# How Good is Good Enough? Strategies for Dealing with Unreliable Segmentation Annotations of Medical Data

Ziyang Wang
St Hilda's College
Department of Computer Science
University of Oxford

A thesis submitted for the degree of
*Doctor of Philosophy*

Michaelmas 2023

# Acknowledgements

My DPhil thesis would not have been possible without the guidance and help of many people.

I would like to express my sincere appreciation to my supervisor Professor Irina Voiculescu. Professor Irina Voiculescu is very knowledgeable, thoughtful, and empathetic. She always can support and guide me with my study and life. She has always been with me to listen, give me detailed advice, and make my study in the right direction. My DPhil study was guided and inspired by her thorough and responsible living mind.

I would like to express my appreciation to St-Hilda's College team, the Computer Science Department team, collaborators, and friends who helped me during my DPhil study.

I would also like to express my gratitude to my parents, and my wife for their strong support of my DPhil study. I am not worried about any difficulties during the 4-year study with their financial and emotional support.

# Abstract

Medical image segmentation is an essential topic in computer vision and medical image analysis, because it enables the precise and accurate segmentation of organs and lesions for healthcare applications. Deep learning has dominated in medical image segmentation due to increasingly powerful computational resources, successful neural network architecture engineering, and access to large amounts of medical imaging data with high-quality annotations. However, annotating medical imaging data is time-consuming and expensive, and sometimes the annotations are unreliable.

This DPhil thesis presents a comprehensive study that explores deep learning techniques in medical image segmentation under various challenging situations of unreliable medical imaging data. These situations include: (1) conventional supervised learning to tackle comprehensive data annotation with full dense masks, (2) semi-supervised learning to tackle partial data annotation with full dense masks, (3) noise-robust learning to tackle comprehensive data annotation with noisy dense masks, and (4) weakly-supervised learning to tackle comprehensive data annotation with sketchy contours for network training.

The proposed medical image segmentation strategies improve deep learning techniques to effectively address a series of challenges in medical image analysis, including limited annotated data, noisy annotations, and sparse annotations. These advancements aim to bring deep learning techniques of medical image analysis into practical clinical scenarios. By overcoming these challenges, the strategies establish a more robust and reliable application of deep learning methods which is valuable for improving diagnostic precision and patient care outcomes in real-world clinical environments.

# Contents

**List of Figures** xi

**List of Abbreviations** xvii

**List of Mathematical Symbols and Notations** xxi

**1 Introduction** 1
- 1.1 Background . . . . . 1
- 1.2 Motivation . . . . . 3
- 1.3 Terminology Disambiguation . . . . . 5
  - 1.3.1 Imprecise Annotated Data . . . . . 5
  - 1.3.2 Low-Quality Annotated Data . . . . . 5
- 1.4 Publications Based on The Thesis . . . . . 6
- 1.5 Thesis Outline . . . . . 8

**2 Deep Learning in Medical Image Segmentation** 11
- 2.1 Architecture Engineering of Segmentation Network . . . . . 11
  - 2.1.1 Backbone Network . . . . . 15
  - 2.1.2 Network Block . . . . . 20
  - 2.1.3 Training Strategy . . . . . 23
  - 2.1.4 Experimental Analysis of the State of the Art . . . . . 24
- 2.2 Supervision of Deep Learning Network . . . . . 28
  - 2.2.1 Supervised Learning . . . . . 28
  - 2.2.2 Semi-Supervised Learning . . . . . 29
  - 2.2.3 Noise-Robust Learning . . . . . 30
  - 2.2.4 Weakly-Supervised Learning . . . . . 31
  - 2.2.5 Further Details . . . . . 32

**3 Medical Image Segmentation Triathlon** 33
- 3.1 Motivation . . . . . 33
- 3.2 Datasets . . . . . 36
- 3.3 Computational Platform . . . . . 41
- 3.4 Practical Segmentation Annotations Simulation Scenarios . . . . . 43

3.4.1 Comprehensive Data Annotation with Full Dense Masks . . 43
3.4.2 Partial Data Annotation with Full Dense Masks . . . . . . . 44
3.4.3 Comprehensive Data Annotation with Noisy Dense Masks . 44
3.4.4 Comprehensive Data Annotation with Sketchy Contours . . 45
3.5 Evaluation Metrics . . . . . . . . . . . . . . . . . . . . . . . . . . 46
3.5.1 Qualitative Analysis . . . . . . . . . . . . . . . . . . . . . 46
3.5.2 Region-based Evaluation Metrics . . . . . . . . . . . . . . . 47
3.5.3 Boundary-based Evaluation Metrics . . . . . . . . . . . . . . 49

**4 Supervised Learning: Comprehensive Data Annotation with Full Dense Masks 53**
4.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 53
4.2 Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55
4.2.1 Supervised Learning Framework Setup . . . . . . . . . . . . 55
4.2.2 Segmentation Backbone Network . . . . . . . . . . . . . . . 55
4.2.3 Network Block . . . . . . . . . . . . . . . . . . . . . . . . 57
4.2.4 Study of Supervised Learning Segmentation . . . . . . . . . 66
4.3 Experiments and Results . . . . . . . . . . . . . . . . . . . . . . . 72
4.3.1 Implementation Details . . . . . . . . . . . . . . . . . . . . 72
4.3.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . . 72
4.3.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . . 72
4.3.4 Ablation Study . . . . . . . . . . . . . . . . . . . . . . . . 73
4.4 Contribution and Discussion . . . . . . . . . . . . . . . . . . . . . 79

**5 Semi-Supervised Learning: Partial Data Annotation with Full Dense Masks 81**
5.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 81
5.2 Literature . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 82
5.2.1 Consistency-Aware Learning . . . . . . . . . . . . . . . . . 82
5.2.2 Adversarial Learning . . . . . . . . . . . . . . . . . . . . . 88
5.2.3 Proposed Advanced Schemes . . . . . . . . . . . . . . . . . 90
5.3 Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 95
5.3.1 Semi-Supervised Learning Framework Setup . . . . . . . . . 95
5.3.2 Triple-View Learning . . . . . . . . . . . . . . . . . . . . . 96
5.3.3 Examiner-Student-Teacher Learning . . . . . . . . . . . . . . 96
5.3.4 Uncertainty-Aware Mean Teacher ViT . . . . . . . . . . . . 99
5.3.5 Computational-Efficient Cross Supervision ViT . . . . . . . 100
5.4 Experiments and Results . . . . . . . . . . . . . . . . . . . . . . . 102
5.4.1 Implementation Details . . . . . . . . . . . . . . . . . . . . 102
5.4.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . . 102

5.4.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . . . . 104
5.4.4 Ablation Study . . . . . . . . . . . . . . . . . . . . . . . . . . 107
5.5 Contribution and Discussion . . . . . . . . . . . . . . . . . . . . . 118

**6 Noise-Robust Learning: Comprehensive Data Annotation with Noisy Dense Masks 119**
6.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 119
6.2 Noise-Robust Learning Framework Setup . . . . . . . . . . . . . . . 121
6.3 Noisy Label Generation . . . . . . . . . . . . . . . . . . . . . . . . 121
6.4 Study of Noise-Robust Segmentation . . . . . . . . . . . . . . . . . 122
6.5 Experiments and Results . . . . . . . . . . . . . . . . . . . . . . . 126
6.5.1 Implementation Details . . . . . . . . . . . . . . . . . . . . . . 126
6.5.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . . . . 126
6.5.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . . . . 127
6.6 Contribution and Discussion . . . . . . . . . . . . . . . . . . . . . 130

**7 Weakly-Supervised Learning: Comprehensive Data Annotation with Sketchy Contours 133**
7.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 133
7.2 Weakly-Supervised Learning Framework Setup . . . . . . . . . . . . 135
7.3 Study of Weakly-Supervised Segmentation . . . . . . . . . . . . . . 136
7.3.1 Scribble Supervision Loss . . . . . . . . . . . . . . . . . . . . 136
7.3.2 Collaborative Hybrid Networks . . . . . . . . . . . . . . . . . 136
7.4 Experiments and Results . . . . . . . . . . . . . . . . . . . . . . . 140
7.4.1 Implementation Details . . . . . . . . . . . . . . . . . . . . . . 140
7.4.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . . . . 140
7.4.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . . . . 141
7.4.4 Ablation Study . . . . . . . . . . . . . . . . . . . . . . . . . . 141
7.5 Conclusion and Discussion . . . . . . . . . . . . . . . . . . . . . . 144

**8 Conclusion and Discussion 147**
8.1 Contribution Summary . . . . . . . . . . . . . . . . . . . . . . . . 147
8.2 Discussion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 149
8.2.1 How Good is Good Enough? . . . . . . . . . . . . . . . . . . . 149
8.2.2 Dealing with Unreliable Annotations . . . . . . . . . . . . . . 149
8.3 Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 150

# List of Figures

1.1 Example Images of CT Spine. (a) CT Image, (b) Ground Truth, (c,d) Imprecise Annotated Data Generated by Erosion, and Dilation. 5
1.2 Example Images of MRI Cardiac. (a) MRI Image, (b) Ground Truth Labeled by a Clinical Expert, (c) Low-Quality Annotated Data. . . 6

2.1 Overview of Architecture Engineering of Deep Learning Network for Image Segmentation. In each group of contribution including backbone network, network block, training strategy, there are various options of 'ingredients' for researchers to utilize. Example studies of 'ingredients' are summarized in Table 2.1. . . . . . . . . . . . . . . . 12
2.2 Overview of Deep Learning Segmentation Network Development. Each network relies on raw data input and outputs a proposed segmentation. Each setup makes a choice of backbone network, network block and training strategy (loss function, optimizer, etc). The relative combinations of these 'ingredients' gives a slightly different network in each case. For the particular example of UNet, some of the options and names are given specifically as a UNet family. 13
2.3 An Example of 2D Convolutional Operation in CNN-based Backbone Network. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
2.4 An Example of 2 Successive CNN Layers in CNN-based Backbone Network. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
2.5 An Example of Self-Attention in ViT-based Layer. . . . . . . . . . . 19
2.6 An Example of 2 Successive Self-Attention-Based Layers in ViT-based Backbone Network. . . . . . . . . . . . . . . . . . . . . . . . . . 19
2.7 Examples of Feature Normalization: Batch Normalization, Layer Normalization, Instance Normalization, Group Normalization. . . . 22
2.8 Overview of Supervised Learning for Image Segmentation . . . . . . 28
2.9 Overview of Semi-Supervised Learning for Image Segmentation. . . 29
2.10 Overview of Noise-Robust Learning for Image Segmentation. . . . . 30
2.11 Overview of Weakly-Supervised Learning for Image Segmentation. . 31

