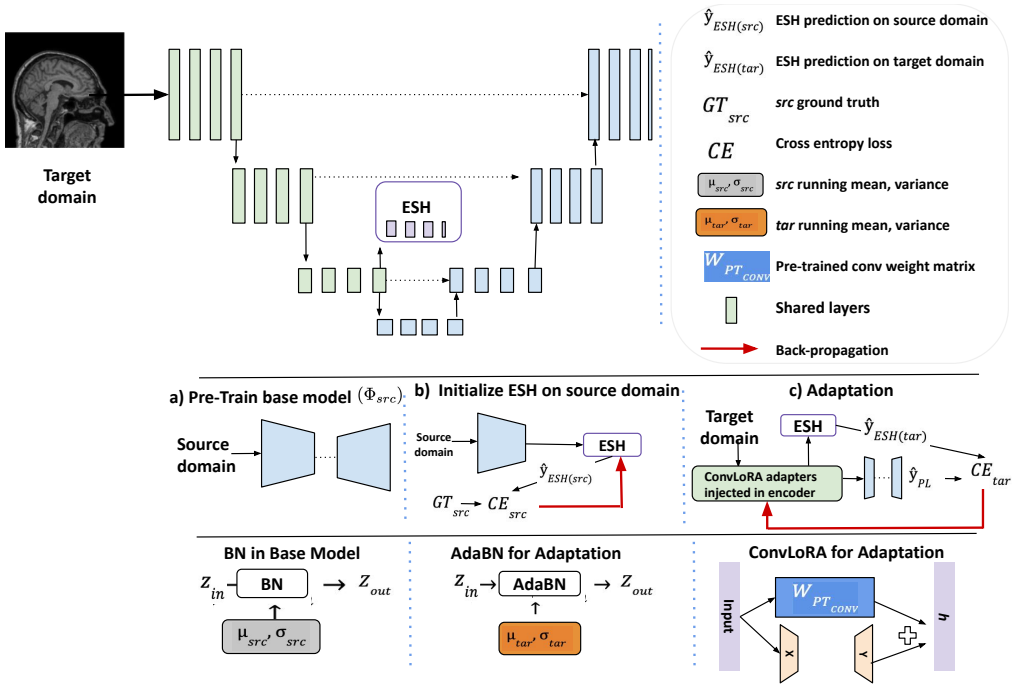


Figure 4.4: 2D U-Net with Early Segmentation Head (ESH): ConvLoRA adapters facilitate parameter efficient adaptation in the encoder, along with AdaBN throughout the network.

coder of the source model (Φ_{src}). Specifically, it is integrated into all convolutional layers within the encoder. An ablation study was conducted to identify which part of the network is most susceptible to domain shift. The results of this experiment indicate that the encoder is the most affected by such shifts (Section 4.5.4).

In our proposed adaptation scheme, all parameters of the source model (Φ_{src}) are frozen, except for the ConvLoRA parameters and the running mean and variance of the batch normalization layers as shown in Figure 4.4.

The target domain images (X_{tar}) are fed to both source model (Φ_{src}) and ESH branches. However, the gradient updates are constrained only to ConvLoRA adapters injected in the encoder as shown in Figure 4.4 (c). The cross-entropy loss is calculated between output of ESH ($\hat{y}_{ESH(tar)}$) and output of the base model ($\hat{y}_{PL}$) as:

$$I(x)_{target} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C}\hat{y}_{PL}^{(c)}(x_n)\log\hat{y}_{ESH}^{(c)}(x_n) \tag{4.5}$$

where N is the number of data points, C is the number of classes, $\hat{y}_{PL}$ is the probability of class $\hat{y}_{PL}^{(c)}$, $\hat{y}_{ESH}^{(c)}(x_n)$ is the is the probability of class c the predicted

by the ESH.

We now leverage the segmentation output ($\hat{y}_{PL}$) from the source model as pseudo-labels to refine predictions from ESH, i.e., ($\hat{y}_{ESH(tar)}$) in a self-training framework as illustrated in Figure 4.4. This process adapts the encoder features by updating the ConvLoRA parameters, leveraging the higher-level information learned by the rest of the network ($\hat{y}_{ESH(tar)}$). The integration of the proposed ConvLoRA solely into the encoder component resulted in superior performance compared to fine-tuning the entire source model. This improvement was achieved with a substantially reduced number of parameters, as detailed in Section 4.5.3, making our proposed method both parameter- and computationally efficient (comparison of trainable parameters is reported in Table 4.4). Furthermore, as the proposed adapters are lightweight, they can be adapted to multi-target domains by modifying only a small subset of ConvLoRA parameters while sharing a single base model.

To further enhance the domain adaptation, our proposed approach utilizes the target domain's running mean and variance, computed through AdaBN [30]. The source domain statistics are updated by computing target-specific batch-wise running statistics as shown in Figure 4.4 "AdaBN for adaptation".

As the running statistics of the batch normalization layer are not learnable parameters, hence adapting them according to the respective target domain throughout the network is achieved in a parameter-free manner. Thus AdaBN does not introduce additional computational overhead and facilitates parameter-free adaptation without extra parameters and components. The experimental results demonstrated that the performance of our proposed ConvLoRA is further complemented by the leveraging AdaBN (Section 4.5).

4.4 Experimental Framework

4.4.1 Datasets

The proposed approach is evaluated on the Calgary-Campinas (CC359) [11] and M&M [12] datasets. CC359 is a multi-vendor (GE, Philips, Siemens), multi-field strength (1.5, 3) magnetic resonance (MR) T1-weighted volumetric brain imaging dataset. It has six different domains and contains 359 3D brain MRI scans, primarily focused on the task of skull stripping. The six different domains are GE 3, GE 1.5, Philips 1.5, Philips 3, Siemens 1.5, and Siemens 3. The name indicates the vendor of the MRI machine and the number indicates the magnetic field strength. Each domain exhibits two levels of domain shift: the MRI vendor and the magnetic field strength. A few samples from each domain are shown in Figure 4.5.

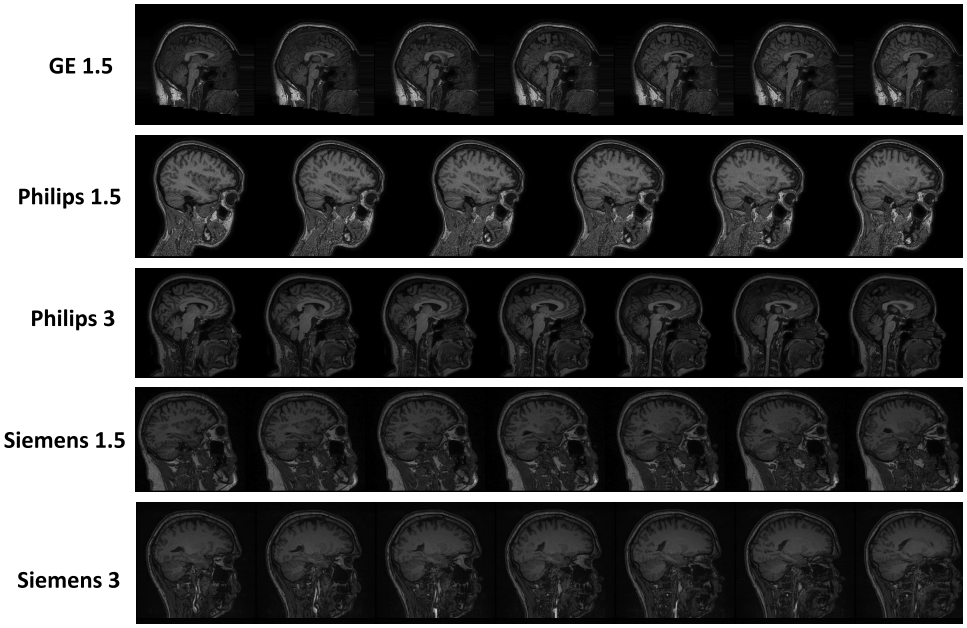


Figure 4.5: CC359 dataset: Different domains [11].

The M&M dataset [12] consists of cardiac MRI scans from a total of 345 subjects. It has four domains, each representing images obtained from different scanner vendors: Siemens, GE, Canon, and Philips. These domains differ in their in-plane resolution, slice thickness, number of slices, and number of time frames. This dataset has three regions of interest: the left ventricle cavity (LV), the right ventricle cavity (RV), and the left ventricle myocardium (MYO) as shown in Figure 4.6.

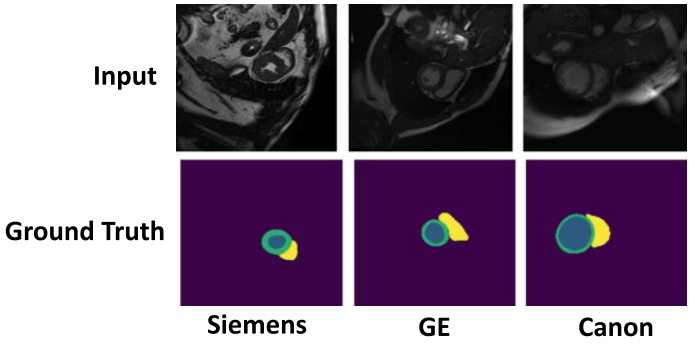


Figure 4.6: M&M dataset [12]: Different domains. (Yellow: RV, Blue: LV, Green: MYO).

4.4.2 Training setup

The source model (Φ_{src}) is pre-trained using source domain (GE 3T for CC359 [11] and Philips for M&M dataset [12]) for 100 epochs using a batch size of 32, a learning rate of 0.001, and optimized with the Adam optimizer using cross-entropy loss. Pre-processing includes the removal of black slices, min-max scaling, and resizing all images to a resolution of 256×256.

The ESH is pre-trained on the source domain for 20 epochs, followed by our proposed adaptation method, where the model is trained for only 5 epochs using 10 samples from the respective target domain with a learning rate of 0.0001. For the adaptation of the ConvLoRA adapter to the downstream segmentation task, we set the rank to $r = 2$, as the original kernel weight matrix has dimension was 3×3. The surface Dice Score (SDS) [196] is used to evaluate the performance. This metric is more informative than volumetric dice as it emphasizes the brain contour over internal volume [196] and it is widely used in methods exploring CC359 dataset [226, 196, 222]. For the M&M dataset [12], dice score is used as the evaluation metric.

The processing pipeline is implemented in Python 3.8.17, and the open-source library PyTorch 2.0.1 is used. All experiments were performed on a desktop computer with the Ubuntu operating system 20.04.6 LTS with CUDA 11.6, NVIDIA GeForce RTX 3090 GPU, and a total of 64 GB RAM.

4.5 Results and Analysis

As we systematically discussed the different stages of our framework in Section 4.3.2, the experimental results are also reported in the same sequential manner.

4.5.1 Source Model

It refers to the base model (Φ_{src}), which is trained exclusively on the source domain, as discussed in Section 4.3.2. For experiments involving CC359 [11], GE 3T is used as the source domain, and the remaining five domains are treated as target domains (mentioned in Section 4.4.1). The source model is trained using 40 subjects from GE 3T, comprising 6,032 slices. The source model's performance is then evaluated on each of the five target domains during the inference phase. To assess the impact on the generalization of the source model due to the domain shift of the target domains and the effectiveness of the proposed unsupervised parameter efficient domain adaptation method, we first evaluate the source model on the target domains without any adaptation.

Since our source domain is GE 3T, we initiated experiments first to evaluate the source model on the target domain, which has the same vendor i.e. GE, but a different field strength (1.5T). Despite the using same vendor, domain shift impacted the results significantly as shown in Figure 4.7. The first row shows the input data, the second row shows the ground truth and the third row shows the performance of the source model. The domain shift observed, even with the same vendor, can be attributed to differences in field strength, which significantly affect image characteristics such as signal-to-noise ratio, contrast, resolution, and artifacts. These variations cause inconsistencies in imaging parameters and scanner performance, leading to notable discrepancies in feature distribution between the source and target domains. As a result, even with the same vendor, these factors contribute to the domain shift, adversely affecting the source model's generalizability when used without adaptation.

We also assessed the performance of the source model on the remaining four

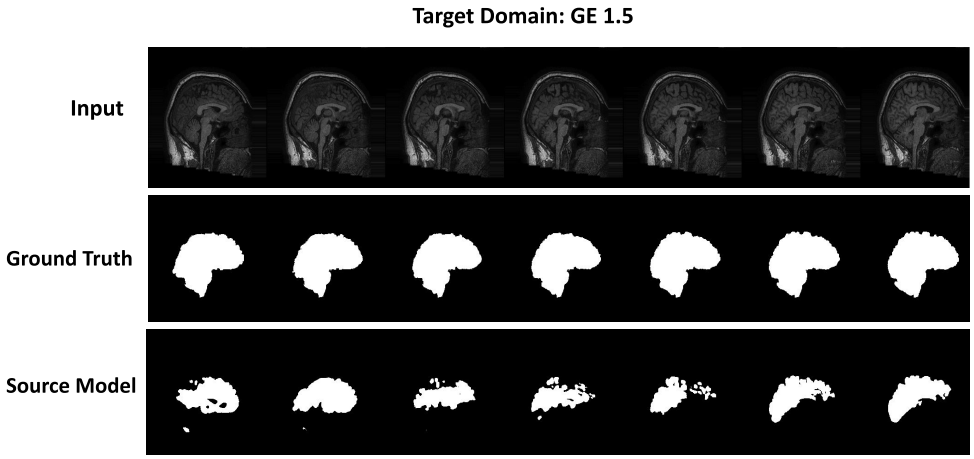


Figure 4.7: Performance of source model without adaptation on the target domain: GE 1.5 [11]

domains, which originate from different vendors and have varying magnetic field strengths (discussed in Section 4.4.1). The results for each domain are shown in Figure 4.8, 4.9, 4.10, 4.11 respectively. As evident from all the qualitative results, when the source model trained exclusively on the source domain is evaluated on the target domains without adaptation, the domain shift between the source and target domain deteriorates the performance of the source model significantly.

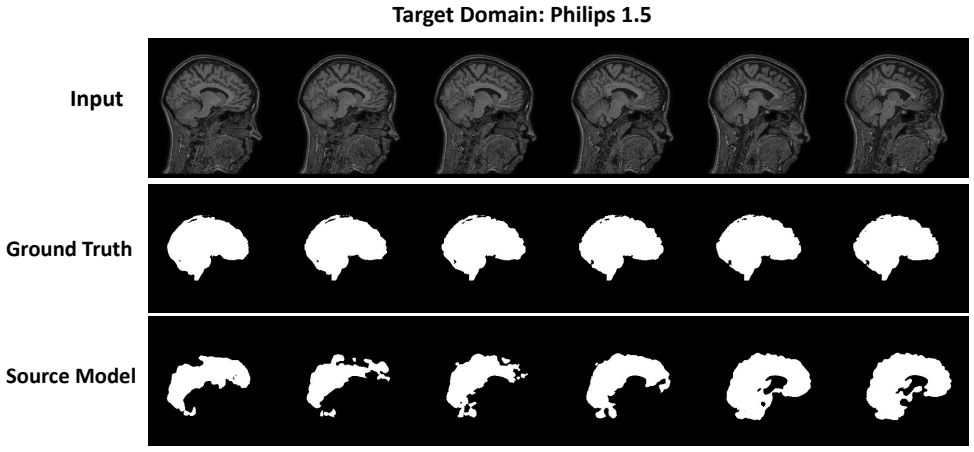


Figure 4.8: Performance of source model without adaptation on the target domain: Philips 1.5 [11].

Quantitative Analysis

A comprehensive quantitative analysis is conducted to evaluate the source model's performance across the entire dataset and examine the impact of domain shift on its

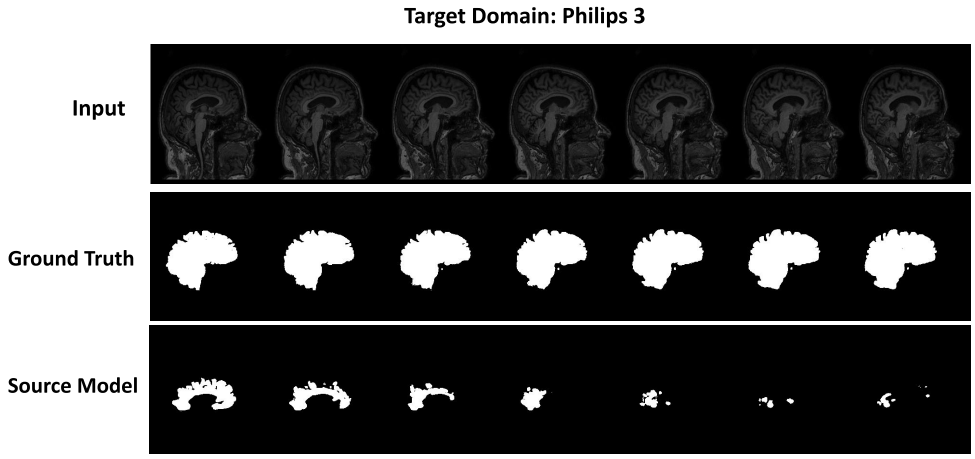


Figure 4.9: Performance of source model without adaptation on the target domain: Philips 3 [11].

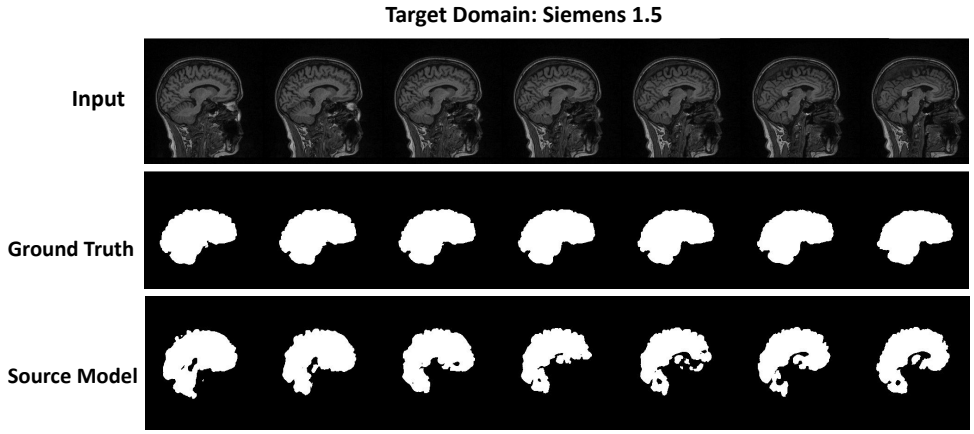


Figure 4.10: Performance of source model without adaptation on the target domain: Siemens 1.5 [11].

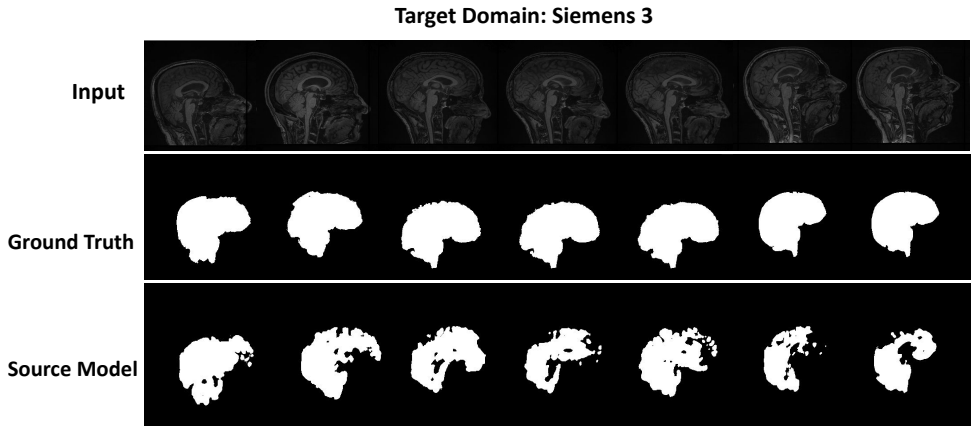


Figure 4.11: Performance of source model without adaptation on the target domain: Siemens 3 [11].

generalization. Table 4.1 presents a quantitative comparison between the proposed method and the baseline [226] on the five target domains. In contrast to the baseline,

which conducted experiments only once, our results are obtained by running the experiments multiple times with different seed values to account for the impact of randomness on the outcomes.

Table 4.1: Evaluation of the source model on the CC359 target domains [11].

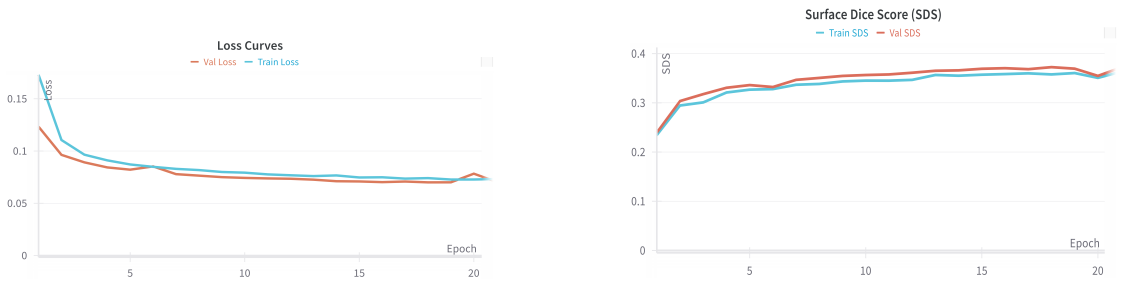
Target Domain	Source Model	
	UDAS [226]	Ours
GE 1.5	0.558	**0.73 ± 0.03**
Philips 1.5	0.749	**0.87 ± 0.02**
Philips 3	0.658	0.61 ± 0.05
Siemens 1.5	0.704	**0.82 ± 0.03**
Siemens 3	0.886	0.84 ± 0.01

The results demonstrate that out of five target domains, superior performance was achieved on three domains, even with the source model. This improvement can be attributed to the pre-processing applied to the CT slices (Section 4.4.2). Furthermore, crucial information regarding the data splits and other hyperparameters is not reported in [226]. It could potentially account for the observed discrepancies in the results. Moreover, in [226], the model adaptation relies exclusively on the training set, which does not accurately reflect whether the predictions are improving. In contrast, our proposed approach employs a separate validation set for robust evaluation, enabling a more reliable assessment of performance based on this set.

4.5.2 Early Segmentation Head Refinement

Following our proposed framework, the next step is to refine the early segmentation head (ESH) using the source domain. ESH is pre-trained using the source domain for 20 epochs. The loss and the corresponding surface dice score (SDS) curves are shown in Figure 4.12a and 4.12b respectively. The SDS is relatively low because the ESH is a compact CNN with only three convolutional layers (discussed in Section 4.3.2 "Early Segmentation Head Refinement"). The primary objective of this stage is to pre-train the ESH on the source domain to establish a robust starting point for subsequent adaptation to the target domain as compared to the random weight

initialization. At this stage, the goal is to prepare ESH for adaptation to the target domain, rather than achieving a high score on the source domain.



(a) **(b)**

Figure 4.12: ESH initialization using the source domain: Loss and surface dice score curves.

4.5.3 ConvLoRA based Parameter Efficient Domain Adaptation

A series of experiments are conducted to evaluate the effectiveness of our proposed ConvLoRA adapters for unsupervised domain adaptation in medical imaging. The first objective was to compare our proposed ConvLoRA adapters with methods that confine adaptation to specific network segments, which we refer to as "constrained adaptation".

In constrained adaptation, only a specific segment of the network is adapted, with all parameters within that segment updated through standard fine-tuning while the rest of the network is frozen. To compare the effectiveness of our proposed parameter-efficient ConvLoRA adapters (Section 4.3.2) against this constrained adaptation approach, ConvLoRA adapters were incorporated exclusively only the initial segment of the U-Net which is shown in Figure 4.2. This configuration is referred to as "constrained ConvLoRA".

The results of this comparative analysis are reported in Table 4.2. It is evident that even in the constrained adaptation setup, our proposed ConvLoRA approach performs on par with constrained adaptation. Notably, ConvLoRA achieves comparable performance with a significantly lower number of trainable parameters. The

reduction in the the number of trainable parameters across various adaptations is reported in Table 4.4. The U-Net model utilized (Figure 4.4) has 14,160 trainable parameters in its initial segment. By incorporating the proposed ConvLoRA adapters into this segment, the number of trainable parameters was reduced from 14,160 to just 3,954. This significant reduction demonstrates that our proposed adaptation is both parameter and computationally efficient.

Table 4.2: Comparative analysis of Constrained adaptation: standard fine-tuning and proposed ConvLoRA adapters.

Target Domain	Base Model	Constrained Adaptation	
		Traditional Fine-tuning	ConvLoRA
GE 1.5	0.729	0.727	**0.736**
Philips 1.5	0.805	0.822	**0.846**
Philips 3	0.624	0.823	0.709
Siemens 1.5	0.793	0.814	**0.820**
Siemens 3	0.830	0.887	0.810

The promising results from the initial experiments highlighted the potential of the proposed ConvLoRA adapters. Building on these findings, the proposed ConvLoRA adapters were integrated into the encoder to achieve parameter-efficient adaptation as it is found to be most susceptible to domain shift (ablation study: Section 4.5.4). To evaluate the effectiveness of the proposed approach, it is compared with various methods:

- **Self-Training** employs pseudo-labels of the target domain to iteratively enhance model performance [227].
- **UDAS:** refers to our baseline which uses self-training to adapt only the initial layers of the network through pseudo-labels [226].
- **UDAS ConvLoRA (Ours):** for a fair comparison with UDAS we incorporated ConvLoRA only to the initial layers.
- **ConvLoRA + AdaBN (Ours):** builds on UDAS but does not restrict adaptation to the initial layers. Instead, the entire encoder is adapted using our

Table 4.3: Comparative analysis of unsupervised domain adaptation approaches.

Target Domain	Source Model	Self-Training [227]	UDAS [226]	UDAS ConvLoRA (Ours)	UDAS ConvLoRA + AdaBN (Ours)
GE 1.5	0.734 ± 0.03	0.530	0.758	0.836 ± 0.038	**0.890 ± 0.019**
Philips 1.5	0.871 ± 0.021	0.725	0.846	0.877 ± 0.005	**0.902 ± 0.010**
Philips 3	0.618 ± 0.005	0.662	0.662	0.719 ± 0.009	**0.825 ± 0.019**
Siemens 1.5	0.825 ± 0.031	0.692	0.824	0.803 ± 0.012	**0.892 ± 0.009**
Siemens 3	0.843 ± 0.012	**0.891**	0.887	0.849 ± 0.002	0.888 ± 0.006

proposed ConvLoRA, with the ESH positioned after the encoder. Additionally, AdaBN is employed to adapt to the target domain's running mean and variance (Section 4.3.1).

Table 4.3 presents a comparative analysis of these methods alongside the proposed approach. When compared to the baseline (UDAS [226]), our method (UDAS ConvLoRA) outperforms in four out of five target domains. While for Siemens 1.5, our method has a slight decrease in SDS (0.2% only) compared to UDAS [226]. It is important to note that the proposed adaptation is achieved with a substantial reduction in trainable parameters, decreasing from 14,160 to just 3,954 — a remarkable 72.07% reduction (Table 4.4).

Additionally, when employing the proposed approach of combining ConvLoRA with AdaBN, it enhanced accuracy for the Siemens 1.5 domain as well as shown in Table 4.3)(UDAS ConvLoRA + AdaBN). The experimental results demonstrate that our proposed approach not only enhances accuracy but also offers significant computational efficiency.

Parameter Efficiency

As shown in Table 4.4, the proposed ConvLoRA-based adaptation achieves a significant reduction in trainable parameters, reducing the original 24.3 million parameters of the U-Net architecture [11] to just 57,714– a 99.80% reduction. Furthermore, when combined with AdaBN (referred to as UDAS ConvLoRA+AdaBN), the ConvLoRA adapter enhances model adaptation and outperforms all other methods, without any additional parameters. This demonstrates that both the standalone ConvLoRA

adapter and the ConvLoRA+AdaBN offer parameter-efficient adaptation while delivering competitive performance compared to other methods.

Table 4.4: Comparison of Trainable Parameters for Different Adaptation Strategies.

Adaptation Strategy	Total Params	Trainable Params	Trainable Params reduction (%)
Full Model Fine-tuning	24.3 M	24.3M	-
Constrained Adaptation	24.3 M	14,160	99.93
Constrained LoRA (Ours)	14,160	**3,954**	72.07
ConvLoRA + AdaBN (Ours)	24.3 M	**57,714**	99.57

Qualitative Results

The qualitative results are shown in Figure 4.13, where the first column displays the target domain input images, the second column shows the corresponding ground truth, the third column illustrates the results achieved with the source model, the fourth column demonstrates the outcomes obtained through constrained adaptation, and the fifth column shows the results attained by our approach. It is evident that the source model (Section 4.3.2 (source model)) does not perform well and lacks generalization as illustrated in column 3. As shown in column 4, constrained domain adaptation (Section 4.5.3) yields improved performance compared to the source model; however, it remains susceptible to the effects of domain shift. It is evident that our proposed method effectively handles the domain shift, surpassing both the base model and the constrained adaptation significantly as shown in the last column. Furthermore, this enhanced performance is attained alongside the added benefit of computational efficiency, further showing our approach's strength (Table 4.4).

Furthermore, we also report the qualitative results after adaptation for the images which were reported to show the impact of domain shift on the source model (Section 4.5.1), as illustrated in Figure 4.14, 4.15, 4.16, 4.17 and 4.18.

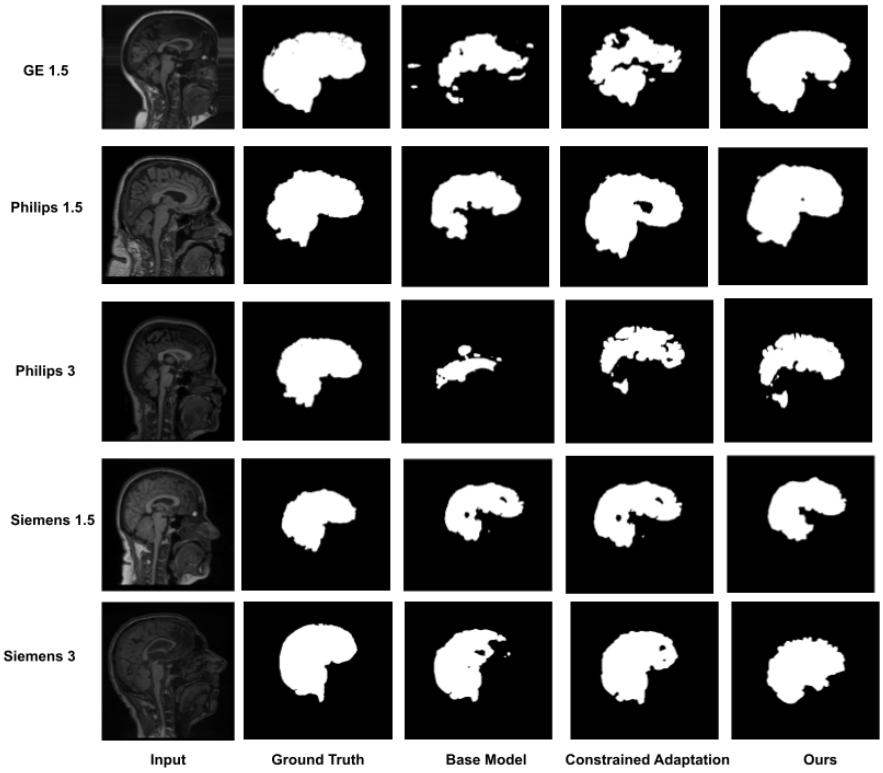


Figure 4.13: Qualitative Results for CC359 [11]

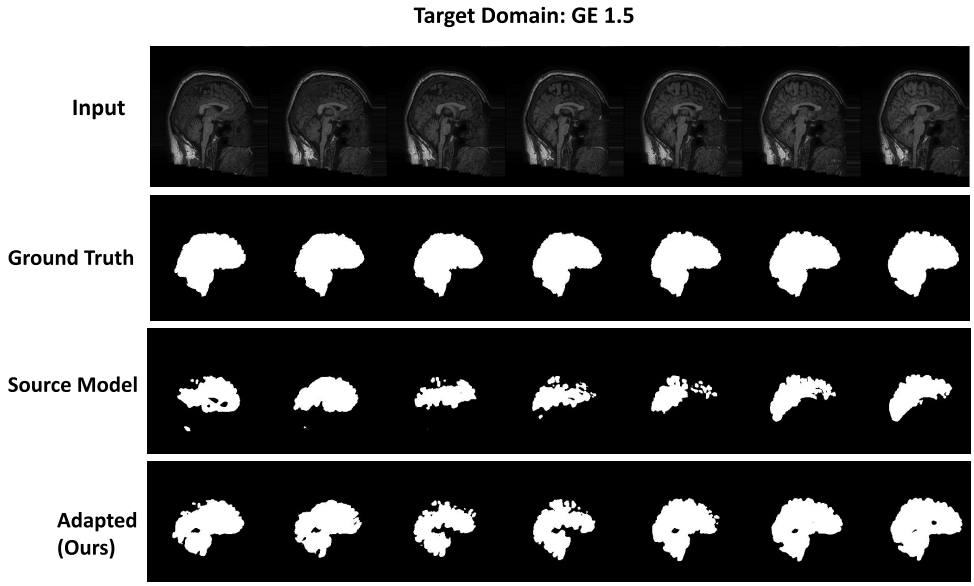


Figure 4.14: Qualitative comparison: source model vs proposed adaptation for target domain: GE 1.5 [11]

4.5.4 Ablations

To identify blocks susceptible to domain shift, the proposed ConvLoRA adapters are integrated into various segments of the network and evaluate the performance.

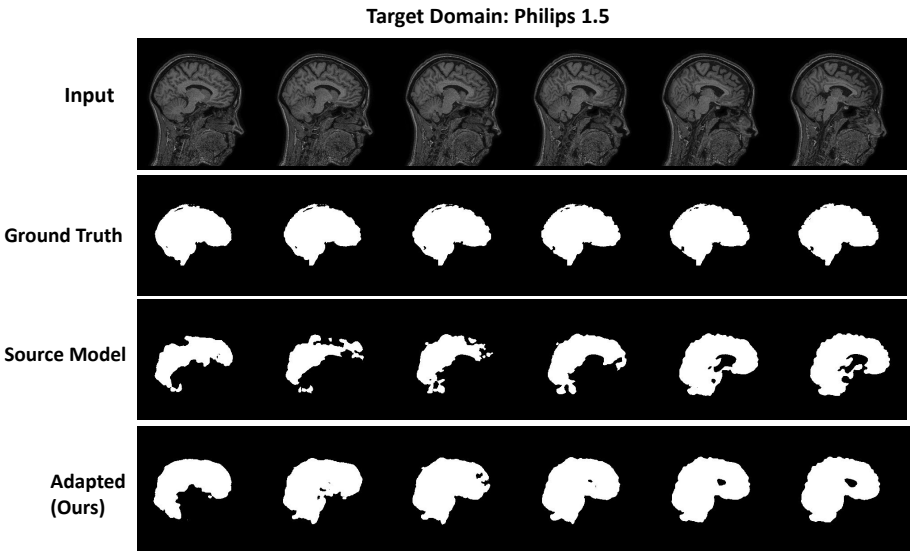


Figure 4.15: Qualitative comparison: source model vs proposed adaptation for target domain: Philips 1.5 [11]

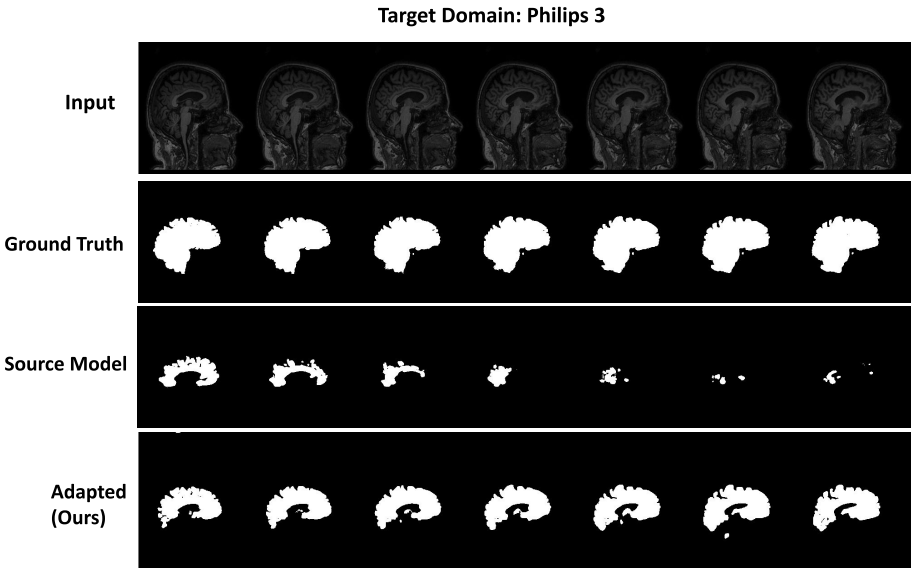


Figure 4.16: Qualitative comparison: source model vs proposed adaptation for target domain: Philips 3 [11].

First, the ConvLoRA adapters are integrated into different segments of the encoder. The U-Net architecture we used has three blocks in the encoder (Figure 4.2). The results of this ablation are reported in Table 4.5. Each column is named "Enc" followed by a number that represents the encoder block. The optimal results were achieved by adapting the full encoder block using the proposed ConvLoRA adapters (Full Enc. Block in Table 4.5). Furthermore, using AdaBN along with ConvLoRA in the encoder, further improved the performance (last column).