3.1 Example CT Spine Images with Segmentation Ground Truth. The White Pixels Representing the Spine, and the Black Pixels Representing Background. . . . . . . . . . . . . . . . . . . . . . . . . 37

3.2 Example CT COVID-19 Images with Segmentation Ground Truth. The White Pixels Representing the Lung, and the Black Pixels Representing Background. . . . . . . . . . . . . . . . . . . . . . . . . 38
3.3 Example MRI Cardiac Images with Segmentation Ground Truth. The White Pixels Representing the Right Ventricle (RV), Myocardium (Myo), and Left Ventricle (LV), and the Black Pixels Representing Background. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38
3.4 Example MRI Brain Tumor Images with Segmentation Ground Truth. The White Pixels Representing the Brain Tumour, and the Black Pixels Representing Background. . . . . . . . . . . . . . . . . . . 39
3.5 Example Ultrasound Nerve Images with Segmentation Ground Truth. 39
3.6 Example Histology Nuclei Images with Segmentation Ground Truth. 40
3.7 GPU Workstation as Computational Resources for Medical Image Segmentation Triathlon. . . . . . . . . . . . . . . . . . . . . . . . 42
3.8 Example CT COVID-19 Segmentation Multi-Class Inference of a Network Against Ground Truth with TP, TN, FP, and FN Pixels on Consolidation, Glass, Lung Other, and Background. . . . . . . . . . 52

4.1 The Conventional U-shaped Encoder-Decoder Segmentation Backbone Network with Various Modified Networks developed by CNN or ViT Network Blocks, including UNet, TransUNet, SwinUNet, Unet3+, and UNet++. The green CNN-based block consists of two successive CNN layers. The yellow ViT-based block consists of two successive ViT layers. . . . . . . . . . . . . . . . . . . . . . . . . . 56
4.2 A Residual Connection Between 2 CNN Layers. . . . . . . . . . . 58
4.3 A Dense Connection Among 3 CNN Layers. . . . . . . . . . . . . . 59
4.4 (a) The Convolutional Attention Network Block. (b) Channel Attention Module. (c) Spatial Attention Module. . . . . . . . . . . . . 60
4.5 The Residual Block Path. . . . . . . . . . . . . . . . . . . . . . . . 61
4.6 The Proposed Network Blocks in Pyramid CNN. . . . . . . . . . . 62
4.7 Example Sequence of 1D Atrous CNN Layers with Different Dilation Rate Setting. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 65
4.8 The Proposed Network Blocks in Pyramid Pooling. . . . . . . . . 66
4.9 The Residual Encoder to Attention Decoder by Residual Block Path for Medical Image Segmentation. . . . . . . . . . . . . . . . . . . 67
4.10 Quadruple Augmented Pyramid Network for Medical Image Segmentation. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 68
4.11 ViT-Improved-Encoder to CNN-based Decoder Network for Medical Image Segmentation. . . . . . . . . . . . . . . . . . . . . . . . . 71

4.12 Pure Swin ViT-based U-Shape Network for Medical Image Segmentation. . . . . 71
4.13 Example Results on CT COVID-19 Test Set. Given a batch of CT images, each of FPN, UNet, and the proposed QAPNet provides the segmentation inference of four classes of ROI including consolidation area, glass area, other part of lung, and background. . . . . 76
4.14 Example Results on CT Spine Test Set. Given a batch of CT images, each of baseline networks and proposed networks provides the segmentation inference of binary classification of spine. . . . . 77
4.15 Box Plot for the Dice-Coefficient Distribution of Prediction by Each Network on CT Spine Test Set. Our proposed modified RARUNet, QAPNet, and NRUNet are more likely to predict segmentation results with high dice score. . . . . 78

5.1 The Framework of Mean Teacher for Medical Image Segmentation, Which Consisting of a Student Network and a Teacher Network. The parameters of Teacher network is updated by Student network, and the prediction of Teacher network can supervise Student network thus expanding unlabeled dataset. . . . . 84
5.2 The Framework of Cross Pseudo Supervision for Medical Image Segmentation, Which Consisting of Two Networks with Same Architecture but Initialize Separately. Two networks collaborate and beneficial each other with their inference as pseudo label. . . . . 85
5.3 The Framework of FixMatch for Medical Image Segmentation, Which Consisting of Strong and Weak Augmentation. The inference from weak augmentation is more likely to be precise than inference from strong augmentation, and can be considered as pseudo label to train network. . . . . 87
5.4 The Framework of Interpolation Consistency Training Data with MixUp for Medical Image Segmentation. The Inference of the Combination of Two Input Images is Similar with the Combination of Inference of Two Input Image Separately. . . . . 88
5.5 The Framework of Adversarial Learning for Medical Image Segmentation, Consisting of a Segmentation Network and a Evaluation Network. Two networks are trained separately (segmentation network is to provide high-quality inference, and evaluation network is to classify the quality of inference is good enough) and against each other as adversarial learning. . . . . 89
5.6 The Example Dynamic Pseudo Label Scheme with Cross Pseudo Label Learning for Medical Image Segmentation, Which Consisting of Two Networks. The pseudo label is generated by both of inference. 91

5.7 The Example Transformation Consistency Scheme with Mean Teacher for Medical Image Segmentation. The inference from rotated input image is similar with the rotated inference from input image. . . . . 94
5.8 The Framework of Multi-View Learning for Medical Image Segmentation, Which Consisting of Multi Networks to Achieve Multi Cross Pseudo Label Supervision. . . . . . . . . . . . . . . . . . . . . . . . 95
5.9 The Illustration of Adversarial Training Between Examiner and Student. In two training stages, Examiner is trained to classify the inference from Student as to whether from labeled or unlabeled data. Student is trained to provide high-quality inference from unlabeled data that make examiner to classify as labeled data. . . . . . . . . . 98
5.10 The Proposed Examiner-Student-Teacher Framework for Medical Image Segmentation. . . . . . . . . . . . . . . . . . . . . . . . . . . 99
5.11 The Example Uncertainty-Aware Scheme with Mean Teacher for Medical Image Segmentation. . . . . . . . . . . . . . . . . . . . . . . 101
5.12 The Example Computational-Efficient Vision Transformer for Semi-Supervised Medical Image Segmentation. . . . . . . . . . . . . . . . 101
5.13 The Example MRI Cardiac Segmentation Inference with 2D CNN network and All SSL Strategies when 10% of Training Set as Labeled Data. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 103
5.14 The Example MRI Cardiac Segmentation Inference with 2D CNN network and All SSL Strategies when 30% of Training Set as Labeled Data. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 104
5.15 The Example MRI Cardiac Segmentation Inference with 2D CNN network and All SSL Strategies when 50% of Training Set as Labeled Data. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 106
5.16 The Example MRI Cardiac Segmentation Inference with 2D ViT network when 10% of Training Set as Labeled Data. . . . . . . . . . 107
5.17 The Example MRI Cardiac Segmentation Inference with 2D ViT network when 30% of Training Set as Labeled Data. . . . . . . . . . 108
5.18 The Example MRI Cardiac Segmentation Inference with 2D ViT network when 50% of Training Set as Labeled Data. . . . . . . . . . 109
5.19 The Example MRI Cardiac Segmentation Inference with Fully Supervised 2D CNN- and 2D ViT-based Segmentation Network. . . . 109
5.20 The Example MRI Brain Tumour Segmentation Inference with 3D CNN Network when 10% of Training Set as Labeled Data. . . . . . 110
5.21 The Example MRI Brain Tumour Segmentation Inference with 3D CNN Network when 30% of Training Set as Labeled Data. . . . . . 111
5.22 The Example MRI Brain Tumour Segmentation Inference with 3D CNN Network when 50% of Training Set as Labeled Data. . . . . . 112

5.23 The Example MRI Cardiac Segmentation Inference with Fully Supervised 3D CNN-based Segmentation Network. . . . . . . . . . . . 112
5.24 Sample Uncertainty Maps, Masks, and Raw Images during the Training Process. In uncertainty map, the yellow denotes the uncertainty area, and blue denotes certainty area. In mask, white denotes as the eligible area of pseudo label to supervise, and black denotes the uncertain area of pseudo label to be filtered. During training process from early stage to end stage, we can find the certain area is growing larger and eligible area of pseudo label to supervise is growing as well, and the only uncertain area is the boundary of region of interest. 113

6.1 The Example MRI Bone Images with Corresponding Ground Truth Segmentation. We generate noisy labels by erosion and dilation, and the level of noise can be higher or lower. . . . . . . . . . . . . . . . . 120
6.2 The Training Process of Adaptive Denoising Learning Strategy. Predictions are generated through segmentation network with a conventional training manner. Labels with higher losses against of network prediction are more likely to be considered as noisy labels. ADL iteratively removes a specific number of labels deemed as noisy labels in each training epoch. Precise (in blue) and imprecise (in green) labels get classified and, if imprecise, discarded. . . . . . . 124
6.3 The Example Input CT Spine Images, and Prediction of NRUNet Against Ground Truth Under Various Data Situations. . . . . . . . 127

7.1 The Illustration of a Multi-Class Scribble-Supervised Segmentation. (a) Input sagittal left-facing MRI, (b) dense ground-truth annotations, (c) sparse annotations via scribble, (d) segmentation inference by dense-label-supervised UNet, (e) segmentation inference by scribble-label-supervised UNet, (f) segmentation inference by proposed scribble-label-supervised CHNets. . . . . . . . . . . . . . . . 134
7.2 The Framework of Scribble-Supervised Collaborative Hybrid Networks for Medical Image Segmentation. . . . . . . . . . . . . . . . 137
7.3 Example Raw Images with Corresponding Inference. (a) Input MRI image, and Inference by (b) pCE, (c) USTM, (d) Scribble2Label, (e) Mumford-Shah loss, (f) GatedCRFLoss and (g) CHNets. . . . . . . 139
7.4 The Ablation Study of the Different Combinations of Pseudo Label Ensembling. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 142

# List of Abbreviations

| | |
|---|---|
| **2D** | Two-Dimensional |
| **3D** | Three-Dimensional |
| **ACC** | Accuracy |
| **ADL** | Adaptive Denoise Learning |
| **AI** | Artificial Intelligence |
| **ASSD** | Average Symmetric Surface Distance |
| **ASPP** | Atrous Spatial Pyramid Network |
| **BCE** | Binary Cross Entropy |
| **BN** | Batch Normalization |
| **BGN** | Batch Group Normalization |
| **CBAM** | Convolutional Block Attention Module |
| **CESS** | Computational-Efficient Semi-Supervised Segmentation |
| **CHNets** | Collaborative Hybrid Networks |
| **CNN** | Convolutional Neural Networks |
| **Conv** | Convolution |
| **COVID-19** | Coronavirus Disease 2019 |
| **CPS** | Cross Pseudo Supervision |
| **CT** | Computerised Tomography |
| **CV** | Computer Vision |
| **CWN** | Centered Weight Normalization |
| **DA** | Dual Attention |
| **DANet** | Dual Attention Network |
| **DBD**$_g$ | Directed Boundary Dice relative to GT |
| **DBD**$_m$ | Directed Boundary Dice relative to MS |
| **DCN** | Deep Co-Training |

| | |
|---|---|
| **DenseNet** | Densely Connected Network |
| **DL** | Deep Learning |
| **DSC** | Dice Coefficient |
| **FCN** | Fully Convolutional Networks |
| **FN** | False Negative |
| **FP** | False Positive |
| **FPN** | Feature Pyramid Networks |
| **GAN** | Generative Adversarial Networks |
| **GELU** | Gaussian Error Linear Unit |
| **GN** | Group Normalization |
| **GPU** | Graphics Processing Unit |
| **GT** | Ground Truth |
| **HD** | Hausdorff Distance |
| **HDD** | Hard Disk Drive |
| **ICT** | Interpolation Consistency Training |
| **IoU** | Intersection Over Union |
| **LN** | Layer Normalization |
| **LV** | Left Ventricle |
| **MB** | Motherboard |
| **MIA** | Medical Image Analysis |
| **MIS** | Medical Image Segmentation |
| **ML** | Machine Learning |
| **MLP** | Multilayer Perceptron |
| **MRI** | Magnetic Resonance Image |
| **MSA** | Multi-Head Self-Attention |
| **MS** | Machine Segmentation |
| **MT** | Mean Teacher |
| **Myo** | Myocardium |
| **MPA** | Mean Pixel Accuracy |
| **MPP** | Mean Pixel Precision |
| **MPR** | Mean Pixel Recall |

| | |
|---|---|
| **MPS** | Mean Pixel Specificity |
| **MDSC** | Mean Dice Coefficient |
| **MIoU** | Mean Intersection Over Union |
| **Net** | Deep Learning Network |
| **PA** | Pixel Accuracy |
| **PP** | Pixel Precision |
| **PR** | Pixel Recall |
| **PS** | Pixel Specificity |
| **PRE** | Precision |
| **PSPNet** | Pyramid Scene Parsing Network |
| **PSU** | Power Supply Unit |
| **RAM** | Random Access Memory |
| **RBN** | Recurrent Batch Normalization |
| **ReLU** | Rectified Linear Unit |
| **RNN** | Recurrent Neural Network |
| **ROI** | Region of Interest |
| **RV** | Right Ventricle |
| **SBD** | Symmetric Boundary Dice |
| **SEN** | Sensitivity |
| **SPE** | Specificity |
| **SSD** | Solid State Drive |
| **SSL** | Semi-Supervised Learning |
| **SOTA** | State-Of-The-Art |
| **SWMSA** | Shift-Window-based Multi-Head Self-Attention |
| **Swin-ViT** | Shift-Window-based Vision Transformer |
| **TN** | True Negative |
| **TP** | True Positive |
| **TriSegNet** | Triple-View Segmentation Network |
| **TVL** | Triple-View Learning |
| **UAMT** | Uncertainty-Aware Mean Teacher |
| **UN** | Uncollected Nodes |

| | |
|---|---|
| **UNet** | U-Shape Network |
| **ViT** | Vision Transformer |
| **WMSA** | Window-based Multi-Head Self-Attention |
| **WSL** | Weakly-Supervised Learning |

# List of Mathematical Symbols and Notations

| | |
|---|---|
| $\mathbf{D}_{train}$ | Labeled dataset for training |
| $\mathbf{D}_{unlabel}$ | Unlabeled dataset for training |
| $\mathbf{D}_{n}$ | Labeled dataset with noise for training |
| $\mathbf{D}_{w}$ | Labeled dataset with scribble for training |
| $\mathbf{D}_{test}$ | Dataset for testing |
| $\boldsymbol{X}_{\mathrm{l}}$ | An image which is labeled for training |
| $\boldsymbol{X}_{\mathrm{u}}$ | An image which is unlabeled for training |
| $\boldsymbol{X}_{\mathrm{t}}$ | An image which is labeled for testing |
| $O$ | Organ |
| $\boldsymbol{Y}_{\mathrm{gt}}$ | Ground truth for an image |
| $\boldsymbol{Y}_{\mathrm{p}}$ | Segmentation prediction for an image |
| $\theta$ | The parameters of network |
| $\overline{\theta}$ | The average parameters of network |
| $\theta_t$ | The parameters of the defined training step |
| $f(\theta)$ | Segmentation network with the defined parameters |
| $f_S(\theta)$ | Student network |
| $f_T(\overline{\theta})$ | Teacher network |
| $P$ | Segmentation prediction on a pixel of an image |
| $\mathcal{L}_{\mathrm{sup}}$ | Segmentation loss under supervised learning |
| $\mathcal{L}_{\mathrm{semi}}$ | Segmentation loss under semi-supervised learning |
| $\mathcal{L}_{\mathrm{weak}}$ | Segmentation loss under weakly-supervised learning |
| $\mathcal{L}_{\mathrm{pCE}}$ | Partial cross-entropy loss |
| $\mathcal{L}_{\mathrm{inter}}$ | Internal consistency loss |
| $\mathcal{L}_{\mathrm{exter}}$ | External consistency loss |

| | |
|---|---|
| $F$ | Feature map |
| $W$ | The width of feature map |
| $H$ | The height of feature map |
| $C$ | The number of channel dimension |
| $\sigma$ | Sigmoid activation |
| $\gamma$ | Deformation scale factor |
| $RF$ | Receptive field |
| $dr$ | Dilation rate |
| $UN$ | Uncollected nodes |
| $ER$ | Evaluation ratio |
| $p$ | 2D patch |
| $E$ | Patch embedding projection |
| $E_{\mathbf{pos}}$ | Position embedding |
| $z$ | Encoded image representation |
| $Q$ | Query |
| $K$ | Key |
| $V$ | Value |
| $\boldsymbol{U}$ | Uncertainty value |
| $\ominus$ | Erosion |
| $\oplus$ | Dilation |

# 1
# Introduction

### Contents

**1.1 Background . . . . . . . . . . . . . . . . . . . . . . . . . . 1**
**1.2 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . 3**
**1.3 Terminology Disambiguation . . . . . . . . . . . . . . . . 5**
  1.3.1 Imprecise Annotated Data . . . . . . . . . . . . . . . . . . 5
  1.3.2 Low-Quality Annotated Data . . . . . . . . . . . . . . . . 5
**1.4 Publications Based on The Thesis . . . . . . . . . . . . 6**
**1.5 Thesis Outline . . . . . . . . . . . . . . . . . . . . . . . . 8**

## 1.1 Background

Medical image segmentation is an essential research topic within the field of computer vision and medical imaging [1–7]. The accurate segmentation of Region of Interest (ROI) including organs, tissues, and lesions is essential for various healthcare applications, such as diagnosis, treatment planning, computer-aided surgery, and monitoring disease progression [8–13]. The importance of medical image segmentation is further heightened by the rapid advancements in medical imaging technology, leading to an increasing amount of high-resolution raw imaging data. These advancements have prompted the need for robust, efficient, and accurate segmentation methods that can keep pace with the growing volume and complexity of medical imaging data.

In recent years, deep learning techniques have emerged as the dominant method for medical image segmentation, surpassing traditional image processing methods [2–7, 14–17]. Machine learning, especially deep learning, has demonstrated remarkable performance in various medical imaging tasks, such as segmentation [3–5, 17], registration [18–20], and detection [21–24]. This success can be attributed to the increasingly powerful computational resources, continuous improvements in neural network architecture engineering, and the increasing availability of large, annotated, high-quality medical imaging datasets.

One of the main obstacles to deploying deep learning in real clinical settings is the acquisition of high-quality reliable annotations for medical imaging data. Annotating medical imaging data, particularly in image semantic segmentation, is a time-consuming and expensive process that often requires input from experienced medical professionals [25–29]. Furthermore, the annotations provided may be subject to human error and inconsistencies such as inter-observer variability, leading to unreliable training data [17, 30, 31].

To address the challenges associated with unreliable annotations, this DPhil thesis presents a comprehensive study exploring deep learning techniques for medical image segmentation under various challenging scenarios. These scenarios include: (1) Comprehensive data annotation with full dense masks; (2) Partial data annotation with full dense masks; (3) Comprehensive data annotation with noisy dense masks; and (4) Comprehensive data annotation with sketchy contours. Specifically, ‘full dense masks’ denote that each pixel of a medical image is precisely labeled as belonging to either ROI or background. ‘Noisy dense masks’ mean that each pixel is labeled similarly, but some pixels have incorrect labels. The primary objective of this research is to develop a series of strategies to address these unreliable annotation situations while maintaining promising and robust segmentation results by deep learning-based networks.

## 1.2 Motivation

In this thesis, we raise the question of ‘how good is good enough’ for the machine learning community when studying medical imaging. This inquiry is rooted in the observation that there may be a potential overemphasis on achieving the best performance on a given dataset within the field. Such a singular focus, while aiming for excellence, can inadvertently lead to overfitting while developing deep learning networks. The focus on achieving State-of-the-Art (SOTA) results can limit the practical application of deep learning networks in clinical settings, where the primary concern is their effectiveness in the real world, not just their performance in laboratory conditions.

This question also brings to another significant aspect: the quality of ground truth in medical imaging. Ground truth annotations, often considered the benchmark for training and evaluating networks, come with their own set of challenges. In real-world scenarios, achieving 100% accuracy in annotations is not only challenging but also resource-intensive. As we venture into the realms of semi-supervised, noise-robust, and weakly-supervised learning, we find that annotations with 90-95% accuracy, albeit less costly and more efficient to produce, can be sufficient. The effectiveness of our proposed strategies demonstrates that even with partially labeled or sparsely annotated data, deep learning networks can achieve commendable performance.

The thesis explores how networks can be trained to be robust to the ‘unreliable annotation’ inherent in less-than-perfect annotations. The unreliable annotation can stem from a variety of sources, such as variability among different annotators or the inherent complexity and ambiguity present in medical images. By developing and implementing strategies that can handle and learn effectively from such data, we can make significant strides in applying machine learning more broadly and effectively in clinical settings.

Thus, the focus of this thesis extends beyond the pursuit of high accuracy on a single dataset to address the practicalities of network training and deployment in the medical field. We emphasize the need for a more realistic and pragmatic approach, where the success of a network is not just measured by its performance on a dataset but also by its adaptability, robustness, and utility in a real-world clinical environment, i.e. unreliable annotations.