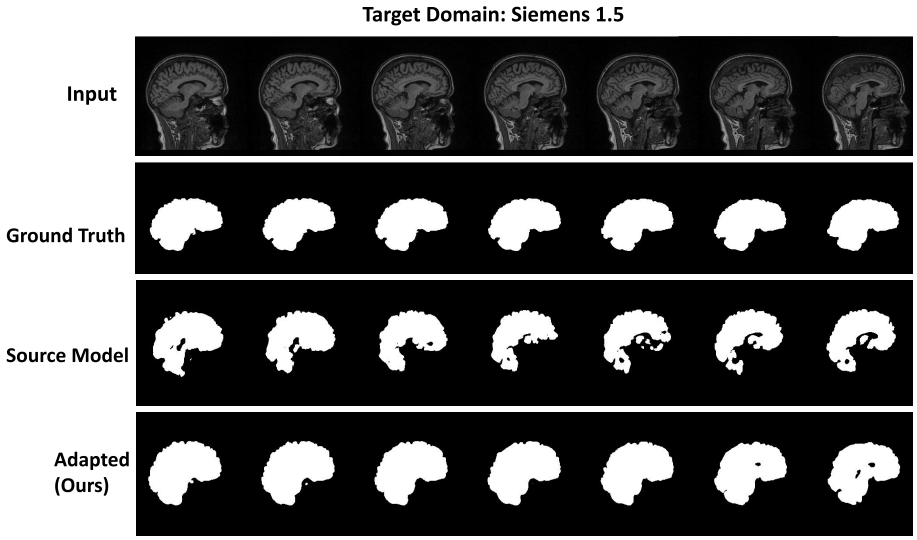


Figure 4.17: Qualitative comparison: source model vs proposed adaptation for target domain: Siemens 1.5 [11].

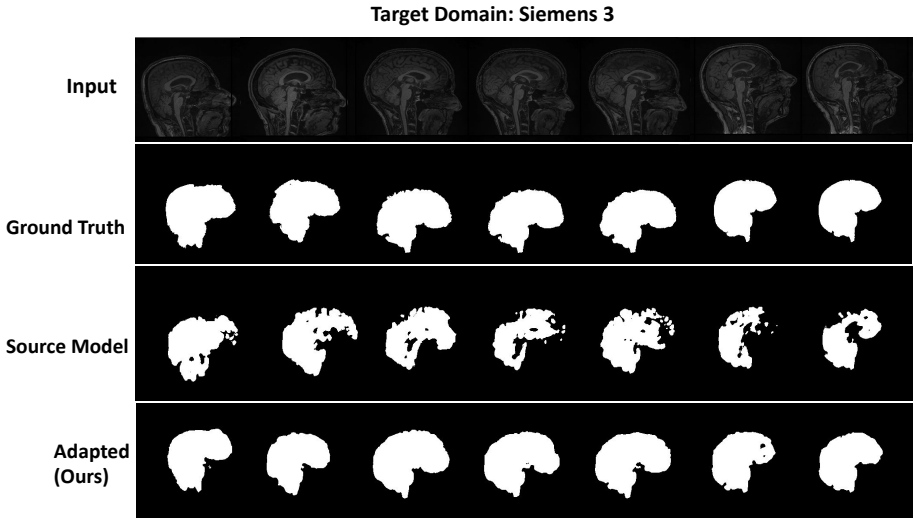


Figure 4.18: Qualitative comparison: source model vs proposed adaptation for target domain: Siemens 3 [11].

To evaluate the potential benefits of integrating ConvLoRA adapters into the decoder, ConvLoRA adapters were incorporated into this part of the network. The evaluation approach followed the same methodology used for assessing ConvLoRA integration into the encoder. The U-Net used three decoder blocks (Figure 4.2). ConvLoRA adapters were incorporated into each of the decoder blocks.

However, unlike the benefits it brought to integration in the encoder, its integration into the decoder proved to be detrimental, resulting in a decline in the surface dice score as reported in Table 4.6. The reason for this decline is that the primary

Table 4.5: Ablation Study: Placement of ConvLoRA adapters in the encoder and respective SDS, (Enc: Encoder).

Target Domain	Enc. Block 1	Enc. Block 2	Enc. Block 3	Full Enc. Block	Full Enc. Block + AdaBN
GE 1.5	0.836 ± 0.038	0.827 ± 0.011	0.808 ± 0.010	0.861 ± 0.044	**0.890 ± 0.019**
Philips 1.5	0.877 ± 0.005	0.832 ± 0.102	0.840 ± 0.038	0.891 ± 0.027	**0.902 ± 0.010**
Philips 3	0.719 ± 0.009	0.738 ± 0.022	0.749 ± 0.014	0.765 ± 0.006	**0.825 ± 0.019**
Siemens 1.5	0.719 ± 0.009	0.852 ± 0.009	0.861 ± 0.028	0.840 ± 0.038	**0.892 ± 0.009**
Siemens 3	0.849 ± 0.002	0.856 ± 0.017	0.868 ± 0.021	0.858 ± 0.013	**0.888 ± 0.006**

function of the decoder in U-Net is to upsample and reconstruct spatial features to match the resolution of the input, effectively translating learned feature representations back into a spatial context [228]. On the other hand, the proposed ConvLoRA adapters, are used to approximate convolutional layers with low-rank decompositions and are optimized for parameter efficiency. Thus the low-rank approximations inherently involve some level of information compression, which can be beneficial in feature extraction stages (like in the encoder) but detrimental in stages where high-resolution details are crucial (like in the decoder) which translates in our results reported in Table 4.6.

Table 4.6: Ablation Study: Placement of ConvLoRA adapters in the decoder and respective SDS, (Dec: Decoder).

Target Domain	Dec. Block 1	Dec. Block 2	Dec. Block 3
GE 1.5	0.631	0.644	0.798
Philips 1.5	0.598	0.790	0.8948
Philips 3	0.598	0.754	0.7565
Siemens 1.5	0.598	0.826	0.8479
Siemens 3	0.849	0.843	0.8449

To evaluate ConvLoRA's effectiveness across the network, a Siamese network is used [229]. A duplicate of the source model, referred as the ConvLoRA model is created. Both models share identical weights, except that ConvLoRA adapters are integrated into the convolutional layers throughout the ConvLoRA model as shown in Figure 4.19.

During adaptation, target domain samples are processed by both the source model and ConvLoRA model. The source model is kept frozen, while only the Con-

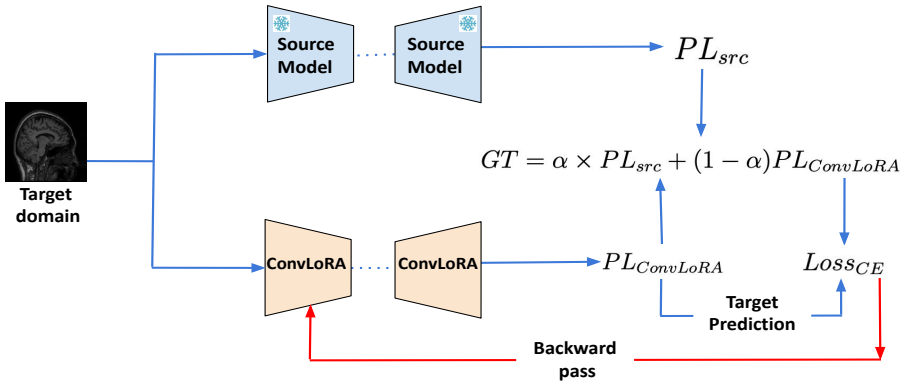


Figure 4.19: ConvLoRA integrated to the entire network (ConvLoRA).

vLoRA adapter parameters in the ConvLoRA model are updated through gradient back-propagation as shown in Figure 4.19. The output from the ConvLoRA model is treated as a target prediction ($PL_{ConvLoRA}$). The ground truth labels are created by mixing pseudo labels from the source model and ConvLoRA model as shown in equation 4.6.

$$GT = \alpha \times PL_{src} + (1-\alpha)PL_{ConvLoRA} \tag{4.6}$$

where α is a hyperparameter that controls the mixup ratio between the two pseudo-labels used to generate the ground truth. PL_{src} represents the pseudo-labels generated by the source model, while $PL_{ConvLoRA}$ refers to the segmentation mask produced by the ConvLoRA model. As training progresses and the ConvLoRA model improves, more weight is given to $PL_{ConvLoRA}$ progressively.

However, applying ConvLoRA across the entire network did not result in better generalization, as shown in Table 4.7. The proposed approach of integrating ConvLoRA in the encoder component (Figure 4.4) termed as "ConvLoRA Encoder", better generalized consistently on all the target domains as reported in Table 4.7.

Furthermore, batch normalization-based adaptation was also evaluated. Specifically domain-specific batch normalization (BN) for unsupervised domain adaptation, following the approach proposed in [210] is used. This method aims to adapt to both source and target domains by using separate batch normalization layers for each domain within convolutional neural networks while keeping all other model parameters

Table 4.7: Impact of ConvLoRA: Comparing Integration into the Full Model vs the Encoder.

Target Domain	ConvLoRA Full Model	ConvLoRA Encoder
GE 1.5	0.778	**0.890**
Philips 1.5	0.598	**0.902**
Philips 3	0.735	**0.825**
Siemens 1.5	0.790	**0.892**
Siemens 3	0.752	**0.888**

shared between the domains. However, the results indicate that this domain-specific BN did not yield performance improvements. Consequently, additional normalization adjustments may be unnecessary or could even have a detrimental impact.

Thus, the experimental results and insights demonstrate that our proposed parameter-efficient unsupervised domain adaptation using ConvLoRA and AdaBN within a self-training framework, not only facilitates better generalization across multiple target domains but also offers significant computational efficiency.

4.5.5 Evaluation on M&M Dataset

Apart from CC359 [11], evaluation is also performed on M&M dataset [12]. Across the three target domains within the M&M (Section 4.4.1) dataset, our proposed adaptation method demonstrated superior performance in two of the target domains in contrast to the source model (baseline- UDAS) as shown in Table 4.8. It highlights the robustness of our approach in handling diverse domain shifts, although further investigation is required to address the performance gap in the third domain.

Table 4.8: ConvLoRA performance of M&M dataset [12] .

Target Domain	Source Model	Self-Training	UDAS	ConvLoRA + AdaBN (ours)
Siemens	0.656 ± 0.095	0.546	0.536	**0.771 ± 0.01**
GE	0.542 ± 0.108	0.373	0.566	**0.655 ± 0.040**
Cannon	**0.654 ± 0.023**	0.664	0.520	0.519 ± 0.101

Following the approach used in the analysis of the CC359 dataset, a thorough

qualitative evaluation is performed on the M&M dataset [12] to assess the adaptability of the proposed approach. The qualitative results are shown in Figure 4.20, 4.21 and 4.22. These figures illustrate the comparison of the segmentation maps achieved by the proposed adaptation and the ground truth. The first row shows the input image, the second row displays the ground truth, and the third row presents the adaptation results obtained using the proposed approach.

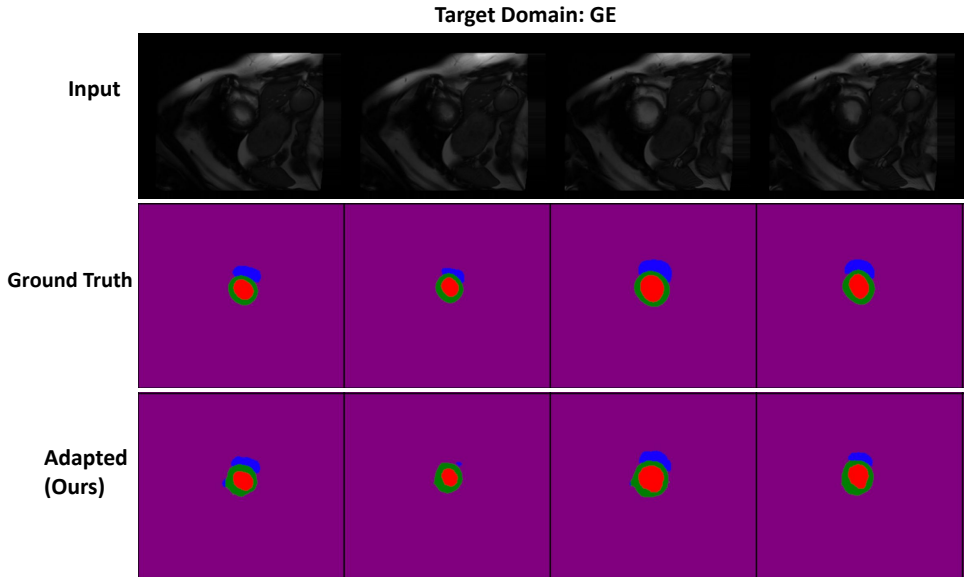


Figure 4.20: Qualitative results target domain: GE [12].

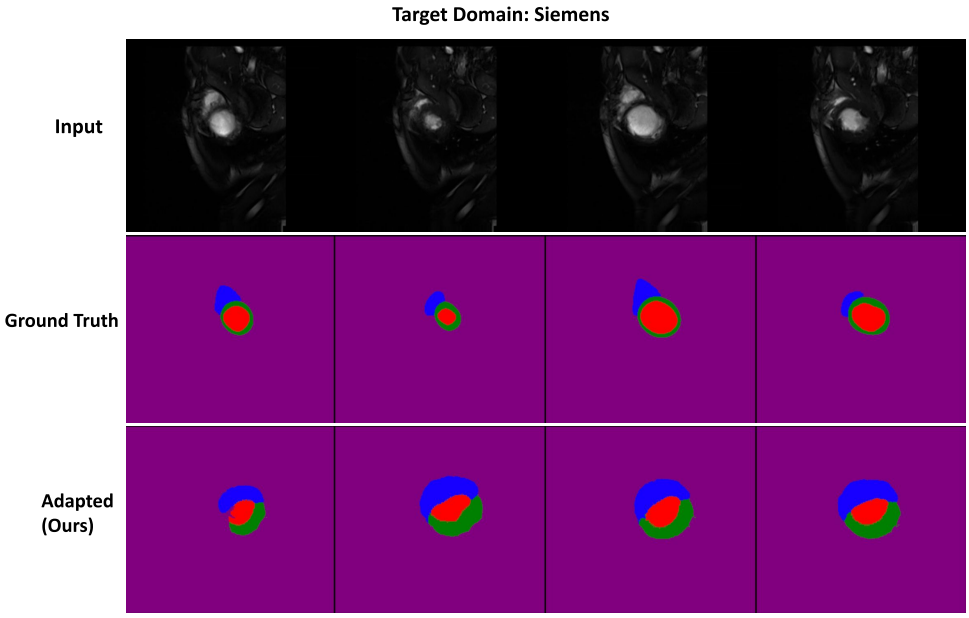


Figure 4.21: Qualitative results target domain: Siemens [12].

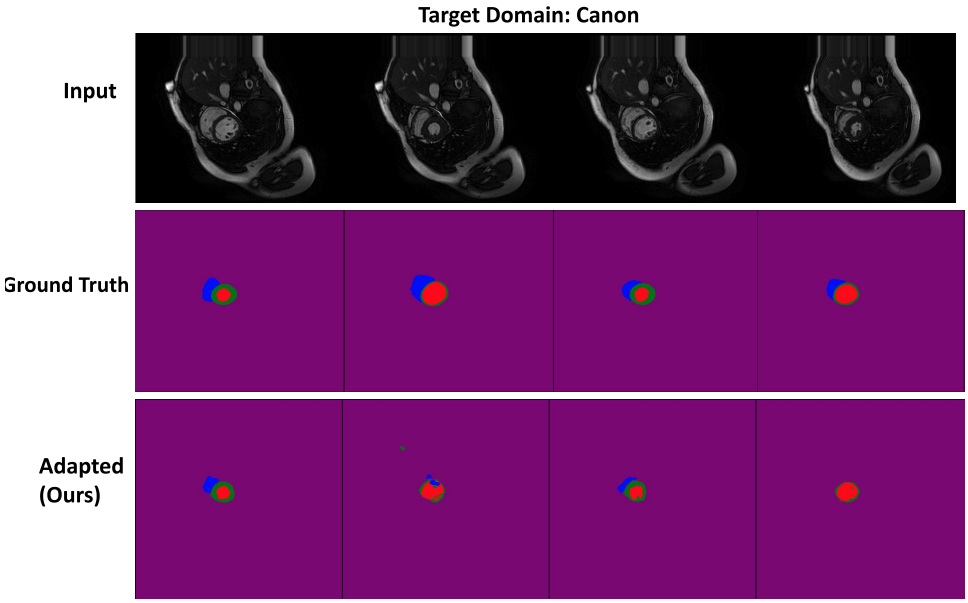


Figure 4.22: Qualitative results target domain: Canon [12].

4.5.6 Analysis and Limitation

The results on the M&M dataset [12] as discussed in the above section highlight a key limitation of the proposed approach: its effectiveness depends on the characteristics of the dataset under consideration. Specifically, the performance of ConvLoRA varies based on the complexity of the dataset.

For the CC359 dataset, the objective is to perform skull stripping on brain MRI scans [11]. The robustness of the proposed approach on CC359 is comprehensively demonstrated through the comprehensive experiments discussed in Section 4.5.3. The proposed approach consistently delivers accurate segmentation results across all domains, regardless of domain shift in the form of the scanner manufacturer and field strength.

In contrast, M&M dataset [12], is a multi-class segmentation dataset and the region of interest is small and subtle as shown in Figure 4.6. This dataset presents a higher level of complexity, featuring three regions of interest. The challenge lies in accurately segmenting these multiple anatomical structures while managing domain shifts. This increased complexity leads to reduced performance compared to the CC359 dataset [11], thereby diminishing the overall effectiveness of the proposed

approach on the M&M dataset.

These findings indicate that while the proposed unsupervised adaptation approach shows promise in adapting to different domains and reducing the computational overhead of supervised methods (Section 4.5), its effectiveness is influenced by the specific characteristics of the dataset. This highlights the need for further optimization and refinement of the proposed approach to improve its generalization to tackle more challenging domain adaptation tasks in medical imaging.

4.6 Summary

This chapter presents our work on unsupervised domain adaptation for multi-target medical imaging domains. A parameter-efficient, convolutional low-rank unsupervised adaptation approach is proposed for adapting convolutional neural networks (Section 4.3.2). Unlike existing methods that focus solely on single-target domain adaptation, our method addresses domain shifts across multiple target domains, enhancing its robustness and applicability in diverse medical imaging settings. It overcomes the limitations of supervised adaptation methods, which typically create separate fine-tuned models dedicated to each target domain and rely heavily on annotated data for supervision.

The proposed approach is evaluated through brain segmentation in brain MRI scans and cardiac structures segmentation from cardiac MRI data. For the first task, it achieved an average dice score of 0.881 across five target domains in the CC359 dataset (Section 4.4.1, 4.5.3). The source U-NET model that is used in this work has 24.3 million parameters (Section 4.3). Our proposed approach achieved these results by adapting only 57,714– a 99.80% reduction. Thus it makes our method computationally efficient, significantly reducing the number of trainable parameters compared to supervised approaches (Section 4.5.3), which adapts the entire model (24.3 million parameters in this case).

For cardiac structures segmentation from MRI scans, an average dice score of 0.75 was achieved across the different target domains in the M&M dataset (Section 4.4.1).

Unlike brain segmentation, the M&M dataset presents a greater challenge due to the complexity of cardiac structures, making accurate segmentation more difficult. Despite this, the experimentation with the M&M dataset [12] helped gain the valuable insights highlighted in the following section.

4.6.1 Insights

While our proposed approach is effective, it has a few limitations as discussed in the above section. To overcome these limitations, there is a need for more robust domain adaptation approaches.

Recent advancements have led to the emergence of "foundation models". These models use large-scale datasets and rely predominantly on self-supervised learning for pre-training [141, 230, 15, 231, 232], enabling better generalization and adaptability across diverse visual tasks [233, 234, 235]. These advancements have changed the research paradigm from the conventional fine-tuning of CNNs to the zero-shot transferability of foundation models. Due to these advancements, the traditional fine-tuning approaches are increasingly being replaced by prompting techniques [233, 234, 235, 236, 237, 37].

These factors motivated us to explore foundation models for domain adaptation with three key objectives:

1. Can the limitations of the proposed parameter-efficient unsupervised adaptation be addressed using foundation models?

2. Is it possible to effectively adapt foundation models trained on natural images to the medical imaging domain fully during test time, without any additional training or fine-tuning?

3. Can visual and language foundation models be aligned for test-time adaptation to alleviate the need for domain knowledge/expertise for medical imaging tasks?

These objectives have further helped shape the research focus and will be explored in depth in the following chapters.

Chapter 5

Test Time Domain Adaptation of Foundation Models for Medical Image Segmentation

This chapter presents our work on test-time adaptation of foundation models to the medical imaging domain. It addresses Research Question 3 (RQ3) of this thesis: "Can test-time adaptation of foundation models provide a more robust alternative to supervised or semi-supervised domain adaptation approaches? Can foundation models be effectively adapted to diverse medical imaging tasks without relying on annotated data, additional training, or specialized domain expertise?" In this context, a novel framework called SaLIP is proposed to adapt foundation models predominantly trained on natural imaging datasets to perform diverse medical imaging tasks. Notably, SaLIP is a test-time adaptation framework that facilitates foundation models adaptation to the medical domain fully at test time, without the need for additional training, fine-tuning, or annotated data. Thus it addresses the challenges posed by the scarcity of medical data and lack of domain expertise in the medical domain. Additionally, SaLIP also addresses the limitations of our proposed unsupervised parameter-efficient domain adaptation through Convolutional Low-Rank Adaptation (ConvLoRA) and Adaptive Batch Normalization [25] (introduced

in Chapter 4). Our proposed SaLIP pipeline is evaluated across diverse medical segmentation tasks: brain segmentation from MRI scans, lung segmentation from chest X-rays, and fetal head segmentation from ultrasound images. The work presented in this chapter has been published in the Computer Vision And Pattern Recognition Conference, Workshop (CVPRW), 2024 [121]. The code to replicate the experiments and results is publicly available at: https://github.com/aleemsidra/SaLIP.

Section 5.1 provides an introduction and the motivation behind adapting foundation models to diverse downstream medical imaging tasks through test-time adaptation. Section 5.2 presents a comprehensive literature review of existing approaches that utilized foundation models for medical imaging tasks. Section 5.3 provides a detailed overview of our proposed framework and its architectural design. Section 5.4 outlines the datasets and experimental evaluation setup. Section 5.5 presents the qualitative and quantitative results, along with ablation studies. Finally, Section 5.6 summarizes the findings of this work and highlights how these findings guided the subsequent research work discussed in Chapter 6.

5.1 Introduction

Segmentation is a crucial task in medical imaging analysis. It focuses on identifying and delineating regions of interest (ROI) in various medical imaging tasks. Depending on the imaging modality, these ROI may include organs, lesions, or tissues [62]. Accurate segmentation is vital for clinical applications such as disease diagnosis, treatment planning, and monitoring disease progression [238, 22, 239]. Deep learning models have demonstrated significant potential in medical image segmentation, due to their ability to learn complex image features and provide highly accurate segmentation results. These models excel across a wide range of tasks from segmenting specific anatomical structures to identifying pathological regions [240].

However, a significant limitation of many current medical image segmentation models is their task-specific nature. These models are typically developed and trained for a specific task, which hampers their generalizability across different do-

mains. As a result, their performance can degrade substantially when applied to new target domains, due to domain shift (Sections: 1.1.1, 1.1.2, 1.1.3). The impact of task-specific nature of the model on its generalizability is also demonstrated in the experimental sections of Chapter 3 and Chapter 4, where task-specific fine-tuning was carried out for each dataset. This restricted form of supervision limits their generalizability and usability, creating a significant barrier to the broader application of these models in clinical practice.

Large language models (LLMs) pre-trained on web-scale datasets are revolutionizing natural language processing with impressive zero-shot and few-shot generalization [237]. These "foundation models" [241] have exceptional ability to generalize to new tasks and data distributions beyond those seen during training. This capability is often implemented using "prompt engineering" where carefully designed text prompts guide the model to generate appropriate responses for a given task. When scaled and trained with abundant text corpora from the web, the zero and few-shot performance of foundation models compares surprisingly well (and even matches in some cases) with fine-tuned models [237, 242]. Empirical evidence shows that this performance improves consistently with increasing model size and dataset diversity [237, 243, 244, 241].

Foundation models, characterized by their substantial size and self-supervised training on diverse datasets, possess remarkable capabilities for generating meaningful representations across multiple domains [241]. These models provide significant advantages, including effective parameter initialization for a wide range of downstream tasks. These models have transformed the landscape of machine learning. Traditional fine-tuning approaches are increasingly being replaced by prompting techniques [233, 234, 235, 236, 237, 37].

The field of computer vision is arguably currently undergoing a similar transformation. Notably, vision-language foundation models have demonstrated exceptional zero-shot capability and strong generalization across a wide range of applications [233, 234, 235]. Visual foundation models can be broadly classified into two cat-

egories: feature encoding models, trained on pretext tasks (e.g., DINO [232, 232, 53], CLIP [15], and BLIP [137]), and models designed for specific tasks, such as segmentation e.g., SAM [13] and SEEM [131].

The transferability of visual language models (VLMs) has shown significant performance improvements to various natural imaging tasks, such as open-vocabulary detection [245, 246], visual grounding [247, 248], and image editing [249, 250]. Typically to adapt VLMs, either prompt engineering is used to facilitate zero-shot transferability, or additional training is conducted to adapt to each specific downstream task [51]. However, medical imaging tasks face significant challenges, primarily due to data scarcity arising from privacy and ethical concerns, as well as the need for domain expertise in prompt engineering [251, 252]. Additionally, domain shift between the data used to pre-train foundation models and the target medical tasks presents further challenges [141].

Current methods for adaptation of foundation models to the medical domain, primarily focus on supervised/parameter-efficient unsupervised approaches (Section 5.2). However, the potential of zero-shot transfer to medical imaging tasks, where foundation models are applied without additional training or fine-tuning remains largely under explored. Developing a framework for effective zero-shot test-time transferability of foundation models could address several key challenges in the medical field, including the need for additional supervised training, data scarcity, task-specific fine-tuning, and specialized domain expertise in prompt engineering.

Segment Anything Model (SAM) is the first promptable segmentation model, pre-trained on a vast dataset of over 1 billion masks. It enables SAM to adapt effectively to a wide range of downstream tasks using interactive prompts [13]. SAM can be utilized in two modes: either to segment everything in an image or to segment a specific region based on the prompts as shown in Figure 5.1. The architectural design of SAM and the respective modes are discussed in detail in Section 5.3.1. SAM has shown impressive results in a broad range of tasks for natural images but its performance has been subpar when directly applied to medical images [58, 56, 126, 57].

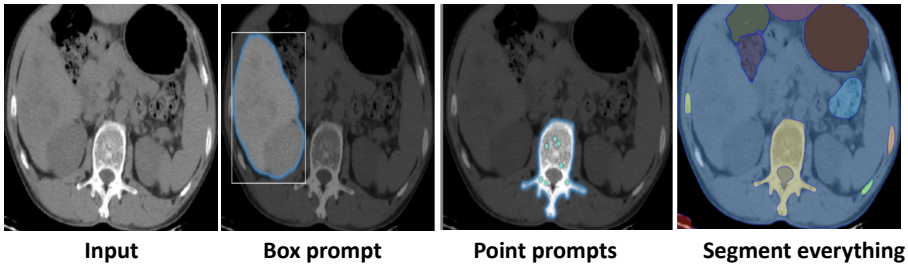


Figure 5.1: Various segmentation modes of the Segment Anything Model.

CLIP (Contrastive Language-Image Pretraining) model, developed by OpenAI, is trained on a dataset of over 400 million image-text pairs. This extensive training enables CLIP to learn rich associations between visual and textual information [132]. By establishing a joint text-vision embedding space, CLIP has been effectively utilized for zero-shot classification, recognition, and retrieval tasks in the natural imaging tasks [18, 120, 253, 51]. The architectural details of CLIP are discussed in detail in Section 5.3.1.

SAM and CLIP have shown remarkable zero-shot transfer capabilities in various downstream tasks for natural images. Despite their success in natural image applications, the combined potential of SAM and CLIP in the complex and challenging medical imaging domain remains largely unexplored [121]. Investigating this potential could lead to significant advancements in medical image analysis, where accurate segmentation and recognition are critical for diagnosis and treatment.

While SAM can effectively segment different regions of an image using prompts like bounding boxes and point prompts (Figure 5.1), it faces inherent limitations to its application in medical image segmentation. One of the key challenges is its reliance on prompts to identify and segment specific regions, which means that the quality of the segmentation results is directly influenced by the prompts used. In medical imaging domain, the effective prompt engineering requires domain expertise or access to annotated data, both of which are often scarce. The limited availability of high-quality labeled medical datasets and the need for specialized knowledge complicates the prompt engineering process in the medical imaging domain.

To address these challenges, several studies have combined SAM with foundation

models such as GroundingDINO [248] and YOLOv8 [254] to create prompts, such as bounding boxes for regions of interest. However, these models are mainly trained on natural imaging datasets, which can lead to generalization challenges due to domain shift when applied to medical datasets (Section 1.1.2). Importantly, to effectively use these models for prompt generation, additional training is required to optimize them for medical imaging tasks [255, 256]. The application of these foundation models in medical imaging faces several challenges:

a) Annotated data is required for fine-tuning and training foundation models, yet such data is often scarce in the medical domain (Section 1.1.4). The model's performance heavily depends on the amount of training data, and it needs careful evaluation and experimentation.

b) Foundation models like GroundingDINO [248] and YOLOv8 [254] need additional textual prompts to generate prompts describing regions of interest (ROIs). The performance of these can vary greatly depending on the quality of these prompts. Creating effective prompts for medical tasks demands specialized domain expertise, which is not readily available in the field of medical imaging. As a result, the lack of specialized knowledge often leads to ineffective prompt engineering.

c) The considerable computational overhead of training foundation models further adds complexity.

Additionally, while SAM's ability to automatically "segment everything" in the image (Figure 5.1), is appealing, there are significant challenges associated with the application of SAM's everything mode to medical imaging [257]. One of the main challenges lies in the inherent variability of required segmentation tasks. For example, when analyzing a CT image of liver cancer, the segmentation task can differ depending on the specific clinical scenario and desired degree of granularity. One clinician may be focused on segmenting the liver tumor, whereas another may require segmentation of the entire liver along with the surrounding organs. Additionally,

clinicians are primarily interested in analyzing specific anatomical organs such as the liver, kidneys, spleen, lesions, etc. It becomes challenging to discern and focus on ROI amidst the growing number of segmented areas. These such challenges impede the direct application of SAM to medical image segmentation.

To overcome the challenges mentioned above, this work proposes a novel unified framework called SaLIP, which harnesses the strengths of SAM and CLIP for zero-shot organ segmentation. SAM can effectively perform organ segmentation when prompts are provided. However, its effectiveness hinges on domain expertise and annotated data for prompt engineering, which is not readily available in the medical domain.

To circumvent these challenges, the proposed framework SaLIP, initially employs SAM's *everything mode* to automatically segment every region within the image as illustrated in the general overview of SaLIP shown in Figure 5.2. This mode operates without any external prompts and does not need manual prompt engineering (Section 5.3.1 (modes of SAM)). While SAM's *everything mode* generates exhaustive segmentation masks for different regions in the image, the resulting masks do not include semantic labels (Section 5.3.2). To extract the relevant ROI mask from the pool of SAM generated masks, the original input image is cropped based on each of these masks. This set of cropped image regions is then processed through CLIP, which retrieves the crop corresponding to ROI in a zero-shot manner using visually descriptive text (VDT) sentences from GPT-3.5, related to the target organ, following the approach proposed in [120] (Appendix A.1). Finally, the retrieved ROI crop is used to generate bounding box prompts. These prompts are eventually used to prompt SAM to segment specific organs as illustrated in Figure 5.2. To evaluate the effectiveness of SaLIP, a thorough experimental evaluation is carried out across three diverse medical imaging datasets encompassing MRI scans, ultrasound, and X-ray (Section 5.5).

Contributions

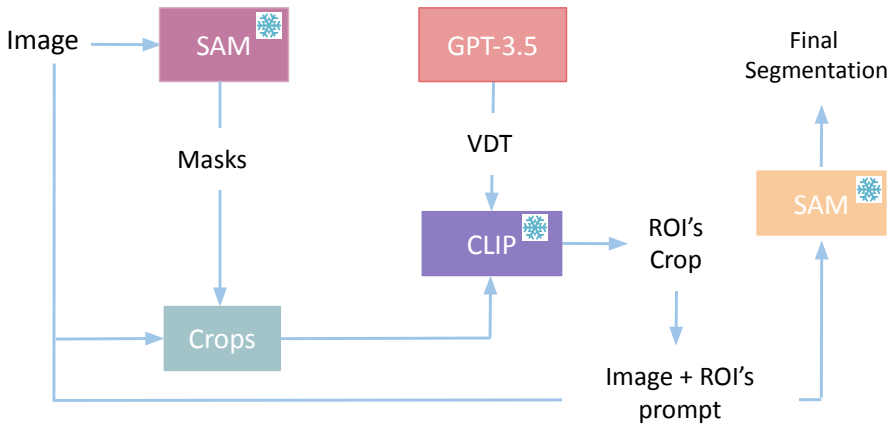


Figure 5.2: The proposed SaLIP framework: The input image is processed through SAM's "everything mode", generating a set of masks for potential regions in the image. The image is then cropped based on the mask coordinates and passed to CLIP's image encoder. GPT-3.5 is used to generate visually descriptive sentences (VDTs) for target ROI. The retrieved ROI crop from CLIP is used to generate a bounding box prompt based on the coordinates of the ROI. This prompt and the input image are then passed to SAM's probabilistic segmentation for final segmentation masks.

- A simple unified framework – SaLIP, is proposed that leverages the combined capabilities of SAM and CLIP for medical image segmentation. It is experimentally demonstrated that the cascade of these foundational models via the proposed SaLIP framework can enhance zero-shot segmentation accuracy in medical imaging.

- SaLIP is training/fine-tuning free and is independent of the specialized domain expertise and labeled data required for prompt engineering.

- To effectively address above mentioned challenges, associated with applying SAM directly to medical imaging and to optimize its utilization for medical image segmentation, both segment everything and promptable segmentation modes of SAM are used. To the best of our knowledge, this is the first work to explore SAM's dual modes for zero-shot medical image segmentation.

- SaLIP is adapted fully at test-time for zero-shot medical image segmentation, thereby efficiently alleviating the training costs and computational overhead associated with these foundation models.

5.2 Related Work

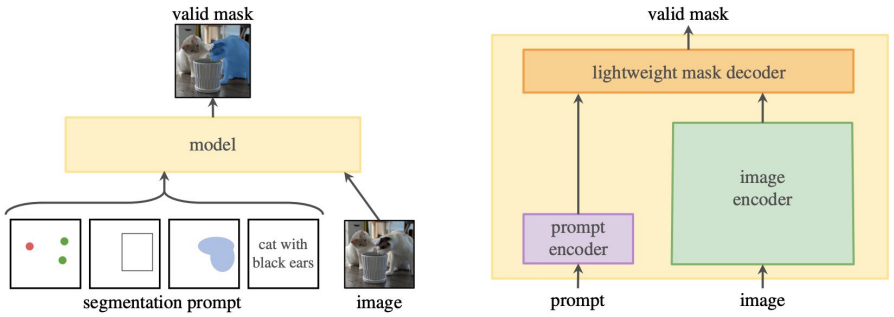
5.2.1 Segment Anything Model

The Segment Anything Model (SAM) is the first promptable segmentation foundation model, trained on the large-scale SA-1B dataset [13]. This extensive training equips SAM with exceptional zero-shot generalization capabilities, enabling it to effectively perform zero-shot segmentation on new data distributions by using various prompts. A prompt is a cue or instruction that guides the model in performing a specific task. It helps define the desired output of the model by providing additional context. For image segmentation, a prompt might specify which object or region in an image to focus on, enabling the model to generate a relevant response or action. There are various types of prompts, which may include spatial information such as a bounding box, point, or mask, as well as textual descriptions that identify an object. Based on the provided prompts, SAM generates a valid segmentation mask, as illustrated in Figure 5.3(a).

SAM has three components: an image encoder, a prompt encoder, and a mask decoder. SAM utilizes a transformer-based architecture [193], which has proven to be highly effective in natural language processing [237] and image recognition tasks [258]. Specifically, SAM adopts an image encoder based on Vision Transformer (ViT) [258] to extract image embeddings, the prompt encoder is used to integrate user interactions via different prompt modes, and a lightweight mask decoder to predict segmentation masks by fusing image embeddings and prompt embeddings as shown in Figure 5.3(b). The architectural details of all three components are discussed in detail in Section 5.3.1.

5.2.2 Adapting SAM for Medical Image Segmentation

Although SAM has shown impressive performance with natural images, its effectiveness in medical image segmentation is limited due to unique challenges arising from domain shift, such as complex anatomical structures, low contrast, and inter-



(a) **Task**: promptable segmentation (b) **Model**: Segment Anything Model (**SAM**)

Figure 5.3: Components of the Segment Anything Model [13].

observer variability (as discussed in Section 1.1.2, 1.1.3). The following approaches have been investigated for the adaptation of SAM to medical image segmentation.