Several factors contribute to the significance of this research:

1. Addressing Clinical Annotation Challenges: Accurate medical image segmentation is important for various clinical applications, including disease diagnosis, treatment planning, and monitoring disease progression. High-quality annotations, however, are time-consuming, expensive, and subject to human error for clinicians. This thesis aims to overcome these annotation challenges by developing deep learning techniques robust to unreliable annotations, thereby enhancing the overall clinical utility of deep learning.

2. Comprehensive Validation: The medical image segmentation triathlon introduced in this research provides a valuable benchmark for evaluating the performance of deep learning techniques under various annotation scenarios. This benchmark is designed to comprehensively compare and validate existing methods, fostering progress in the field and facilitating the development of more effective segmentation algorithms.

3. Methodological Advancements: By proposing novel strategies to tackle unreliable annotation situations in medical image segmentation, this research contributes to the ongoing methodological advancements in the field of deep learning and medical image analysis.

## 1.3 Terminology Disambiguation

The vast majority of work in computer vision concerns the identification of features in the absence of full dense masks. The community uses a variety of terms for slightly different technical problems, and the vocabulary around these problems is still unsettled. This thesis deals with a variety of scenarios where full dense masks are not available.

The choice of terminology in each chapter follows the following pattern:

### 1.3.1 Imprecise Annotated Data

These illustrate a scenario where a clinician sets out to draw complete closed contours around an anatomical feature but they are imprecise, and part of the annotation contains false information due to lacked experience or uncertainty about the situation. The masks used in the labels therefore do not follow the precise contour of the feature, but can move slightly outside or slightly inside the ideal mask.

An example raw image, ground truth, and imprecise annotations are briefly sketched in Figure 1.1. This scenario is named noisy labels and is dealt with in Chapter 6.

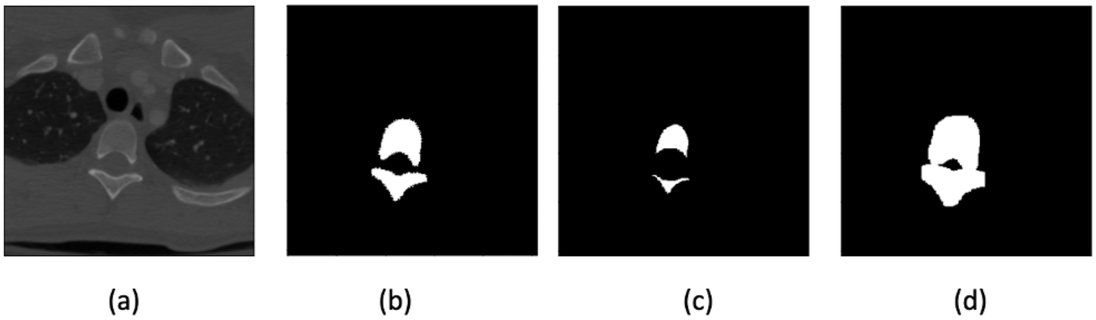


**Figure 1.1:** Example Images of CT Spine. (a) CT Image, (b) Ground Truth, (c,d) Imprecise Annotated Data Generated by Erosion, and Dilation.

### 1.3.2 Low-Quality Annotated Data

A separate scenario where a clinician draws simple lines or ‘scribbles’ over the areas of the image that they want to label and have not got the the time to draw accurate contours. The scribble helps guide the process of identifying and labeling specific regions of interest within an image. An example raw image, ground truth, and low-quality annotation are sketched in Figure 1.2. This scenario is named scribble and is dealt with in Chapter 7.

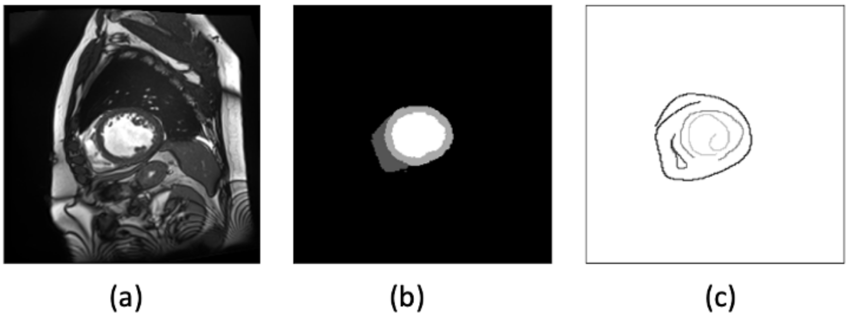


**Figure 1.2:** Example Images of MRI Cardiac. (a) MRI Image, (b) Ground Truth Labeled by a Clinical Expert, (c) Low-Quality Annotated Data.

## 1.4 Publications Based on The Thesis

The publications as the past work presented in this DPhil thesis are as follows.

1. Wang, Z. and Voiculescu, I., 2023, October. Exigent examiner and mean teacher: An advanced 3d cnn-based semi-supervised brain tumor segmentation framework. In Workshop on Medical Image Learning with Limited and Noisy Data (pp. 181-190). Cham: Springer Nature Switzerland.

2. Wang, Z. and Voiculescu, I., 2023, October. Weakly supervised medical image segmentation through dense combinations of dense pseudo-labels. In MICCAI Workshop on Data Engineering in Medical Imaging (pp. 1-10). Cham: Springer Nature Switzerland. [Best Paper Award]

3. Wang, Z., Dong, N. and Voiculescu, I., 2022, October. Computationally-efficient vision transformer for medical image semantic segmentation via dual pseudo-label supervision. In 2022 IEEE International Conference on Image Processing (ICIP) (pp. 1961-1965). IEEE.

4. Wang, Z. and Voiculescu, I., 2022, September. Triple-view feature learning for medical image segmentation. In MICCAI Workshop on Resource-Efficient Medical Image Analysis (pp. 42-54). Cham: Springer Nature Switzerland.
5. Wang, Z., Zheng, J.Q. and Voiculescu, I., 2022, July. An uncertainty-aware transformer for MRI cardiac semantic segmentation via mean teachers. In Annual Conference on Medical Image Understanding and Analysis (pp. 494-507). Cham: Springer International Publishing.
6. Wang, Z., Zhang, Z. and Voiculescu, I., 2021, September. RAR-U-Net: a residual encoder to attention decoder by residual connections framework for spine segmentation under noisy labels. In 2021 IEEE International Conference on Image Processing (ICIP) (pp. 21-25). IEEE.
7. Wang, Z. and Voiculescu, I., 2021, November. Quadruple augmented pyramid network for multi-class COVID-19 segmentation via CT. In 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) (pp. 2956-2959). IEEE.
8. Wang, Z. and Voiculescu, I., 2023. Dealing with unreliable annotations: a noise-robust network for semantic segmentation through a transformer-improved encoder and convolution decoder. Applied Sciences, 13(13), p.7966.

## 1.5 Thesis Outline

This thesis begins with a comprehensive literature review of deep learning techniques in medical image segmentation, providing a solid foundation for understanding the development of SOTA deep learning segmentation networks and identifying potential areas of improvement. Next, a medical image segmentation triathlon is introduced, which includes various medical imaging modalities (CT, MRI, ultrasound, and histology images), data preprocessing techniques, the computational platform for experiments, and evaluation metrics utilized in this thesis. The core of the thesis is to develop novel segmentation network and network training strategies to tackle the aforementioned challenging annotation scenarios. Finally, the experimental results demonstrate that the proposed strategies outperform other methods in the literature in terms of segmentation accuracy and robustness. By addressing the challenges associated with unreliable annotations, the proposed medical image segmentation strategies have the potential to advance deep learning in medical image analysis and contribute to the improvement of healthcare applications in real clinical scenarios.

This DPhil thesis is organized as follows:

Chapter 2: Literature Review - This chapter provides a comprehensive review of the existing literature on deep learning techniques for medical image segmentation, focusing on the key challenges, methodologies, and advancements in network development and training strategy. The literature review serves as a foundation for the subsequent chapters and helps to identify potential areas for improvement.

Chapter 3: Medical Image Segmentation Triathlon - This chapter introduces the medical image segmentation triathlon, which includes dataset with various medical imaging modalities, data preprocessing techniques, the computational experimental platform, and evaluation metrics used in this research. The triathlon serves as a benchmark for comprehensively and fairly evaluating the performance of the proposed deep learning techniques under different annotation scenarios.

Chapter 4: Conventional Supervised Learning - This chapter investigates the performance of deep learning techniques for medical image segmentation in a conventional supervised learning setting, using datasets assumed to be sufficient and perfect.

Chapter 5: Semi-Supervised Learning - This chapter explores semi-supervised learning techniques for medical image segmentation, focusing on situations where limited annotated data and a large amount of raw data are available for network training.

Chapter 6: Noise-Robust Learning - This chapter studies noise-robust learning techniques for medical image segmentation, addressing scenarios with large amounts of annotated data containing incorrect labels for network training.

Chapter 7: Weakly-Supervised Learning - This chapter studies weakly-supervised learning techniques for medical image segmentation, targeting situations where all medical data is sparsely annotated, such as with scribbles, for network training.

Chapter 8: Conclusion and Discussion - This final chapter summarizes the main contributions and discusses potential directions for future research in the area of deep learning for medical image segmentation under unreliable annotation scenarios.

# 2

# Deep Learning in Medical Image Segmentation

### Contents

**2.1 Architecture Engineering of Segmentation Network . . 11**
  2.1.1 Backbone Network . . . . . . . . . . . . . . . . . . . . . . 15
  2.1.2 Network Block . . . . . . . . . . . . . . . . . . . . . . . 20
  2.1.3 Training Strategy . . . . . . . . . . . . . . . . . . . . . . . 23
  2.1.4 Experimental Analysis of the State of the Art . . . . . . 24
**2.2 Supervision of Deep Learning Network . . . . . . . . . 28**
  2.2.1 Supervised Learning . . . . . . . . . . . . . . . . . . . . . 28
  2.2.2 Semi-Supervised Learning . . . . . . . . . . . . . . . . . 29
  2.2.3 Noise-Robust Learning . . . . . . . . . . . . . . . . . . . . 30
  2.2.4 Weakly-Supervised Learning . . . . . . . . . . . . . . . 31
  2.2.5 Further Details . . . . . . . . . . . . . . . . . . . . . . . . 32

## 2.1 Architecture Engineering of Segmentation Network

Medical image segmentation networks aim to classify each pixel of an input image, distinguishing between ROI and background elements. It is an essential study topic in computer vision, as it is the foundational technique to the concept of scene understanding and explaining the global context of an image. This section reviews the advancements and methodologies in the architecture engineering of deep learning networks for image segmentation. The architecture engineering of segmentation network is categorized into three groups based on the general contributions: (1) Fundamental image segmentation backbone networks, such as FCN [3], UNet [4], SegNet [32], Deeplab [33], RefineNet [34], and SegFormer [7]; (2) Segmentation network blocks, such as attention mechanisms [35–39], normalization techniques [40–43], and multi-scale studies [22, 33, 44]; and (3) Segmentation network training strategies, including the design of loss functions [45, 46], optimizer [47, 48], and learning rate settings.

An overview of architecture engineering of deep learning segmentation networks including backbone network, network block, and training strategy is shown in Figure 2.1, and some of essential are detailed in Table 2.1.

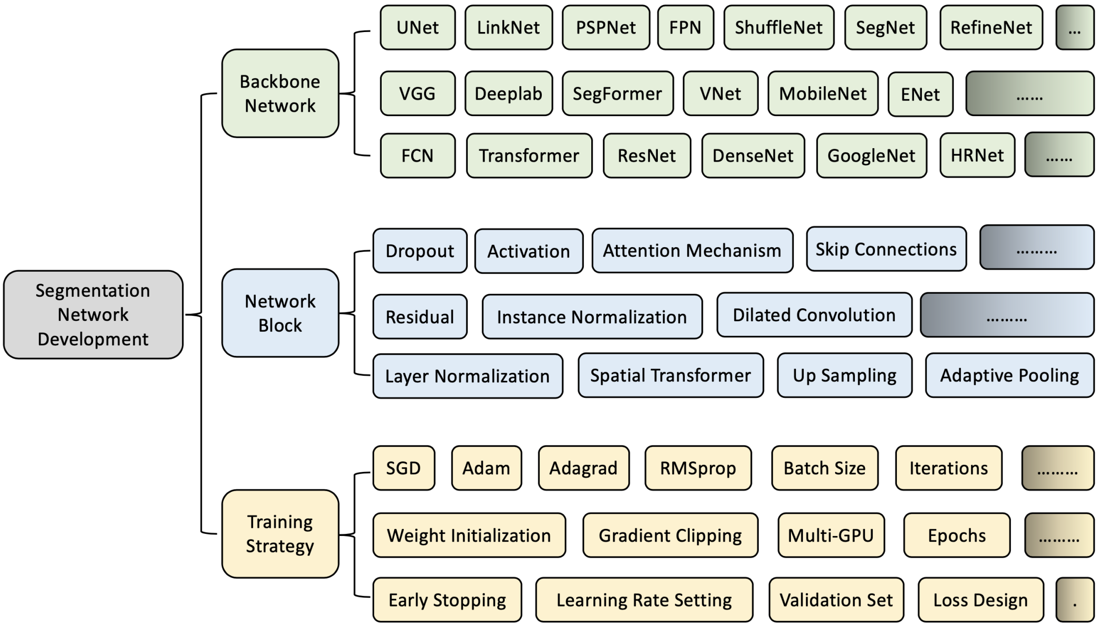


**Figure 2.1:** Overview of Architecture Engineering of Deep Learning Network for Image Segmentation. In each group of contribution including backbone network, network block, training strategy, there are various options of 'ingredients' for researchers to utilize. Example studies of 'ingredients' are summarized in Table 2.1.

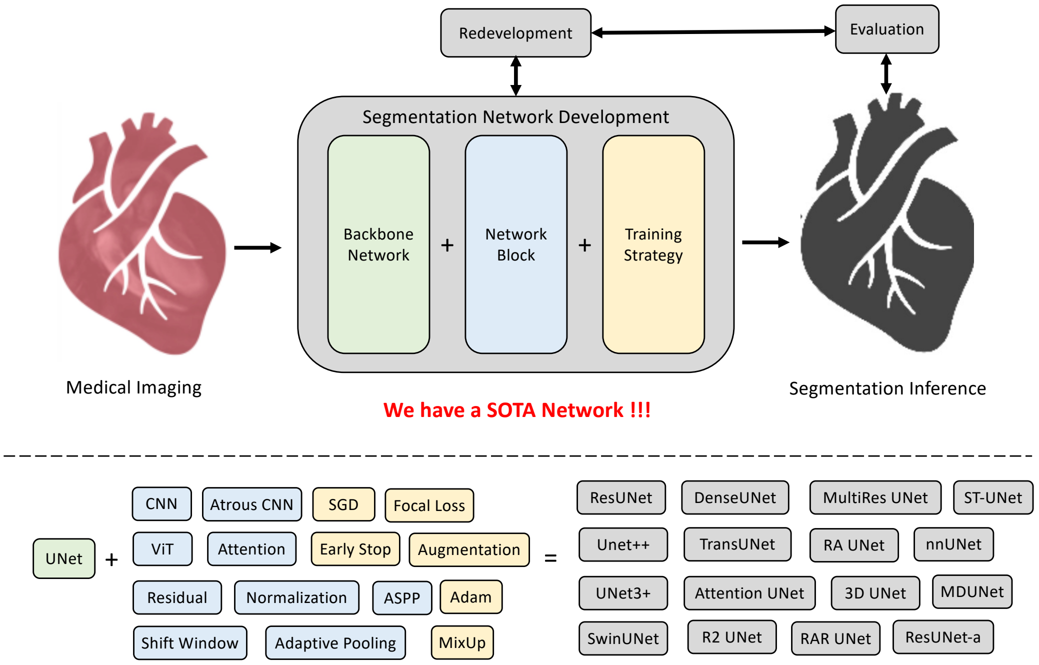


**Figure 2.2:** Overview of Deep Learning Segmentation Network Development. Each network relies on raw data input and outputs a proposed segmentation. Each setup makes a choice of backbone network, network block and training strategy (loss function, optimizer, etc). The relative combinations of these 'ingredients' gives a slightly different network in each case. For the particular example of UNet, some of the options and names are given specifically as a UNet family.

**Table 2.1:** Example Groups of Contributions of Deep Learning Network Architecture Engineering, Illustrating what Backbone Plus Network Block Plus Strategy Combination Gives Rise to Which Network.

| | |
|---|---|
| CNN | Residual Block [49] , Bottle-neck, Split-Attention block [50], Hourglass Module [51], Spatial Transformer [52]<br>group convolution[53], dilated convolution [54], octave convolution [55], Bottleneck Residual Block [56]<br>Depth-wise separable [57], recurrent convolution [58] , ghost convolution[59], PnP [60], DPN Block [61] |
| Attention | SE [62], Non-local [35], CCNet [63], Additive Att [64], Strided Att[65], Dot-Product Att[66], SAGAN [67]<br>Dual Att [68], Spatial Att [38], Channel Att [69], Fixed Factorized Att [65]<br>Residual Att [70], Self-Att [39], Scaled Dot-Product Att [39], Visual Att [71], RAM[72], Sliding Window[73] |
| Multi-scale | ASPP [33], PPM [74], DenseASPP [75], RetinaNet [46], PyramidNet [76], Res2Net [77], AutoFocus [78]<br>FPN [22], SwinViT [79], Scale Aggregation Block [80], MUSIQ [81], CrossViT [82]<br>PSP [44], Xception [57], MnasFPN [83], SNIPER [84], MA-Net [85], ML-Images [86], Yang [87], Yu [54] |
| Loss | Focal [46], Dice [88] , BCE loss, Weightloss, Triplet [89], Cycle Consistency [90], WGAN-GP [91]<br>Boundary loss, Active Countour loss [92], Normalized Temperature-scaled Cross Entropy Loss [93], InfoNCE [94]<br>Adversarial network Perturbation [95], Sharpness-Aware Minimization [96], Additive Angular Margin Loss [97] |
| Pooling & Activation | Max Pooling [98], Average Pooling, Random Pooling, GELU [99], ELU [100], Hard Sigmoid [101]<br>Global Average Pooling[102], Learnable Pooling [103], Swish [104], Shifted Softplus [105], CReLU [106]<br>TanhExp [107], Stochastic pooling [108], Sigmoid Linear Unit [109], Squared ReLU [110], Kernel Activation Function [111] |
| Normalization | Batch Normalization [43], Layer Normalization [40], Instance Normalization [112], Group Normalization [42]<br>Weight Normalization [113], Iterative Normalization [114], Switchable [115], Mix Normalization [116]<br>Decorrelated Batch Normalization [117], Batch Group Normalization [118], Batch Renormalization [119] |
| Training Strategy | StepLR, MultiStepLR, ExponentialLR , CosineAnnealingLR , ReduceLROnPlateau<br>MixUp [120], RandAugment [121], CutMix [122], Cutout [123], Fast AutoAugment [124]<br>AutoAugment [125], simple Copy-Paste [126], AugMix [127], Random Erasing [128], Patch AutoAugment [129] |
| Optimization | BGD [47], SGD [130], AdamW [131] , Adam [48], RMSProp[132], Gravity [133], Gradient Sparsification[134], Adabelief [135]<br>LAMB [136], Deep Ensembles [137], SAGA [138], AMSGrad [139], Dynamic Algorithm Configuration [140], Apollo [141]<br>Adafactor [142], Mixing Adam and SGD [143], Local SGD [144], Stochastic Weight Averaging [145] |
| Backbone | LeNet [146] , ResNet [56], DenseNet [147] , VGGNet [148], GoogleNet [149], Darknet [150], ShuffleNet v2 [151]<br>SqueezeNet [152] ,ShuffleNet [153], MobileNet [154], MobileNetV2 [155], AlexNet [156], EfficientNetV2 [157]<br>EfficientNet [158], ResNeXt, InceptionNet [159], HRNet [160], WideResNet [49], Inception-ResNet-v2 [161] |

In past studies of medical image segmentation, different combinations of the above novel contributions on a specific medical dataset can be considered as a network with SOTA performance (Seen in Figure 2.2) including CT [17, 162–166], MRI [8, 167, 168], histology [4, 169, 170], and Ultrasound [13, 160, 171, 172]. For example, many of past studies explore advanced network blocks for UNet segmentation backbone network resulting in a 'UNet Family': VNet [5], 3DUNet [173], H-DenseUNet [162], nnUNet [6], GP-UNet [174], UNet++[175], RA-UNet [176], MultiResUNet [177], Attention UNet [178], U2-Net [179], RAR-Unet [17], UNet3+ [180], TransUNet [181], and SwinUNet [182].