Fine Tuning based Adaptation

The most straightforward approach for adapting SAM to medical image segmentation is to directly fine-tune it for the specific task. Hu et al. [259] conducted a fine-tuned SAM for skin cancer segmentation, showing a significant improvement in the dice similarity coefficient (DSC) from 81.25% to 88.79%. PolypSAM is designed specifically for segmenting polyps in the colon [260]. This method fine-tunes all components of the SAM. The components of SAM are discussed in Section 5.3.1. This approach achieved performance on five public datasets with dice scores all above 88%. MedSAM [257] is introduced for universal medical image segmentation. It adapted SAM by curating a diverse and comprehensive dataset containing more than one million medical image mask pairs of 11 modalities. MedSAM surpasses the performance of the U-Net models.

Updating all parameters of SAM is a time-consuming, computationally intensive, and challenging process, making it less feasible for widespread deployment. Consequently, many researchers focus on fine-tuning a small fraction of the parameters of SAM using various parameter-efficient fine-tuning (PEFT) techniques. SAMed [130] adopts a low-rank-based fine-tuning strategy (LoRA) [10] and trains a default prompt for all images in the dataset. Medical SAM Adapter (MSA) [55] uses adapter modules for fine-tuning.

While the fine-tuning methods for SAM demonstrate great potential, they require substantial labeled data for supervised training and have not yet fully leveraged the prompting ability of SAM (discussed in Section 5.3.2), which is the main strength of SAM. Moreover, the effectiveness of PEFT approaches is reliant on the type of the dataset and segmentation task in hand as experimentally demonstrated in Chapter 4. Furthermore, as a foundation model, SAM has computational overload during fine-tuning due to its substantial size, and high resource requirements and is prone to overfitting when trained with limited data.

In contrast, our proposed approach effectively overcomes these challenges by adapting SAM for zero-shot organ segmentation without additional training, task-specific fine-tuning, and prompt engineering. Our framework facilitates the adaptation of foundation models entirely at test time, which helps to mitigate issues related to the scarcity of medical data, lack of specialized domain expertise for prompt engineering, and the computational overload typically associated with supervised adaptation of SAM to downstream tasks.

Auto Prompting Adaptation

SAM typically requires high quality prompts (i.e., points, boxes, and masks) to achieve effective segmentation performance in medical image segmentation tasks. These prompts are typically generated from the ground truth annotations [55, 257, 261]. However, creating accurate and reliable prompts requires domain-specific knowledge. It is particularly challenging in the context of medical imaging, as domain expertise is often not readily available and difficult to obtain. To tackle these challenges, several methods have employed automatic prompt generation techniques to create prompts that can be provided to SAM for segmentation.

The YOLOv8 model [262] is employed to identify the regions of interest in X-rays, CT scans, and ultrasound images [255]. The bounding box of the identified region of interest i.e. lungs, brain, and the fetal head is passed as a prompt for SAM, enabling fully automated segmentation. Grounding DINO [248] is used to detect

the bounding box corresponding to the polyp in colon [256]. For this purpose, first, the textual prompt describing "polyp" has to be provided to Grounding DINO. MedLSAM [263] utilizes a few-shot localization process to identify 3D bounding boxes around anatomical structures of interest in 3D medical images, based on the premise that images with similar pixel distributions correspond to the same region across different individuals. From these 3D boxes, 2D boxes are projected onto each slice, directing SAM to automatically segment the target anatomy.

However, object detection foundation models like YOLOv8 [262] and GroundingDino [248] are highly sensitive to the textual prompts provided to generate the bounding prompts for ROI. This prompt engineering requires specialized domain knowledge particularly in the medical domain. Engineering effective prompts to recognize the object needs a lot of evaluation and testing. It is even more challenging in the medical imaging task which often leads to the failure to recognize the correct ROI. Moreover, these models are optimized for natural imaging tasks and cannot be used directly; they require training to optimize their performance for medical imaging like polyp segmentation [262].

In contrast, our proposed method facilitates zero-shot test-time adaptation for organ segmentation in medical imaging without requiring specialized domain knowledge for prompt engineering. Instead, it effectively adapts SAM to medical imaging segmentation by harnessing the capabilities of both SAM's segment everything mode and promptable mode (discussed in Section 5.3.1), using CLIP as the bridge between the two (Section 5.3.2).

5.2.3 Contrastive Learning Image Pre-training

CLIP [15] is a pre-trained large visual language model known for its strong generalizability and impressive zero-shot domain adaption capabilities. In CLIP, the classifier is constructed by plugging the class name into a predetermined prompt template like 'a photo of {class name}' [15].

Prompt engineering is an effective technique for generating prompts that adapt

CLIP to various domains by typically incorporating relevant semantic details related to the specific target task [120]. CLIPSeg [264] extends the CLIP model with a transformer-based decoder that facilitates dense prediction. MedCLIP [133] fine-tunes the CLIP model by separating medical images and texts to expand the available training data exponentially at a low cost. CXR-CLIP [265] improves its performance in chest X-ray classification tasks by fine-tuning the CLIP image and text encoders using samples from image-text and image-label datasets. These methodologies require supervised fine-tuning of medical image-text pairs. Other studies such as [266, 267, 268] have demonstrated that incorporating text embeddings learned from CLIP into medical segmentation models achieves state-of-the-art results.

However, these medical image-text pairs are collected under guidelines and with the support of domain experts. Although Aa few publicly available datasets contain medical image-text pairs such as RadImageNet [46], MIMIC-CXR [47], CheXpert [48], and ARCH [49]. However, these datasets are primarily skewed toward radiology.

To generate textual prompts for medical organs for CLIP's textual encoder, our proposed framework employs prompt ensembles of visually descriptive sentences (VDTs) generated using GPT-3.5 for each class [120] (Appendix A.1.1).

5.3 Methodology

This section first presents the preliminaries, including the Segment Anything Model and Contrastive Learning Image Pre-Training in Section 5.3.1. Following this, our proposed SaLIP framework is discussed in detail in Section 5.3.2.

5.3.1 Preliminaries

Segment Anything Model (SAM)

SAM is a prompt-driven segmentation foundation model. It consists of three main components: an image encoder, a prompt encoder, and a lightweight mask decoder

as illustrated in Figure 5.4.

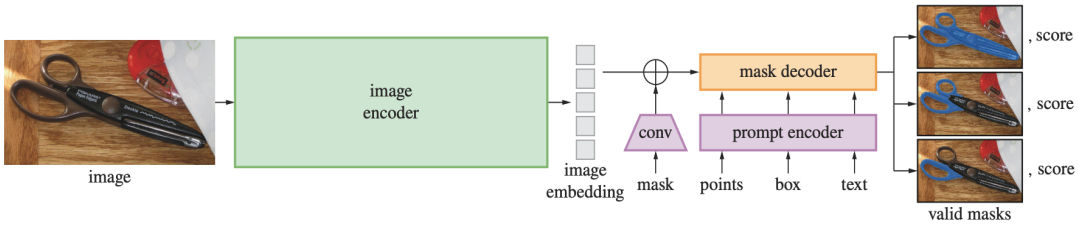


Figure 5.4: Architecture of Segment Anything Model [13].

1. **Image Encoder:**

   At the highest level, SAM employs a pre-trained masked auto-encoder (MAE) is used as its image encoder [14]. MAE has demonstrated outstanding recognition capabilities in natural imaging. During the training of MAE, a large random subset of image patches (approximately 75%) is masked out. The encoder processes a small subset of visible patches. The full set of encoded patches and mask tokens is processed by a lightweight decoder that reconstructs the original image in pixels as shown in Figure 5.5.

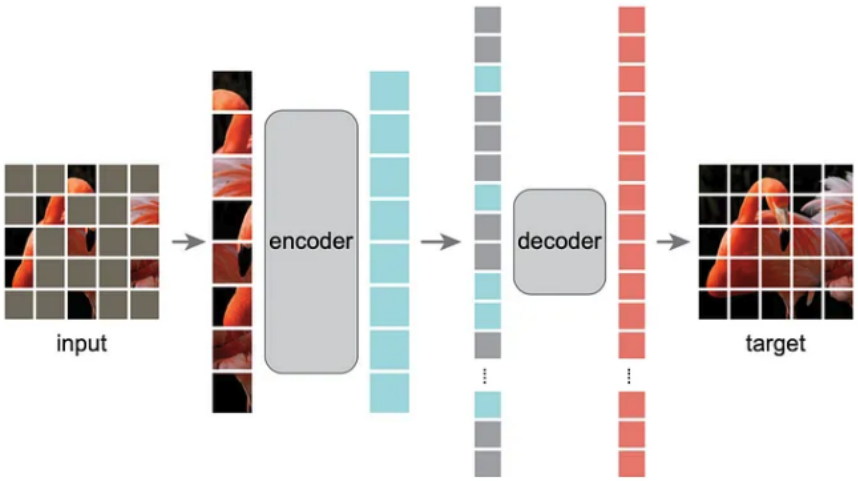


Figure 5.5: Architecture of Masked Auto-encoder [14].

After pre-training, the decoder is discarded and the encoder processes uncorrupted images (full sets of patches) for recognition tasks.

SAM uses the pre-trained encoder from MAE to generate image embeddings as shown in Figure 5.4. These embeddings are generated only once per image and

produced before prompting the SAM model allowing for seamless integration into the segmentation process.

2. **Prompt Encoder:**

   The prompt encoder can encode various types of prompts, such as background points, masks, bounding boxes, and textual inputs, into an embedding vector in real-time [13]. SAM supports two sets of prompts: sparse (points, boxes, text) and dense (masks). Points and boxes are encoded using positional encoding [269] and summed with learned image embeddings from pre-trained MAE auto-encoder (discussed in the above section), for each prompt type. Dense prompts (i.e., masks) are embedded using convolutions and summed element-wise with the image embedding as shown in Figure 5.4.

3. **Mask Decoder:**

   A lightweight mask decoder predicts segmentation masks by utilizing the embeddings generated from both the image and prompt encoders, as illustrated in Figure 5.4. SAM employs a modified decoder block followed by a dynamic mask prediction head, drawing inspiration from existing Transformer decoder blocks [193, 270, 271]. The modified decoder block uses prompt self-attention and cross-attention in two directions (prompt-to-image embedding and vice-versa) [13].

Having described all the components of SAM, the next step is to explain how SAM works: Let an input image be $I \in R^{H \times W \times 3}$ and input visual prompt be $P \in R^N$, where $H \times W$ are the spatial dimensions and N is the number of prompts. The SAM's image encoder encodes an image into dense features: F_{SAM}. The prompt encoder encodes prompts P into sparse prompts Q_{sp}. P can either be sparse, such as points, boxes, or text, or dense, like masks as shown in Figure 5.3. The points and boxes are represented by positional encoding [269] summed with learned embeddings for each prompt type. Currently, SAM does not directly process text prompts and the text-to-mask task is still in its exploratory stages and is not entirely robust [13].

The mask decoder efficiently maps the encoded image features F_{SAM}, Q_{sp}, and an output token to a mask. It uses prompt self-attention and cross-attention in two directions (prompt-to-image embedding and vice-versa) to update all embeddings. After running two blocks, a multilayer perceptron (MLP) maps the output token to a dynamic linear classifier, which then computes the mask foreground probability at each image location.

Modes of SAM:

SAM can operate in two distinct modes: segment everything mode and the promptable segmentation mode as shown in Figure 5.1.

1. **Segment Everything Mode (SAM_{EM}):** can segment everything in the image without using any externally provided prompts. Instead, a grid of key points is generated on the image, and this set of key points is used as prompts to segment everything in the image as shown in Figure 5.1 ("segment everything"). Specifically, SAM_{EM} is prompted by a default 32×32 regular grid of points, predicting a set of masks for each point that may correspond to valid objects [13]. If a point lies on a part or subpart, SAM returns the masks for the subpart, part, and whole object.

2. **Promptable segmentation mode** (SAM_{PSM})**:** segments a specific region of interest based on the prompts given to SAM. The prompts can be bounding boxes, points, or free-form text as shown in Figure 5.1 (box, points).

Our proposed framework utilizes both modes of SAM i.e. SAM_{EM} and SAM_{PSM} with CLIP as a bridge between them as reported in detail in Section 5.3.2.

Contrastive Learning Image Pre-training (CLIP)

Contrastive pre-trained large vision language models like CLIP have revolutionized visual representation learning by providing good performance on downstream datasets. Models such as CLIP are pre-trained on web-scale datasets comprising over 400 million image-text pairs, resulting in a highly generalizable model with effective zero-shot domain adaptation capabilities [15]. Using contrastive pre-training on

large image-text datasets, CLIP performs image classification. While vanilla supervised training is performed on a closed set of concepts or classes, CLIP pre-training uses natural language. This results in a joint text-vision embedding space that is not constrained to a fixed set of classes. CLIP aligns image and text modalities within a shared embedding space as shown in Figure 5.6(1).

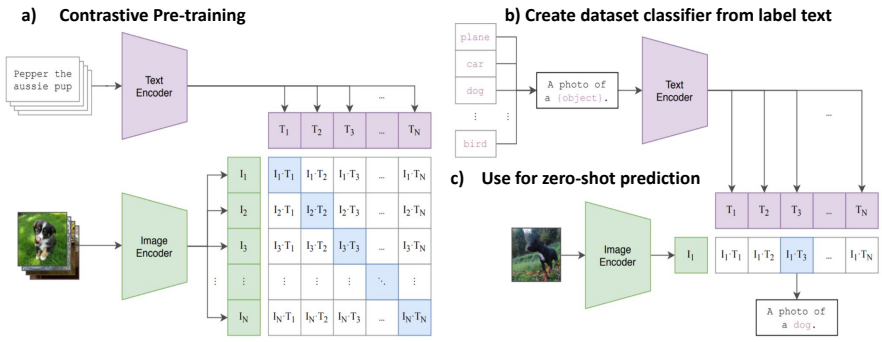


Figure 5.6: Architecture of the CLIP model [15].

After pre-training, CLIP directly performs image classification on the target dataset without any fine-tuning. For an image $I \in R^{H \times W \times C}$, where $H \times W \times C$ denotes spatial dimension, the vision encoder (f) maps I into a joint embedding space to get the image features $E \in D$ with dimension D.

During inference, a prompt template such as *'A photo of classname'* is used to generate sentences for K different classes and passed through the text-encoder to yield classifier weight matrix $W \in R^{D \times K}$ as shown in Figure 5.6(2). Prediction probabilities are then calculated by multiplying image feature f and W and applying a softmax function as shown in Figure 5.6(3).

To construct textual prompts and process these prompts through CLIP's textual encoder, our proposed framework uses ensembles of visually descriptive (VDT) sentences to describe the organ to be segmented following the approach proposed in [120]. These VDTs are generated using GPT-3.5 [52]. A detailed description of this prompt generation process is provided in Appendix A.1.

5.3.2 Proposed Approach: SaLIP

This section presents a comprehensive overview of our proposed framework– SaLIP, which is a test time-adaptation approach for adapting foundation models for medical organ segmentation without any training/fine-tuning and domain expertise. Specifically, SaLIP achieves this without any form of supervision, such as ground truth or domain expertise/knowledge for prompt engineering.

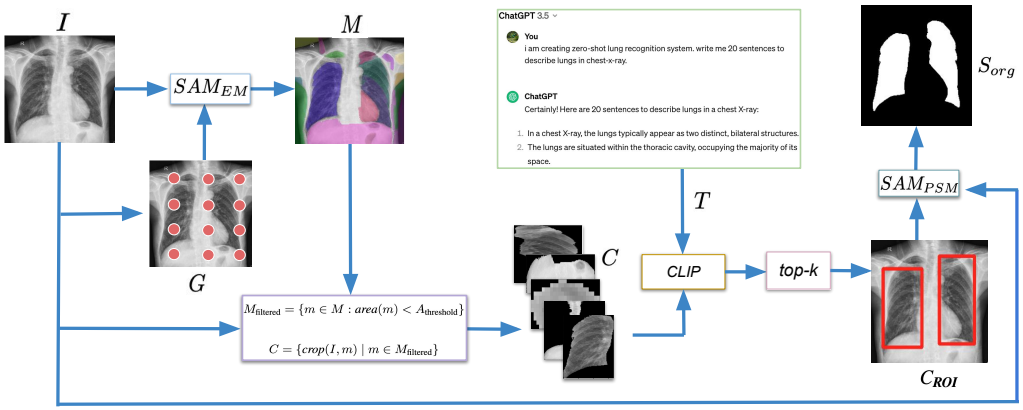


Figure 5.7: Architecture of our proposed SaLIP framework.

As illustrated in Figure 5.7, initially, SaLIP utilizes SAM's *everything mode* (SAM_{EM}) to generate region proposals in the image. These region proposals are part-based segmentation masks for different parts of the input image. SAM_{EM} does not require external prompts; instead, it generates a grid of keypoints $G \in R^{g^2 \times 2}$ on the input image, where g is the point number along one side of the image [13]. These points are used as prompts and if a point lies on a part or subpart, SAM_{EM} will return the subpart, part, and whole object as shown in Figure 5.8. Mask generation by SAM_{EM} using the grid of key points is performed as follows:

$$M = SAM_{EM}(I, G) \tag{5.1}$$

where $I \in R^{3 \times H \times W}$ is the input image, G is the grid of key points, and $M \in R^{N \times H \times W}$ is the set of all the part-based generated masks, each having the same spatial dimension as that of I, N refers to the number of SAM_{EM} generated part based masks and $H \times W$ is the spatial dimension.

Thus, a pool of segmentation masks is generated without any external prompts, supervision, or domain knowledge as shown in Figure 5.8. Instead, the exceptional SAM's sophisticated ambiguity-aware design is used to annotate the input images autonomously.

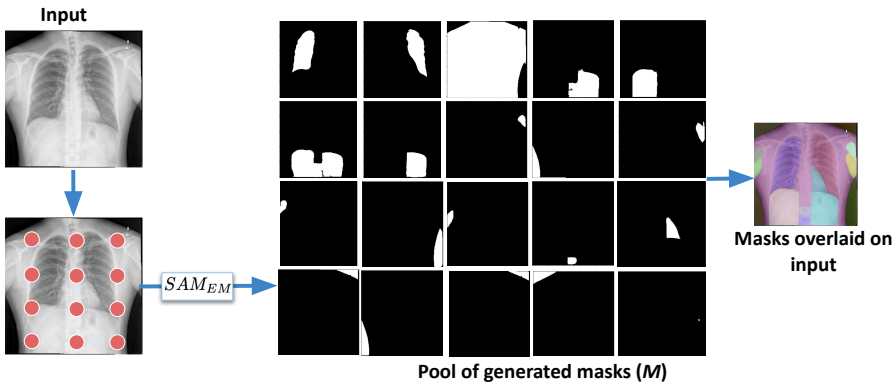


Figure 5.8: Pool of region proposal masks predicted by SAM_{EM} using grid-wise point prompts (red).

The mask generation process by SAM_{EM} is extremely sensitive and is highly influenced by the choice of hyper-parameters used in the SAM's generator module [51]. Our experiments also revealed that this choice significantly impacts the generation of part-based masks, as discussed in Sections 5.5.2 and 5.5.4.

To address this issue, SaLIP implements a random hyperparameter search to select optimal hyperparameters for the SAM_{EM}'s mask generator module. Specifically, SaLIP uses five randomly selected images and conducts a random search to identify the best hyper-parameters for the SAM_{EM} generator module and evaluates the part-based segmentation achieved using each set of hyper-parameters. The combination of hyperparameters which yields the highest average dice score for the randomly chosen five images is eventually used as the final configuration of SAM_{EM}. This optimized configuration is then used to generate part-based masks for the entire dataset. The potential benefits of this hyperparameter optimization are experimentally demonstrated in detail in Section 5.5.2.

The next step in the SaLIP pipeline involves using CLIP to identify the mask corresponding to the region of interest (ROI) from the set of SAM_{EM} generated masks (M) (Figure 5.7). As shown in Figure 5.8, each mask has the same spatial

dimension as that of the input image. To guide CLIP in selecting the correct ROI, the original input image is cropped according to each mask from SAM_{EM} generated pool of masks. This results in several cropped regions of the image as shown in Figure 5.9. CLIP then analyzes these cropped regions to identify which crop corresponds to the desired ROI.

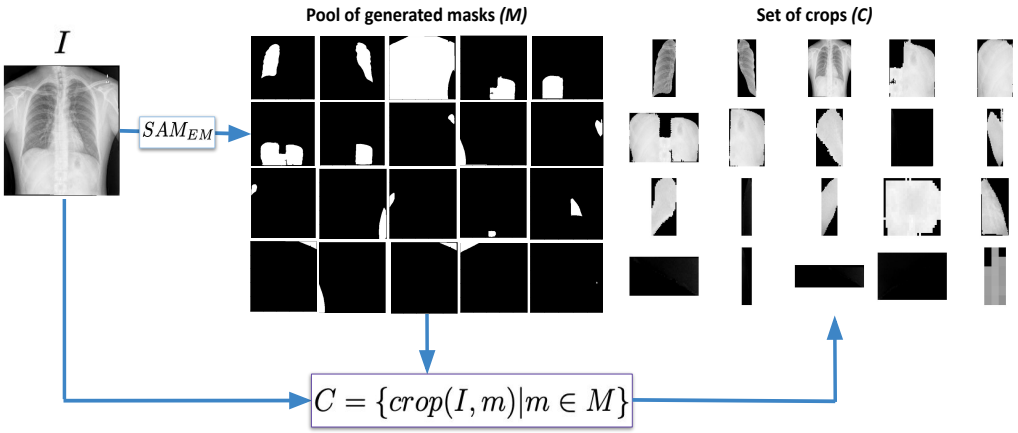


Figure 5.9: Cropped regions of the original input image based on SAM_{EM} masks.

However, as SAM_{EM} employs a grid-wise set of key points to generate masks for different parts of an image, the resulting pool of generated masks may include masks corresponding to the background or larger regions. In such instances, when the input image is cropped, the resulting crop will have the same spatial dimension as the input image. One such case is depicted in Figure 5.9, where the third crop in the first row in the set of crops (C), has the same spatial dimension as that of the original input (I). Such instances can result in miss-classification by CLIP, as it may erroneously recognize the whole I as ROI as illustrated in Figure 5.10. Therefore, it is crucial to manage these masks effectively to prevent miss-classification and ensure that CLIP selects the correct mask corresponding to the ROI.

To address this issue, SaLIP does not directly feed the entire set of SAM_{EM} generated masks (M) to CLIP. Instead, it first filters out the masks $\{m \in M\}$ that likely correspond to the background/larger regions. SaLIP achieves this by applying area-based filtering to each mask in M as illustrated in Figure 5.7.

To determine the optimal threshold for area-based filtering, SaLIP performs a

random hyper-parameter search within the space defined by the areas of SAM_{EM} generated masks. This search is performed concurrently with the hyper-parameter optimization for SAM_{EM}, using the same approach as discussed above. The area filtering on SAM_{EM} generated masks is performed as follows:

$$M_{filtered} = \{m \in M : area(m) < A_{threshold}\} \tag{5.2}$$

where M is the set SAM_{EM} generated masks, $A_{threshold}$ is the area value determined through hyper-parameter search for filtering M, m represents a mask from M, $M_{filtered}$ is the set of masks after removing the m potentially corresponding to background/ larger regions encompassing ROI. The benefits of using area-based filtering on the segmentation results are experimentally demonstrated in Section 5.5.4. After removing the mask potentially corresponding to the background, SaLIP utilizes SAM_{EM} generated masks to crop input image (I), thereby producing a series of crops as :

$$C = \{\text{crop}(I, m) \mid m \in M_{\text{filtered}}\} \tag{5.3}$$

where I is the input image, $crop(I, m)$ represents the function used to crop I according to $m \in M_{filtered}$, and C refers to the set of generated crops.

SaLIP then feeds the set of crops (C) to CLIP to identify the specific crop corresponding to the ROI. As shown in Figure 5.6, CLIP is a vision-language model,

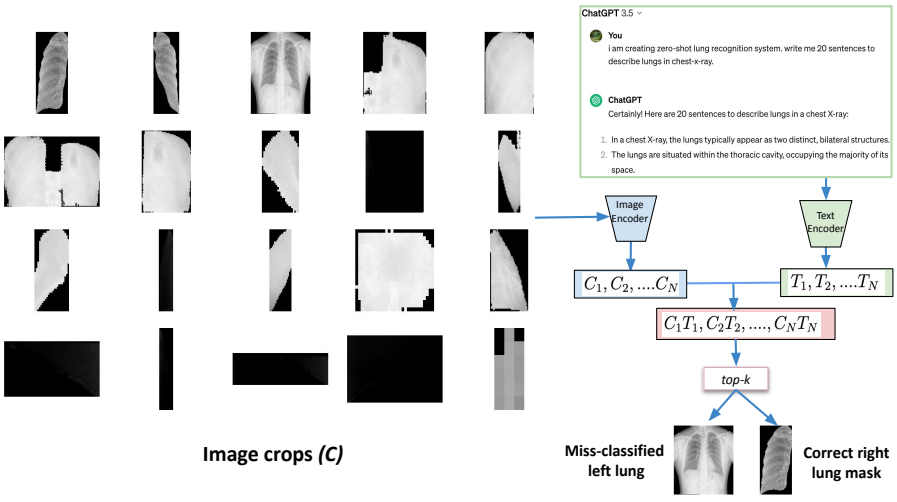


Figure 5.10: Retrieval of the relevant mask using CLIP

to retrieve the relevant image from a set of input images, it needs the textual prompts defining the target class.

To construct textual prompts describing the region of interest (organs in our case), SaLIP uses prompt ensembling. It is a technique that constructs several sentences to define each class and subsequently averages the classification vectors. SaLIP uses prompt ensembles for each class generated following the approach proposed in [120]. Specifically, in our cases, SaLIP utilizes prompt ensembles containing visually descriptive text (VDT) information for the target organ to be segmented (e.g., lungs, brain, fetal head, etc.). These VDT sentences are generated using GPT-3.5 and the process of this prompt engineering is discussed in detail in Appendix A.1.

These VDT prompts are then processed through CLIP's textual encoder to obtain the text embeddings and averaged to obtain a single text prototype W_T for the organ under consideration. Now all the image crops in C are passed through CLIP's vision encoder to obtain vision embeddings (E_c) as shown in Figure 5.7. Subsequently, the mask corresponding to ROI is retrieved as:

$$C_{ROI} = \text{topk}\left(\arg\max_{c \in C} S(\boldsymbol{E}_c, \boldsymbol{W}_T)\right) \tag{5.4}$$

where $S(E_c, W_T)$ represents a similarity function that computes cosine similarity between any embeddings E_c of any crop $\{c \in C\}$ and the text embeddings W_T. k denotes the number of ROIs and it varies depending on the number of ROIs in the image. C_{ROI} is the crop corresponding to ROI.

Finally, SaLIP computes the bounding box prompts using the minimum and maximum X, Y co-ordinates of the retrieved C_{ROI} and uses it to prompt SAM_{PSM} as:

$$S_{org} = SAM_{PSM}(I, P) \tag{5.5}$$

where $P \in R^{N \times 4}$ is the bounding box prompt computed from co-ordinates of C_{ROI}, N is the number of box prompts which varies according to ROI and S_{org} is the final segmentation map of ROI generated by SAM_{PSM} as shown in Figure 5.7.

5.4 Experimental Framework

5.4.1 Datasets and Metrics

The proposed SaLIP framework is evaluated across three diverse medical imaging modalities, including two datasets focused on single-organ segmentation and one more challenging dataset involving the segmentation of two distinct regions of interest. Calgary-Campinas (CC359) [11] is a multi-vendor (GE, Philips, Siemens), multi-field strength (1.5, 3) magnetic resonance (MR) T1-weighted volumetric brain imaging dataset. It has six different domains and contains 359 3D brain MR image volumes. The CC359 dataset is primarily used for the task of brain segmentation in head MRI scans. The HC18 [17] consists of 2D fetal head ultrasound images obtained throughout all trimesters of pregnancy. The region of interest in this dataset is the fetal head. X-ray Masks and Labels dataset [16] consists of 2D chest X-ray images. The target organs to be segmented are the left and right lungs. The evaluation metrics used are the dice similarity coefficient (DSC) and the mean intersection over union (mIoU).

5.4.2 Implementation Details

There are three different variants of the Segment Anything Model (SAM), primarily differentiated by encoder size [13]. The proposed SaLIP pipeline uses the huge variant of SAM (ViT-H). Similarly, CLIP has several variants, and SaLIP employs its large variant (ViT-L/14) from OpenAI's CLIP framework. The choice of specific model variants is based on an ablation conducted to assess the impact of different variants on performance (discussed in Section 5.5.6).

For CLIP, visually descriptive text prompts describing the organ to be segmented are generated using GPT-3.5 [52]. The detailed process for generating these prompts with GPT-3.5 is outlined in the appendix of this thesis, in Section A.1.

The U-Net architecture, used for comparison with our proposed test-time adaptation framework, is trained for 100 epochs using cross-entropy loss, with a learning

rate of 1×10^{-5} and a batch size of 32.

The SaLIP framework is implemented in PyTorch [272] using the SAM codebase [1]. All experiments were conducted on a desktop running Ubuntu 20.04.6 LTS with CUDA 11.6 and an NVIDIA GeForce RTX 3090 GPU. To ensure reproducibility, a random seed of 1234 was used.

5.5 Results and Analysis

5.5.1 Comparative Analysis of SaLIP and Other Methods

Our proposed SaLIP framework is compared against the following methods:

- **U-Net:** is a prominent architecture widely adopted for medical image segmentation [273]. To compare it with our proposed framework, U-Net undergoes training separately on each of the three datasets (details are reported in Section 5.4.1). Task-specific fine-tuning is required to optimize the performance of U-Net for the specific segmentation task at hand.

- **GT-SAM:** Prompts relevant to the target organ are derived directly from ground truth annotations and provided to the Segment Anything Model. It represents an ideal scenario where annotated data is available and can be used for prompt engineering. This configuration setup serves as a performance upper bound.

- **Un-prompted SAM:** As our proposed SaLIP framework is adapted at test time without using any labeled data. Thus, to ensure a fair evaluation of the SaLIP framework and accurately reflect real-world medical imaging scenarios where annotated data and domain expertise are often unavailable, an un-prompted version of SAM is used. Unlike GT-SAM, this un-prompted version does not rely on prompts derived from ground truth data. Instead, it uses SAM's default pre-trained prompt embeddings. This setup enables evaluation

[1] https://github.com/facebookresearch/segment-anything, Accessed: [15.06.2024]

of SAM's performance in challenging applications, like medical imaging, where obtaining precise prompts is difficult or impractical.

- **Supervised Adaptation:** The entire SAM model is fine-tuned using annotated data and ideal prompts derived from ground truth.

- **Parameter-Efficient Adaptation:** Specific modules of the SAM model are adapted using low-rank adaptation [10], with prompts based on ground truth.

The primary difference between prompted SAM, unprompted SAM, and our proposed SaLIP lies in how the prompts for each of them are generated, as mentioned above.

The quantitative comparative analysis of our proposed approach with the above-mentioned methods is reported in Table 5.1. Although the U-Net model outperformed other methods, it is important to highlight that it followed a standard training procedure using annotated data and underwent task-specific fine-tuning for each of the three datasets (Section 5.4.1). In contrast, our proposed SaLIP framework facilitates unsupervised test-time adaptation. It is adapted to diverse medical imaging segmentation tasks without additional training or task-specific fine-tuning. Instead, it is adapted entirely at test time without any supervision and domain knowledge, as detailed in Section 5.3.2. Consequently, SaLIP effectively circumvents the need for extensive annotated data. Although the SaLIP framework leverages foundation models, it avoids the computational overhead of training or fine-tuning. As a result, SaLIP significantly streamlines the segmentation workflow, enhancing robustness and efficiency, particularly in medical imaging applications where annotated datasets are scarce and costly to obtain. Its adaptability at test time across diverse medical domains eliminates the need for task-specific training/fine-tuning, making it a more practical solution.

The un-prompted SAM notably has a poor generalization in contrast to the other evaluated methods as reported in Table 5.1. These results provided valuable insights into SAM's strong reliance on prompts for optimal segmentation and highlighted its

Table 5.1: Comparison of SaLIP with other methods.

ROI	Dataset	U-Net		GT-SAM ∗		Un-prompted SAM		SaLIP (Ours)	
		DSC	mIoU	DSC	mIoU	DSC	mIoU	DSC	mIoU
Brain	GE 1.5	0.98	0.93	0.95	0.91	0.33	0.29	**0.92**	**0.87**
	Philips 1.5	0.97	0.95	0.96	0.93	0.41	0.31	**0.94**	**0.85**
	Philips 3	0.95	0.92	0.93	0.89	0.40	0.39	**0.89**	0.80
	Siemens 1.5	0.97	0.95	0.95	0.91	0.39	0.26	**0.90**	**0.81**
	Siemens 3	0.98	0.92	0.96	0.90	0.41	0.32	**0.93**	**0.85**
Lungs	X-ray	0.98	0.95	0.94	0.90	0.47	0.31	**0.83**	**0.76**
Fetal head	Ultrasound	0.95	0.91	0.95	0.91	0.55	0.40	**0.81**	**0.72**

∗ **Note:** GT-SAM uses the perfect prompts extracted from ground truth.

poor generalization to downstream medical tasks, particularly in scenarios where prompts are unavailable. In such cases, when SAM's default prompt embeddings are used, its segmentation performance is poor due to the domain shift between the features learned during pre-training and those needed for medical imaging tasks.

These limitations pose challenges in the application of the Segment Anything Model in real-world scenarios, particularly in the medical domain, where annotated data is often unavailable and there is a lack of domain expertise for prompt engineering. In contrast, our SaLIP framework consistently outperformed the un-prompted SAM for all the evaluated segmentation tasks as reported in Table 5.1. Notably, SaLIP facilitates robust adaptation across diverse medical imaging tasks without specialized domain expertise for prompt engineering and annotated ground truth for crafting perfect prompts. Instead, our proposed method autonomously generates prompts as discussed in detail in Section 5.3.2.

As reported in Table 5.1, for brain segmentation in the CC359 dataset [11], our proposed approach achieved an average of 0.94 dice similarity coefficient(DSC), significantly outperforming the un-prompted SAM's average DSC of 0.31. When evaluated for lung segmentation, SaLIP significantly enhances the generalization, increasing the initial DSC from 0.31 achieved by unprompted SAM to 0.83. SaLIP achieves an average DSC of 0.81 for segmenting the fetal head on HC18 [17], compared to the unprompted SAM's average DSC of 0.55. Thus, SaLIP consistently generalizes well across diverse medical imaging segmentation tasks, achieving ex-

ceptional zero-shot performance. It demonstrates significant improvements over un-prompted SAM, with improvements of 63.46% for the brain, 50.11% for the lungs, and 30.82% for fetal head segmentation compared to unprompted SAM as reported in Table 5.1.

To gain deeper insights into the significantly lower performance of un-prompted SAM, a thorough qualitative analysis is conducted. This analysis revealed that un-prompted SAM tends to segment larger regions in the image, which often does not correspond to the regions of interest. This behavior was consistently observed across all three datasets examined. As illustrated in Figure 5.11, un-prompted SAM segments the entire upper body region in the X-ray image, while the primary region of interest (ROI) in this case is the lungs. Similarly, for the HC18 dataset [17], the ROI is the fetal head in ultrasound images; however, un-prompted SAM incorrectly segments the entire ultrasound image instead. In the case of the CC359 dataset [11], which includes axial MRI scans, un-prompted SAM segments the entire head region rather than ROI i.e. brain in this case.

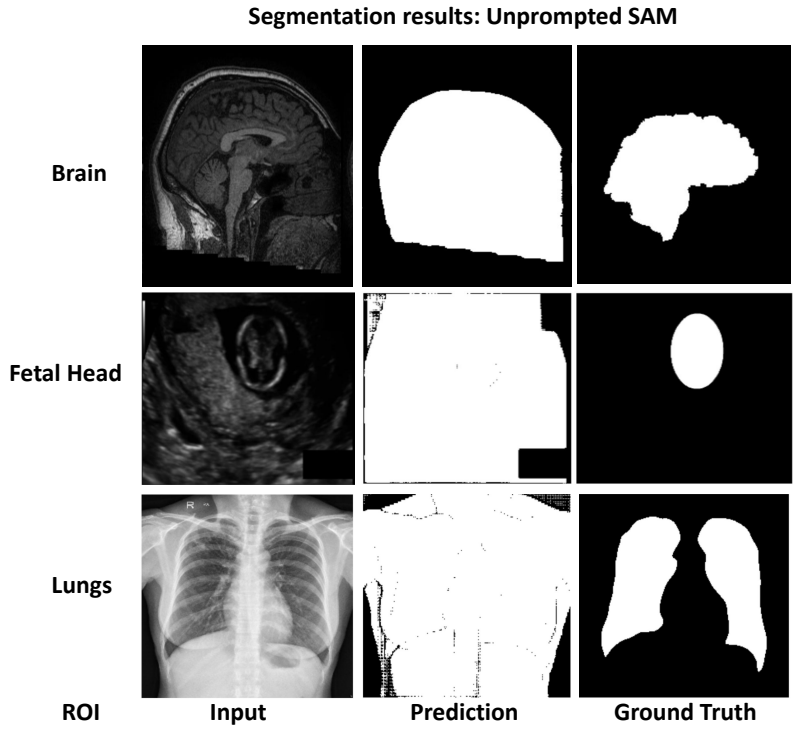


Figure 5.11: Qualitative Analysis of Un-prompted SAM's Performance.

This qualitative analysis reveals that the consistent failure of un-prompted SAM

across various medical imaging segmentation tasks is due to SAM's strong reliance on the prompts provided. Thus, the lack of annotated data or domain expertise for prompt engineering hampers SAM's ability to accurately identify specific anatomical structures and effectively segment ROIs. Hence, SAM's applicability to medical imaging scenarios is limited, where obtaining domain expertise and annotated data for prompt engineering is challenging. In contrast, our proposed SaLIP approach achieves accurate segmentation while alleviating these challenges, as it does not require annotated data or domain expertise to segment the specific anatomical organs as demonstrated in Figure 5.12 and discussed in detail in Section 5.3.2.