### 2.1.1 Backbone Network

**CNN-based Segmentation Backbone Network**

The Convolutional Neural Network (CNN) serves as a primary deep learning architecture for image processing tasks. It comprises a series of artificial neural layers that employ convolutional operations within a limited-sized receptive field. An example convolutional operation, illustrated in Figure 2.3, is defined as:

$$\mathbf{Y} = \mathbf{K} \times \mathbf{X} \tag{2.1}$$

where $\mathbf{X}$ is the input feature, $\mathbf{K}$ is the $3 \times 3$ kernel, and $\mathbf{Y}$ is the output feature. The convolutional operation can be defined mathematically as:

$$(\mathbf{K} \times \mathbf{X})_{(i,j)} = \sum_{m=-1}^{1} \sum_{n=-1}^{1} \mathbf{K}_{m+1,n+1} \cdot \mathbf{X}_{i-m,j-n} \tag{2.2}$$

where $(i, j)$ represents the position of the output pixel, and $m$ and $n$ represent the positions within the $3 \times 3$ kernel.

The deep CNN-based network is frequently used for feature extraction in computer vision tasks as CNN-based backbone network. Certain CNN-based backbones have become milestones in network architecture design including LeNet in 1998 [146], AlexNet in 2012 [183], VGG in 2014 [148], InceptionNet in 2014 [159], ResNet in 2015 [56] and DenseNet in 2017 [147].

LeNet [146] is the first deep neural network study utilizing gradient-based learning, featuring a 5 × 5 CNN, 2 × 2 Pooling, and a fully connected network at the end of the network. It is the first efficient CNN-based network for handwritten number classification. AlexNet [183] introduces a deeper CNN through two separate branches, achieving higher accuracy compared to traditional methods on ImageNet. VGG [148] explores further by employing smaller-sized but more than 15 layers of CNN, while ResNet [56] introduces residual learning to address the vanishing gradient problem, making it possible to develop networks with hundreds of CNN layers. In recent years, InceptionNet [159] has explored the use of multiple convolutional filters of different sizes simultaneously to handle different scales of image details. SqueezeNet [152] reduces the number of channels using a 1 × 1 CNN and fewer pooling layers, thereby increasing the receptive field. MobileNet [154] introduces depthwise separable convolution, which efficiently decreased trainable parameters. ShuffleNet [153] employs group convolution, dividing the feature map by channel shuffle and pointwise group convolution.

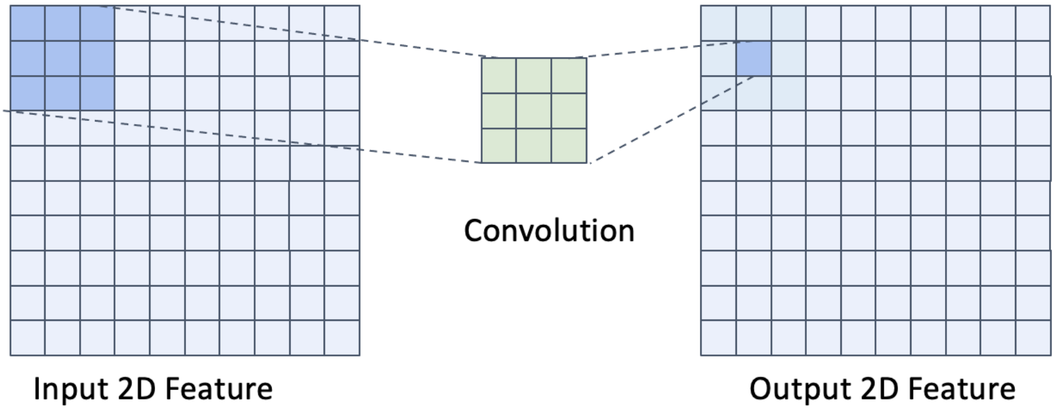


**Figure 2.3:** An Example of 2D Convolutional Operation in CNN-based Backbone Network.

More specifically, for pixel-level classification in semantic segmentation, encoder-decoder architectures have shown great promise. The encoder reduces the spatial dimensions while extracting features, and the decoder restores these dimensions to output a segmentation map that matches the input size. Notable architectures include FCN [3], UNet [4], and Deeplab [184]. FCN is the first CNN-based image

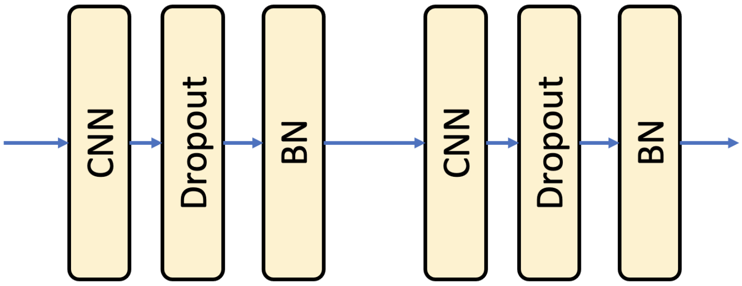


**Figure 2.4:** An Example of 2 Successive CNN Layers in CNN-based Backbone Network.

segmentation work that utilize a deconvolution in the end of network to enable the output size is same with the input image size. Deeplab's main contribution is the design of atrous CNN. Atrous, also known as dilated, CNN is introduced in 2018 [33, 54]. Atrous convolution embeds zeros between non-zero filter taps to sample the feature map. The proposing of atrous CNN directly lead to a series of state-of-the-art segmentation network such as DeeplabV1 [184], DeeplabV2 [33], PSPNet [44], and Deeplabv3 [185]. DeeplabV1 explores on atrous CNN with VGG [148] to enlarge the receptive field, and DeeplabV2 [33] studies on atrous CNN with ResNet by atrous spatial pyramid network (ASPP). ASPP consists of parallel atrous convolutions with varying rates to accommodate different object scales. Finally, DeeplabV3 [185] further explores atrous CNN through multi-grid, maintaining a constant stride for each convolution, thereby enlarging the field-of-view without increasing the number of parameters or the computational workload. UNet [4] is one of the most promising segmentation networks with symmetric encoder-decoder style architecture. The encoder is to collect pixel location features, and then the decoder restores the spatial dimension and pixel location features. To recover the pixels' location information lost in the pooling operation, encoder-decoder architecture based networks are proposed introducing opposite operations including convolution and deconvolution (or transpose convolution), down sampling and up sampling. Therefore, the information of feature and pixel location can be fully retained and analyzed.

### ViT-based Segmentation Backbone Network

Image semantic segmentation, a dense prediction task, is one of a most challenging computer vision tasks, and CNN has been mainly studied since 2015 [3]. These methods, however, are limited by their finite receptive field [33, 54], and until the advent of ViT [186] demonstrates exceptional performance [186].

The success of self-attention mechanisms in natural language processing [39] has inspired their application in image processing tasks, treating images as sequences for sequence-to-sequence problems. ViT demonstrates the efficacy of self-attention in image recognition [186], conceptualizing the input image as a 'sentence' split into patch 'words', thus modeling global dependencies via multi-head self-attention (MSA). Various purely ViT-based backbone networks for dense prediction, such as Swin-ViT [79] and DeiT [187], have been proposed for diverse image processing tasks [15, 188–192]. In medical image segmentation, the combination of UNet and ViT has been particularly influential [181, 193, 194]. TransUnet is a ViT-based medical image segmentation network integrating ViT encoders with CNN tokenized image patches and decoders for upsampling encoded features [181]. SwinUNet then is explored by replacing the whole decoder with Swin-Transformer blocks, creating a ViT-based encoder-decoder U-shape network [193].

As shown in Figures 2.5 and 2.6, a sequence of patches $\mathbf{X}' = [x'_1 \cdots x'_N]^\top \in \mathbb{R}^{N \times P^2}$ is extracted from a medical image $\mathbf{X} \in \mathbb{R}^{h \times w}$, where $P$ is the patch size, and $N = \frac{h \times w}{P^2}$ represents the total number of patches per image. Each patch is flattened into a 1D vector and projected via patch embedding $\mathbf{X}_0 = [E_1 \cdots E_N]^\top, E_{1 \cdots N} \in \mathbb{R}^{D \times P^2}$. Positional embeddings $pos = [pos_1 \cdots pos_N]^\top \in \mathbb{R}^{N \times D}$ are added to capture positional information, forming the final input sequence of tokens for the encoder $\mathbf{Z}_0 = \mathbf{X}_0 + pos$. The Transformer encoder includes a MSA block and a two-layer point-wise MLP block. Each block applies residual connections and layer normalization (LN). The details of MSA and MLP blocks for feature learning are outlined in Equations 2.3 & 2.4, with $i \in \{1 \cdots L\}$, where $L$ is the number of layers in the encoder. The self-attention mechanism comprises three point-wise linear layers mapping tokens to intermediate representations: queries $\mathbf{Q}$, keys $\mathbf{K}$,