GT-SAM achieves high dice scores of 0.95, 0.94, and 0.91 for brain, lung, and fetal head segmentation respectively, as reported in Table 5.1. This strong performance is due to its use of "perfect prompts" which are directly derived from ground truth data, giving it a significant advantage.

While GT-SAM leverages prompts derived from annotated data, our proposed approach does not require annotated truth for prompt engineering and is independent of domain expertise for prompt engineering and annotated data. Our proposed method achieves results comparable to GT-SAM and operates independently of external prompts, as shown in Figure 5.12 and Table 5.1. The effectiveness of SaLIP compared to other methods is demonstrated qualitatively in Figure 5.12.

Thus both qualitative and quantitative results highlight the generalization of our proposed SaLIP framework to real-world scenarios where annotated data or expert knowledge is limited or unavailable. Thus our proposed SaLIP approach is a general segmentation framework, which enables zero-shot segmentation of the target ROIs without traditional supervised training, annotated data, task-specific fine-tuning, or specialized domain expertise for prompt engineering.

In addition, the proposed test-time adaptation framework is compared with standard adaptation approaches. Supervised adaptation refers to the scenario where the entire SAM model is adapted. However, when attempting to adapt the entire model, our GPU ran out of memory. As a result, in this supervised adaptation

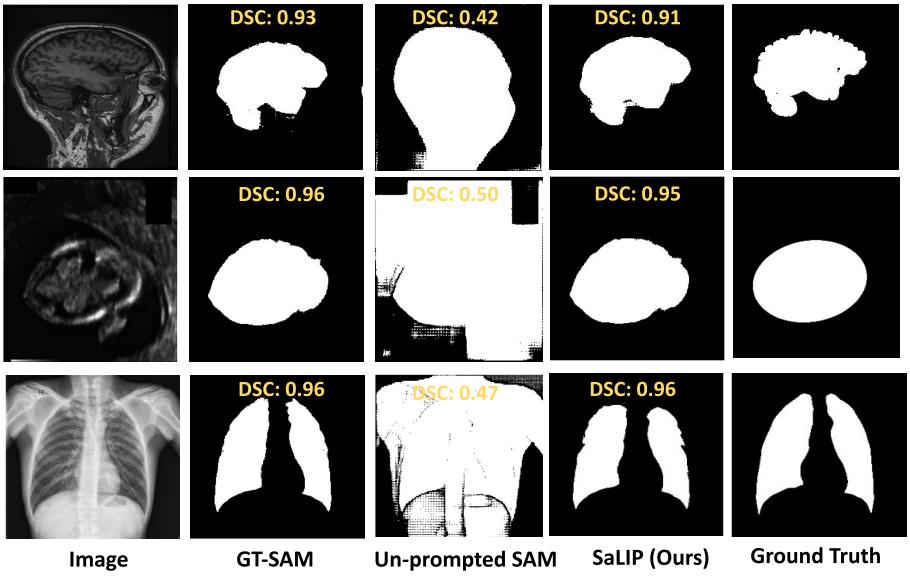


Figure 5.12: Qualitative Analysis for Performance Comparison: GT-SAM (Upper Bound), Un-prompted SAM, and SaLIP (Ours).

mode, we were only able to adapt the SAM decoder. We also evaluated parameter-efficient adaptation methods, where we used LoRA to adapt both the encoder and the decoder. For both of these adaptation settings, the bounding box prompts are extracted using the ground truth. The results of these supervised adaptation approaches with our SaLIP framework are presented in Table 5.2. This evaluation is carried out on CC359 dateset 5.4.1.

Our method is computationally efficient, taking only 30 minutes for inference on the CC359 dataset (Section 5.4.1), compared to the 24, 15, and 8 hours required by other approaches. Additionally, the slightly better performance of these methods can be attributed to their use of perfect prompts extracted from ground truth. In contrast, our approach is entirely independent of additional training, labeled data, or manually created prompts. Instead, prompts are automatically generated, eliminating the need for domain-specific knowledge. Furthermore, the performance of SaLIP is comparable to that of these approaches and can potentially be enhanced by implementing the alternatives suggested in Section 7.3.

Additionally, to provide clearer insights into the outcomes of the different stages of the proposed framework, step-wise qualitative results of our SaLIP framework are presented in Figure 5.13 and 5.14. The first column displays the input image,

Table 5.2: Comparison of SaLIP with supervised adaptation approaches.

Dataset	Supervised Adaptation	PEFT- LoRA		SaLIP (Ours)
	Decoder	Encoder	Decoder	Dice score
GE 1.5	0.93	0.94	0.91	0.92
Philips 1.5	0.95	0.96	0.93	0.94
Philips 3	0.92	0.93	0.89	0.89
Siemens 1.5	0.95	0.93	0.91	0.90
Siemens 3	0.92	0.94	0.90	0.93
Compute. time	24h	15h	8h	**30 (mins)**

while the second column presents the set of region proposals generated using the Segment Anything Model's "everything mode" (SAM_{EM}) (Section 5.3.1– Modes of SAM). The third column (also fourth in Figure 5.13) shows the retrieved crop corresponding to a region of interest (ROI) that CLIP identified from the generated masks. The next column shows the bounding box (BBOX) coordinates on the input image calculated using the retrieved ROI crop (Section 5.3.2). Following that, the next column shows the organ segmented by SAM's promptable segmentation mode (SAM_{PSM}) by promoting it with the BBOX of the retrieved crop. Finally, the last column presents the ground truth segmentation. It is evident that for both datasets, SaLIP achieves segmentation results comparable to the ground truth.

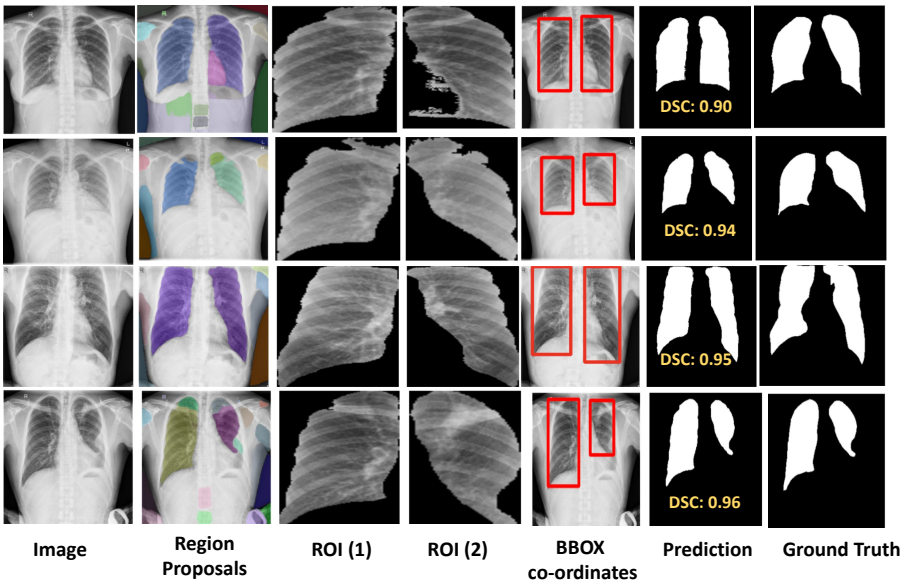


Figure 5.13: SaLIP Qualitative Results: X-ray labels and masks dataset [16].

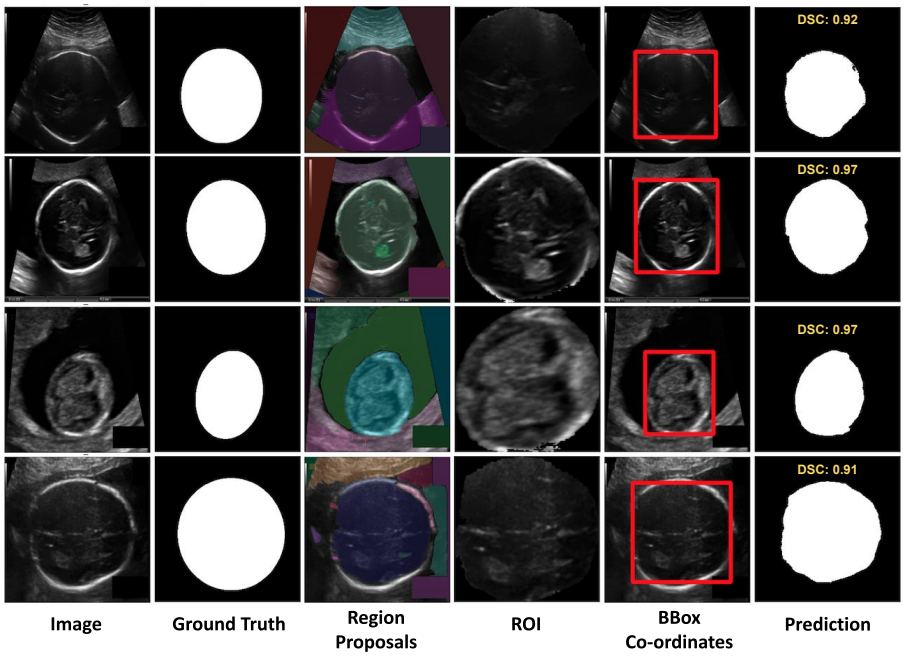


Figure 5.14: SaLIP Qualitative Results: HC18 dataset [17].

5.5.2 Hyper-parameter Optimization

The process of mask generation for potential regions in an image using Segment Anything Model's "everything mode" (SAM_{EM}) is highly sensitive. SAM_{EM}'s hyperparameters are vital in determining how effectively it can segment various regions and features within an image.

For example, one of the hyperparameters in the SAM_{EM} is the number of crop layers ($int : crop\_n\_layers$) [13]. It sets the number of layers on the image to run, where each layer has $2 * *ilayer$ number of image crops. If $crop\_n\_layers > 0$, mask prediction will be run again on crops of the image. The impact of changing values for $crop\_n\_layers$ on the respective mask predictions is illustrated in Figure 5.15. The last column in Figure 5.15 illustrates how different sub-crops are generated for an image. The various colored lines represent the different crop regions created by SAM, with a predicted region shown within each crop.

These results demonstrate that even a small change to a hyperparameter "crop layer" from 0 to 1 can significantly impact the SAM_{EM}'s region proposals, leading to substantial changes in the pool of masks predicted by SAM_{EM} for the same image as evident in Figure 5.15.

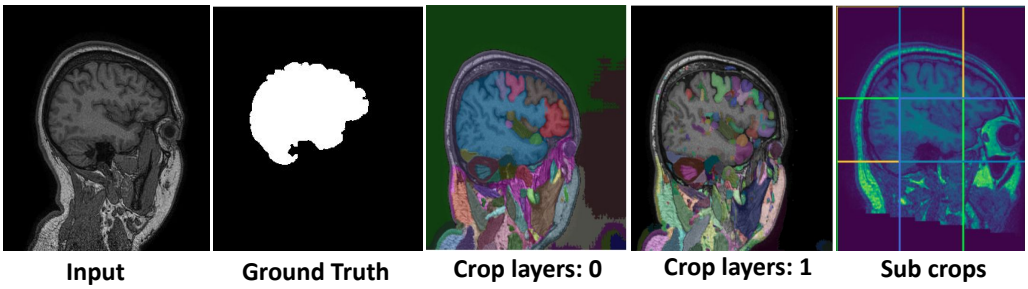


Figure 5.15: Effect of hyperparameters on region proposals generated by SAM_{EM} in an image.

Additionally, in the majority of real-world cases, using SAM's default hyperparameters configuration fails to segment the region of interest correctly or segments irrelevant areas, which complicates the overall segmentation and stability of the results. Figure 5.16 demonstrates that the choice of hyperparameters for SAM_{EM} can lead to failures in generating masks for the region of interest. The first column displays the input image, while the second column shows the ground truth. The third column features results from SAM's online demo; however, the specific hyperparameters used in this version are not publicly available [274]. The fourth column demonstrates the default hyperparameters from SAM's official repository [275].

It is evident in Figure 5.16 (fourth column), with SAM's default settings, that the region of interest (e.g., the brain in this example) is not predicted. Instead, the entire head region from the axial scan is segmented (shown in green). Thus, proper optimization of these hyperparameters is crucial; without careful adjustment, it can lead to cases where no masks correspond to the regions of interest, as seen in the fourth column. The last column demonstrates the effectiveness of our hyperparameter optimization, where the region of interest is correctly segmented and highlighted in gray (discussed in Section 5.3.2).

5.5.3 ROI Mask Retrieval

The proposed SaLIP pipeline leverages CLIP to identify the mask corresponding to the target organ from the pool of masks generated by SAM's everything mode (Section 5.3.2). The decision to use CLIP was informed by our initial experiments,

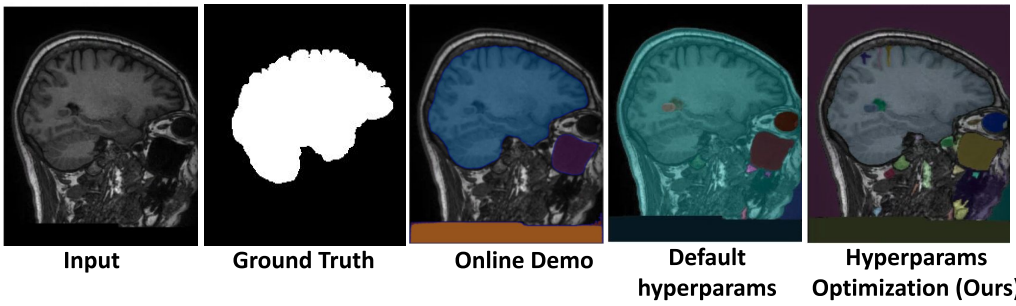


Figure 5.16: Effect of hyperparameter optimization on mask generation by SAM_{EM}.

during which alternative approaches were evaluated (as discussed in the following section) but yielded unsatisfactory results.

Before leveraging CLIP in the proposed method, we explored manual area-based filtering to retrieve the ROI from the SAM-generated region proposals. However, applying a specific threshold, in this case, retrieved multiple masks from the pool of SAM-generated masks, as many predicted masks lie within a similar area range, as shown in Figure 5.17. The first image is the input image, while the second shows the ground truth. The third image illustrates the masks generated by the Segment Anything Model's 'everything mode' (SAM_{EM}). In the fourth column, the set of masks retrieved by manual area-based filtering as ROI from the pool of SAM_{EM}'s generated masks are depicted. The three predicted masks for the particular example shown in Figure 5.17 ("area based filtering") are: a brain mask in purple and a background mask in brown, which eventually decreases the segmentation accuracy as in this case only the brain mask corresponds to the region of interest while threshold based method retrieved two irrelevant regions as well.

These results highlight the inherent variability in organ morphology, which lacks fixed dimensions and can further fluctuate, especially under pathological conditions. Consequently, selecting a single area-based threshold to reliably isolate the region of interest becomes challenging. While one threshold may work effectively for larger anatomical structures (e.g., lungs), it may fail to capture smaller, more subtle pathological features (e.g., tumors).

In contrast, our proposed framework employs CLIP for retrieving ROI masks

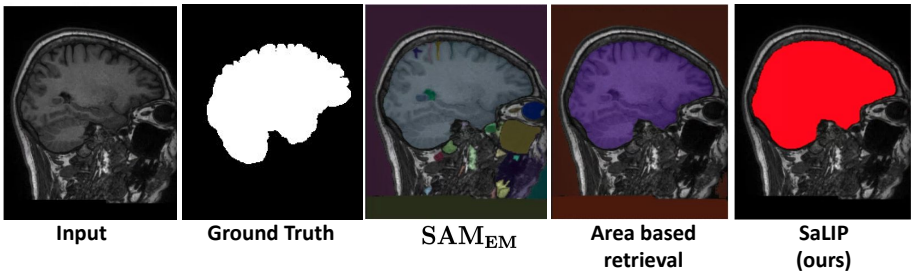


Figure 5.17: Comparison of various techniques for ROI mask retrieval.

from the pool SAM_{EM} predicted masks. The last column in Figure 5.17, illustrates that our proposed approach effectively leverages CLIP to accurately retrieve the region of interest, which is the brain, highlighted in red.

In addition to the traditional area-based filtering approach, we also evaluated large language models (LLMs) for visual inference in medical imaging. For this purpose, we used ViperGPT– a framework that leverages code generation models to compose vision and language models into subroutines to produce a result for any query [276]. ViperGPT achieves this by using large language models to generate modular programs to perform a specific task. This approach has proven highly effective in natural imaging, for visual question-answering tasks, such as: 1) find the children, and the muffins in the image, 2) count how many there are of each, and 3) reason that 'fair' implies an even split, hence divide.

Motivated by its benefits in above mentioned scenarios, we employed ViperGPT to create a sub-routine to automatically generate bounding box prompts for the brain in MRI scans. As illustrated in Figure 5.18, this process begins with a query that is given to ViperGPT i.e. "generate a bounding box prompt for identifying the brain in a head MRI scan". ViperGPT then outputs the modular code for this task, referred to as "generated code" in Figure 5.18. This sub-routine is subsequently applied to the input image in a step labeled as "execution". Through this process, ViperGPT facilitated autonomous prompt generation for our specific use case i.e., bounding box prompt for brain region without requiring annotated data or specialized domain knowledge.

However, ViperGPT has a significant limitation when applied to medical imag-

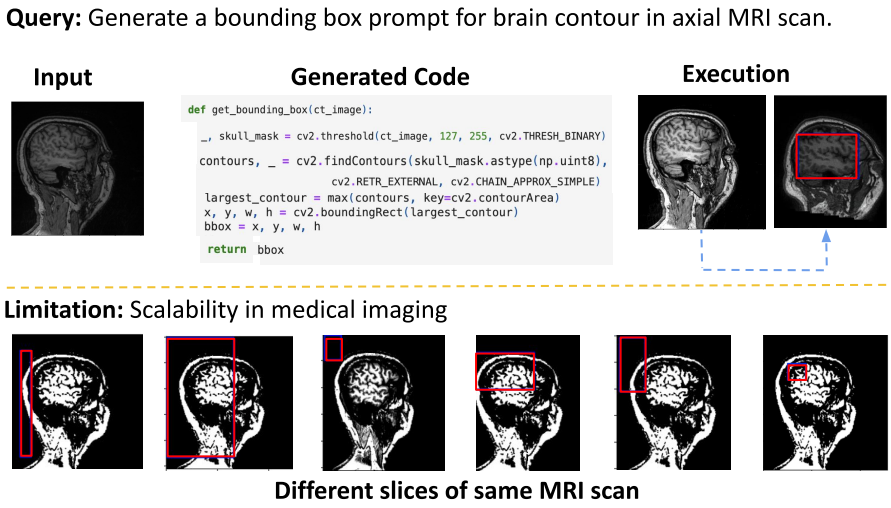


Figure 5.18: Limitation of foundation models to perform domain-specific tasks in medical domain [16].

ing tasks due to the complex nature of medical data. For example, as shown in Figure 5.18, the generated sub-routine created the correct box prompt for the brain region (highlighted in red) for the slice of an MRI scan depicted under the label of "execution". However, when applied to other slices of the same MRI scan, the generated subroutine often failed to accurately detect the brain contour, resulting in incorrect bounding box prompts (shown in red) for these cases, as illustrated in Figure 5.18 (Limitation).

Due to these inconsistencies, ViperGPT lacks scalability for medical imaging tasks. Given that the brain is relatively large and structurally distinct compared to other anatomical regions, we initially expected ViperGPT to perform well. However, its poor generalization, even in these seemingly straightforward cases (as shown in Figure 5.18) indicates that LLMs can encounter even greater challenges when applied to more complex anatomical structures with variable morphology.

5.5.4 Area-based Mask Filtering

CLIP may erroneously retrieve the crop corresponding to the background or an extended area that encompasses the region of interest (ROI), rather than precisely identifying the ROI itself, as illustrated in Figure 5.19. As discussed in Section 5.3.2, our pipeline- SaLIP, reduces the likelihood of this miss-classification in such scenarios

by applying area-based filtering before passing them to CLIP (Figure 5.7).

To address this issue, area-based filtering is applied to exclude masks that may be erroneously classified as regions of interest (ROI). The hyperparameter optimization process for our filtering approach is detailed in Section 5.3.2. A quantitative comparison of results between our proposed area-based filtering and using all the SAM_{EM}'s generated region masks directly is presented in Table 5.3. Our approach shows a 3% improvement in brain segmentation accuracy [11] and approximately a 10% improvement in lung and fetal head segmentation [16].

Table 5.3: Comparative Analysis: impact of area-filtering on ROI Mask Retrieval.

Dataset	No Filtering	Filtering (Ours)
CC359 [11]	0.91	**0.94**
X-ray [16]	0.75	**0.83**
HC18 [17]	0.71	**0.81**

Furthermore, the strength of our area-based filtering is demonstrated qualitatively in Figure 5.19. The Figure 5.19(a) shows the results without area filtering, while Figure 5.19(b) illustrates that our proposed approach removes the masks encompassing ROI, thereby reducing the likelihood of miss-classification by CLIP in the proposed SaLIP pipeline.

5.5.5 Limitations and Potential Solutions

Although our proposed SaLIP framework demonstrates effective performance, an in-depth analysis highlighted two key limitations: one at the SAM level and the other at the CLIP level.

Region Proposals Generated by SAM

It refers to the instances where SAM's everything mode (SAM_{EM}) (Section 5.3), fails to generate a mask for the ROI. In both cases shown in Figure 5.20, it is evident that SAM_{EM} did not generate masks for ROI. The first row shows the instance where no mask was generated for the fetal head, while the second row shows that

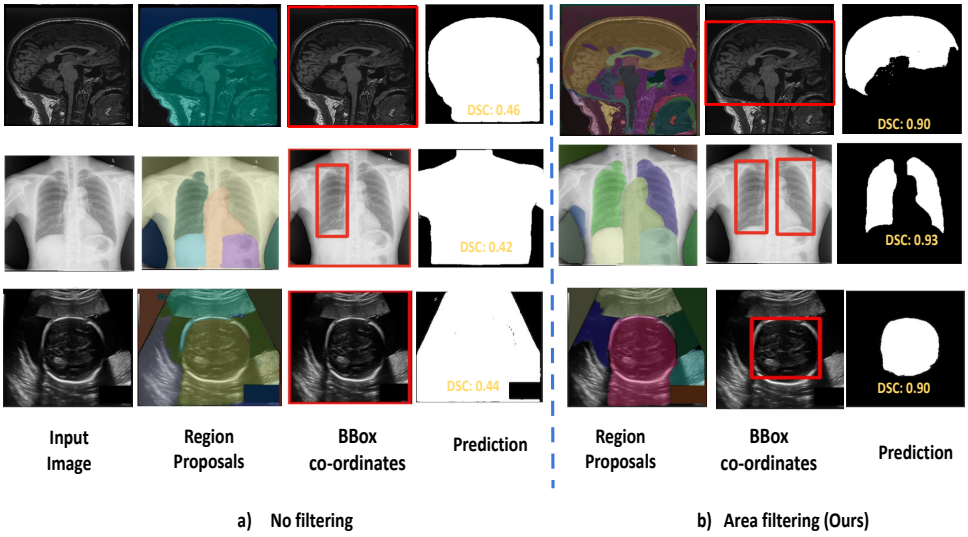


Figure 5.19: Qualitative Results: a) No Filtering : All SAM-generated region proposals are fed to CLIP, leading to miss-classification of the ROI. b) Area filtering (ours): applies area filtering to SAM-generated region proposals to remove the masks encompassing ROI, thereby reducing the likelihood of miss-classification by CLIP.

although a mask was produced for one of the ROI (the right lung), SAM_{EM} failed to predict the mask for the other ROI i.e. the left lung. In both cases, when the original image crops generated using these SAM_{EM} generated masks are passed to CLIP (discussed in Section 5.3.2), it incorrectly classifies the wrong mask as the ROI, as shown in the segmentation results in the fourth column of Figure 5.20.

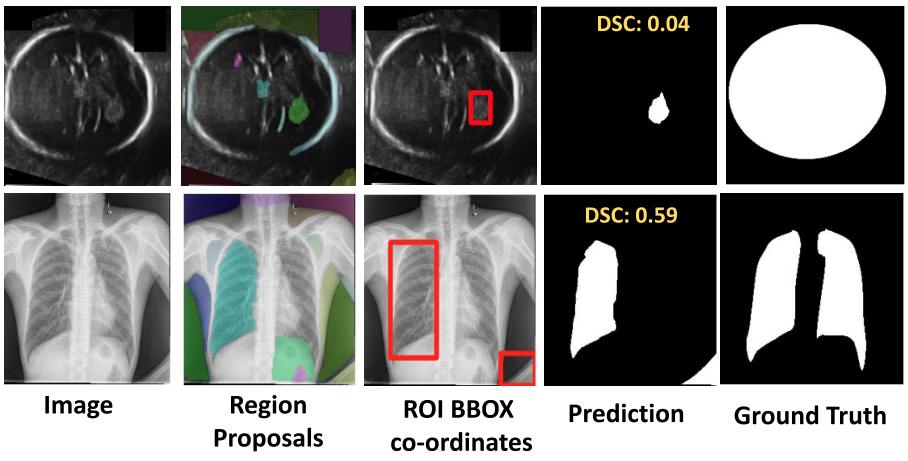


Figure 5.20: SAM failure cases: First row: SAM_{EM} fails to generate a mask for the fetal head, resulting in miss-classification by CLIP. Second row: SAM_{EM} generates a mask for the right lung but fails to generate a mask for the left lung, eventually CLIP retrieves the wrong crop as ROI.

These issues often arise because the SAM_{EM} mask generator module is highly

sensitive to hyperparameters, which can lead to poor performance. To address this, our proposed approach incorporates hyperparameter optimization (Section 5.5.2). This optimization has led to improved performance and the potential benefits are discussed in Section 5.5.2. However, there are still instances where SAM_{EM} fails to generate masks for relevant regions of interest (ROIs) as illustrated in Figure 5.20.

Mask Retrieval by CLIP

These instances refer to the situations where SAM_{EM} generates masks corresponding to ROIs, but CLIP fails to retrieve the correct ROI. As shown in Figure 5.21, the third column displays two cases where SAM_{EM} accurately generated the masks for the ROIs i.e. fetal head (first image: ROI is highlighted in purple, second image: ROI is highlighted in green). However, as shown in the fourth column, CLIP erroneously retrieves an incorrect image crop from the pool of crops generated using SAM_{EM}'s masks (Section 5.3.2) as the ROI despite the SAM_{EM} had generated masks for both cases.

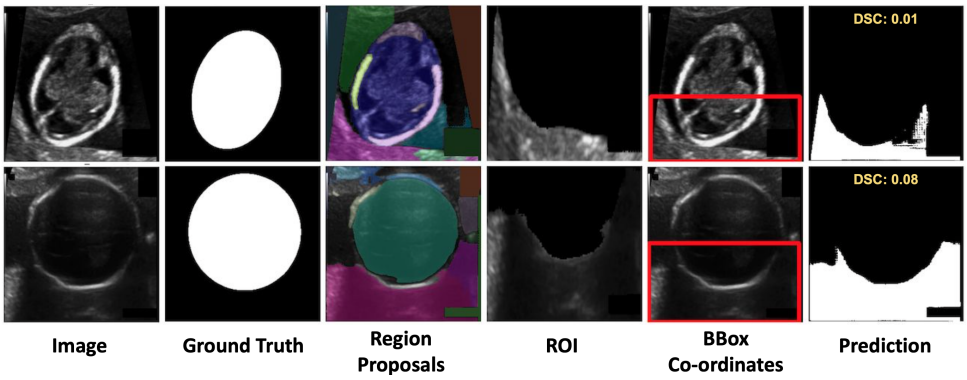


Figure 5.21: Qualitative Results: CLIP Failure cases for HC18 [17]: SAM_{EM} predicts a mask for the fetal head ("region proposal column"). However, CLIP does not retrieve the correct mask.

For lung segmentation analysis from the X-ray dataset [16], two primary limitations were identified. The first issue arises due to SAM_{EM} generates multiple masks for a single image region. In such cases, CLIP sometimes retrieves the masks corresponding to the same lung region for both the left and right lung, as illustrated in Figure 5.22 (first row). The second issue arises when CLIP correctly identifies the

crop corresponding to one of the lungs from the pool of image crops, but erroneously classifies a non-lung region as the ROI for the other lung, as shown in Figure 5.22 (second row).

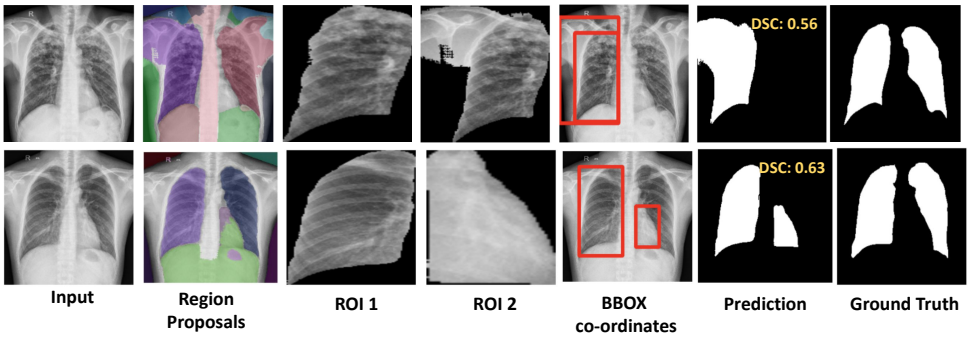


Figure 5.22: CLIP failure cases: First row: SAM_{EM} generates multiple masks for both ROIs (left and right lung). CLIP while correctly recognizing the right lung, identifies a second mask for the same lung region and fails to retrieve the crop corresponding to the left lung. Second row: CLIP did not retrieve the left lung crop.

We explored potential solutions based on the performance benefits reported in the literature. These solutions are discussed in the following sections.

Potential Solution 1: Separate Prompts for Different ROIs

To retrieve the left and right lungs correctly from the pool of various image region crops using CLIP, our proposed method uses a single set of visually descriptive sentences that describe both lungs in a chest X-ray and passes these prompts to CLIP's text encoder (Appendix A.1.1). These prompts consist of general, visually descriptive sentences about different attributes of the lungs. These sentences do not include information on spatial alignment or the distinguishing features of the left and right lungs.

To address the limitations mentioned above, we evaluated using a different set of textual prompts, each specifically describing the left and right lung. By providing detailed descriptions that highlight the spatial locations and distinct attributes of each lung, we aimed to improve the differentiation between the two lungs, thereby reducing the risk of misclassification by CLIP. This misclassification can occur due to multiple mask generations for a single image, as shown in Figure 5.22 (first

row). The process of generating the separate prompts for the left and right lung is described in Section A.1.2.

Table 5.4 provides a quantitative comparison between the results obtained using a single set of prompts with general visual descriptions of lung attributes (Section A.1.1) and those using separate sets of prompts, each describing the distinct features of the two lungs (Section A.1.2). SaLIP achieves 0.83 DSC for both lung segmentation using the single set (labeled as "Combined (Ours)", thereby outperforming the separate prompts for both lungs, which achieved 0.67 and 0.28 DSC for the left and right lung, respectively.

Table 5.4: Quantitative Analysis: CLIP retrieval performance with Separate Prompts vs Combined Prompts

	Right Lung	Left Lung	Combined (Ours)
DSC	0.67	0.28	**0.83**

Contrary to expectations, using separate prompts for each ROI to improve CLIP's classification performance and aid in retrieving the correct regions did not yield the anticipated benefits. However, the results reported in Table 5.4 demonstrated that CLIP has limited performance in precise localization and fine-grained recognition tasks. It lacks semantic knowledge in distinguishing regions based on their spatial alignment i.e. "left" and "right" for lungs in this context. Consequently, employing separate sets of prompts to describe lungs based on their spatial alignment and distinct features did not improve CLIP's ability to distinguish between them.

One of the future works is the integration of inference mechanisms to detect such failures and prevent their propagation to the subsequent steps in the pipeline. It will help mitigate the occurrence of such failures and improve performance further.

Potential Solution 2: Visual Prompting

Recent research indicates that for precise location and recognition tasks, CLIP's performance can be enhanced through the utilization of visual prompting [51, 18,

253, 277]. Visual prompting (VPT) involves the addition of visual markers like colorful boxes or circles directly onto an image, aiding in highlighting specific targets in image-language tasks [51]. VPT directs the attention of visual language models toward desired targets while maintaining the global context. A few of the examples of VPT are shown in Figure 5.23. The visual prompt in these examples is a red circle. Multiple visual prompts are drawn on the same image and CLIP is tasked to choose the correct one relevant to the given caption [18].

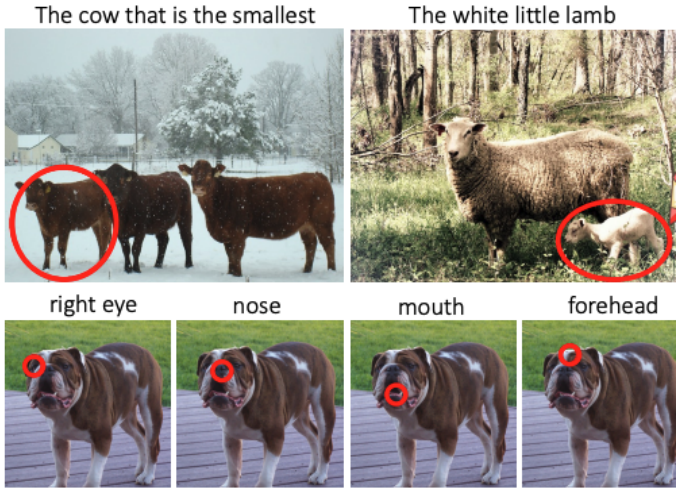


Figure 5.23: Visual Prompt Engineering: Multiple visual prompts i.e., red circle (in this case) are drawn over an image and CLIP is tasked to choose the correct one given a caption. The image is taken from [18].

Based on the potential benefits of VPT on CLIP's ability for precise localization and recognition tasks in natural imaging, we investigated the potential of visual prompts to enhance CLIP's ability to distinguish between the left and right lungs. Specifically, we incorporated visual markers onto the original image at spatial locations corresponding to the coordinates of the masks generated by SAM_{EM} (Section 5.3.2). This set of images is then fed to CLIP's image encoder. Specifically, three different visual prompts are evaluated: a red bounding box marker, contour delineation, and a reverse gray box highlighting the masked area while blurring the rest of the image [51] as shown in Figure 5.24.

However, unlike the advantages VPT brings to natural imaging, it failed to perform well on medical datasets as evident from cases reported in Figure 5.24. For

all three reported cases, although SAM_{EM} generated masks for both lungs (shown in the region proposal column), however, visual prompting did not aid CLIP in correctly recognizing the left and right lungs. With the gray reverse mark and contour visual prompt, the problem of retrieving the same mask for both lungs persists as shown in columns ROI 1 and ROI 2 of Figure 5.24. While using red bounding box prompts, the problem was worse as CLIP classified non-lung regions as lungs. Consequently, none of the visual prompts proved effective in aiding CLIP for localization tasks within the medical imaging domain.

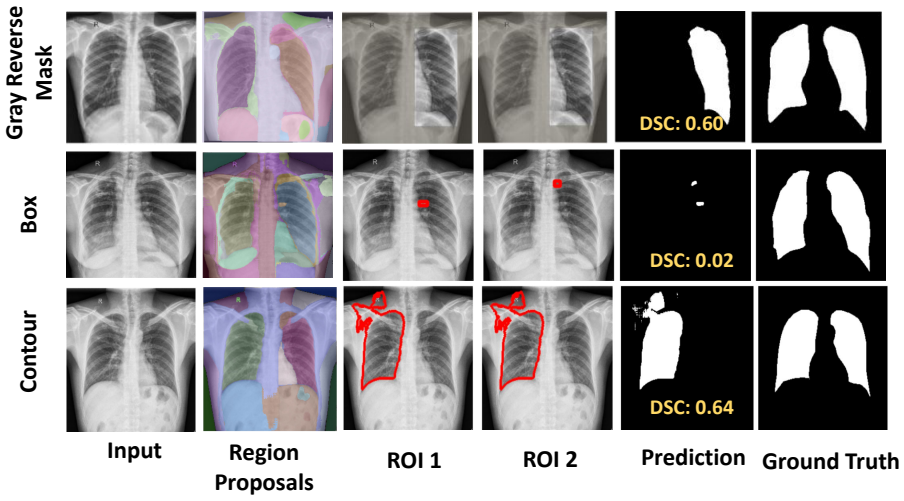


Figure 5.24: VPT results on X-ray labels and masks dataset [16]: Although SAM_{EM} generated masks for both lungs, VPTs did not facilitate CLIP in accurately retrieving ROIs.

Furthermore, when these three VPT methods are compared with the proposed SaLIP framework which does not use visual prompting, instead, it feeds the set of crops of the original image to CLIP which are generated according to masks generated by SAM_{EM} (discussed in Section 5.3.2), our framework still achieves a superior DSC of 0.65 as compared to other prompts as reported in Table 5.5.

These results indicate that, unlike the advantages of visual prompting in the natural imaging domain, achieving similar benefits in the medical imaging domain is challenging. These insights shaped the direction of the subsequent research presented in Chapter 6.

Table 5.5: Evaluation of visual prompting to enhance CLIP's recognition performance.