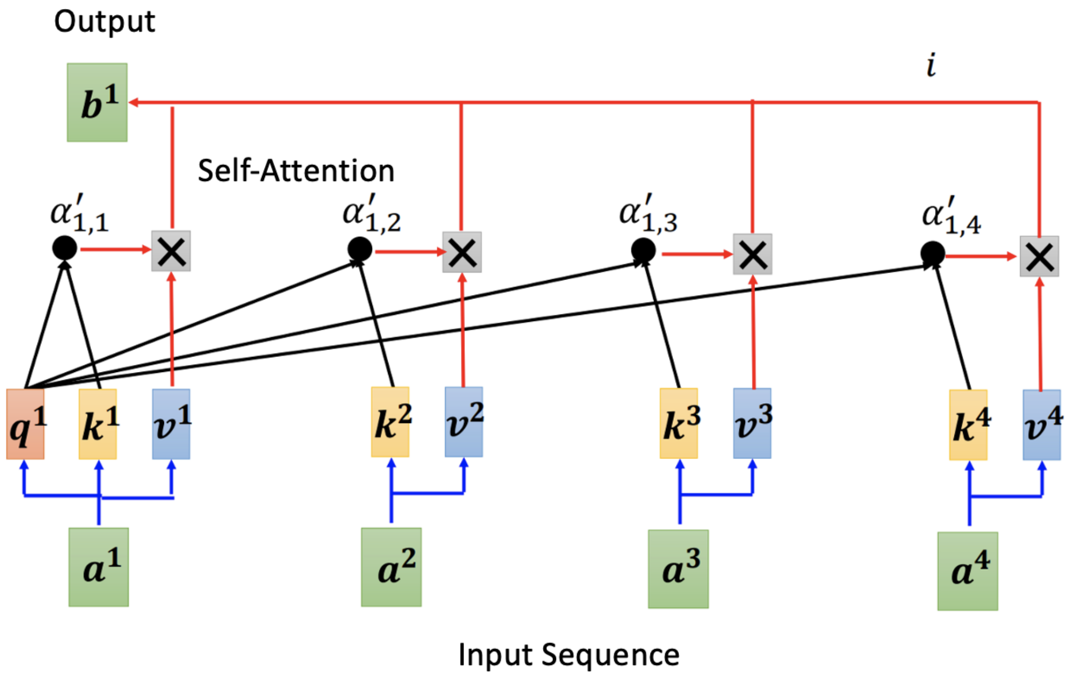


**Figure 2.5:** An Example of Self-Attention in ViT-based Layer.

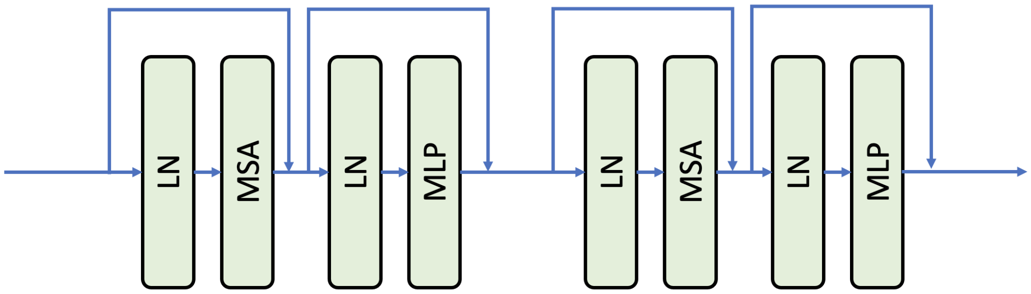


**Figure 2.6:** An Example of 2 Successive Self-Attention-Based Layers in ViT-based Backbone Network.

and values $\mathbf{V}$, as described in Equation 2.5. This process enables the Transformer encoder to map the input sequence $\mathbf{Z}_0 = [z_{0,1} \cdots z_{0,N}]$ with positional information to $\mathbf{Z}_L = [z_{L,1}, ..., z_{L,N}]$. These settings follow the previous work detailed in [186], facilitating the comprehensive utilization of rich semantic features in the encoder.

$$\boldsymbol{A}_{i-1} = \mathrm{MSA}(\mathrm{LN}(\boldsymbol{Z}_{i-1})) + \boldsymbol{Z}_{i-1} \tag{2.3}$$

$$\boldsymbol{Z}_{i} = \mathrm{MLP}(\mathrm{LN}(\boldsymbol{A}_{i-1})) + \boldsymbol{A}_{i-1} \tag{2.4}$$

where MSA is calculated by:

$$\mathrm{MSA}(\boldsymbol{Z}') = \mathrm{softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}}{\sqrt{D}})\,\boldsymbol{V}, \tag{2.5}$$

and the $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ are given by:

$$\boldsymbol{Q} = \mathrm{Linear_Q}(\boldsymbol{Z}'), \boldsymbol{K} = \mathrm{Linear_K}(\boldsymbol{Z}'), \boldsymbol{V} = \mathrm{Linear_V}(\boldsymbol{Z}') \tag{2.6}$$

The sequence of $Z_L$ is subsequently decoded into a dense map $\mathbf{S} \in \mathbb{R}^{h \times w \times k}$, representing the segmentation results, where $k$ denotes the number of classes. The decoder functions by mapping the patch sequence from the encoder and upsampling it to generate pixel-level probability maps for each class [15]. This output patch sequence is reshaped into a 2D mask and bi-linearly upsampled to match the original image size, thus forming the prediction results. In the transformer mask decoder, class embeddings and patch sequences are processed conjointly, enabling the final inference of the semantic segmentation mask.

### 2.1.2 Network Block

In recent years, there has been significant research interest in network blocks that can be attached to backbone networks to improve performance. Some of the widely studied network blocks include attention mechanisms [35–39], feature normalization [40–43], dropout [195], residual networks [56], and densely connected networks [147]. These advanced network blocks are generally simple yet effective in improving the performance of various backbone networks. Two popular network blocks, attention mechanisms and feature normalization, are selected and discussed in this section.

**Attention mechanisms** aim to enhance the performance and explainability of the network by enabling the CNN to focus on key information from the feature maps [35–39]. Visual attention blocks typically assign weights to the feature map, allowing the network to selectively extract features according to their importance. The most classical attention mechanisms include non-local [35], channel attention [38], and dual attention [68]. Non-local attention is a kind of attention mechanism that is first utilized in image analysis [35]. The non-local operation, which establishes long-range dependencies (e.g., between two pixels that are a certain distance apart), is detailed in Equation 2.7.

$$Y_i = \frac{1}{C(X)} \sum_{\nu j} f(X_i, X_j) g(X_j) \tag{2.7}$$

where $X$ is the input features, such as feature map in computer vision task, $Y_i$ is the output and $i$ is the index of location (space location, sequence of time), a pairwise function $f(X_i, X_j)$ computes a scalar (representing relationship such as affinity) between $i$ and all $j$, $g(X_j)$ is the feature extracted on $j$, and $C(X)$ is a normalization function. In general, $i$ is the representation of local feature, $j$ is the representation of global feature, and the non-local weighting is calculated as $Y_i$.

Squeeze-and-Excitation is the first channel attention that adaptively adjusts the response values of each channel [62]. To facilitate feature re-calibration, global pooling is used to selectively emphasize important features while suppressing less useful ones. The channel information is detailed in Equation 2.8:

$$F_{sq}(U_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} U_c(i, j) \tag{2.8}$$

where $H$, $W$, $c$ illustrate the dimension of feature map, $U_c$ is the global average pooling, and $F_{sq}$ is to extract global spatial information into channel, so that the output size is $1 \times 1 \times C$. Convolutional Block Attention Module (CBAM) is an attention module that can be integrated into CNN [38]. CBAM achieve channel attention and spatial attention by cascaded connecting. The Dual Attention Network (DANet) incorporates two types of attention including channel attention and position attention module in conjunction with Atrous FCN for enhanced image semantic segmentation [68]. This integration captures rich contextual relationships by focusing on relevant features across both spatial and channel dimensions.

**Feature Normalization** is to adjust numeric values within a dataset to a common scale without distorting their range or losing information. Typical normalization layer in deep learning network is producing normalized activation maps by subtracting the mean and dividing by the standard deviation aiming at the output to be a normal distribution with a mean of 0 and a variance of 1.

Normalization is beneficial to network because it can improve convergence rate of networks with gradient descent, and improve the accuracy of network. Classical normalization includes Batch Normalization (BN) [43], Layer Normalization (LN) [40], Instance Normalization (IN) [112], and Group Normalization (GN) [42] shown in Figure 2.7, and illustrated in Equation 2.9.

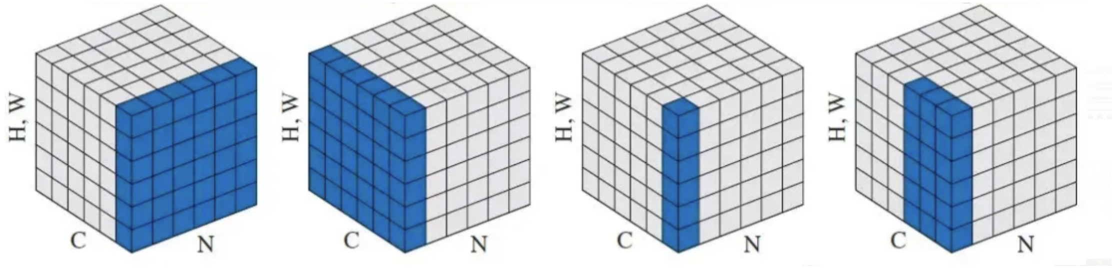


**Figure 2.7:** Examples of Feature Normalization: Batch Normalization, Layer Normalization, Instance Normalization, Group Normalization.

Each normalization address their concern and is advanced under different conditions. BN process on N, H, W dimension and the dimension of the channel is retained, however it require a proper batch sizes. LN address the limited number of batch concern and can also be utilized on RNN. GN is to control the number of feature instances used so that offer neither noisy nor confused statistic for different batch sizes.

$$y = \gamma(\frac{x - \mu(x)}{\sigma(x)}) + \beta \tag{2.9}$$

where $\mu(x)$ is the mean, $\sigma(x)$ is the standard deviation, and $\gamma$ is linear mapping/ re-scale parameter, and $\beta$ is re-shift parameter.

Batch Normalization is defined as follows:

$$\mu_i = \frac{1}{NHW} \sum_{n=1}^{N} \sum_{h=1}^{H} \sum_{w=1}^{W} x_{nchw} \tag{2.10}$$

$$\sigma_i = \sqrt{\frac{1}{NHW} \sum (x_{nchw} - \mu_i)^2} \tag{2.11}$$

Layer Normalization is defined as follows:

$$\mu_i = \frac{1}{vHW} \sum x_i \tag{2.12}$$

$$\sigma_i = \sqrt{\frac{1}{vHW}\sum(x_i - \mu_i)^2} \tag{2.13}$$

Instance Normalization is defined as follow:

$$\mu_n c = \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W} x_{nchw} \tag{2.14}$$

$$\sigma_{nc} = \sqrt{\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw} - \mu_{nc})^2} \tag{2.15}$$

Group Normalization is defined as follow:

$$\mu_{ng} = \frac{1}{(C/G)HW}\sum_{c=gC/G}^{(g+1)C/G}\sum_{h=1}^{H}\sum_{w=1}^{W} x_{nchw} \tag{2.16}$$

$$\sigma_{ng} = \sqrt{\frac{1}{(C/G)HW}\sum_{c=gC/G}^{(g+1)C/G}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw} - \mu_{ng})^2} \tag{2.17}$$

Latest studies have further explored the feature normalization strategy to improve network performance. Centered Weight Normalization (CWN) introduces an additional trainable parameter to the feature normalization [113]. Recurrent Batch Normalization (RBN) applies BN to the hidden transformations within RNN, optimizing performance across sequential data [41]. Moving Average Batch Normalization focuses on refining batch statistics during backward propagation, ensuring more stable updates [196]. The recent study named Batch Group Normalization (BGN) addresses the challenges posed by small or extremely large batch sizes in Batch Normalization by integrating across channel, height, and width dimensions. This method demonstrates superior performance compared to traditional normalization methods such as BN, LN, GN, and IN [118].

### 2.1.3 Training Strategy

The training strategy is a crucial step in engineering a deep learning network architecture for segmentation. It involves various aspects, including dataset augmentation, preprocessing, splitation, loss function, optimizer, batch sizes, distributed learning, etc. Along with the backbone network and network block, the training strategy plays a critical role in determining the network's performance, such as loss design. Typical segmentation loss includes cross entropy [197], weighted cross entropy, Dice-efficient-based loss [88], tversky loss [198], boundary based loss [199] and etc. Regarding the optimizer in machine learning, it is able to tweak and change the trainable parameters of network in the training process to minimize the loss function. Commonly used optimizer includes SGD [130], AdaGrad [200], RMSProp [132], and Adam [48]. The learning rate needs to be set appropriately for effective training. A common strategy is to use ReduceLROnPlateau, where the learning rate is decreased by a factor after a specified number of epochs if there's no improvement in performance. Another approach is CosineAnnealingLR, which applies cosine annealing to modify the learning rate. Additionally, early stopping is a prevalent regularization technique used to prevent overfitting. It halts training when there's no improvement in the network's performance during iterative training processes. The details of classical techniques of network architecture engineering are summarized in Table 2.1.