VPT	Dice Score
Box	0.49
Reverse blur	0.60
Contour	0.61
Crops (ours)	**0.65**

5.5.6 Ablations

Different SAM Variants

There are three different variants of SAM, primarily differentiated by the type of encoder employed. The encoder is a masked auto-encoder (MAE) [14], pre-trained on ViT [258]. The three variants are: Base version has ViT-B, SAM-Larg has ViT-L, and SAM-H has ViT-H (huge). The quantitative results of our proposed SaLIP framework with each of these variants are reported in Table 5.6. Notably, SAM-H demonstrates superior performance due to the enhanced capabilities of the ViT-H encoder. As a result, SaLIP strategically leverages the SAM-H variant to maximize segmentation accuracy.

Importantly, our proposed method, SaLIP, is a test-time adaptation framework that enables zero-shot segmentation without requiring additional training or fine-tuning (Section 5.3.2). As a result, integrating the ViT-H encoder into the SaLIP framework incurs no additional computational overhead associated with training large foundation models, ensuring operational efficiency while significantly boosting performance.

Table 5.6: Ablation: Comparison of SAM's variant.

Dataset	SAM-B	SAM-L	SAM-H
CC359 [11]	0.80	0.89	**0.94**
X-ray [16]	0.71	0.76	**0.83**
HC18 [17]	0.66	0.76	**0.81**

Table 5.7: Ablation: Performance comparison between SAM-CLIP and SaLIP (ours).

Dataset	SAM-CLIP	SaLIP
CC359 [11]	0.89	**0.94**
X-ray [16]	0.80	**0.83**
HC18 [17]	0.78	**0.81**

SaLIP vs SAM-CLIP

Our proposed SaLIP framework follows the sequence $SAM_{EM} \rightarrow CLIP \rightarrow SAM_{PSM}$ as detailed in the Section 5.3.2. To assess the benefit of this sequence, an ablation study is conducted using only the $SAM_{EM} \rightarrow CLIP$.

With $SAM_{EM} \rightarrow CLIP$, SAM_{EM} first generates masks corresponding to different image regions with spatial dimensions that match those of the input image. Based on the predicted regions, image crops are created, and CLIP is used to retrieve the crop corresponding to the ROI. The SAM_{EM}'s generated mask corresponding to the retrieved ROI is then considered the final segmentation prediction. It may include irrelevant or extraneous information outside the target region as demonstrated by the dice score reported in Table 5.7.

In contrast, our proposed method uses the ROI crop retrieved by CLIP to obtain the bounding box coordinates corresponding to the crop within the original input image. This box prompt is processed by SAM's prompt encoder (Figure 5.4). It helps to refine the segmentation process by guiding the SAM's decoder to focus solely on the relevant features within the bounding box. This approach enhances segmentation precision by minimizing the small surrounding areas within the bounding box that are outside the actual region of interest, thereby improving segmentation, as demonstrated in Table 5.7. Thus, the proposed sequence $SAM_{EM} \rightarrow CLIP \rightarrow SAM_{PSM}$ allows for a more targeted approach to segmentation, ensuring that the output accurately reflects the desired objects or areas within the image, leading to improved performance and reliability of the proposed SaLIP framework.

5.6 Summary

This chapter presents our work on test-time adaptation of foundation models for zero-shot medical organ segmentation. It comprehensively investigated the challenges in the adaptation of foundation models – trained predominantly on natural images – to the downstream medical imaging tasks.

Prompt engineering is an integral component of foundation models and the performance of these models is heavily influenced by the quality of prompts (Section 5.5.1). In the natural imaging domain, prompt engineering is often straightforward because there is an abundance of annotated data to extract meaningful insights and it does not need specialized domain knowledge. However, in medical imaging, there is a scarcity of labeled data and domain expertise, which complicates prompt engineering.

Additionally, the existing task-specific supervised methods for the adaptation of foundation models in the medical domain have limitations. These methods require a large amount of labeled data and do not fully exploit prompting capabilities which is the primary strength of foundation models (Section 5.2.2). On the other hand, adaptation using parameter-efficient fine-tuning largely depends on the nature of the dataset and the specific segmentation task, as demonstrated experimentally in Chapter 4. Additionally, fine-tuning foundation models has considerable computational challenges due to their substantial size and intensive resource requirements.

To address the aforementioned challenges, a new pipeline called SaLIP is proposed to perform test-time adaptation foundation models for the medical imaging domain. SaLIP leverages a cascade of foundation models pre-dominantly trained on natural imaging to perform zero-shot medical organ segmentation. Notably, SaLIP eliminates the need for additional training, annotated data, task-specific fine-tuning or specialized domain knowledge for prompt engineering, thereby simplifying the adaptation of foundation models to downstream medical segmentation tasks (Section 5.3.2).

The proposed approach has been evaluated for organ segmentation across diverse

medical imaging modalities: brain from MRI scans, lung from X-rays, and fetal head from ultrasound. Despite the differences and lack of correlation between these tasks, SaLIP consistently demonstrated strong performance, achieving dice scores of 0.94, 0.83, and 0.81, respectively (Section 5.5). These results highlight the robustness and generalization capabilities of our method.

Furthermore, as SaLIP is fully adapted at test time, it considerably reduces the computational overhead typically associated with the adaptation of foundation models in supervised settings (Section 5.3.2). By leveraging large language models (LLMs), SaLIP ensures efficient and effective segmentation without additional complexity, making it accessible for a wide range of applications in medical imaging.

5.6.1 Insights

While SaLIP demonstrated its strengths in medical organ segmentation, its experimental evaluation gave us valuable insights:

1. Vision-language models like CLIP demonstrate impressive performance in global image-level tasks, but their effectiveness is limited when it comes to instance-level tasks in the medical imaging domain. This limitation is evident in our experiments which involve recognizing lung regions based on spatial location (i.e., left or right lung- Section 5.5.5, Appendix A.1.2). This challenge is even more pronounced in fine-grained medical imaging tasks, such as recognizing pathological structures, which often have variable spatial locations and morphology.

2. In contrast to natural imaging, where LLMs have demonstrated impressive performance in tasks like visual perception [276], their application in medical imaging has been less effective. This limitation stems from the complexity and domain-specific challenges unique to medical imaging. LLMs, being primarily designed for general tasks, are not inherently equipped to address the specialized requirements of medical image analysis as experimentally demonstrated in Sections 5.5.5, 5.5.3 and Figure 5.18.

3. SaLIP generalizes well across different medical organ segmentation tasks, as demonstrated by comprehensive experimental evaluations (Section 5.5). Disease diagnostics pose unique challenges due to the small size and variability of affected regions. Therefore, it is essential to explore the effectiveness of foundation models in challenging medical disease diagnostics tasks, particularly regarding their ability to accurately identify and segment small disease regions.

These insights helped shape the further research, which is subsequently discussed in Chapter 6.

Chapter 6

Adaptation of Foundation Models for Fine-Grained Medical Imaging Analysis

This chapter presents our work on adapting foundation models to perform challenging fine-grained medical imaging tasks. It addresses Research Question 4 (RQ4): "Can foundation models be effectively adapted to challenging fine-grained medical imaging tasks?" While foundation models' transferability is typically assessed on coarse/global image-level tasks, medical imaging analysis often demands precise, highly specific, and granular analysis to address detailed diagnostic and clinical requirements. In this context, a new framework called SaLIP-V is proposed to adapt foundation models to fine-grained medical imaging tasks. It has been evaluated on two fine-grained medical imaging tasks: (a) localization/recognition and segmentation of anatomical structures based on spatial location; and (b) recognition and segmentation of pathological structures, which exhibit considerable variability in shape, morphological structure, and spatial location. Specifically, this work achieved improved performance on the first task compared to our approach introduced in Chapter 5. The code to replicate the experiments is available at: https://github.com/aleemsidra/SaLIP-V.

Section 6.1 provides an introduction and our motivation for evaluating the performance of foundation models and the need for their adaptation to fine-grained medical tasks. Section 6.2 presents a literature review of existing approaches. Section 6.3 outlines the proposed framework and provides a detailed explanation of its architectural design. Section 6.4 outlines the datasets used and evaluation procedures. Section 6.5 presents the results, analysis, and ablation studies. Finally, Section 6.6 summarizes the findings and insights gained from this work and highlights potential directions for future research.

6.1 Introduction

Vision-Language Models (VLMs), such as CLIP [132] and ALIGN [278], have demonstrated impressive zero-shot transferability on image-level visual perception. These foundation models are predominantly trained on large-scale image caption corpora for image-level supervision [279, 280, 281]. As a result, they excel in global image-level tasks, such as differentiating between domains like "Chest X-ray" or "Chest CT". Consequently, these models demonstrate limited generalization in fine-grained tasks that demand precise recognition and classification, such as differentiating between specific sub-regions of an image and identifying those that contain pathological structures (e.g., tumors) [51, 18].

VLMs also face significant challenges in medical imaging tasks that require semantic understanding, such as spatial reasoning to interpret the relationships between different image regions or organs. This limitation is demonstrated in our experimental evaluation in Chapter 5, where CLIP performed well in recognizing larger structures, such as organs, which occupy a substantial portion of the image. However, CLIP showed poor performance in recognizing lungs based on spatial location (e.g., "left" or "right" Section 5.5.5). This poor generalization can be attributed to CLIP's lack of semantic knowledge and thus it fails to reliably distinguish between regions based on relative spatial locations, such as "left" and "right" lung.

Similarly, LLMs are not inherently suited for instance-level tasks in the medical

imaging domain. This is experimentally demonstrated in Chapter 5, where we task LLMs to generate textual prompts describing lungs according to the distinct features and spatial location of the lungs (Appendix A.1.2). However, the generated prompts proved ineffective for the task at hand, as evidenced by the results presented in Section 5.5.5. In a second task, we evaluated LLMs for generating subroutines to create bounding box prompts for the brain region in MRI scans. In this case, LLMs faced scalability challenges due to the diverse and complex nature of medical tasks, as discussed in Section 5.5.3.

Another major challenge is VLMs show limited performance in fine-grained medical imaging analysis such as recognizing pathological structures due to their subtle nature, small size (e.g., tumors), inconsistent morphology (which can further alter with disease progression), and high variability in appearance. Collecting large-scale high-quality datasets for every visual task, specifically for challenging fine-grained medical tasks, is labor-intensive and too expensive to scale [282]. Eventually, it results in poor generalization of VLMs to fine-grained medical tasks.

A common method for encoding location information is to crop the image around the desired area, creating a zoomed-in visual representation [278, 283, 121]. However, this approach often discards the global context essential for fine-grained medical tasks, such as disease recognition. One such example is illustrated in Figure 6.1, where the colonoscopic image containing a polyp is cropped, and the resulting image loses the global context needed to correctly recognize the polyp. Therefore, for fine-grained medical tasks, processing only the cropped region can lead to misclassification due to the loss of essential contextual information (Section 6.5.1).

Recent research in natural imaging has explored visual prompting as an alternative to cropping, aiming to enhance the zero-shot performance of VLMs. Visual prompting is a technique used in image-language tasks, where visual markers such as colorful boxes or circles are added directly onto an image to highlight specific regions within the image [18, 51, 284]. These approaches hypothesize that the model has encountered the selected visual markers during training, allowing it to under-

stand the underlying concepts. An example of visual prompting specifically, a red bounding box to highlight the polyp region in a colonoscopy image is shown in Figure 6.1 (visual prompt). Also, in medical imaging analysis, unlike cropping the ROI,

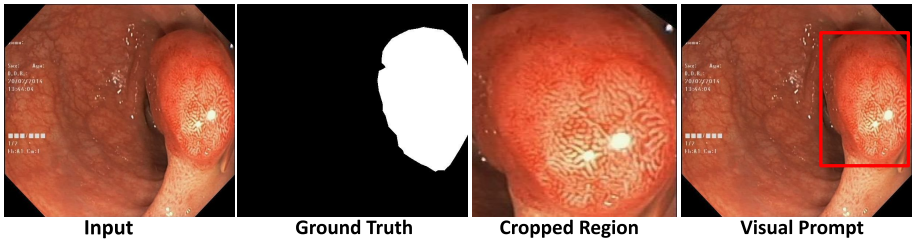


Figure 6.1: A polyp in the colon [19]. Cropping ROI in the image, results in the loss of global context necessary for fine-grained medical imaging tasks. While the visual prompting preserves the global context.

the visual prompt maintains the global context necessary for disease recognition as shown in Figure 6.1 (visual prompt).

Currently, the application of visual prompting particularly for fine-grained medical tasks largely remains unexplored. To address the above-mentioned challenges, a new framework called SaLIP-V is proposed. It adapts foundation models to fine-grained medical imaging tasks. SaLIP-V has two main phases:

1. **Classification of fine-grained image sub-regions:** In the first phase, a lightweight linear classifier is trained in a few-shot setting to classify fine-grained sub-regions of an image, as illustrated in Figure 6.2 (a). Our framework autonomously generates masks for various regions within the image by utilizing the Segment Anything Model (SAM). These region proposals are used to overlay visual prompts (VPT) on the image, resulting in a set of images– labeled as "Images + VPT" in Figure 6.2 (a), each highlighting a sub-region of the image. The images in this set are classified as either regions of interest (ROI) or irrelevant areas. To optimize the classification, this set is first processed through CLIP's frozen visual encoder (CLIP-V) to generate robust feature representations. These extracted features are then adapted by a lightweight linear classifier, which is trained in a few-shot setting to specifically distinguish ROIs from irrelevant regions from the set "image + VPT" as

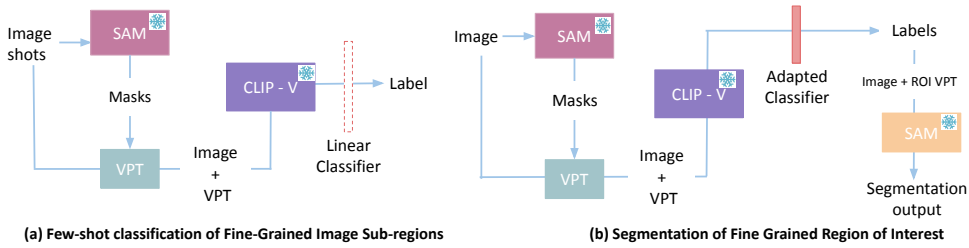


Figure 6.2: The proposed SaLIP-V framework: a) few-shot classification of fine-grained image sub-regions: a linear classifier is trained in a few-shot setting to classify different sub-regions of the image. b) segmentation of fine-grained regions: the adapted classifier is used to identify the correct ROI from the pool of SAM-generated various sub-regions, and the selected ROI is then segmented.

illustrated in Figure 6.2 (a). This few-shot training is done only once to initialize the linear classifier for the effective classification of fine-grained image sub-regions.

2. **Segmentation of the fine-grained region of interest:** images are first processed through SAM to generate masks for potential regions in the images as shown in Figure 6.2 (b). VPTs are then applied to the original input image based on these identified regions. The resulting set of images (Image + VPT) is then processed by CLIP-V to extract meaningful visual representations, to facilitate the identification of fine-grained regions. These features are then classified into distinct categories, such as ROI or irrelevant, using the adapted linear classifier from the first phase (Figure 6.2(a)). The image labeled as ROI, its VPT coordinates, and the original image is processed through the SAM to segment the fine-grained ROI, as illustrated in Figure 6.2(b).

Contributions

- To the best of our current knowledge, our proposed framework called – SaLIP-V is the first work focused on adapting foundation models to fine-grained medical imaging tasks such as anatomical structure localization based on spatial context, and segmentation of pathologies with varying spatial locations and morphology.

- SaLIP-V effectively addresses the challenge of labeling fine-grained regions within medical images. An automated approach for labeling fine-grained anatomical structures and pathologies is proposed that does not need manual labeling or domain expertise.

- The SaLIP-V framework leverages a few-shot adaptation strategy to adapt foundation models to fine-grained medical analysis tasks. It achieves this by adapting their features with a straightforward linear classifier while keeping the foundation models frozen.

6.2 Related Work

The related work on segmentation of medical imaging using foundation models, specifically visual language models, is discussed in detail in Chapter 5 in Section 5.2. This section will primarily focus on the work related to visual prompting to enhance CLIP's recognition capability, which is a key component of our proposed SaLIP-V framework, as outlined in (Section 6.3).

Colorful Prompt Tuning (CPT) colors different regions of an image and uses a captioning model to predict which object in an image an expression refers to by predicting its color [253]. While generating textual prompts to refer to specific objects in natural images is feasible, creating effective textual prompts for specific regions in medical images presents challenges. Medical terminology is complex and specialized, making prompt engineering for these applications more difficult as experimentally demonstrated in Section 6.5.2. CPT [253] creates crops of different image regions and assigns them semantic labels using colored prompts. However, cropping images to encode location information results in a loss of global context, which is essential for fine-grained medical tasks, as discussed in Section 6.1.

RedCircle [18] demonstrated that drawing red circles around objects within an image can effectively distinguish instances by enclosing them in inscribed ellipses derived from proposal boxes. This approach was evaluated on the CUB-200-2011

(CUB) dataset [285], which already includes key point annotations, and on the SPair71k [286] animal image dataset, for which manual annotation was performed. Since the SPair71k dataset consists of animal images, the visual prompt engineering did not require specialized domain expertise for this dataset. Fine-grained visual prompt engineering is achieved by introducing a Blur Reverse Mask to highlight specific image regions [51]. A general segmentation model is first used to generate masks for different regions within the image. For each region, the remaining areas of the image are blurred while the target region is highlighted. This pool of Blur Reverse Mask images is then fed to CLIP, which is tasked with selecting the image containing the region of interest. However, medical imaging modalities, such as lung imaging, often have a predominantly gray appearance, making the use of reverse gray blur masking less effective. This limitation is experimentally validated with results reported in Section 5.5.5.

Alpha-CLIP is an enhanced version of CLIP with an auxiliary alpha channel to suggest attentive regions and fine-tuned with constructed millions of RGBA region-text pairs [287]. Alpha-CLIP not only preserves the visual recognition ability of CLIP but also enables precise control over the emphasis of image contents. While Alpha-CLIP demonstrates effective performance in various scenarios requiring region focus, its current structure and training process limits its ability to model relationships between multiple objects.

While the application of visual prompting for fine-grained tasks in natural imaging has been explored, to the best of our knowledge it is not widely adapted for fine-grained medical imaging analysis tasks. One study utilized visual prompting for lung cancer classification [279], however, the visual prompts are engineered using annotated data. Furthermore, lung cancer is a prevalent disease with a significantly larger body of research compared to other pathological conditions, and abundant annotated data is available, which can facilitate visual prompt engineering for this task. The evaluation of this approach on more complex fine-grained medical imaging analysis tasks is necessary to assess its generalization.

To tackle these challenges, a new framework SaLIP-V is proposed, designed specifically for adapting foundation models for fine-grained medical tasks, as discussed in the following section.

6.3 Methodology

This section presents our proposed framework named SaLIP-V, which is designed to adapt foundation models specifically for fine-grained medical imaging tasks. SaLIP-V consists of two main phases: (a) Few-shot setup for fine-grained image sub-regions classification: Training a linear classifier on the top of the frozen CLIP visual branch to enable classification across various fine-grained image regions and instances as shown in Figure 6.2 (a). This few-shot training is done only once to effectively initialize the linear classifier for the effective classification of fine-grained image regions. (b) After training the linear classifier, it is used to classify different sub-regions of the image into distinct classes. The regions labeled as regions of interest are then segmented, as illustrated in Figure 6.2 (b).

The key preliminaries for understanding SaLIP-V are the Segment Anything Model (SAM) and CLIP, which are discussed in detail in Section 5.3.1.

6.3.1 Foundation Models Adaptation

The zero-shot adaptation of the foundation model showed poor performance on fine-grained medical tasks (Sections 5.5.4, 5.5.5,and 6.5.1). The first step of our proposed approach is to use visual prompting and a linear probe as a lightweight adaptation strategy to effectively adapt foundation models to classify fine-grained sub-regions in medical imaging. The linear probe is a trainable linear layer, which is trained on the features extracted from the foundation model. It is trained in a few-shot setting while keeping the foundation model itself fixed.

The primary objective of the linear layer is to accurately classify different regions within an image, with a specific focus on correctly identifying the ROI. However,

several challenges arise in this context: **a) Fine-grained mask acquisition:** obtaining masks for different sub-regions in medical imaging is extremely challenging. **b) Lack of region-level semantic labels:** in medical imaging, the annotated data is not readily available specifically precise labels for fine-grained regions within the image. **c) Loss of global context:** to isolate a specific ROI, the input image is typically cropped, and a VLM is tasked to retrieve the crop corresponding to ROI. However, cropping can lead to a loss of global context necessary for fine-grained medical tasks as illustrated in Figure 6.1. It eventually leads to sub-optimal performance for fine-grained medical tasks (experimentally demonstrated in Section 6.5.1).

To address the lack of region-level annotations for medical images, we employed our approach proposed in Chapter 5. It automates the process of mask generation for the different potential regions in the image by leveraging the Segment Anything Model's "everything mode" (SAM_{EM}) (Section 5.3). Notably, this mask-generation process is fully automated and does not require labeled data or domain expertise.

The generated region proposals lack semantic labels. To assign each proposed sub-region to a specific category (ROI or irrelevant), an automated labeling approach is proposed. This approach computes the dice score for each of the SAM generated region proposals by comparing it with the ground truth of the input image. Masks with a dice score above a specified threshold are classified as regions of interest (ROIs), while the remaining are labeled as "irrelevant" class as shown in Figure 6.3 (a). This region-based labeling process can be mathematically represented as follows:

$$L_i^{(k)} = \begin{cases} \textit{'polyp'} & \text{if Dice}(M_i, GT) \geq \tau \\ \textit{'irrelevant'} & \text{if Dice}(M_i, GT) < \tau \end{cases} \tag{6.1}$$

where k denotes the number of shots for which labels are generated, M_i is the set of potential sub-regions in an image generated by SAM_{EM}, GT is the ground truth, τ is the threshold value used for comparing the dice score to classify a mask as either ROI or irrelevant.

The next step is to classify different SAM_{EM} generated image sub-regions and

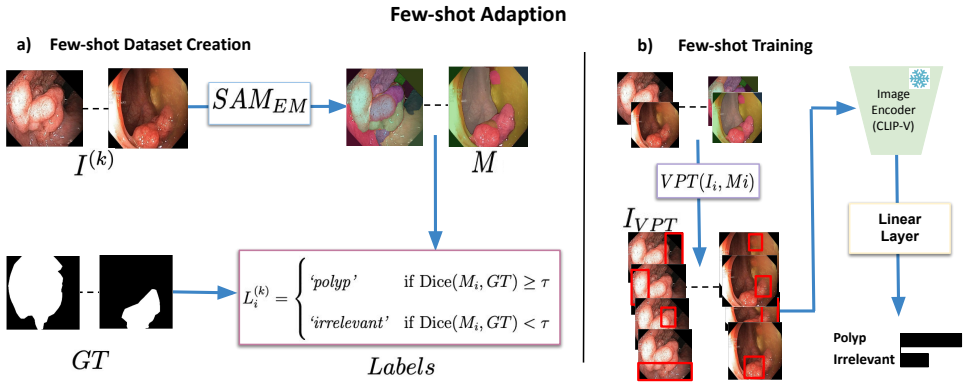


Figure 6.3: Visual prompting and linear classifier adaptation: a) Few-shot dataset creation: SAM_{EM} generates region proposals for various regions in the image (M). These sub-regions are labeled with specific class labels by comparing (M) with the ground truth (GT). b) Few-shot training: Visual prompts (VPT) are overlaid on the input image to create I_{VPT}. This set is processed through frozen CLIP-V, and a linear layer is trained using few-shot examples to classify sub-regions as either ROI or irrelevant.

retrieve the regions corresponding to ROI. To optimize this classification, visual prompting is employed to preserve the overall global context necessary for the classification of fine-grained regions within the image as shown in Figure 6.1 (visual prompt). However, for the medical domain, the annotated data is not readily available for visual prompt engineering (Section 1.1.4).

To address the challenge of visual prompt engineering for fine-grained medical imaging tasks, our proposed approach leverages SAM_{EM} generated pool of region proposals and draws "red bounding box" visual prompts on the input image, positioned according to the specific coordinates of each proposed region as illustrated in Figure 6.3 (b) labeled as (I_{VPT}). The selection of this particular type of visual prompt is based on an ablation conducted to evaluate the effectiveness of different visual prompts for medical imaging tasks (Section 5.5.5). The process of applying visual prompt markers to the image can be mathematically represented as follows:

$$I_{VPT} = VPT(I, M_i) \tag{6.2}$$

where I is the input image, M_i is the set of masks generated by SAM_{EM} for I, VPT is the function that draws visual prompts on the input image according to

the coordinates of each mask in M_i. This results in a pool of images each having an overlaid prompt highlighting different regions of the image (I_{VPT}) as shown in Figure 6.3 (b).

While visual prompting enhances the zero-shot recognition capabilities of CLIP, it is not sufficient for complex, fine-grained medical tasks (demonstrated by the experimental results in Section 6.5.1). To optimize foundation models for these fine-grained tasks some form of adaptation is required.

To accomplish this, our proposed method trains a simple linear layer in a few-shot setting as shown in Figure 6.3 (b). Specifically, this linear layer adapts the features extracted from CLIP's visual branch (CLIP-V). To prevent overfitting, CLIP-V is kept frozen and is used as a feature extractor to obtain rich visual representations from I_{VPT} that can aid in classifying various image regions. The linear layer is then trained on these extracted features to categorize images in I_{VPT} into distinct classes/regions (e.g. "polyp" or "irrelevant") as illustrated in Figure 6.3 (b). This few-shot adaptation is performed only once. After adaptation, the linear layer is integrated into the second phase of our framework as shown in Figure 6.4.

Our proposed framework does not utilize CLIP's textual branch. This decision is based on the experimental evaluation that showed poor performance when applying it to complex, fine-grained tasks requiring precise localization of anatomical structures (see Section 6.5.2). This sub-optimal performance stems from the fact that CLIP has primarily been trained on image captions for coarse image-level tasks, which hinders fine-grained perception, specifically for medical tasks. Furthermore, CLIP-V is selected over other vision encoders based on an ablation study demonstrating its effectiveness for medical image analysis (Section 6.5.3).

6.3.2 Fine-Grained Segmentation

The second phase of our proposed SaLIP-V framework is used for segmenting fine-grained regions in medical imaging in a zero-shot setting. The process begins by leveraging SAM_{EM} (Section 5.3.2) to generate the masks for different regions in an

image (M). These generated masks are then used for visual prompt engineering, which involves drawing red bounding box prompts on the input image based on the coordinates of the generated masks using the VPT function in Eq. 6.2. This process creates a set of images with visual prompts, referred to as (I_{VPT}) as shown in Figure 6.4.

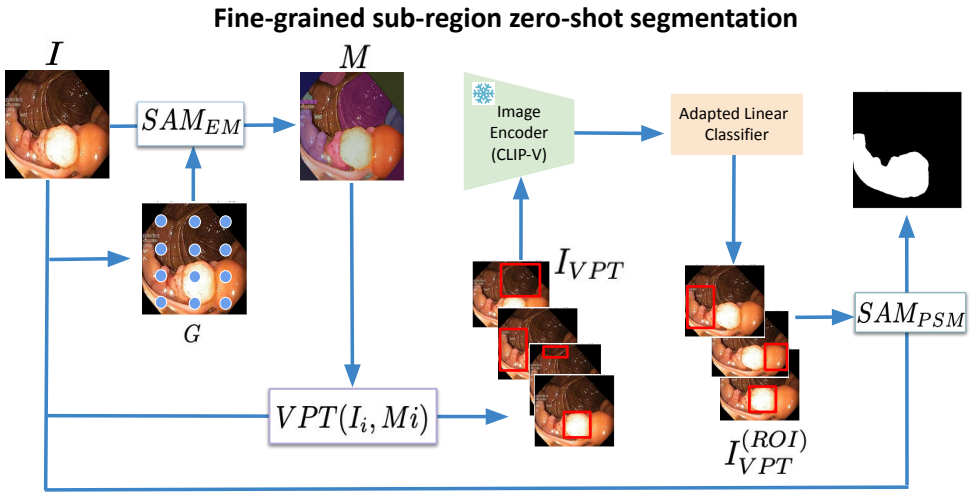


Figure 6.4: Architecture of our proposed SaLIP-V framework: SAM_{EM} segments image sub-regions using a grid of key-points (G), red bounding box visual prompts are overlaid on the input image I according to the resulting sub-regions (M) generating a pool of images I_{VPT}. These images are then processed through the frozen CLIP-V model and classified by the adapted linear classifier. The images categorized as ROI by the classifier are subsequently passed to SAM_{PSM} to get the final segmentation mask.

The pool of images with overlaid visual prompts highlighting different image sub-regions (I_{VPT}) are processed through the frozen CLIP-V to extract rich features, which are then passed to an adapted linear classifier (from the first phase of our method, discussed in Section 6.3.1). This adapted linear classifier categorizes the images into distinct classes such as "polyp" or "irrelevant" for the polyp segmentation task. Based on the linear classifier's logits, each image in (I_{VPT}) that are classified as regions of interest are selected from I_{VPT} as illustrated in Figure 6.4. This process can be mathematically represented as:

$$I_{VPT}^{(ROI)} = \{I_i \in I_{VPT} \,|\, C_i = \text{“polyp”}\} \tag{6.3}$$

where I_{VPT} is a set of images each with an overlaid visual prompt each highlighting a different image sub-region, I_i refers to i^{th} image in I_{VPT}, $I_{VPT}^{(ROI)}$ is the image which is categorized as ROI by the classifier, C_i is ROI class label ("polyp" in this case).

Finally, the bounding box prompts for the retrieved $I_{VPT}^{(ROI)}$ and the original input image are passed to SAM's promptable segmentation mode (SAM_{PSM}) (Section 5.3). In this mode SAM segments the regions enclosed by the visual prompt as illustrated in Figure 6.4.

6.4 Experimental Framework

6.4.1 Datasets and Metrics

Our proposed approach is evaluated on two distinct fine-grained medical imaging tasks: a) The localization and segmentation of anatomical structures based on their spatial location. For this task, SaLIP-V is assessed using the X-ray Masks and Labels dataset for lungs (described in Section 5.4.1). b) The recognition and segmentation of pathological structures that have varying morphology and spatial location. This evaluation is specifically done to recognize and segment polyps in colonoscopy images. For this purpose, the Kvasir-seg dataset [19] is utilized, which contains 1,000 images from inside the gastrointestinal tract during colonoscopy, featuring polyps alongside. The evaluation metrics used are accuracy for classification and dice similarity coefficient (DSC) for the segmentation.

6.4.2 Implementation Details

To adapt foundation models for fine-grained tasks, the first phase of our proposed approach involves few-shot classification (Section 6.3.1). For this few-shot setup, the extracted features from the models are adapted by training a linear classifier (Figure 6.3). For lung classification, 20 shots are used to train the linear classifier, while for polyp classification, 40 shots are used as it more challenging task than the former. The linear classifier is trained for 100 epochs using cross-entropy loss with

a learning rate of 1×-4 and a batch size of 32.

SaLIP-V utilizes the "huge" variant of SAM (ViT-H) from the official SAM repository [1] and incorporates the large variant of CLIP (ViT-L/14) from OpenAI's CLIP framework.

6.5 Results and Analysis

6.5.1 Zero-shot Performance of CLIP on Fine-Grained Medical Tasks

First, we evaluated the potential of visual prompting in improving the CLIP's zero-shot recognition performance for fine-grained medical tasks. The textual prompts for CLIP's text encoder are generated using GPT-3.5 [52]. This process of textual prompt engineering to construct textual sentences describing the different image sub-regions is detailed in Appendix A.1. The images are processed through CLIP in two different setups to evaluate the potential benefits of visual prompting:

- **Crop:** different image sub-regions are cropped according to SAM_{EM} generated image sub-region proposals (Section 5.3.2). CLIP's vision encoder then processes these zoomed-in crops of different image sub-regions (Figure 6.1-image crops).
- **Visual prompting (VPT):** different image sub-regions are highlighted using visual prompts, which are overlaid on the original image according to SAM_{EM} generated image sub-region proposals. The resulting set of images with overlaid visual prompts is passed to CLIP's vision encoder (Section 6.3.1).

The comparative analysis of these two configurations on CLIP's zero-shot classification performance is reported in Table 6.1. When CLIP was tasked to retrieve the crop corresponding to the region of interest (ROI) from the pool of crops of different sub-regions of a chest X-ray image, it only identified 1 out of 40 left lung

[1] https://github.com/facebookresearch/segment-anything, Accessed: [12.07.2024]

crops correctly, and for right lung 13 out of 40 crops were correctly classified, as reported in Table 6.1. Thus, cropping different regions of the image (Chest X-ray in our case) did not prove to be effective, as the global context to recognize the lungs based on spatial location is lost (Section 6.1).

In contrast, using VPT to highlight different sub-regions, as discussed above, improved CLIP's zero-shot classification performance. Specifically, for the left lung, CLIP correctly classified 32 out of 40 images, whereas, with the crop configuration, only 1 out of 40 was correctly identified, as reported in Table 6.1. However, a similar pattern was not observed for the right lung classification as only 1 out of 40 right lung samples was correctly classified using VPT as shown in Table 6.1.

Table 6.1: Zero-Shot Classification Performance of CLIP: Crops vs. BBOX VPT.

Class	No of. Images	Crop		BBOX VPT	
		Correct	Accuracy (%)	Correct	Accuracy (%)
Left Lung	40	1	2.5	32	80
Right Lung	40	13	32.5	1	2.5
Irrelevant	590	540	91.5	335	56.8

The key insight from this experiment is that while VPT can enhance CLIP's recognition capabilities, it alone is insufficient for fine-grained tasks such as the recognition of lungs based on spatial location. As shown in Table 6.1, while VPT boosted CLIP performance for left lung classification, it exhibited inconsistent performance for the right lung.

Therefore, to tackle complex fine-grained medical imaging tasks, especially in the absence of domain expertise and annotated data for visual prompt engineering, it is essential to supplement visual prompting with additional methods to optimize the performance of foundation models in such scenarios.

6.5.2 Few-Shot Adaptation of CLIP using Adapters

Although visual prompting improved CLIP's recognition performance compared to cropping image regions (Section 6.5.1), further optimizations are necessary to en-

hance performance in fine-grained medical imaging tasks due to inconsistencies in the results, as reported in Table 6.1.

Fine-tuning the full CLIP model in a few-shot setting is ineffective due to its large number of parameters, the limited availability of training examples, and the associated risk of overfitting [132]. To better adapt CLIP, we kept the CLIP's backbone frozen and instead leveraged lightweight adapters to adapt the features extracted from CLIP. These adapters are trained in a few-shot setting. We evaluate three different adapters:

- **Linear Adapter**: An MLP adapter consisting of two linear layers is trained to adapt the features extracted from CLIP's image and/or text encoders.
- **CLIP-A [200]:** a small number of additional learnable bottleneck linear layers are trained to adapt CLIP's textual and image branches while keeping the original CLIP backbone frozen. To prevent overfitting, CLIP-A further adopts residual connections to dynamically blend the fine-tuned knowledge with the original knowledge from CLIP's backbone.
- **CLIP-A-Self [120]:** applies a self-attention mechanism on the set of all textual sentences for any class that is fed to CLIP's text encoder. It learns to select and aggregate the best subset of visual descriptive sentences for the dataset from the few-shot training set.

To evaluate the impact of using the aforementioned adapters on improving CLIP's performance for fine-grained classification, we adapted CLIP using three different settings: **1)** Adaptation of CLIP's textual branch, **2)** Adaptation of CLIP's visual branch, and **3)** Joint adaptation of both textual and visual branches.

The experimental evaluation for each setup is detailed in the subsequent Sections.

1) CLIP's Textual Branch Adaptation

In this setup, only the CLIP's textual branch features are adapted using adapters, while the image branch remains frozen. In contrast to the zero-shot setting (Sec-

tion 6.5.1), all three adapters improved CLIP's classification performance as reported in Table 6.2. The column "corr" shows the number of instances that were correctly classified and "acc" shows the class-wise accuracy.

Table 6.2: Few-shot adaptation of CLIP's textual branch using various adapters.

Class	Zero-shot		Linear		CLIP-A (0.2)		CLIP-A (0.5)		CLIP-A-Self	
	Corr	Acc	Corr	Acc	Corr	Acc	Corr	Acc	Corr	Acc
Left Lung	32	80	7	17.5	39	97.5	34	85.0	7	17.5
Right Lung	1	2.5	14	35	18	45.0	17	42.5	2	5
Irrelevant	335	56.7	433	73.4	497	73.4	491	83.2	533	90.3
Avg. Acc	46.4		67.8		**82.7**		80.9		80.9	

Specifically, CLIP-A with residual ($\alpha = 0.2$) outperformed the other adapters. α controls residuals style feature blending with the original pre-trained CLIP's features. CLIP-A significantly improves classification accuracy, improving from 31% in the zero-shot setting to 82.7%.

This suggests that few-shot training of lightweight adapters to adapt the foundation model's extracted features, combined with visual prompts, enhances performance while enabling efficient adaptation. Notably, this approach requires no modification of the foundation model, as only the adapters are fine-tuned (Section 6.3.1).

The second phase of the proposed method involves segmenting the ROI, where the adapted classifier is utilized to identify the image sub-region corresponding to the ROI (Section 6.3.2). Given that CLIP-A outperformed other adapters (Table 6.2), it was integrated our SaLIP-V pipeline for this purpose (Figure 6.4).

- **SaLIP [121]:** it is our proposed approach for test-time adaptation of foundation models for zero-shot organ segmentation (introduced in Chapter 5). It is used as a baseline to evaluate the performance improvement with few-shot adaptation of CLIP via light-weight adapters. The crops showing the zoomed-in image sub-regions are processed through CLIP within the SaLIP framework (Section 5.3.2).

- **SaLIP-VPT:** Instead of using image crops, visual prompts are overlaid to

the original image to highlight different regions based on masks generated by SAM_{EM} (Figure 6.4 (I_{VPT})). These sets of images with visual prompts are then processed through CLIP within the SaLIP framework (Figure 5.7).

- **SaLIP-A**: The above mentioned adapted CLIP-A is integrated into our proposed SaLIP-V framework (Section 6.3.2). The image with visual prompts is processed through SaLIP-V, where adapted CLIP-A is used to categorize these images into specific classes.

The quantitative results of this evaluation are reported in Table 6.3. The column labeled "Class Acc" reports the average classification accuracy for the left and right lungs achieved by CLIP on few-shot data, while the column "Seg (DSC)" shows the corresponding average segmentation dice similarity coefficient (DSC) across the entire dataset for both lungs.

CLIP-A significantly enhanced CLIP's classification performance, increasing its average accuracy from 46.4% (zero-shot) to 82.7% (few-shot) in classifying all categories i.e., left, right, and irrelevant—across a pool of images with various visual prompts corresponding to different regions (Table 6.2). However, when CLIP-A is integrated into SaLIP-A (mentioned above), the final segmentation DSC is 0.654, which is significantly lower than the DSC of 0.834 achieved by our proposed test-time adaptation SaLIP pipeline (Chapter 5). Thus, CLIP-A did not prove effective when applied to classify different sub-regions across the entire dataset, as indicated by the corresponding segmentation DSC in Table 6.3.

Table 6.3: Performance comparison of different SaLIP-V configurations on the chest X-ray dataset.

SaLIP	Class Acc	Seg (DSC)
SaLIP-VPT	0.31	0.56
SaLIP-A	**0.82**	0.65
SaLIP [121]	-	**0.83**

Analysis: Swapping Left and Right Lung Prompts

One potential reason for the lower segmentation DSC of SaLIP-A (Table 6.3) could be attributed to the naming conventions used in medical image descriptions. In medical terminology, the left side of an image typically represents the patient's right side, and vice versa. Specifically, our evaluation focuses on lung segmentation, and regions of interest within the X-ray are 'left' and 'right' lungs. The textual prompts are according to conventional left and right orientation in natural imaging (reported in Appendix A.1.1). This mismatch of conventions may have contributed to the lower performance of SaLIP-A, as reported in Table 6.3.

To address this issue, prompts for "left" and "right" lungs were adapted to align with medical conventions by swapping the labels for the left and right lungs accordingly. All three adapters are then re-evaluated after swapping prompts for "left" and "right" lungs (Appendix A.1.2). The results of this re-evaluation are reported in Table 6.4.

Table 6.4: CLIP's textual branch adaptation branch using various adapters with swapped prompts.

Class	Linear		CLIP-A (0.2)		CLIP-A (0.5)		CLIP-A-Self	
	Corr	Acc	Corr	Acc	Corr	Acc	Corr	Acc
Left Lung	22	17.5	39	97.5	34	85.0	7	17.5
Right Lung	22	35	18	45.0	17	42.5	2	5
Irrelevant	317	73.4	497	73.4	491	83.2	533	90.3
Avg. Acc	44.3		**52.7**		51.8		39.7	

Contrary to expectations, adhering to medical domain naming conventions did not enhance performance. The adapters with swapped prompts failed to improve CLIP's results and significantly underperformed as reported in Table 6.4. In contrast, the original prompts, which followed naming conventions from general natural imaging, delivered better outcomes. This is reflected in the Avg. Acc" row in Table 6.2 and Table 6.4. Specifically, the original prompts that followed the natural imaging naming convention for describing spatial location "left" and "right" resulted in an average classification accuracy of 82.7% while with swapped prompts

the average classification accuracy was 52.7%. Thus, swapping prompts to follow the medical imaging naming conventions, did not improve CLIP's recognition based on spatial location.

2) CLIP's Visual Branch Adaptation

In this setup, only the features extracted from CLIP's visual branch are adapted using adapters, while the textual branch remains frozen. The results for this evaluation are reported in Table 6.5.

Table 6.5: CLIP's visual branch adaptation: comparison of different adapters.

Class	Zero-shot		Linear		CLIP-A ($\alpha : 0.2$)		CLIP-A ($\alpha : 0.5$)	
	Corr	Acc	Corr	Acc	Corr	Acc	Corr	Acc
Left Lung	32	80	39	97.5	16	40.0	11	27.5
Right Lung	1	2.5	0	0.0	10	25.0	23	57.5
Irrelevant	335	56.7	258	48.5	327	55.4	313	53.1
Avg. Acc	46.4		**69.4**		52.7		52.5	

The results indicate a significant performance disparity between the adaptation of CLIP's textual (reported in the previous section) and visual branch in terms of classification accuracy. For visual branch adaptation, a linear adapter outperformed and achieved an average accuracy of 69.4% as reported in Table 6.5. While for textual branch adaptation, CLIP-A outperformed and achieved an average classification accuracy of 82% on few-shot data (Table 6.2). However, the integration of this pre-trained CLIP-A into SaLIP-A did not yield satisfactory results on the full dataset for lung segmentation, as demonstrated by segmentation DSC reported in Table 6.3.

3) CLIP's Visual and Textual Branch Adaptation

In this setup, both textual and visual branch features are adapted. The CLIP-A (0.2) enhanced the textual branch features (Table 6.2) and the linear adapter improved the visual branch features (see Table 6.5). Based on these results, these

adapters are therefore utilized to adapt CLIP's visual and textual branche features respectively. The result of this adaptation is reported in Table 6.6.

Table 6.6: CLIP visual and textual features adaption in few-shot setting: comparison of different adapters.

Class	Zero shot		Linear		CLIP-A (0.2)	
	Corr	Acc	Corr	Acc	Corr	Acc
Left Lung	32	80	40	17	6	15
Right Lung	1	2.5	0	0	29	72.5
Irrelevant	335	56.7	0	288	243	243
Avg. Acc	46.4		6.0		41.5	

However, adapting both the visual and textual branches also did not enhance classification performance. Compared to adapting only the visual or textual branch (Table 6.5 and 6.2) individually, this approach led to an even further decline in classification performance as reported in Table 6.6.

Analysis

Extensive experimentation with CLIP adaptation is conducted in both zero-shot (Section 6.5.1) and few-shot settings (Section 6.5.2). For-few shot setup, adaptations of the textual branch, the visual branch, or both are evaluated. The results highlighted the limitations of CLIP's recognition capabilities for fine-grained medical tasks, particularly in accurately identifying lung structures.

Though CLIP's textual branch features adaptation using CLIP-A achieved an average classification accuracy of 0.82 on few-shot data, a similar pattern is not observed when CLIP-A is integrated into SaLIP-A to evaluate segmentation in the entire dataset as reported in Table 6.3.

The results demonstrated that CLIP is highly sensitive to textual information, posing challenges in achieving consistent and reliable outcomes, particularly for complex fine-grained medical tasks. This sensitivity was evident in its limited ability to accurately recognize lung structures based on spatial location, as evaluated in the preceding sections (Section 6.5.1, 6.5.2). Furthermore, these findings highlight

a significant limitation: CLIP's embeddings lack sufficient medical domain knowledge. As a result, its performance on the evaluated fine-grained medical tasks i.e., recognizing lungs based on spatial location, was suboptimal.

Additionally, leveraging LLMs to create accurate textual prompts for anatomical structures related to spatial location proved to be both challenging and ineffective. LLMs lack the intrinsic knowledge required to generate accurate textual prompts for such tasks. For example, generating prompts to describe lungs based on their spatial location and distinctive visual features was particularly difficult, as evidenced by the results presented in Sections 5.5.5, 6.5.2. The process of this prompt generation is outlined in Appendix A.1.2.

6.5.3 Evaluation of the Proposed SaLIP-V Framework on Fine-Grained Medical Tasks

The strategies outlined in Section 6.5.2, rely on adapting either CLIP's textual or visual branch while keeping the other branch frozen but still utilized for extracting the respective features. These configurations did not prove effective and resulted in poor performance in recognition of lungs based on their spatial location. Additionally, crafting precise textual prompts for fine-grained medical regions proved complex and less effective.

To address these challenges and improve CLIP's recognition performance for fine-grained medical tasks, we exclusively utilized the CLIP visual encoder (CLIP-V) for extracting rich visual features for fine-grained regions in images highlighted by visual prompts. The extracted CLIP-V features are adapted using a simple linear classifier as shown in Figure 6.3 (b).

Task 1: Spatial Localization and Segmentation of Lungs

To evaluate the effectiveness of CLIP-V in fine-grained tasks, we compared its performance with DINOv2. DINOv2 is a vision-only foundation model that learns robust visual features without any form of supervision, enabling it to excel in vari-

ous visual recognition tasks [248]. DINOv2 has proven to be effective in contrast to CLIP [141, 282]. CLIP-V and DINOv2 are evaluated and compared as follows:

- **SaLIP-V:** This approach begins by autonomously generating masks for different image sub-regions using SAM_{EM}, which are used to create visual prompts to highlight different image sub-regions. The resulting set of images with visual prompts is then processed by the CLIP visual branch, as illustrated in Figure 6.3 (b).

- **SAM-DINO:** This method also begins by generating masks for image sub-regions using SAM_{EM}. Subsequently, the images, overlaid with visual prompts derived from the SAM_{EM} generated regions, are processed by DINOv2.

The extracted features from DINOv2 and CLIP-V are individually adapted using a linear classifier as shown in Figure 6.3 (b). The comparative analysis between these two setups for lung classification and segmentation is reported in Table 6.7. The column "Class Acc" shows the classification accuracy of the linear classifier on the few-shot dataset's test split. The column "Seg (DSC)" shows the segmentation dice score, when the classifier individually trained on CLIP-V and DINOv2 features, is integrated into the SaLIP-V framework (Figure 6.4).

Table 6.7: Few-Shot Adaptation Comparisons: CLIP-V vs. DINOv2.

Classification Approach	Class Acc (%)	Seg (DSC)
SaLIP-A	0.820	0.650
SAM-DINO	0.881	0.747
SaLIP (Ours) [121]	0.906	0.839
SaLIP-V (Ours)	**0.946**	**0.874**

Contrary to our expectations, CLIP-V outperformed DINOv2, despite not leveraging CLIP's textual branch. This indicates that CLIP-V effectively extracts detailed visual features critical for analyzing complex medical images. It also suggests that CLIP's textual embedding space may not be well-suited for fine-grained

medical imaging tasks, given the complexity and specificity of medical terminology, as demonstrated in the results (Section 6.5.2). Additionally, in the context of fine-grained medical images, crafting precise textual prompts to accurately describe anatomical structures and pathologies presents a significant challenge.

By adapting only CLIP-V features, our proposed approach significantly improved the performance i.e., the dice score significantly improved from 0.650 (using CLIP's textual features- SaLIP-A) to 0.874, as reported in Table 6.7. In comparison, DINOv2 achieved a dice score of 0.747. Notably, our proposed method, SaLIP-V, not only outperformed DINOv2 but also surpassed the baseline SaLIP, achieving a notable improvement in the dice score from 0.839 to 0.874.

The extensive experimental evaluation in Sections 6.5.1, 6.5.2, 6.5.3 and results reported in Table 6.7, demonstrate that our proposed SaLIP-V approach excels in spatial localization tasks, achieving impressive performance without the need for specialized domain expertise in prompt engineering or annotated data.

Task 2: Fine-Grained Polyp Classification and Segmentation

Building on the promising results of our proposed approach for the first fine-grained task outlined in Section 6.5.3, we further evaluated it on a more challenging task: recognition/localization and segmentation of tumors that have varying spatial locations, lack consistent anatomical morphology and are often subtle and small in size. Specifically, for this task, the proposed approach is evaluated on polyp recognition and segmentation in colonoscopic images from the Kvasir-seg dataset (Section 6.4.1).

To evaluate the effectiveness of our proposed approach for polyp classification, CLIP-V is used in two different setups:

- **Spatial Average of Patch Embeddings:** transformer-based models preserve the spatial information/structure of the input by using patch embeddings. Each patch embedding corresponds to a specific location in the original image. By using patch embeddings, the model focuses on the local features of each patch rather than a global representation of the entire image.

- **CLS Token:** is a special class (CLS) token in transformer-based models like vision transformers (ViT) [258]. It aggregates information from the entire input image for tasks like classification. The CLS token captures the global context.

After extracting features from CLIP-V within the above mentioned embedding spaces, the classification of fine-grained image sub-regions is performed by adapting the extracted features using the linear classifier (Section 6.3.1 illustrated in Figure 6.3). Table 6.8 presents the linear classifier's performance, the column "spatial avg" refers to the results obtained from the spatial average of CLIP-V's patch embeddings, while "CLS token" presents the classification results achieved using the CLIP-V's CLS token embeddings. The classification performance of the linear classifier did not vary significantly with either of the CLIP-V's embeddings. With spatial average and CLS token embeddings, the classifier achieved an average classification accuracy of 0.828 and 0.79, respectively, as reported in Table 6.8 ("Avg. Acc" row).

Although the classifier achieved an average classification accuracy of 0.79 using CLS token embeddings for the first phase of our proposed approach (i.e., few-shot classification of fine-grained regions, see Section 6.3.1). However, when this classifier was integrated into the second phase of our method for fine-grained region segmentation (Section 6.3.2), the dice score significantly declined to only 0.50, as reported in Table 6.8.

Table 6.8: Evaluation of SAM-CLIP-V: A comparison of classification performance using the spatial average of patch embeddings vs CLS token embeddings.

Class	No of. Images	Spatial Avg		CLS Token	
		Corr	Acc	Corr	Acc
Polyp	109	101	0.93	96	0.88
Irrelevant	953	771	0.84	744	0.78
Avg. Acc	-	**0.82**		0.79	
SaLIP-V (DSC)	-	0.49		**0.50**	

The decline in the DSC can be attributed to the linear classifier being trained

in a few-shot setting. For polyps, 40 shots were used to train the linear classifier for few-shot image sub-region classification. These 40 shots were processed through SAM_{EM}, autonomously generating 559 masks, of which 50 were labeled as polyps and 509 as irrelevant regions (see Section 6.3.1). From this pool, only 40 images (20 representing polyps and 20 irrelevant regions) were selected for few-shot training, which may have led to overfitting.

As a result, both approaches achieved high classification accuracies of 0.828 and 0.79 in the first phase, where the adapted classifier was evaluated on the test split of the few-shot dataset (Section 6.3.1). However, when the adapted linear classifier was incorporated into the second phase (Section 6.3.2), overfitting caused poor generalization, as demonstrated by the results in Table 6.8.

To address this issue, the entire set of image sub-region proposals generated by SAM_{EM} was utilized, rather than selecting a few-shot subset from the general pool (Section 5.3.1). This approach still remains few-shot in nature because only a limited number of samples (i.e., 40 shots) are utilized to generate masks for different regions in the images using SAM_{EM} (Figure 6.3). However, instead of selecting a subset of the generated image sub-regions for training the linear classifier, the entire set was utilized.

For the original 40 shots used for polyp classification, SAM_{EM} generated 559 masks, of which 50 were labeled as polyps and 509 as irrelevant regions using our proposed labeling approach, which is discussed in detail in Section 6.3.1. Given the inherent class imbalance in the data, two sampling strategies were evaluated to mitigate bias toward the majority class.

- **Weighted Random Sampler:** Assigns sampling probabilities based on class weights. The probabilities are calculated as the inverse of the frequency of each class, which increases the likelihood of selecting minority classes within each batch during training.
- **Balanced Sampler:** ensures that each class is equally represented in batches during training, regardless of the original class distribution in the dataset.

The results of leveraging the full pool of SAM_{EM} generated image sub-region proposals for training the linear classifier are reported in Table 6.9. The column "without sampler" presents the results achieved without using any sampler. In this case, the linear classifier achieves a high overall average classification accuracy of 0.916 on the test split of the few-shot dataset. However, an evaluation of class-wise accuracy indicates that without the sampler, the linear classifier exhibits a significant bias towards the majority class. Out of a total of 1,062 masks generated by SAM_{EM} for the test split of the few-shot data, 953 samples belong to the majority class i.e., irrelevant regions. Thus the linear classifier achieves a high accuracy of 0.964 for the majority class. In contrast, the model's performance on the minority class–"polyp", which is the primary focus of this analysis, is considerably lower, with an average classification accuracy of only 0.495. These results indicate that class imbalance created a bias toward the majority class and eventually the classifier fails to generalize well on the minority class.

Table 6.9: Performance Improvement Analysis with Samplers

Class	**No. of images**	**Accuracy**		
		Without Sampler	**Weighted Sampler**	**Balanced Sampler**
Polyp	109	0.495	0.688	0.706
Irrelevant	953	0.964	0.927	0.926
Avg. Acc	-	0.916	0.903	0.904
SaLIP-V (DSC)	-	-	0.484	0.481

In contrast, employing samplers improved classification accuracy for the minority class, from 0.495 to 0.688 with the weighted sampler and to 0.706 with the balanced sampler, as shown in Table 6.9. However, when the adapted classifier, was integrated into the second phase of our proposed framework (Section 6.3.2) to segment the recognized polyp regions in the images, the DSC remained low at 0.484 (Table 6.9 (SaLIP-V))

To investigate the reasons for this persistent poor performance, a qualitative analysis was conducted, as detailed in the following section.

Qualitative Analysis

The proposed SaLIP-V framework consists of two phases (Section 6.3), evaluation of both phases is conducted to investigate the reasons for poor performance for polyps classification and segmentation.

For the first phase of our method (i.e., the few-shot classification setup – Section 6.3.1), the proposed labeling approach, along with the classification performance of the linear classifier is evaluated. For the second phase of our method (i.e., segmentation of fine-grained region – Section 6.3.2), the performance of both SAM_{EM} for generating masks of various fine-grained image sub-region as well as the adapted linear classifier in recognizing and classifying different classes is evaluated.

Diversity and Variability of Fine-Grained Regions

One of the key factors that adversely affected the performance of our proposed SaLIP-V framework in polyp localization is the wide diversity and variability in polyp morphology and spatial location. The influence of this diversity on SaLIP-V's performance is demonstrated through a few examples shown in Figure 6.5. Across the reported images, significant variations in polyp morphology, spatial location, and shape are clearly visible, as shown in the ground truth displayed in the second column. The red bounding box in the fourth column of Figure 6.5 (bbox coordinates) shows the region of the image classified as ROI by our method, from the region proposal generated by SAM_{EM} (Section 6.3.2) as shown in Figure 6.5. The last column shows the segmentation output from our method.

In the first row of Figure 6.5, the highlighted region (red bounding box) is classified as a polyp. This prediction appears accurate when evaluated independently, as the predicted region resembles a pathological structure. However, it is an incorrect prediction. Similarly, in the second row, the detected ROI appears accurate due to its strong resemblance to the polyp structure in the first-row image. However, it is incorrect when compared to the actual ground truth.

This variability in polyp's morphology and spatial location poses significant chal-

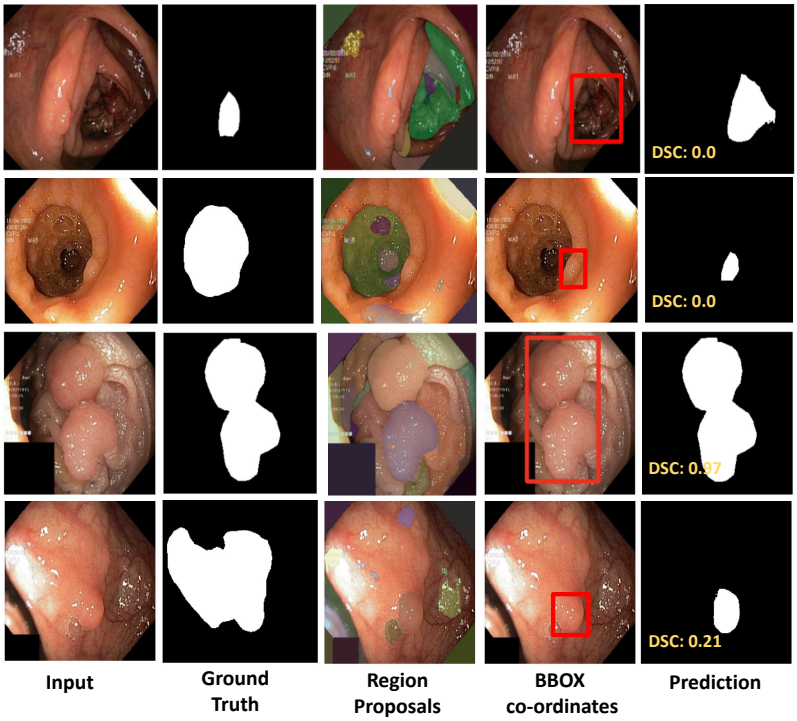


Figure 6.5: Impact of polyp variability on SaLIP-V performance.

lenges in accurately classifying ROI which impacts the generalization of our proposed method. The miss-classification eventually impacts the segmentation result of fine-grained regions corresponding to polyps as evident in Figure 6.5. Therefore, the variability inherent in fine-grained medical imaging analysis tasks, such as the challenging tasks of polyp localization and segmentation, presents significant challenges and needs further improvement.

To address these challenges, a potential direction for future work is to implement data augmentations and generate synthetic data from the few-shot dataset to increase the volume of available data. By leveraging these augmentations, both diversity and dataset size could be enhanced, leading to improved classification accuracy and ultimately enhancing the segmentation performance of the proposed framework.

Region Proposals Generated by SAM

In some cases, polyps exhibit a camouflaged morphological appearance with indistinct boundaries, as demonstrated in the examples presented in Figure 6.6. Conse-

quently, SAM often fails to generate accurate masks in these cases. The first column presents the input image, the second column displays the ground truth, and the third column shows the region proposals generated by SAM. In each of the three cases, a comparison with the ground truth reveals that SAM failed to generate masks for the polyp region, which is camouflaged in these reported cases.

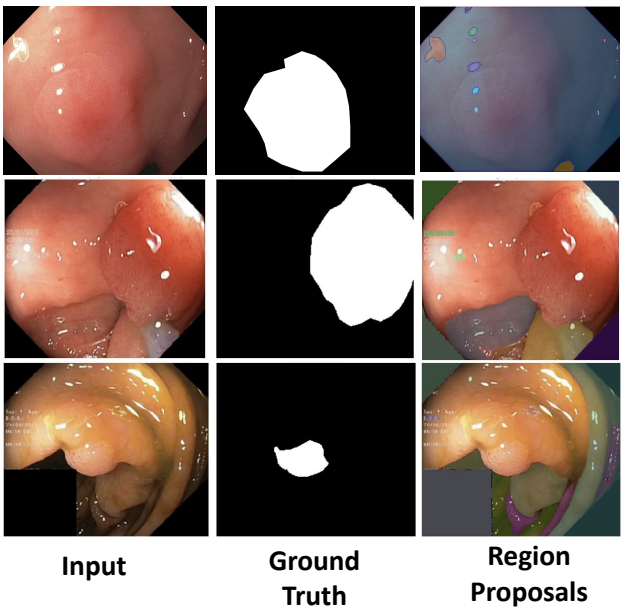


Figure 6.6: SAM Limitation: No masks are generated for camouflaged polyp regions. Region proposals: Masks generated by SAM.

Unlike the first evaluated fine-grained task, which focuses on lung spatial localization and segmentation involving relatively consistent anatomical structures (Section 6.5.3), the second task i.e., polyp detection and segmentation—presents considerably greater challenges.

This analysis provided valuable insights into the limitations of foundation models in fine-grained medical tasks, where the region of interest may have varying morphological structures, inconsistent spatial locations, and camouflaged areas that blend with the surrounding features and lack clear boundaries, such as polyps. This variability complicates the segmentation process, as demonstrated in Figure 6.5 and 6.6.

Few-shot Dataset Creation

The first phase of our proposed approach is few-shot classification of fine-grained regions (Section 6.3.1). The proposed labeling approach for the few-shot dataset impacts the second phase of our method i.e., segmentation of the region of interest picked by the adapted classier from the first phase (Section 6.3.1 (a)).

In the first evaluated fine-grained task involving localization and segmentation of lungs (Section 6.5.3), the labeling process was straightforward, as the lungs constitute the major region. Therefore, selecting the region from the SAM_{EM} generated image sub-regions that had the highest dice coefficient compared to the ground truth proved effective for label creation.

In contrast for polyps recognition/classification, a single image may contain multiple polyps, as illustrated in Figure 6.7 (first row). We initially used the $argmax$ function to select the mask for the region of interest with the highest dice score from the pool of SAM-generated region proposals. However, this approach resulted in the selection of only a single polyp and resulted in reduced dice scores for the instances where multiple polyps are present (Figure 6.7– first row).

Additionally, SAM often generates multiple masks corresponding to different sub-parts of the same polyp region, rather than producing a single cohesive mask that encompasses the entire polyp region. In such scenarios, using $argmax$ function selects only a sub-part of the polyp, leading to a reduced dice score, as shown in Figure 6.7 (second row, third column).

To address these issues, we revised our labeling process for creating the few-shot dataset and used a threshold-based labeling approach (Section 6.3.1). Each sub-region generated by SAM_{EM} with a dice coefficient greater than 0.5, when compared to the ground truth, is labeled as a "polyp". The last column in Figure 6.7 illustrates that our revised labeling approach effectively recognizes and segments multiple polyps in the image (Figure 6.7– first row, fifth column). It also successfully segments the entire polyp region, which was initially predicted as individual sub-parts of the image by SAM_{EM} as shown in Figure 6.7 (second row, fifth column).

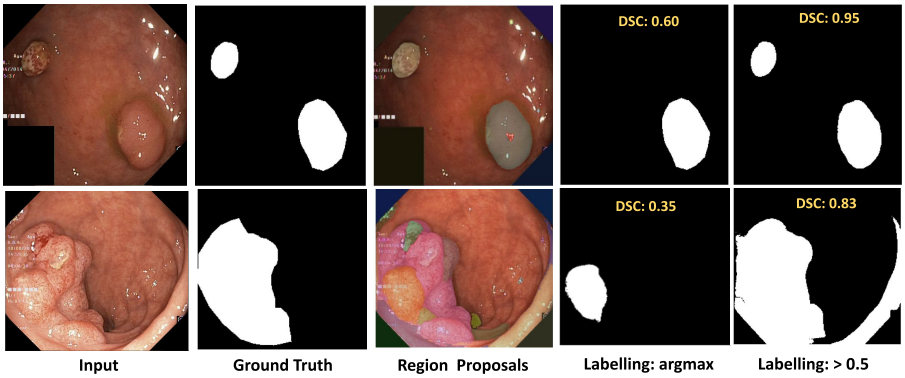


Figure 6.7: Few-shot dataset creation: comparison of labeling approaches.

The revised labeling approach highlighted another limitation of SAM: it often segments only partial regions of polyps, failing to generate masks for the remaining sub-regions that constitute the polyps, as illustrated in Figure 6.8. This partial segmentation results in only a portion of the polyp being labeled as "polyp" during the label creation process (Section 6.3.1). Consequently, this incorrect labeling adversely affects the performance of the overall SaLIP-V framework, as the model does not receive sufficient information to accurately classify different image regions and retrieve the correct ROI.

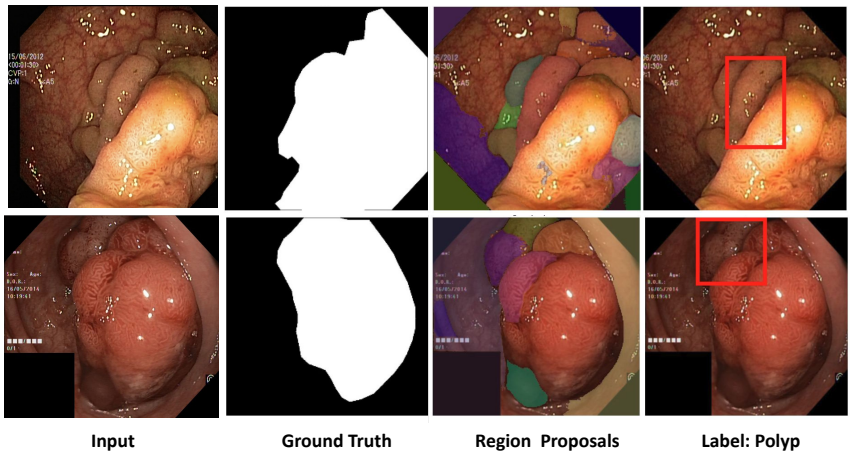


Figure 6.8: SAM Limitation: Partial segmentation of the region of interest.

This analysis provided insights into several limitations of the SAM foundation model in fine-grained medical imaging analysis tasks. Specifically, it struggles to segment larger, continuous anatomical structures as a single cohesive region (Figure 6.8). Instead, it may generate multiple masks for sub-parts of the same structure

(Figure 6.8, first row) or produce partial segmentation masks of anatomical regions (Figure 6.8, second row). Such inconsistencies create challenges for our labeling process, making it challenging for the linear classifier to accurately categorize the SAM_{EM} generated region proposals. As a result, these limitations pose significant challenges to the application of foundation models in fine-grained medical tasks that require precise recognition and segmentation of complex pathologies, such as polyps.

6.6 Summary

This chapter presents our work on the adapting foundation model to complex fine-grained medical imaging tasks. We introduced a new framework called SaLIP-V which facilitates light-weight adaptation of foundation models in a few-shot setting. In contrast to existing approaches that primarily leverage foundation models for coarse/global image-level tasks, SaLIP-V specifically targets fine-grained medical imaging analysis. These tasks involve the analysis of anatomical structures or pathologies often have varying spatial locations and complex morphological features and require fine-grained visual perception.

SaLIP-V is the first approach to employ visual prompting specifically for fine-grained medical imaging tasks. Instead of relying on annotated data for visual prompt engineering, it automatically generates visual prompts in a zero-shot manner for different regions in medical images without any labeled data and domain expertise (Section 6.3.2).

SaLIP-V is evaluated on two different fine-grained medical imaging tasks. First, it is evaluated on the recognition and segmentation of anatomical structures based on spatial locations and distinct features (Section 6.5.3 (Task 1)). This evaluation specifically focuses on localization/recognition of lung in chest X-ray followed by segmentation. SaLIP-V achieved a classification accuracy of 0.946 and a dice score of 0.874 for segmentation (Section 6.5.3), achieving a 4% improvement over the SaLIP framework introduced in Chapter 5.

The second fine-grained task involves the localization, recognition, and segmen-

tation of polyps. For this task, SaLIP-V achieved a classification accuracy of 0.79 and a dice score of 0.50 for segmentation (Table 6.8). The performance of SaLIP-V in this task is notably lower as compared to the first tasks. It is primarily because polyps are small, subtle, have varying morphological structures, inconsistent spatial locations (Figure 6.5), and are often camouflaged (Figure 6.6) which presents significant challenges for accurate identification (Section 6.5.3 (Task 2)). These are our preliminary results, and we plan to further enhance SaLIP-V for this task in the future.

A comprehensive analysis of SaLIP-V is conducted to evaluate the reasons behind its limited performance on the second task (Section 6.5.3). This analysis provided valuable insights into several limitations of foundation models for complex fine-grained medical tasks (see Sections 6.5.2, and 6.5.3). These limitations suggest potential research directions. A few of these include incorporating augmentation techniques into the few-shot classification setup and applying generative methods to synthesize additional data samples from the limited few-shot data. Such approaches can provide SaLIP-V with diverse training examples, thereby enhancing its ability to learn and adapt to the diverse nature of polyps.

6.6.1 Insights

SaLIP-V's experimental evaluation on challenging fine-grained tasks, such as the recognition of pathological structures that possess diverse spatial locations and varying morphological structures, has provided the following valuable insights:

1. While segmentation foundation models like SAM have shown exceptional zero-shot transferability in segmenting natural images and large organs in medical imaging (Chapter 5), they often struggle with fine-grained segmentation. Specifically, SAM fails to segment complex pathological structures, such as tumors, which are often subtle and very small (Figure 6.5). Additionally, SAM encounters challenges when tumors are camouflaged or lack distinct boundaries, a common issue in complex diseases like tumors, as illustrated in Fig-

ure 6.6.

2. CLIP has demonstrated impressive performance across a wide range of recognition and classification tasks. Additionally, visual prompting further enhances its zero-shot transferability in the natural imaging domain (Section 6.2). However, CLIP's embedding space lacks the domain-specific semantic knowledge required for fine-grained medical tasks. It makes CLIP less suited for such tasks. The complex and domain-specific terminology of the medical domain further poses challenges for CLIP's generalized embeddings. Thus these challenges limit its effectiveness for tasks requiring precise anatomical localization (Section 6.5.2).

3. Large language models (LLMs) like GPT-3.5 [52] have been widely used for textual prompt engineering in image-level tasks [120]. It has also shown effectiveness in medical applications (Chapter 5), for prompt engineering to describe organs with promising results (Section 5.5). However, prompt engineering for fine-grained medical imaging tasks is more complex, and LLMs often perform poorly in this context. This limitation stems from the inherent complexity and domain-specific challenges of medical imaging. While LLMs are designed for general tasks, they are not well-equipped to address the specialized demands of fine-grained medical image analysis, such as creating prompts for anatomical structures based on spatial locations and distinguishing features (Appendix A.1.2), or for complex pathologies that are subtle, vary in spatial location, and have intricate morphological features. As a result, LLMs lead to poor performance in these fine-grained medical imaging tasks (Section 6.5.2, Appendix A.2.1).

Chapter 7

Conclusion

Neural networks have shown exceptional performance across medical imaging tasks, becoming state-of-the-art tools in clinical workflows. However, this success is based on certain requirements: the training and testing data are typically assumed to be identically and independently distributed. Domain shift, however, can severely impact a model's generalizability, presenting significant challenges in maintaining robust performance across diverse clinical settings. Adding to this challenge, neural networks require large amounts of annotated data to achieve top results. However, acquiring such datasets for medical applications is both costly and resource-intensive, as labeling medical data typically requires expert knowledge. Moreover, privacy concerns and restrictions on data sharing between institutions further exacerbate the issue.

The research presented in this thesis addresses these challenges: Chapters 3 and 4, address model generalization issues caused by domain shifts across diverse medical imaging domains, and propose alternatives to reduce reliance on strong supervision during training. Chapter 5 proposes an efficient, supervision-free test-time adaptation framework for adapting foundation models to diverse medical imaging tasks. Chapter 6 focuses on developing a pipeline for adapting foundation models to perform fine-grained medical tasks.

In particular, Chapter 3 investigates the impact of domain shift on model generalization, emphasizing the limitations of supervised learning approaches in adapting

to diverse medical imaging domains. The evaluation is conducted on raw medical data from diverse domains, sourced from multiple hospitals, reflecting variations in acquisition protocols. These differences pose significant challenges within a supervised training framework. To address these challenges, an ensemble deep learning approach combined with test-time data augmentation is proposed to improve model generalization across domains, with the limited available data.

In Chapter 4 we eased the supervision restriction and assumed the unavailability of labeled samples from the target domain, i.e., unsupervised domain adaptation. In this context, the proposed approach tackles two key challenges: creating dedicated models for each downstream task and the computational overhead inherent in supervised domain adaptation methods. Specifically, a parameter-efficient, self-supervised domain adaptation strategy is proposed to adapt convolutional neural networks to multiple target domains. This chapter demonstrates that, when properly regularized, parameter-efficient adaptation in an unsupervised manner can achieve performance comparable to that of supervised domain adaptation.

Chapter 5 explores the test-time adaptation of foundation models for a wide range of medical imaging tasks. This research addresses several key challenges: the scarcity of labeled data, the need for domain expertise for prompt engineering, and the significant computational cost of adapting foundation models to downstream tasks. A novel framework, SaLIP is proposed to facilitate zero-shot adaptation of foundation models for medical organ segmentation, without requiring labeled data or specialized domain expertise for prompt engineering. Furthermore, we provide valuable insights into how foundation models can generalize to medical imaging tasks in scenarios where labeled data and domain-specific knowledge are limited or not readily available.

Chapter 6, investigates the adaptation of foundation models for complex, fine-grained medical tasks within a few-shot learning framework. It leverages visual prompting techniques to guide foundation models in accurately identifying and analyzing specific pathological features. In particular, a framework called SaLIP-V

is introduced as a lightweight alternative designed to adapt foundation models for fine-grained medical tasks.

In this chapter, Section 7.1 addresses the hypothesis described in Chapter 1, and how the research presented in Chapters 3 to 6 addresses the hypothesis through the research questions also introduced in Chapter 1. Section 7.2 summarizes the research contributions of this thesis. Section 7.3 elaborates on the suggestions for future research introduced in the main chapters of the thesis. Finally, Section 7.4 provides the closing remarks for this thesis.

7.1 Hypothesis and Research Questions

The hypotheses introduced in Chapter 1 are discussed in this section with respect to the research presented in the corresponding chapters. Each of the research questions associated with each hypothesis is addressed to provide a more concise notion of the contribution of this thesis.

Hypothesis 1

In medical imaging scenarios with multiple target domains, low-rank adapters can facilitate parameter-efficient adaptation of convolutional neural networks. It provides an alternative to training separate dedicated networks for each domain and achieves performance similar to full model adaptation while reducing computational overhead.

- **Research Question 1:** ***What are the key challenges and limitations of supervised adaptation approaches when applied to diverse medical imaging datasets? Specifically, how do domain shifts and data scarcity affect the generalization of neural networks for medical imaging tasks?***

  The experiments described in Chapter 3 investigate the effect of domain shift on the generalization of neural networks using data collected from different acquisi-

tion devices and hospitals. To address the challenges of domain shift and limited data availability, our proposed approach utilizes an ensemble of neural networks combined with test-time augmentation. This method is evaluated on the STOIC 2021 dataset [161], which consists of raw CT scan images collected from multiple hospitals over a period of time. While the proposed approach performed well, securing fourth place in the STOIC challenge, the findings offered valuable insights. Specifically, the results emphasize that supervised adaptation approaches struggle to generalize effectively due to domain shifts and the scarcity of labeled data. Furthermore, the experiments highlight that task-specific supervision alone is not sufficient for tackling the diverse and complex nature of medical imaging tasks.

- **Research Question 2:** ***How could the parameter-efficient adaptation approach be enforced in the unsupervised adaptation of convolutional neural networks? Could convolutional neural networks benefit from the features learned through self-supervised training when using parameter-efficient adaptation?***

  The experiments outlined in Chapter 4 demonstrate that convolutional neural networks can be effectively adapted to multi-target domains in a parameter-efficient manner within a self-supervised framework. Our proposed convolutional low-rank adaptation approach offers a parameter-efficient alternative to traditional supervised adaptation methods. By adapting significantly fewer parameters, our method provides several advantages: enhanced model generalization across multi-target domains, reduced dependence on strong supervision, mitigation of over-fitting risks associated with adapting the entire model with limited data, and alleviation of the computational constraints tied to creating separate dedicated fine-tuned models for each target domain through supervised training. In particular, we conclude that parameter-efficient adaptation in a self-supervised setup is an effective alternative to supervised domain adaptation, particularly for multi-target domain scenarios. Notably, the success of parameter-efficient adaptation

techniques is influenced by task-specific characteristics, and regularization is essential to achieve optimal performance.

Hypothesis 2

In the absence of annotated data or domain expertise, the test-time adaptation of foundation models (FMs) can enable efficient adaptation to diverse medical image tasks. FM-extracted features can be adapted for fine-grained analysis without requiring large datasets.

- **Research Question 3:** ***Can test-time adaptation of foundation models provide a more robust alternative to supervised or semi-supervised domain adaptation approaches? Can foundation models be effectively adapted to diverse medical imaging tasks without relying on annotated data, additional training, or specialized domain expertise?***

  Experiments described in Chapter 5, focus on exploring the inherent limitations and challenges associated with adapting natural foundation models to the medical imaging domain. Our proposed test-time adaptation framework is specifically designed to address the constraints of supervised/semi-supervised adaptation methods and to overcome the inherent limitations of directly applying foundation models to medical tasks. Our proposed test-time adaptation framework is evaluated through zero-shot organ segmentation across various medical imaging modalities, demonstrating the robustness and effectiveness of our method. In conclusion, our framework facilitates the test-time adaptation of foundation models for medical organ segmentation, eliminating the need for additional supervised training, task-specific fine-tuning, annotated data, or specialized domain expertise for prompt engineering.

- **Research Question 4:** ***Can foundation models be effectively adapted to challenging fine-grained medical imaging tasks?***

Experiments conducted in Chapter 6 explore the adaptation of natural foundation models to fine-grained medical imaging tasks. A simple yet effective approach is introduced to utilize the embedding space of foundation models, enabling the extraction of features that can be efficiently adapted with a lightweight linear classifier in a few-shot learning scenario. The results of these experiments demonstrate that the foundation models can be effectively adapted for fine-grained tasks that involve relatively stable, well-defined features, such as specific anatomical locations. However, foundation models are less effective for more complex tasks, particularly those involving pathological structures with varying spatial locations and morphology, or have camouflaged features. Furthermore, as experimentally demonstrated in Chapter 6, large language models are not well-suited for fine-grained medical tasks, such as textual prompt engineering for complex pathological structures. In these contexts, LLMs tend to hallucinate which results in poor generalization.

7.2 Research Contributions and Proposed Solutions

The contributions of this research are summarized in the following list:

- Chapter 3: An Ensemble Approach with Test-Time Augmentations

    1. The challenges of supervised learning, particularly regarding domain shift and model generalizability, are assessed using real-world raw chest CT volumes acquired from diverse hospitals and medical domains.
    2. The proposed ensemble approach combined with test-time augmentation offers a simple yet effective solution and secured fourth place in the STOIC COVID-19 AI Challenge.

- Chapter 4: Parameter efficient adaptation of convolution neural networks

1. A new unsupervised parameter-efficient adaptation framework is proposed for multi-target domain adaptation. Specifically, a new method called Convolutional Low-Rank Adapter (ConvLoRA) is proposed to adapt the convolutional neural network to multi-target medical imaging domains. It offers an effective adaptation alternative to traditional supervised domain adaptation techniques.
2. Our ConvLoRA method is implemented in a self-supervised setting, which facilitates the adaptation to target domains without relying on labeled data.
3. ConvLoRA is a generic parameter-efficient adaptation method and can be easily integrated into deep neural networks having convolutional layers, thereby facilitating effective domain adaptation.
4. Experimental results demonstrate that our method is complementary to existing approaches. Combining ConvLoRA with other domain adaptation techniques further enhances model generalizability to diverse target domains.

- Chapter 5: Test Time Adaptation of Foundation Models

1. A novel test-time adaptation framework called SaLIP is proposed to adapt natural foundation models for the medical imaging domain. SaLIP utilizes a cascade of foundation models to enable zero-shot medical organ segmentation, effectively bridging the gap between general-purpose models and specialized medical tasks.
2. SaLIP effectively addresses the key challenges associated with adapting foundation models to medical imaging tasks. It does not need supervised training, task-specific fine-tuning, and is independent of specialized domain expertise for prompt engineering, thus streamlining the adaptation process.
3. SaLIP is evaluated across a range of medical imaging modalities, exper-

imentally demonstrating its robustness and strong generalization across diverse domains.

4. Through comprehensive analysis, we provide valuable insights into the adaptation of foundation models without annotated data, highlighting their relevance and applicability in real-world medical scenarios.

- Chapter 6: Few-shot Adaptation of Foundation Models

1. The adaptation of foundation models to complex, fine-grained medical imaging tasks is evaluated through comprehensive experimental evaluation.
2. A few-shot approach is proposed as a lightweight adaptation alternative, designed to effectively leverage foundation models for extracting robust and adaptable features for fine-grained medical imaging tasks.
3. Experimental results provided valuable insights into the inherent limitations of current foundation models in performing fine-grained medical imaging tasks. These limitations include challenges in the recognition and localization of small, subtle pathological structures– such as tumors – that have varying spatial locations, morphologies that change over time, and, in some cases, are camouflaged. These challenges impede the effective adaptation of foundation models to complex, fine-grained tasks.

We also highlight potential research directions for each chapter, aiming to advance the field toward more realistic scenarios and enhance the applicability of neural networks to real-world challenges. These insights are further elaborated in Section 7.3.

7.3 Recommendations and Future Work

In this section, we outline potential future research directions based on the limitations identified in each chapter. These limitations with the possible solutions and

their respective outcomes are discussed in detail throughout the chapters. Here, we provide a concise summary of these limitations and propose the potential research direction for future exploration.

- Scalability and Robustness of Parameter Efficient Adaptation
  - A limitation of the proposed approach is that, while it demonstrated robustness across five different target domains for brain segmentation using the CC359 dataset [11], it was less effective for the segmentation of complex cardiac structures in M&M dataset [12]. Although the method improved model generalization for brain skull segmentation, its performance on cardiac structures was relatively less effective.
  - Future research could focus on architectural modifications of our proposed method to enhance its robustness and scalability across a broader range of diverse datasets.

- Test-Time Adaptation of Foundation Models
  - While our proposed framework demonstrated strong and consistent performance across a variety of medical imaging modalities, there were instances where the Segment Anything Model (SAM) [13] struggled to generalize effectively and failed to accurately segment the region of interest (Section 5.5.5).
  - At the time of our research on test-time adaptation, the original SAM had just been released [13]. Since then, its successor, SAM-2, has been introduced and is reported to outperform the original model [288]. Future work could explore whether SAM-2 enhances the performance of our proposed framework.
  - Since the proposed test-time framework relies on a cascade of foundation models, a failure in any one component can lead to error propagation throughout the pipeline. A promising avenue for future research

involves integrating uncertainty estimation mechanisms into the proposed pipeline. Such mechanisms could help detect failure points, prevent error propagation, and enhance robustness by providing confidence estimates for model predictions.

- Few-shot adaptation of foundation models for fine-grained medical imaging tasks
  - The proposed approach is our initial attempt at adapting foundation models to fine-grained medical imaging tasks. While it showed promising results on one of the evaluated tasks, challenges persist in detecting pathological structures with varying spatial locations and morphologies.
  - Further improvements are necessary to optimize foundation model adaptation while preserving the proposed approach's few-shot nature. Potential solutions may include leveraging additional synthetic data, enabling the proposed adaptation approach to learn more universal robust features required for fine-grained tasks.
  - Data augmentation techniques can enhance the generalization of the proposed framework. While current augmentation methods are primarily image-specific, we see significant value in developing domain-agnostic approaches that can be generalized across various data types.
  - Instead of developing specialized foundation models for individual tasks, a more impactful research direction is to optimize existing models for a broader range of domains. This strategy would improve the foundation model's robustness and generalizability across a diverse set of applications.

7.4 Closing Remarks

Recent advancements in computer vision, particularly driven by deep learning techniques, have enabled significant progress in medical image analysis. The research

conducted during the development of this thesis focused on the applicability of deep learning methods in medical imaging, particularly in the context of challenging real-world conditions: the reduced generalizability of models caused by discrepancies between training and testing data distributions due to domain shifts, and the limited availability of annotated data. The motivation for the former stems from the challenge that models are often trained with the assumption that the training and testing datasets follow the same distribution. Moreover, model generalization is often evaluated by creating splits within the same dataset, which may not accurately reflect real-world scenarios. The latter, however, is driven by the lack of sufficient medical data, which is crucial for effectively training deep learning models.

The approaches explored and developed in this thesis improve model robustness and generalization, addressing challenges such as domain shift and the limited availability of labeled medical data. In particular, our parameter-efficient adaptation within a self-supervised learning framework offers significant benefits in scenarios where multiple target domains exist with varying degrees of domain shift. These challenges are especially pronounced in fields like medical imaging, where expert annotations are scarce due to the specialized knowledge required, or in self-driving cars, where large volumes of data are readily available but costly and time-consuming to annotate. Test-time adaptation of foundation models would be a valuable solution in scenarios where source domain data is limited, overfitting is a risk, and computational resources are constrained. For example, in the medical domain, it eliminates the need for task-specific supervision and specialized domain expertise, and in the research itself, where comparing different computationally expensive algorithms often slows down the exploration of the field. Additionally, in smart environments, it can enable models to be trained directly on edge devices. Few-shot adaptation of foundation models would be especially beneficial in fields that require a fine-grained understanding of data enabling models to quickly adapt to new tasks with minimal labeled examples. For example in medical imaging to detect rare diseases or conditions with limited labeled data and in autonomous driving, few-shot learning

allows vehicles to quickly adapt to new or rare road conditions, weather patterns, or geographical environments with limited labeled data.

Concretely, this thesis explores robust domain adaptation methods to improve model generalizability across diverse target domains, particularly in the presence of domain shift, limited data, and a lack of domain expertise. The author finds it particularly interesting that models trained on a source domain often struggle to generalize when evaluated on diverse target domains, resulting in poor performance. However, by leveraging our proposed unsupervised parameter-efficient adaptation approach, which involves adapting a significantly smaller number of parameters, the model can generalize effectively across a wide range of target domains, allowing an effective alternative to supervised training. In particular, the proposed test-time adaptation method addresses the substantial performance degradation typically observed in foundation models when evaluated on medical imaging datasets. Continued research for adapting foundation models to the challenging scenarios outlined earlier will be beneficial in developing robust domain adaptation techniques capable of effectively handling real-world complexities. This exploration has the potential to enhance model generalization and reliability across diverse practical applications.

Appendix A

Appendix Title

A.1 Textual Prompt Engineering for SaLIP using GPT-3.5

A.1.1 Lungs

In this study, we use a two-step prompting strategy to identify relevant characteristics that aid in the detection of crops having the desired organ, following the approach proposed in [120].

Example of the first prompt for the lung dataset:
I am creating a zero-shot lung recognition system in a chest x-ray. Write me 20 attributes to describe the lungs in the chest x-ray.

Creating a zero-shot lung recognition system involves identifying and describing features that can uniquely identify and describe the lungs in a chest X-ray. Here are 20 attributes that could be relevant for such a task:

Lung Size: The overall size of the lungs, noting whether they are expanded, normal, or show signs of constriction or atelectasis. Lung Shape: The general shape of the lungs, including any deviations from the typical butterfly shape. Opacity: The degree of darkness observed in the lung fields, indicating the presence of fluids, masses,

or other abnormalities. Texture: The internal texture of the lung fields, whether it appears smooth, granular, or shows specific patterns such as reticular, nodular, or cystic formations. Symmetry: The comparative symmetry between the right and left lungs, noting any significant differences. Lung Borders: The clarity and integrity of the lung borders, particularly against the heart and diaphragm. Hilum Position and Appearance: The location and appearance of the lung hila, which may be displaced or show signs of enlargement. Pleural Line: The visibility and condition of the pleural line, including any thickening, plaques, or pneumothorax (air in the pleural space). Costophrenic Angles: The sharpness and clarity of the costophrenic angles, noting any blunting which may suggest pleural effusion. Vascular Markings: The visibility and pattern of vascular markings within the lung fields. Air Bronchograms: The presence of air-filled bronchi (tubular structures) against a background of opaque lung, indicating consolidation. Tracheal Position: The position of the trachea, noting if it is centered or displaced. Diaphragm Appearance: The contour and position of the diaphragm, including any signs of elevation or abnormal curvature. Cardiac Silhouette: The size and shape of the heart shadow as it relates to the lungs. Mediastinal Width: The width of the mediastinum, noting any widening that may suggest mediastinal masses or lymphadenopathy. Interstitial Markings: The presence of lines and markings that may indicate interstitial lung disease. Lung Apex: The condition and appearance of the lung apex, looking for masses, fibrosis, or other abnormalities. Presence of Calcifications: The presence and distribution of calcifications within the lung fields, which may suggest old infections or tumors. Pleural Effusion: The presence of fluid in the pleural space, visible as an area of increased opacity. Foreign Bodies or Devices: The presence of any foreign bodies, surgical clips, pacemaker leads, or other medical devices. These attributes are critical for a detailed analysis and recognition of lung health and disease in chest X-rays. Each attribute can help differentiate between normal lung anatomy and various pathologies, aiding in the development of a comprehensive zero-shot lung recognition system.

Example of the second prompt for the lung dataset which uses GPT-3.5's re-

sponse for the first prompt as part of the prompt:

Now write me sentence about each of these attributes with "lung" in the sentences and return the answer as a Python dictionary, with lungs as a key.

The response of the second prompt constitutes the visual descriptive (VDT) information about lungs.

"Lungs:" *["The X-ray shows the shape of the lungs, whether they appear rounded or irregular." "The X-ray indicates the relative size of the lungs within the chest cavity, providing insights into lung health.", "The X-ray helps assess the proper position of the lungs within the chest cavity.", "The X-ray reveals the density of the lung areas, indicating the presence of air or tissue within the lungs.", "The X-ray evaluates how clear the lung fields appear by assessing the transparency of the lungs.", "The X-ray shows the texture of the lungs, whether they appear smooth or have irregularities.", "The X-ray provides insights into the vascular pattern within the lungs, aiding in the assessment of lung circulation.", "The X-ray examines the borders of the lungs, assessing for normal anatomy or potential abnormalities.", "The X-ray checks the presence and integrity of lung fissures, the thin lines separating lung lobes.", "The X-ray evaluates structures at the lung hilum, such as lymph nodes and bronchi, for any abnormalities.", "The X-ray assesses the contour of the diaphragm where it meets the lungs, ensuring proper lung function.", "The X-ray checks the angles formed by the diaphragm and the chest wall, assessing for effusion or pneumothorax in the lungs.", "The X-ray examines the appearance and distribution of bronchi and blood vessels within the lungs.", "The X-ray detects fine linear or reticular opacities within the lung fields, indicating interstitial lung disease.", "The X-ray assesses the space between the lung and the chest wall for any effusion or pneumothorax.", "The X-ray evaluates the deviation of the trachea from its normal position, indicating potential mediastinal shift or mass effect on the lungs.", "The X-ray examines the appearance of ribs surrounding the lungs, checking for integrity and potential fractures.", "The X-ray shows the outline of the heart within the chest, which can influence the appear-*

ance of the lung fields.", "The X-ray assesses the structures within the mediastinum, providing insights into their impact on the adjacent lungs."]

A.1.2 Prompt Engineering based on Spatial Location of Lungs

Example of the first prompt for the lung dataset:

I am creating a zero shot right and left lung recognition system in a chest x-ray. Write me 10 attributes to describe the left and right lungs in the chest x-ray.

Attributes of the Left Lung:

"Position: Situated on the left side of the thoracic cavity." "Size: Slightly smaller than the right lung to accommodate the heart." "Lobes: Consists of two lobes - the superior and inferior lobes, separated by the oblique fissure." "Function: Main function is to exchange oxygen and carbon dioxide with the blood during respiration." "Medical Significance: Annotated images showing the left lung are essential in medical imaging for diagnosing various respiratory conditions." "Blood Circulation: Receives oxygenated blood from the pulmonary veins and sends deoxygenated blood to the pulmonary arteries." "pH Regulation: Plays a role in regulating blood pH through the removal of carbon dioxide." "Development: Undergoes morphological changes during development, including the formation of bronchial buds and bronchopulmonary segments." "Susceptibility: Susceptible to infections such as pneumonia and tuberculosis, which can cause inflammation and compromise respiratory function."

Attributes of the Right Lung:

"Position: Located on the right side of the thoracic cavity." "Size: Slightly larger than the left lung due to the presence of the heart." "Lobes: Divided into three lobes - the superior, middle, and inferior lobes, separated by the horizontal and oblique fissures." "Function: Plays a crucial role in oxygenating the blood and removing carbon dioxide from the body." "Medical Significance: Medical annotations on images showing the right lung aid in diagnosing diseases such as pneumonia,

lung cancer, and tuberculosis." "Blood Circulation: Receives deoxygenated blood from the systemic circulation via the pulmonary arteries and returns oxygenated blood to the heart through the pulmonary veins." "Aspiration Risk: More prone to injury from aspiration due to its more vertical orientation of the main bronchus compared to the left lung." "Diseases: Susceptible to diseases such as emphysema, bronchitis, and pulmonary embolism, which can impair respiratory function." "Development: Undergoes branching morphogenesis during embryonic development to form the bronchial tree and alveoli." "Gas Exchange: Has a larger capacity for gas exchange compared to the left lung due to its larger surface area."

Example of the second prompt for the lung dataset which uses GPT-3.5's response for the first prompt as part of the prompt:

Now write me a sentence about each of these attributes with "left lung" and "right lung" in the sentences of each respective class and return the answer as a Python dictionary, with right and left lung key labels.

The response of the second prompt constitutes the VDT information about left lung and right lung.

"lungs:" { "left lung": "A left lung is situated on the left side of the thoracic cavity.", "It is slightly smaller than the right lung to accommodate the heart.", "The left lung consists of two lobes: the superior and inferior lobes, which are separated by the oblique fissure.", "Its main function is to exchange oxygen and carbon dioxide with the blood during respiration.", "Annotated images showing the left lung are essential in medical imaging for diagnosing various respiratory conditions.", "The left lung receives oxygenated blood from the pulmonary veins and sends deoxygenated blood to the pulmonary arteries.", "In addition to its respiratory function, the left lung also plays a role in regulating blood pH through the removal of carbon dioxide.", "Certain diseases, such as pulmonary edema and pulmonary fibrosis, can affect the left lung's ability to function properly.", "During development, the left

lung undergoes morphological changes, including the formation of bronchial buds and the establishment of bronchopulmonary segments.", "The left lung is susceptible to infections, such as pneumonia and tuberculosis, which can cause inflammation and compromise its respiratory function." ,
"right lung": "A right lung is located on the right side of the thoracic cavity.", "It is slightly larger than the left lung due to the presence of the heart.", "The right lung plays a crucial role in oxygenating the blood and removing carbon dioxide from the body.", "Medical annotations on images showing the right lung aid in diagnosing diseases such as pneumonia, lung cancer, and tuberculosis.", "The right lung is divided into three lobes: the superior, middle, and inferior lobes, separated by the horizontal and oblique fissures.", "It receives deoxygenated blood from the systemic circulation via the pulmonary arteries and returns oxygenated blood to the heart through the pulmonary veins.", "The right lung is more prone to injury from aspiration due to its more vertical orientation of the main bronchus compared to the left lung.", "Diseases affecting the right lung include emphysema, bronchitis, and pulmonary embolism, which can impair its respiratory function.", "During embryonic development, the right lung undergoes branching morphogenesis to form the bronchial tree and alveoli.", "The right lung has a larger capacity for gas exchange compared to the left lung due to its larger surface area." }

A.1.3 Fetal Head

Example of the first prompt for the fetal head segmentation:

I am creating a fetal head recognition system from the transthalamic and transventricular ultrasound planes; write me 20 attributes to describe the fetal head.

Head Circumference: Measurement of the circumference of the fetal head, indicating overall size. Biparietal Diameter (BPD): Distance between the two parietal bones, a key indicator of fetal head size and gestational age. Head Shape: Describing

the overall shape of the fetal head, which can vary from round to more elongated. Skull Density: Density of the fetal skull bones, influencing ultrasound image contrast. Fontanelle Size: Size and appearance of the fontanelles, indicating the stage of skull ossification. Cerebral Ventricles: Presence and size of cerebral ventricles within the fetal brain. Cranial Bone Integrity: Assessment of the integrity of the cranial bones, looking for any fractures or abnormalities. Midline Structures: Identification of midline structures such as the falx cerebri and cavum septi pellucidi. Cerebellar Vermis: Presence and size of the cerebellar vermis, important for assessing brain development. Third Ventricle: Measurement and appearance of the third ventricle, indicative of normal brain development. Cerebral Hemispheres: Assessment of the size and symmetry of the fetal cerebral hemispheres. Occipitofrontal Diameter (OFD): Measurement from the occipital bone to the frontal bone, providing another indicator of fetal head size. Nuchal Fold Thickness: Thickness of the nuchal fold, which can indicate chromosomal abnormalities. Facial Features: Identification and characterization of facial features such as the nose, lips, and orbits. Fetal Neck: Assessment of the fetal neck for any abnormalities or masses. Eyes and Eye Orbit: Identification of the eyes and assessment of the eye orbits for normal development. Temporal Bone Anatomy: Examination of the temporal bones for any abnormalities or ossification issues. Brain Symmetry: Assessment of symmetry in brain structures, including the hemispheres and ventricles. Brain Parenchyma Texture: Evaluation of the texture and appearance of the fetal brain parenchyma. Fetal Presentation: Determination of the fetal presentation (cephalic, breech, transverse), which can impact the segmentation process.

Example of the second prompt for the fetal head dataset which uses GPT-3.5's response for the first prompt as part of the prompt:

Now write me sentences about each prompt of these attributes with "fetal head" in the sentences, and return the answer as a python dictionary, with fetal head as

key.

The response of the second prompt constitutes the VDT information about fetal head.

"fetal head" : *["Fetal head circumference is a crucial metric measured to assess overall growth and development.", "Biparietal diameter (BPD) provides insight into fetal head size and gestational age.", "The fetal head's shape varies, ranging from round to more elongated forms.", "Skull density contributes to the clarity and contrast of structures within the fetal head.", "Fontanelle size offers information about the fetal head's ossification stages and cranial development.", "Cerebral ventricles within the fetal head are assessed to ensure normalcy and appropriate size.", "Cranial bone integrity is essential for the protection and development of the fetal head.", "Midline structures, such as the falx cerebri, contribute to the overall architecture of the fetal head.", "The cerebellar vermis, located within the fetal head, influences motor coordination and balance.", "The third ventricle within the fetal head is examined to evaluate central nervous system development.", "Symmetry of the cerebral hemispheres is crucial for the overall morphology of the fetal head.", "Occipitofrontal diameter (OFD) measurement aids in assessing the proportionality of the fetal head.", "Nuchal fold thickness measurement provides insights into potential abnormalities affecting the fetal head.", "Facial features within the fetal head region contribute to the overall assessment of anatomical development.", "Evaluation of the fetal neck ensures proper alignment and structural integrity of the fetal head and neck.", "Eyes and eye orbits within the fetal head region are assessed to evaluate ocular development.", "Temporal bone anatomy within the fetal head is evaluated to detect any abnormalities or ossification issues.", "Brain symmetry within the fetal head is assessed to ensure proper development and alignment of structures.", "Texture and appearance of the brain parenchyma provide insights into overall health and development within the fetal head.", "Fetal presentation, whether cephalic, breech, or transverse, influences the positioning and segmentation of the fetal head during imaging."]*

A.2 Textual Prompt Engineering for Fine-grained Medical Tasks

A.2.1 Polyps

Example of first prompt

I am creating a zero shot polyp recognition system in colonoscopy image. Write me 10 attributes to describe the polyp in the colonscopic image.

Attributes of Polyps:

Shape: A polyp in the colon often takes the shape of a small, rounded bump or mushroom-like protrusion. Size: The size of a polyp can vary greatly, ranging from a tiny few millimeters to several centimeters in diameter. Color: A polyp typically has a reddish-pink color, though it may appear paler or darker depending on its blood supply. Surface Texture: The surface texture of a polyp can be smooth, granular, or have a lobulated appearance. Attachment Type: A polyp may be sessile with a flat base, or pedunculated, attached to the colon wall by a thin stalk. Vascularization: Fine blood vessels may be visible on the surface of a polyp, indicating its vascularization. Margins: The margins of a polyp can be well-defined or may blend into the surrounding tissue. Consistency: A polyp often has a softer consistency than the surrounding tissue and may appear slightly translucent. Occurrence: While a polyp can occur in isolation, it's not uncommon to find multiple polyps during a colonoscopy. Impact on Mucosa: A polyp can disrupt the normal mucosal fold patterns of the colon lining.

Now write me a sentence about each of these attributes with "polyp" in the sentences of each respective class and return the answer as a Python dictionary, with polyp as key.

The response of the second prompt constitutes the VDT information about polyps.

"polyp":["A polyp in the colon often takes the shape of a small, rounded bump or mushroom-like protrusion.", "The size of a polyp can vary greatly, ranging from a tiny few millimeters to several centimeters in diameter.", "A polyp typically has a reddish-pink color, though it may appear paler or darker depending on its blood supply.", "The surface texture of a polyp can be smooth, granular, or have a lobulated appearance.", "A polyp may be sessile with a flat base, or pedunculated, attached to the colon wall by a thin stalk.", "Fine blood vessels may be visible on the surface of a polyp, indicating its vascularization.", "The margins of a polyp can be well-defined or may blend into the surrounding tissue.", "A polyp often has a softer consistency than the surrounding tissue and may appear slightly translucent.", "While a polyp can occur in isolation, it's not uncommon to find multiple polyps during a colonoscopy.", "A polyp can disrupt the normal mucosal fold patterns of the colon lining."]

Bibliography

[1] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017. x, 3, 4, 8

[2] Anirudh Choudhary, Li Tong, Yuanda Zhu, and May D Wang. Advancing medical imaging informatics by deep learning-based domain adaptation. *Yearbook of medical informatics*, 29(01):129–138, 2020. x, xi, 5, 30, 73

[3] Neerav Karani, Krishna Chaitanya, Christian Baumgartner, and Ender Konukoglu. A lifelong learning approach to brain mr segmentation across scanners and protocols. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I*, pages 476–484. Springer, 2018. x, 8

[4] Christian S Perone, Pedro Ballester, Rodrigo C Barros, and Julien Cohen-Adad. Unsupervised domain adaptation for medical imaging segmentation with self-ensembling. *NeuroImage*, 194:1–11, 2019. x, 7, 8, 33, 37

[5] Andrej Karpathy and Fei-Fei Li. Neural networks and deep learning. `https://cs231n.github.io/neural-networks-1/`, 2016. Accessed: 2024-10-18. x, 20

[6] Tim Hartley. When parallelism gets tricky: Accelerating floyd-steinberg on the mali gpu. *ARM Community*, 11, 2014. x, 21

[7] François Chollet. A comprehensive guide to convolutional neural networks — the eli5 way. *Towards Data Science*, 2018. URL https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a5 xi, 22

[8] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009. xi, 26, 27

[9] USC Media Communications Lab. Mcl research on domain adaptation, 2018. URL https://mcl.usc.edu/news/2018/12/16/mcl-research-on-domain-adaptation/. Accessed: 2024-10-21. xi, 29

[10] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. xi, 12, 74, 75, 76, 79, 80, 82, 117, 132

[11] R. Souza et al. An open, multi-vendor, multi-field-strength brain mr dataset and analysis of publicly available skull stripping methods agreement. *NeuroImage*, 170:482–494, 2018. xii, xvii, 87, 88, 89, 90, 91, 92, 95, 97, 98, 99, 102, 104, 130, 133, 134, 143, 150, 151, 198

[12] Victor M Campello, Polyxeni Gkontra, Cristian Izquierdo, Carlos Martin-Isla, Alireza Sojoudi, Peter M Full, Klaus Maier-Hein, Yao Zhang, Zhiqiang He, Jun Ma, et al. Multi-centre, multi-vendor and multi-disease cardiac segmentation: the m&ms challenge. *IEEE Transactions on Medical Imaging*, 40(12):3543–3554, 2021. xii, xiii, xvii, 87, 88, 102, 103, 104, 106, 198

[13] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen

Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. xiii, 11, 39, 111, 116, 117, 121, 122, 123, 125, 130, 138, 198

[14] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. xiii, 121, 150

[15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. xiii, 38, 106, 111, 119, 123, 124

[16] Nikhil Pandey. Chest x-ray masks and labels. https://www.kaggle.com/datasets/nikhilpandey360/chest-xray-masks-and-labels/data, 2019. xiii, xiv, xv, 130, 137, 142, 143, 145, 149, 150, 151

[17] Thomas L A van den Heuvel, Dagmar de Bruijn, Chris L de Korte, and Bram van Ginneken. Automated measurement of fetal head circumference using 2D ultrasound images. *PLoS One*, 13(8):e0200412, 2018. xiii, xiv, 130, 133, 134, 138, 143, 145, 150, 151

[18] Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. What does clip know about a red circle? visual prompt engineering for vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11987–11997, 2023. xiv, 112, 147, 148, 156, 157, 160

[19] Debesh Jha, Pia H Smedsrud, Michael A Riegler, Pål Halvorsen, Thomas De Lange, Dag Johansen, and Håvard D Johansen. Kvasir-seg: A segmented polyp dataset. In *MultiMedia modeling: 26th international conference, MMM*

2020, Daejeon, South Korea, January 5–8, 2020, proceedings, part II 26, pages 451–462. Springer, 2020. xv, 158, 167

[20] luukboulogne. STOIC2021- COVID-19 AI Challenge. https://stoic2021.grand-challenge.org/stoic2021/, 2022. [Online; accessed 22-Feb-2022]. xvii, 44, 45, 47, 68

[21] Suruchi Kumari and Pravendra Singh. Deep learning for unsupervised domain adaptation in medical imaging: Recent advancements and future perspectives. *Computers in Biology and Medicine*, page 107912, 2023. 2

[22] Jeffrey De Fauw, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, et al. Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature medicine*, 24(9):1342–1350, 2018. 2, 109

[23] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017. 2, 7, 9

[24] Yinuo Wang, Kai Chen, Weimin Yuan, Zhouping Tang, Cai Meng, and Xiangzhi Bai. Samihs: adaptation of segment anything model for intracranial hemorrhage segmentation. In *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1–5. IEEE, 2024. 3, 12, 73

[25] Sidra Aleem, Julia Dietlmeier, Eric Arazo, and Suzanne Little. Convlora and adabn based domain adaptation via self-training. *arXiv preprint arXiv:2402.04964*, 2024. 3, 7, 72, 73, 108

[26] Y. Chen et al. Domain adaptive faster R-CNN for object detection in the wild. In *CVPR*, 2018. 3, 33, 73

[27] Y. Luo et al. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *CVPR*, pages 2507–2516, 2019. 3, 73

[28] Hao Guan and Mingxia Liu. Domain adaptation for medical image analysis: a survey. *IEEE Transactions on Biomedical Engineering*, 69(3):1173–1185, 2021. 3, 6, 7

[29] Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3819–3824. IEEE, 2018. 3

[30] Y. Li et al. Revisiting batch normalization for practical domain adaptation. *arXiv:1603.04779*, 2016. 3, 75, 78, 81, 86

[31] Joaquin Quiñonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset shift in machine learning*. Mit Press, 2022. 3

[32] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*, 2017. 3

[33] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5, 8, 11, 27, 31, 62

[34] Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5):1122–1131, 2018. 5

[35] Jung-Hua Cheng, Dong Ni, Yung-Hui Chou, Jing Qin, Chui-Me Tiu, Yu-Cheng Chang, and Chung-Ming Chen. Computer-aided diagnosis with deep learn-

ing architecture: applications to breast lesions in us images and pulmonary nodules in ct scans. *Scientific reports*, 6(1):1–13, 2016. 5

[36] Richard Bitar, General Leung, Richard Perng, Sameh Tadros, Alan R Moody, Josee Sarrazin, Caitlin McGregor, Monique Christakis, Sean Symons, Andrew Nelson, et al. Mr pulse sequences: what every radiologist wants to know but is afraid to ask. *Radiographics*, 26(2):513–537, 2006. 7

[37] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 8, 11, 77, 106, 110

[38] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 8

[39] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015. 8, 9

[40] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 8, 45

[41] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, 34(11):8135–8153, 2022. 9

[42] Curtis Northcutt, Lu Jiang, and Isaac Chuang. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70: 1373–1411, 2021. 9

[43] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning.

In *2020 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2020. 9, 36, 37

[44] Gaël Varoquaux and Veronika Cheplygina. Machine learning for medical imaging: methodological failures and recommendations for the future. *NPJ digital medicine*, 5(1):48, 2022. 9

[45] Geert Litjens, Robert Toth, Wendy Van De Ven, Caroline Hoeks, Sjoerd Kerkstra, Bram van Ginneken, Graham Vincent, Gwenael Guillard, Neil Birbeck, Jindang Zhang, et al. Evaluation of prostate segmentation algorithms for mri: the promise12 challenge. *Medical image analysis*, 18(2):359–373, 2014. 9

[46] Xueyan Mei, Zelong Liu, Philip M Robson, Brett Marinelli, Mingqian Huang, Amish Doshi, Adam Jacobi, Chendi Cao, Katherine E Link, Thomas Yang, et al. Radimagenet: an open radiologic deep learning research dataset for effective transfer learning. *Radiology: Artificial Intelligence*, 4(5):e210315, 2022. 9, 120

[47] Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317, 2019. 9, 40, 120

[48] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019. 9, 40, 120

[49] Jevgenij Gamper and Nasir Rajpoot. Multiple instance captioning: Learning representations from histopathology textbooks and articles. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16549–16559, 2021. 9, 120

[50] Eyad Elyan et al. Computer vision and machine learning for medical image analysis: recent advances, challenges, and way forward. *Artificial Intelligence Surgery*, 2(1):24–45, 2022. 9

[51] Lingfeng Yang, Yueze Wang, Xiang Li, Xinlong Wang, and Jian Yang. Fine-grained visual prompting. *Advances in Neural Information Processing Systems*, 36, 2024. 11, 111, 112, 126, 147, 148, 156, 157, 161

[52] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 11, 42, 124, 130, 168, 189

[53] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 11, 42, 111

[54] Wenxuan Wang, Jiachen Shen, Chen Chen, Jianbo Jiao, Yan Zhang, Shanshan Song, and Jiangyun Li. Med-tuning: Exploring parameter-efficient transfer learning for medical volumetric segmentation. 2023. 12, 75

[55] J. Wu et al. Medical sam adapter: Adapting segment anything model for medical image segmentation. *arXiv:2304.12620*, 2023. 12, 39, 75, 117, 118

[56] Ruining Deng, Can Cui, Quan Liu, Tianyuan Yao, Lucas W Remedios, Shunxing Bao, Bennett A Landman, Lee E Wheless, Lori A Coburn, Keith T Wilson, et al. Segment anything model (sam) for digital pathology: Assess zero-shot segmentation on whole slide imaging. *arXiv preprint arXiv:2304.04155*, 2023. 13, 39, 111

[57] Chuanfei Hu, Tianyi Xia, Shenghong Ju, and Xinde Li. When sam meets medical images: An investigation of segment anything model (sam) on multi-

phase liver tumor segmentation. *arXiv preprint arXiv:2304.08506*, 2023. 13, 111

[58] Maciej A Mazurowski, Haoyu Dong, Hanxue Gu, Jichen Yang, Nicholas Konz, and Yixin Zhang. Segment anything model for medical image analysis: an experimental study. *Medical Image Analysis*, 89:102918, 2023. 13, 111

[59] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. URL http://www.deeplearningbook.org. 20

[60] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015. 22, 80, 81

[61] Siyuan Qiao, Huiyu Wang, Chenxi Liu, Wei Shen, and Alan Yuille. Micro-batch training with batch-channel normalization and weight standardization. *arXiv preprint arXiv:1903.10520*, 2019. 22

[62] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021. 25, 109

[63] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)*, pages 1–7. IEEE, 2020. 25

[64] Stevo Bozinovski and Ante Fulgosi. The influence of pattern similarity and transfer learning upon training of a base perceptron b2. In *Proceedings of symposium informatica*, volume 3, pages 121–126, 1976. 26

[65] Ling Shao, Fan Zhu, and Xuelong Li. Transfer learning for visual categorization: A survey. *IEEE transactions on neural networks and learning systems*, 26(5):1019–1034, 2014. 26

[66] Jing Zhang, Wanqing Li, Philip Ogunbona, and Dong Xu. Recent advances in transfer learning for cross-dataset visual recognition: A problem-oriented perspective. *ACM Computing Surveys (CSUR)*, 52(1):1–38, 2019. 26

[67] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020. 27, 60

[68] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y Ng. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th international conference on Machine learning*, pages 759–766, 2007. 27

[69] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27, 2014. 27

[70] Gabriela Csurka. A comprehensive survey on domain adaptation for visual applications. *Domain adaptation in computer vision applications*, pages 1–35, 2017. 28, 44

[71] Wouter M Kouw and Marco Loog. A review of domain adaptation without target labels. *IEEE transactions on pattern analysis and machine intelligence*, 43(3):766–785, 2019. 28, 44

[72] Jee Seok Yoon, Kwanseok Oh, Yooseung Shin, Maciej A Mazurowski, and Heung-Il Suk. Domain generalization for medical image analysis: A review. *Proceedings of the IEEE*, 2024. 30

[73] Tribikram Dhar, Nilanjan Dey, Surekha Borra, and R Simon Sherratt. Challenges of deep learning in medical image analysis—improving explainability and trust. *IEEE Transactions on Technology and Society*, 4(1):68–75, 2023. 30

[74] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2022. 30

[75] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010. 31

[76] Mohsen Ghafoorian, Alireza Mehrtash, Tina Kapur, Nico Karssemeijer, Elena Marchiori, Mehran Pesteie, Charles RG Guttmann, Frank-Erik de Leeuw, Clare M Tempany, Bram Van Ginneken, et al. Transfer learning for domain adaptation in mri: Application in brain lesion segmentation. In *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III 20*, pages 516–524. Springer, 2017. 31

[77] Ravi K Samala, Heang-Ping Chan, Lubomir Hadjiiski, Mark A Helvie, Caleb Richter, and Kenny Cha. Cross-domain and multi-task transfer learning of deep convolutional neural network for breast cancer diagnosis in digital breast tomosynthesis. In *Medical Imaging 2018: Computer-Aided Diagnosis*, volume 10575, pages 172–178. SPIE, 2018. 31

[78] Naimul Mefraz Khan, Nabila Abraham, and Marcia Hon. Transfer learning with intelligent training data selection for prediction of alzheimer's disease. *IEEE Access*, 7:72726–72735, 2019. 31

[79] Asmaa Abbas, Mohammed M Abdelsamea, and Mohamed Medhat Gaber. Detrac: Transfer learning of class decomposed medical images in convolutional neural networks. *IEEE Access*, 8:74901–74913, 2020. 32

[80] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 32, 52, 53, 55, 61, 62, 63, 64, 65, 66, 67, 69, 81

[81] Deepak Kumar, Chetan Kumar, and Ming Shao. Cross-database mammographic image analysis through unsupervised domain adaptation. In *2017 IEEE international conference on big data (Big Data)*, pages 4035–4042. IEEE, 2017. 32

[82] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE transactions on neural networks*, 22(2):199–210, 2010. 32

[83] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, pages 443–450. Springer, 2016. 32, 33

[84] Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016. 32, 33

[85] Jindong Wang, Yiqiang Chen, Shuji Hao, Wenjie Feng, and Zhiqi Shen. Balanced distribution adaptation for transfer learning. In *2017 IEEE international conference on data mining (ICDM)*, pages 1129–1134. IEEE, 2017. 32

[86] Nina Linder, Juho Konsti, Riku Turkki, Esa Rahtu, Mikael Lundin, Stig Nordling, Caj Haglund, Timo Ahonen, Matti Pietikäinen, and Johan Lundin. Identification of tumor epithelium and stroma in tissue microarrays using texture analysis. *Diagnostic pathology*, 7:1–11, 2012. 32

[87] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010. 32

[88] Y. Ganin et al. Domain-adversarial training of neural networks. *JMLR*, 17 (1):2096–2030, 2016. 33, 73

[89] Artem Rozantsev, Mathieu Salzmann, and Pascal Fua. Beyond sharing weights

for deep domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 41(4):801–814, 2018. 33

[90] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4893–4902, 2019. 33

[91] Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 447–463, 2018. 33

[92] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 34

[93] Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. *arXiv preprint arXiv:1706.05208*, 2017. 34

[94] Ilja Manakov, Markus Rohm, Christoph Kern, Benedikt Schworm, Karsten Kortuem, and Volker Tresp. Noise as domain shift: Denoising medical images by unpaired image translation. In *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data: First MICCAI Workshop, DART 2019, and First International Workshop, MIL3ID 2019, Shenzhen, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13 and 17, 2019, Proceedings 1*, pages 3–10. Springer, 2019. 34

[95] Amir Gholami, Shashank Subramanian, Varun Shenoy, Naveen Himthani, Xiangyu Yue, Sicheng Zhao, Peter Jin, George Biros, and Kurt Keutzer. A novel domain adaptation framework for medical image segmentation. In *Brainlesion:*

Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II 4, pages 289–298. Springer, 2019. 34

[96] Joseph Paul Cohen, Margaux Luck, and Sina Honari. Distribution matching losses can hallucinate features in medical image translation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I*, pages 529–536. Springer, 2018. 34

[97] Nripendra Kumar Singh and Khalid Raza. Medical image generation using generative adversarial networks: A review. *Health informatics: A computational perspective in healthcare*, pages 77–96, 2021. 34

[98] Muhammad Yaqub, Feng Jinchao, Kaleem Arshid, Shahzad Ahmed, Wenqian Zhang, Muhammad Zubair Nawaz, and Tariq Mahmood. Deep learning-based image reconstruction for different medical imaging modalities. *Computational and Mathematical Methods in Medicine*, 2022(1):8750648, 2022. 34

[99] Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017. 34

[100] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018. 34

[101] Amirhossein Kazerouni, Ehsan Khodapanah Aghdam, Moein Heidari, Reza Azad, Mohsen Fayyaz, Ilker Hacihaliloglu, and Dorit Merhof. Diffusion models in medical imaging: A comprehensive survey. *Medical Image Analysis*, 88: 102846, 2023. 34

[102] Sophie Starck, Vasiliki Sideri-Lampretsa, Bernhard Kainz, Martin Menten, Tamara Mueller, and Daniel Rueckert. Diff-def: Diffusion-generated defor-

mation fields for conditional atlases. *arXiv preprint arXiv:2403.16776*, 2024. 34

[103] Xin Wang, Hong Chen, Zihao Wu, Wenwu Zhu, et al. Disentangled representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 35

[104] Junlin Yang, Nicha C Dvornek, Fan Zhang, Julius Chapiro, MingDe Lin, and James S Duncan. Unsupervised domain adaptation via disentangled representations: Application to cross-modality liver segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*, pages 255–263. Springer, 2019. 35

[105] Li-Ming Zhao, Xu Yan, and Bao-Liang Lu. Plug-and-play domain adaptation for cross-subject eeg-based emotion recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 863–870, 2021. 35

[106] Eunjin Jeon, Wonjun Ko, Jee Seok Yoon, and Heung-Il Suk. Mutual information-driven subject-invariant and class-relevant deep representation learning in bci. *IEEE Transactions on Neural Networks and Learning Systems*, 2021. 35

[107] Xiao Liu, Pedro Sanchez, Spyridon Thermos, Alison Q O'Neil, and Sotirios A Tsaftaris. Learning disentangled representations in the imaging domain. *Medical Image Analysis*, 80:102516, 2022. 35

[108] Yu-Chu Yu and Hsuan-Tien Lin. Semi-supervised domain adaptation with source label adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24100–24109, 2023. 35

[109] Jeff Heaton. Ian goodfellow, yoshua bengio, and aaron courville: Deep learning: The mit press, 2016, 800 pp, isbn: 0262035618. *Genetic programming and evolvable machines*, 19(1):305–307, 2018. 36, 79, 81

[110] Zeyu Han, Chao Gao, Jinyang Liu, Sai Qian Zhang, et al. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024. 36, 74

[111] Ali Madani, Mehdi Moradi, Alexandros Karargyris, and Tanveer Syeda-Mahmood. Semi-supervised learning with generative adversarial networks for chest x-ray classification with ability of data domain adaptation. In *2018 IEEE 15th International symposium on biomedical imaging (ISBI 2018)*, pages 1038–1042. IEEE, 2018. 36

[112] Joris Roels, Julian Hennies, Yvan Saeys, Wilfried Philips, and Anna Kreshuk. Domain adaptive segmentation in volume electron microscopy imaging. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 1519–1522. IEEE, 2019. 36

[113] Sebastian Ruder and Barbara Plank. Strong baselines for neural semi-supervised learning under domain shift. *arXiv preprint arXiv:1804.09530*, 2018. 36

[114] Krishna Chaitanya, Ertunc Erdil, Neerav Karani, and Ender Konukoglu. Local contrastive loss with pseudo-label based self-training for semi-supervised medical image segmentation. *Medical image analysis*, 87:102792, 2023. 36

[115] Patrick Kage, Jay C Rothenberger, Pavlos Andreadis, and Dimitrios I Diochnos. A review of pseudo-labeling for computer vision. *arXiv preprint arXiv:2408.07221*, 2024. 36

[116] Xi Wang, Hao Chen, Huiling Xiang, Huangjing Lin, Xi Lin, and Pheng-Ann Heng. Deep virtual adversarial self-training with consistency regularization for semi-supervised medical image classification. *Medical image analysis*, 70: 102010, 2021. 37

[117] Ferran Prados, John Ashburner, Claudia Blaiotta, Tom Brosch, Julio Carballido-Gamio, Manuel Jorge Cardoso, Benjamin N Conrad, Esha Datta,

Gergely Dávid, Benjamin De Leener, et al. Spinal cord grey matter segmentation challenge. *Neuroimage*, 152:312–329, 2017. 38

[118] Wei Li, Yifei Zhao, Xi Chen, Yang Xiao, and Yuanyuan Qin. Detecting alzheimer's disease on small dataset: a knowledge transfer perspective. *IEEE journal of biomedical and health informatics*, 23(3):1234–1242, 2018. 38

[119] Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014. 38

[120] Mayug Maniparambil, Chris Vorster, Derek Molloy, Noel Murphy, Kevin McGuinness, and Noel E O'Connor. Enhancing clip with gpt-4: Harnessing visual descriptions as prompts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 262–271, 2023. 38, 112, 114, 120, 124, 129, 170, 189, 202

[121] Sidra Aleem, Fangyijie Wang, Mayug Maniparambil, Eric Arazo, Julia Dietlmeier, Kathleen Curran, Noel EO' Connor, and Suzanne Little. Test-time adaptation with salip: A cascade of sam and clip for zero-shot medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5184–5193, 2024. 38, 109, 112, 157, 171, 172, 177

[122] Dongjie Cheng, Ziyuan Qin, Zekun Jiang, Shaoting Zhang, Qicheng Lao, and Kang Li. Sam on medical images: A comprehensive study on three prompt modes. *arXiv preprint arXiv:2305.00035*, 2023. 39

[123] Sheng He, Rina Bao, Jingpeng Li, Jeffrey Stout, Atle Bjornerud, P Ellen Grant, and Yangming Ou. Computer-vision benchmark segment-anything

model (sam) in medical images: Accuracy in 12 datasets. *arXiv preprint arXiv:2304.09324*, 2023. 39

[124] Sovesh Mohapatra, Advait Gosai, and Gottfried Schlaug. Brain extraction comparing segment anything model (sam) and fsl brain extraction tool. *arXiv preprint arXiv:2304.04738*, 2023. 39

[125] Peilun Shi, Jianing Qiu, Sai Mu Dalike Abaxi, Hao Wei, Frank P-W Lo, and Wu Yuan. Generalist vision foundation models for medical imaging: A case study of segment anything model on zero-shot medical segmentation. *Diagnostics*, 13(11):1947, 2023. 39

[126] Tao Zhou, Yizhe Zhang, Yi Zhou, Ye Wu, and Chen Gong. Can sam segment polyps? *arXiv preprint arXiv:2304.07583*, 2023. 39, 111

[127] Zhongxi Qiu, Yan Hu, Heng Li, and Jiang Liu. Learnable ophthalmology sam. *arXiv preprint arXiv:2304.13425*, 2023. 39

[128] Yichi Zhang, Zhenrong Shen, and Rushi Jiao. Segment anything model for medical image segmentation: Current applications and future directions. *Computers in Biology and Medicine*, page 108238, 2024. 39

[129] Jun Ma and Bo Wang. Segment anything in medical images. *arXiv preprint arXiv:2304.12306*, 2023. 39

[130] Kaidong Zhang and Dong Liu. Customized segment anything model for medical image segmentation. *arXiv preprint arXiv:2304.13785*, 2023. 39, 117

[131] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *Advances in Neural Information Processing Systems*, 36, 2024. 40, 111

[132] Alec Radford. Improving language understanding by generative pre-training. 2018. 40, 112, 156, 170

[133] Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. Medclip: Contrastive learning from unpaired medical images and text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language (EMNLP)*, pages 3876–3887, 2022. 40, 120

[134] Tawsifur Rahman, Amith Khandakar, Yazan Qiblawey, Anas Tahir, Serkan Kiranyaz, Saad Bin Abul Kashem, Mohammad Tariqul Islam, Somaya Al Maadeed, Susu M Zughaier, Muhammad Salman Khan, et al. Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images. *Computers in biology and medicine*, 132:104319, 2021. 40

[135] George Shih, Carol C Wu, Safwan S Halabi, Marc D Kohli, Luciano M Prevedello, Tessa S Cook, Arjun Sharma, Judith K Amorosa, Veronica Arteaga, Maya Galperin-Aizenberg, et al. Augmenting the national institutes of health chest radiograph dataset with expert annotations of possible pneumonia. *Radiology: Artificial Intelligence*, 1(1):e180041, 2019. 40

[136] H Mehta Yang, T Duan, D Ding, A Bagul, C Langlotz, K Shpanskaya, et al. Chexnet: radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017. 40

[137] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 41, 111

[138] Shih-Cheng Huang, Liyue Shen, Matthew P Lungren, and Serena Yeung. Gloria: A multimodal global-local representation learning framework for label-efficient medical image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3942–3951, 2021. 41

[139] Sheng Zhang, Yanbo Xu, Naoto Usuyama, Hanwen Xu, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, et al.

Biomedclip: a multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs. *arXiv preprint arXiv:2303.00915*, 2023. 41

[140] Sheng Zhang, Yanbo Xu, Naoto Usuyama, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, Cliff Wong, et al. Large-scale domain-specific pretraining for biomedical vision-language processing. *arXiv preprint arXiv:2303.00915*, 2(3):6, 2023. 41

[141] Joana Palés Huix, Adithya Raju Ganeshan, Johan Fredin Haslum, Magnus Söderberg, Christos Matsoukas, and Kevin Smith. Are natural domain foundation models useful for medical image classification? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7634–7643, 2024. 42, 106, 111, 177

[142] Sidra Aleem, Mayug Maniparambil, Suzanne Little, Noel O'Connor, and Kevin McGuinness. An ensemble deep learning approach for covid-19 severity prediction using chest ct scans. *arXiv preprint arXiv:2305.10115*, 2023. 45

[143] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 45

[144] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 45

[145] Tongxue Zhou, Su Ruan, and Stéphane Canu. A review: Deep learning for medical image segmentation using multi-modality fusion. *Array*, 3:100004, 2019. 45

[146] Justin Ker, Lipo Wang, Jai Rao, and Tchoyoson Lim. Deep learning applications in medical image analysis. *Ieee Access*, 6:9375–9389, 2017. 45

[147] Yin Dai, Yifan Gao, and Fayu Liu. Transmed: Transformers advance multimodal medical image classification. *Diagnostics*, 11(8):1384, 2021. 45

[148] Hedvig Hricak, David J Brenner, S James Adelstein, Donald P Frush, Eric J Hall, Roger W Howell, Cynthia H McCollough, Fred A Mettler, Mark S Pearce, Orhan H Suleiman, et al. Managing radiation use in medical imaging: a multifaceted challenge. *Radiology*, 258(3):889–905, 2011. 45

[149] Richard B Gunderman and Philip K Wilson. Exploring the human interior: The roles of cadaver dissection and radiologic imaging in teaching anatomy. *Academic Medicine*, 80(8):745–749, 2005. 45

[150] Andrea Borghesi, Angelo Zigliani, Roberto Masciullo, Salvatore Golemi, Patrizia Maculotti, Davide Farina, and Roberto Maroldi. Radiographic severity index in covid-19 pneumonia: relationship to age and sex in 783 italian patients. *La radiologia medica*, 125:461–464, 2020. 46

[151] Beatriz Böger, Mariana M Fachi, Raquel O Vilhena, Alexandre F Cobre, Fernanda S Tonin, and Roberto Pontarolo. Systematic review with meta-analysis of the accuracy of diagnostic tests for covid-19. *American journal of infection control*, 49(1):21–29, 2021. 46

[152] Adam Bernheim, Xueyan Mei, Mingqian Huang, Yang Yang, Zahi A Fayad, Ning Zhang, Kaiyue Diao, Bin Lin, Xiqi Zhu, Kunwei Li, et al. Chest ct findings in coronavirus disease-19 (covid-19): relationship to duration of infection. *Radiology*, 295(3):685–691, 2020. 46

[153] Tao Ai, Zhenlu Yang, Hongyan Hou, Chenao Zhan, Chong Chen, Wenzhi Lv, Qian Tao, Ziyong Sun, and Liming Xia. Correlation of chest ct and rt-pcr testing for coronavirus disease 2019 (covid-19) in china: a report of 1014 cases. *Radiology*, 296(2):E32–E40, 2020. 46

[154] Thomas C Kwee and Robert M Kwee. Chest ct in covid-19: what the radiologist needs to know. *Radiographics*, 40(7):1848–1865, 2020. 46

[155] Ioannis D Apostolopoulos and Tzani A Mpesiana. Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and engineering sciences in medicine*, 43:635–640, 2020. 46

[156] Mohammad Rahimzadeh and Abolfazl Attar. A modified deep convolutional neural network for detecting covid-19 and pneumonia from chest x-ray images based on the concatenation of xception and resnet50v2. *Informatics in medicine unlocked*, 19:100360, 2020. 46

[157] AL Aswathy, Anand Hareendran, and Vinod Chandra SS. Covid-19 diagnosis and severity detection from ct-images using transfer learning and back propagation neural network. *Journal of Infection and Public Health*, 14(10): 1435–1445, 2021. 46

[158] Muhammad Farooq and Abdul Hafeez. Covid-resnet: A deep learning framework for screening of covid19 from radiographs. *arXiv preprint arXiv:2003.14395*, 2020. 46

[159] Linda Wang, Zhong Qiu Lin, and Alexander Wong. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *Scientific reports*, 10(1):1–12, 2020. 46

[160] Ioannis D Apostolopoulos, Sokratis I Aznaouridis, and Mpesiana A Tzani. Extracting possibly representative covid-19 biomarkers from x-ray images with deep learning approach and image data related to pulmonary diseases. *Journal of Medical and Biological Engineering*, 40:462–469, 2020. 46

[161] Marie-Pierre Revel, Samia Boussouar, Constance de Margerie-Mellon, Inès Saab, Thibaut Lapotre, Dominique Mompoint, Guillaume Chassagnon, Audrey Milon, Mathieu Lederlin, Souhail Bennani, et al. Study of thoracic ct in

covid-19: the stoic project. *Radiology*, 301(1):E361–E370, 2021. 47, 48, 52, 55, 193

[162] Docker, Inc. What is a container?, 2024. URL https://docs.docker.com/get-started/docker-concepts/the-basics/what-is-a-container/. Accessed: 2025-02-08. 47, 48

[163] Mohammad Rahimzadeh, Abolfazl Attar, and Seyed Mohammad Sakhaei. A fully automated deep learning-based network for detecting covid-19 from a new and large lung ct scan dataset. *Biomedical Signal Processing and Control*, 68: 102588, 2021. 48

[164] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 630–645. Springer, 2016. 48

[165] Tahereh Javaheri, Morteza Homayounfar, Zohreh Amoozgar, Reza Reiazi, Fatemeh Homayounieh, Engy Abbas, Azadeh Laali, Amir Reza Radmard, Mohammad Hadi Gharib, Seyed Ali Javad Mousavi, et al. Covidctnet: an open-source deep learning approach to diagnose covid-19 using small cohort of ct images. *NPJ digital medicine*, 4(1):1–10, 2021. 48

[166] Haibo Qi, Yuhan Wang, and Xinyu Liu. 3d regnet: Deep learning model for covid-19 diagnosis on chest ct image. *arXiv preprint arXiv:2107.04055*, 2021. 49

[167] Nick Schneider, Florian Piewak, Christoph Stiller, and Uwe Franke. Regnet: Multimodal sensor registration using deep neural networks. In *2017 IEEE intelligent vehicles symposium (IV)*, pages 1803–1810. IEEE, 2017. 49

[168] Hossein Aboutalebi, Maya Pavlova, Hayden Gunraj, Mohammad Javad Shafiee, Ali Sabri, Amer Alaref, and Alexander Wong. Medusa: Multi-scale

encoder-decoder self-attention deep neural network architecture for medical image analysis. *arXiv preprint arXiv:2110.06063*, 2021. 49

[169] Zhao Wang, Quande Liu, and Qi Dou. Contrastive cross-site learning with redesigned net for covid-19 ct classification. *IEEE Journal of Biomedical and Health Informatics*, 24(10):2806–2813, 2020. 49

[170] Eduardo Soares, Plamen Angelov, Sarah Biaso, Michele Higa Froes, and Daniel Kanda Abe. Sars-cov-2 ct-scan dataset: A large dataset of real patients ct scans for sars-cov-2 identification. *MedRxiv*, pages 2020–04, 2020. 49

[171] C Glide-Hurst, D Chen, H Zhong, and IJ Chetty. Changes realized from extended bit-depth and metal artifact reduction in ct. *Medical physics*, 40 (6Part1):061711, 2013. 50

[172] Stern E J, Frank M S, and J D Godwin. Chest computed tomography display preferences. survey of thoracic radiologists. *Invest Radiol*, 99:106906, 1995. 50

[173] Radiopaedia. Windowing (ct). *Radiopaedia.org*, 2024. URL https://radiopaedia.org/articles/windowing-ct. Accessed: 2024-11-10. 50

[174] Xibin Dong, Zhiwen Yu, Wenming Cao, Yifan Shi, and Qianli Ma. A survey on ensemble learning. *Frontiers of Computer Science*, 14:241–258, 2020. 52

[175] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019. 52, 53, 55, 61, 62, 63, 64, 65, 66, 67, 69

[176] PyTorch Contributors. torch.amax — pytorch 2.0 documentation, 2024. URL https://pytorch.org/docs/stable/generated/torch.amax.html. https://pytorch.org/docs/stable/generated/torch.amax.html. 53

[177] Masanari Kimura. Understanding test-time augmentation. In *International Conference on Neural Information Processing*, pages 558–569. Springer, 2021. 54

[178] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 55, 58, 61, 62, 63, 67, 70

[179] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 56, 65

[180] Zongwei Zhou, Vatsal Sodha, Md Mahfuzur Rahman Siddiquee, Ruibin Feng, Nima Tajbakhsh, Michael B Gotway, and Jianming Liang. Models genesis: Generic autodidactic models for 3d medical image analysis. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part IV 22*, pages 384–393. Springer, 2019. 60, 62

[181] Samuel G. III Armato, Geoffrey McLennan, Luc Bidaut, Michael F. McNitt-Gray, Charles R. Meyer, Anthony P. Reeves, Laurence P. Clarke, Byron Y. Croft, and Keyvan Farahani. LIDC-IDRI: The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI), 2024. URL https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI. Accessed: 2024-08-21. 60

[182] National Institutes of Health Clinical Center. Nih chest x-ray dataset. https://nihcc.app.box.com/v/ChestXray-NIHCC, 2017. Accessed: 2024-08-21. 60

[183] Michael P LaValley. Logistic regression. *Circulation*, 117(18):2395–2399, 2008. 62

[184] Samira Pouyanfar, Saad Sadiq, Yilin Yan, Haiman Tian, Yudong Tao, Maria Presa Reyes, Mei-Ling Shyu, Shu-Ching Chen, and Sundaraja S Iyengar. A survey on deep learning: Algorithms, techniques, and applications. *ACM computing surveys (CSUR)*, 51(5):1–36, 2018. 64

[185] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019. 65

[186] Alhassan Mumuni and Fuseini Mumuni. Data augmentation: A comprehensive survey of modern approaches. *Array*, 16:100258, 2022. 65

[187] Sergey P Morozov, AE Andreychenko, NA Pavlov, AV Vladzymyrskyy, NV Ledikhova, VA Gombolevskiy, Ivan A Blokhin, PB Gelezhe, AV Gonchar, and V Yu Chernina. Mosmeddata: Chest ct scans with covid-19 related findings dataset. *arXiv preprint arXiv:2005.06465*, 2020. 68

[188] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018. 68

[189] HJWL Aerts, Emmanuel Rios Velazquez, RT Leijenaar, Chintan Parmar, Patrick Grossmann, Sara Carvalho, and Philippe Lambin. Data from nsclc-radiomics. the cancer imaging archive, 2015. 68

[190] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021. 68

[191] Simon Friederich. Fine-tuning. *The Stanford encyclopedia of philosophy*, 2017. 73

[192] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12104–12113, 2022. 73

[193] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 73, 116, 122

[194] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2, 2019. 73

[195] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 73

[196] B. Shirokikh et al. First U-Net layers contain more domain specific information than the last ones. In *MICCAI Workshops DART and DCL*. Springer, 2020. 74, 79, 88

[197] E.B. Zaken et al. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *Association for Computational Linguistics*, 2021. 74

[198] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018. 74

[199] Zihao Fu, Haoran Yang, Anthony Man-Cho So, Wai Lam, Lidong Bing, and Nigel Collier. On the effectiveness of parameter-efficient fine-tuning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 12799–12807, 2023. 74

[200] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965, 2022. 74, 170

[201] Haoyu He, Jianfei Cai, Jing Zhang, Dacheng Tao, and Bohan Zhuang. Sensitivity-aware visual parameter-efficient fine-tuning. In *Proceedings of*

the IEEE/CVF International Conference on Computer Vision, pages 11825–11835, 2023. 74

[202] Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235, 2023. 74

[203] Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu Lee Wang. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *arXiv preprint arXiv:2312.12148*, 2023. 74

[204] R. Dutt et al. Parameter-efficient fine-tuning for medical image analysis: The missed opportunity. *arXiv:2305.08252*, 2023. 75

[205] N. Houlsby et al. Parameter-efficient transfer learning for NLP. In *ICML*, pages 2790–2799. PMLR, 2019. 76

[206] Zhaojiang Lin, Andrea Madotto, and Pascale Fung. Exploring versatile generative language model via parameter-efficient transfer learning. *arXiv preprint arXiv:2004.03829*, 2020. 76

[207] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *ArXiv e-prints*, pages arXiv–1607, 2016. 76

[208] David Vos, Till Döhmen, and Sebastian Schelter. Towards parameter-efficient automation of data wrangling tasks with prefix-tuning. In *NeurIPS 2022 First Table Representation Workshop*, 2022. 77

[209] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *International Joint Conference on Natural Language Processing*, 2021. 77

[210] Woong-Gi Chang, Tackgeun You, Seonguk Seo, Suha Kwak, and Bohyung Han. Domain-specific batch normalization for unsupervised domain adapta-

tion. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 7354–7362, 2019. 78, 101

[211] Massimiliano Mancini, Lorenzo Porzi, Samuel Rota Bulo, Barbara Caputo, and Elisa Ricci. Boosting domain adaptation by discovering latent domains. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3771–3780, 2018. 78

[212] S. Choi et al. Improving test-time adaptation via shift-agnostic weight regularization and nearest source prototypes. In *ECCV*, 2022. 78

[213] Q. Wang et al. Continual test-time domain adaptation. In *CVPR*, 2022. 78

[214] Y. Liu et al. Ttt++: When does self-supervised test-time training fail or thrive? *NIPS*, 2021. 78

[215] J-Y Zhu et al. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 78

[216] J. Hoffman et al. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv:1612.02649*, 2016. 78

[217] Y. Li et al. Bidirectional learning for domain adaptation of semantic segmentation. In *CVPR*, 2019. 78

[218] H. Huang et al. Domain transfer through deep activation matching. In *ECCV*, 2018. 78

[219] D. Demner-Fushman et al. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2), 2016. 78

[220] K. Kushibar et al. Supervised domain adaptation for automatic sub-cortical brain structure segmentation with minimal user interaction. *Scientific reports*, 9(1):6742, 2019. 78

[221] V. V. Valindria et al. Domain adaptation for MRI organ segmentation using reverse classification accuracy. *arXiv:1806.00363*, 2018. 79

[222] Xiahai Zhuang and Juan Shen. Multi-scale patch and multi-modality atlases for whole heart segmentation of mri. *Medical image analysis*, 31:77–87, 2016. 79, 88

[223] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv preprint arXiv:2012.13255*, 2020. 79

[224] Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. *arXiv preprint arXiv:1804.08838*, 2018. 79

[225] M.J. Mirza et al. The norm must go on: Dynamic unsupervised domain adaptation by normalization. In *CVPR*, 2022. 81

[226] Rasha Sheikh and Thomas Schultz. Unsupervised domain adaptation for medical image segmentation via self-training of early features. In *MIDL*. PMLR, 2022. 82, 83, 88, 91, 92, 94, 95

[227] Y. Zou et al. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018. 94, 95

[228] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 100

[229] Anfeng He, Chong Luo, Xinmei Tian, and Wenjun Zeng. A twofold siamese network for real-time object tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4834–4843, 2018. 100

[230] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. 106

[231] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 106

[232] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 106, 111

[233] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23390–23400, 2023. 106, 110

[234] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369, 2023. 106, 110

[235] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 106, 110

[236] John George Allen MacGregor Willes. Open-world few shot recognition. Master's thesis, University of Toronto (Canada), 2021. 106, 110

[237] Ben Mann, N Ryder, M Subbiah, J Kaplan, P Dhariwal, A Neelakantan, P Shyam, G Sastry, A Askell, S Agarwal, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1, 2020. 106, 110, 116

[238] David Ouyang, Bryan He, Amirata Ghorbani, Neal Yuan, Joseph Ebinger, Curtis P Langlotz, Paul A Heidenreich, Robert A Harrington, David H Liang, Euan A Ashley, et al. Video-based ai for beat-to-beat assessment of cardiac function. *Nature*, 580(7802):252–256, 2020. 109

[239] Guotai Wang, Maria A Zuluaga, Wenqi Li, Rosalind Pratt, Premal A Patel, Michael Aertsen, Tom Doel, Anna L David, Jan Deprest, Sébastien Ourselin, et al. Deepigeos: a deep interactive geodesic framework for medical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 41(7):1559–1572, 2018. 109

[240] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, et al. The medical segmentation decathlon. *Nature communications*, 13(1):4128, 2022. 109

[241] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021. 110

[242] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023. 110

[243] Jungseok Hong, Michael Fulton, and Junaed Sattar. Trashcan: A semantically-segmented dataset towards visual detection of marine debris. *arXiv preprint arXiv:2007.08097*, 2020. 110

[244] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 110

[245] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 111

[246] Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. Open-vocabulary detr with conditional matching. In *European Conference on Computer Vision*, pages 106–122. Springer, 2022. 111

[247] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022. 111

[248] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 111, 113, 118, 119, 177

[249] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 111

[250] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *European conference on computer vision*, pages 707–723. Springer, 2022. 111

[251] Christos Matsoukas, Johan Fredin Haslum, Magnus Söderberg, and Kevin Smith. Pretrained vits yield versatile representations for medical images. *arXiv preprint arXiv:2303.07034*, 2023. 111

[252] Christos Matsoukas, Johan Fredin Haslum, Moein Sorkhei, Magnus Söderberg, and Kevin Smith. What makes transfer learning work for medical images: Feature reuse & other factors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9225–9234, 2022. 111

[253] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Colorful prompt tuning for pre-trained vision-language models. *AI Open*, 5:30–38, 2024. 112, 148, 160

[254] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolo, January 2023. URL https://github.com/ultralytics/ultralytics. 113

[255] Sumit Pandey, Kuan-Fu Chen, and Erik B Dam. Comprehensive multimodal segmentation in medical imaging: Combining yolov8 with sam and hq-sam models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2592–2598, 2023. 113, 118

[256] Risab Biswas. Polyp-sam++: Can a text guided sam perform better for polyp segmentation? *arXiv preprint arXiv:2308.06623*, 2023. 113, 119

[257] Jun Ma, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang. Segment anything in medical images. *Nature Communications*, 15(1):654, 2024. 113, 117, 118

[258] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 116, 150, 179

[259] Mingzhe Hu, Yuheng Li, and Xiaofeng Yang. Skinsam: Empowering skin cancer segmentation with segment anything model. *arXiv preprint arXiv:2304.13973*, 2023. 117

[260] Yuheng Li, Mingzhe Hu, and Xiaofeng Yang. Polyp-sam: Transfer sam for polyp segmentation. In *Medical Imaging 2024: Computer-Aided Diagnosis*, volume 12927, pages 759–765. SPIE, 2024. 117

[261] Weijia Feng, Lingting Zhu, and Lequan Yu. Cheap lunch for medical image segmentation by fine-tuning sam on few exemplars. *arXiv preprint arXiv:2308.14133*, 2023. 118

[262] Moahaimen Talib, Ahmed HY Al-Noori, and Jameelah Suad. Yolov8-cab: Improved yolov8 for real-time object detection. *Karbala International Journal of Modern Science*, 10(1):5, 2024. 118, 119

[263] Wenhui Lei, Xu Wei, Xiaofan Zhang, Kang Li, and Shaoting Zhang. Medlsam: Localize and segment anything model for 3d medical images. *arXiv preprint arXiv:2306.14752*, 2023. 119

[264] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7086–7096, June 2022. 120

[265] Kihyun You, Jawook Gu, Jiyeon Ham, Beomhee Park, Jiho Kim, Eun K Hong, Woonhyuk Baek, and Byungseok Roh. CXR-CLIP: Toward large scale chest x-ray Language-Image pre-training. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, pages 101–111. Springer Nature Switzerland, 2023. 120

[266] Jie Liu, Yixiao Zhang, Jie-Neng Chen, Junfei Xiao, Yongyi Lu, Bennett A Landman, Yixuan Yuan, Alan Yuille, Yucheng Tang, and Zongwei Zhou. Clip-driven universal model for organ segmentation and tumor detection. In

Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 21152–21164, October 2023. 120

[267] Yixiao Zhang, Xinyi Li, Huimiao Chen, Yaoyao Yuille, Alan Land Liu, and Zongwei Zhou. Continual learning for abdominal multi-organ and tumor segmentation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, pages 35–45. Springer Nature Switzerland, 2023. 120

[268] Shaoteng Zhang, Jianpeng Zhang, Yutong Xie, and Yong Xia. TPRO: Text-Prompting-Based weakly supervised histopathology tissue segmentation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, pages 109–118. Springer Nature Switzerland, 2023. 120

[269] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33: 7537–7547, 2020. 122

[270] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 122

[271] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in neural information processing systems*, 34:17864–17875, 2021. 122

[272] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 131

[273] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241. Springer International Publishing, 2015. 131

[274] Segment Anything Team. Segment anything: Demo, 2024. URL https://segment-anything.com/demo. Accessed: 2024-02-07. 139

[275] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. https://github.com/facebookresearch/segment-anything, 2023. Accessed: 2023-11-14. 139

[276] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11888–11898, 2023. 141, 153

[277] Sanjay Subramanian, William Merrill, Trevor Darrell, Matt Gardner, Sameer Singh, and Anna Rohrbach. Reclip: A strong zero-shot baseline for referring expression comprehension. *arXiv preprint arXiv:2204.05991*, 2022. 148

[278] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 156, 157

[279] Stefan Denner, Markus Bujotzek, Dimitrios Bounias, David Zimmerer, Raphael Stock, Paul F Jäger, and Klaus Maier-Hein. Visual prompt engineering for medical vision language models in radiology. *arXiv preprint arXiv:2408.15802*, 2024. 156, 161

[280] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's" up" with vision-language models? investigating their struggle with spatial reasoning. *arXiv preprint arXiv:2310.19785*, 2023. 156

[281] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European Conference on Computer Vision*, pages 696–712. Springer, 2022. 156

[282] Dongsheng Jiang, Yuchen Liu, Songlin Liu, Jin'e Zhao, Hao Zhang, Zhen Gao, Xiaopeng Zhang, Jin Li, and Hongkai Xiong. From clip to dino: Visual encoders shout in multi-modal large language models. *arXiv preprint arXiv:2310.08825*, 2023. 157, 177

[283] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1307–1315, 2018. 157

[284] Razieh Rezaei, Masoud Jalili Sabet, Jindong Gu, Daniel Rueckert, Philip Torr, and Ashkan Khakzar. Learning visual prompts for guiding the attention of vision transformers. *arXiv preprint arXiv:2406.03303*, 2024. 157

[285] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010. 161

[286] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019. 161

[287] Zeyi Sun, Ye Fang, Tong Wu, Pan Zhang, Yuhang Zang, Shu Kong, Yuanjun Xiong, Dahua Lin, and Jiaqi Wang. Alpha-clip: A clip model focusing on wherever you want. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13019–13029, 2024. 161

[288] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 198