### 2.1.4 Experimental Analysis of the State of the Art

In recent studies, numerous segmentation networks have been proposed, each claiming SOTA performance within specific experimental setting, such as particular datasets. Even for similar ROI segmentation tasks, e.g. CT COVID-19 segmentation [163, 164, 201–204], researchers employ different training strategies, loss functions, optimizers, and learning rates to achieve what they each claim to be 'SOTA' performance. Table 2.2 summarises these combinations of these 'ingredients', and the evaluation metrics used (mostly Accuracy, Dice and IoU). The existence of numerous SOTA networks complicates the selection process for clinicians, who must understand the network's practical effectiveness in clinical settings. Moreover, the risk of overfitting to one dataset while underperforming in practical clinical environments remains a significant concern.

Since these algorithms are available publicly, the most logical way to compare them against each other is to apply them to the same dataset, in the same conditions, and evaluate them with the same set of metrics. Table 2.3 summarises the outcome of our preliminary experiments carried out in this controlled manner. We employ residual learning [56], attention mechanism [38], densely connected [147], dilated CNN [33], and some other network blocks on segmentation backbone networks including UNet [4], LinkNet [205], and FPN [21]. As shown in Table 2.2 and Table 2.3, deep learning network architecture engineering lacks clear formatting rules, and the performance of proposed different combinations of network blocks, backbone networks and training strategies may perform better or worse on another dataset. This situation leading researchers to typically select different combinations of advanced technique,and experiment in an iterative manner to identify the best network for a specific dataset, as illustrated in Figure 2.2. Researchers have investigated the generalization of segmentation networks across different modalities, such as transferring knowledge from CT to MRI bone segmentation, a process known as domain adaptation. While effective in specific cases, achieving broad generalization across all datasets and modalities remains a significant challenge. Everyone working in this area is caught in an overfitting race, which further leads to difficulties in applying machine learning research to actual clinical practice.

**Table 2.2:** Example Segmentation Frameworks with Contribution on Same COVID-19 Segmentation Tasks.

| Reference | Contribution | | | Dataset Size | Results |
|---|---|---|---|---|---|
| | Backbone Network | Network Block | Training Strategy | | |
| [201] | UNet | 3D CNN | pre-training | 630 cases | AUC: 0.959, Acc: 0.901 |
| [10] | UNet | 2D 3D CNN, ResNet50 | pre-training | 157 cases | AUC:0.996, Sen:0.982 |
| [203] | UNet++ | 3D CNN | multi-task learning | 1136 cases | Sen:0.974, Spe:0.922 |
| [203] | Attention ResNet 50 | | Data Augmentation | 1136 cases | AUC:0.983 |
| [203] | ResNet | | | 1136 cases | AUC:0.991 |
| [203] | GoogleNet | | | 1136 cases | AUC:0.972 |
| [206] | VB-Net | | | 249 cases | Dice: 0.916 |
| [11] | UNet | noisy robust, leaky relu | novel NR-loss function | 558 cases | Dice: 0.79 |
| [11] | UNet | | novel CE loss function | 558 cases | Dice: 0.77 |
| [11] | UNet | | novel Focal CE loss function | 558 cases | Dice: 0.75 |
| [11] | UNet | | Focal and Dice loss function | 558 cases | Dice: 0.77 |
| [11] | UNet | MAE | Dice loss function | 558 cases | Dice: 0.78 |
| [207] | UNet | Attention, Atrous CNN | | 473 slices | Dice:0.83, Sen:0.87 |
| [208] | ResNet-50 | | Data Augmentation | 45 cases | Auc:0.96 |
| [204] | AlexNet | rescaling | | 50 cases | Acc:0.98 |
| [209] | VGG | DenseNet | | 50 cases | Acc:0.90 |
| [24] | MobileNet, | SqueezeNet | Social Mimic Optimizer | 295 cases | Acc:0.99 |
| [165] | ResNet | XceptionNet | | 108 cases | Sen:0.98, Acc: 0.99 |
| [210] | InceptionV3 | Clus-HMC for prediction | | 90 cases | Dice:0.83 |
| [23] | EfficientNet | Instance Normalization | | 645 cases | Acc:0.93, Sen:0.97 |
| [211] | DenseNet | | | 145 cases | Acc:0.95 |
| [212] | TRiple ResNet18 | multi channel | transfer learning | 184 cases | Acc:0.95, F1:0.96 |
| [164] | SqueezeNet | | | 536 slices | Sen:97.5 |
| [163] | ShuffleNet | | | 521 cases | Acc:0.91, Sen:0.91 |

**Table 2.3:** The Direct Comparison of Existing Methods, Illustrating Their Relative Performance on the Same CT Spine Dataset. Although they each claimed SOTA performance, this is not necessarily justifiable at the time of their publication due to the different experimental conditions. The best performance is highlighted with **Bold**.

| Network | Dice | Acc | Pre | Sen | Spe |
|---|---|---|---|---|---|
| 2D UNet | 0.8360 | 0.9863 | 0.8832 | 0.7936 | 0.9952 |
| 2D Residual-UNet | 0.8810 | 0.9898 | **0.9097** | 0.8540 | 0.9961 |
| 2D Densely-UNet | 0.8316 | 0.9860 | 0.8832 | 0.7857 | 0.9952 |
| 2D M-UNet | **0.9478** | 0.9954 | 0.9512 | 0.9444 | **0.9978** |
| 2D M-Densely-UNet | 0.9517 | **0.9958** | 0.9524 | **0.9508** | **0.9978** |
| 2D UNet-VGG16 | 0.9138 | 0.9925 | 0.9235 | 0.9043 | 0.9966 |
| 2D UNet-VGG19 | 0.9024 | 0.9914 | 0.9029 | 0.9019 | 0.9955 |
| 2D UNet-ResNet34 | 0.6626 | 0.9689 | 0.6333 | 0.6947 | 0.9815 |
| 2D UNet-SE-ResNet34 | 0.7306 | 0.9762 | 0.7265 | 0.7347 | 0.9873 |
| 2D UNet-ResNeXt101 | 0.7597 | 0.9765 | 0.6909 | 0.8438 | 0.9826 |
| 2D UNet-DenseNet121 | 0.7982 | 0.9811 | 0.7526 | 0.8498 | 0.9872 |
| 2D UNet-InceptionV3 | 0.8109 | 0.9837 | 0.8250 | 0.7972 | 0.9922 |
| 2D UNet-MobilenetV2 | 0.5671 | 0.9586 | 0.5240 | 0.6179 | 0.9742 |
| 2D UNet-EfficientNet | 0.8358 | 0.9857 | 0.8431 | 0.8286 | 0.9929 |
| 2D MultiRes-UNet | 0.8542 | 0.9864 | 0.8094 | 0.9043 | 0.9902 |
| 2D LinkNet | 0.8958 | 0.9908 | 0.8919 | 0.8999 | 0.9950 |
| 2D FPN | 0.8804 | 0.9893 | 0.8675 | 0.8936 | 0.9937 |
| 3D UNet | 0.8078 | **0.9874** | 0.7788 | 0.8390 | 0.9922 |
| 3D Residual-UNet | 0.7757 | 0.9850 | 0.7360 | 0.8198 | 0.9904 |
| 3D Densely-UNet | 0.7921 | 0.9860 | 0.7450 | 0.8456 | 0.9906 |
| 3D Attention UNet | **0.8623** | 0.9870 | 0.8129 | 0.9182 | 0.9902 |
| 3D UNet-ResNet34 | 0.6114 | 0.9749 | 0.4424 | **0.9896** | 0.9746 |
| 3D UNet-VGG16 | 0.7377 | 0.9770 | 0.7237 | 0.7523 | 0.9871 |
| 3D UNet-MobileNet | 0.7731 | 0.9830 | 0.6465 | 0.9614 | 0.9837 |
| 3D UNet-Inceptionv3 | 0.7148 | 0.9799 | 0.5634 | 0.9775 | 0.9800 |
| 3D UNet-DenseNet121 | 0.4773 | 0.9693 | 0.3144 | 0.9910 | 0.9689 |
| 3D LinkNet-ResNet34 | 0.7468 | 0.9803 | 0.6514 | 0.8749 | 0.9839 |
| 3D LinkNet-VGG16 | 0.8185 | 0.9859 | 0.7092 | 0.9678 | 0.9865 |
| 3D LinkNet-MobileNet | 0.6776 | 0.9777 | 0.5240 | 0.9584 | 0.9782 |
| 3D LinkNet-Inceptionv3 | 0.5267 | 0.9711 | 0.3604 | 0.9780 | 0.9710 |
| 3D LinkNet-DenseNet121 | 0.5046 | 0.9699 | 0.3432 | 0.9620 | 0.9702 |
| 3D FPN-ResNet34 | 0.7074 | 0.9796 | 0.5517 | 0.9855 | 0.9795 |
| 3D FPN-VGG16 | 0.8191 | 0.9861 | 0.7011 | 0.9848 | 0.9862 |
| 3D FPN-MobileNet | 0.6775 | 0.9778 | 0.5227 | 0.9623 | 0.9781 |
| 3D FPN-Inceptionv3 | 0.7484 | 0.9813 | 0.6231 | 0.9369 | 0.9826 |
| 3D FPN-DenseNet121 | 0.8081 | 0.9836 | 0.7718 | 0.8481 | 0.9893 |
| 3D Atrous-ResUNet | 0.8493 | 0.9861 | **0.8734** | 0.8265 | **0.9940** |

## 2.2 Supervision of Deep Learning Network

In order to address the various data situations encountered in real clinical scenarios, this thesis proposes various advanced supervision strategies of deep learning segmentation networks, including supervised learning, semi-supervised learning, noise-robust learning, and weakly-supervised learning, each addressing specific challenges in real clinical data situations.

### 2.2.1 Supervised Learning

Supervised learning is a machine learning strategy where a network is exclusively trained using labeled data available for the entire training set. In medical image segmentation, this strategy involves labeling every pixel of a medical image with its corresponding class, such as tissue, organ, tumor, or background. The goal is to train a deep learning network $f_\theta : \boldsymbol{X} \mapsto \boldsymbol{Y}$ that maps input $\boldsymbol{X}$ to output $\boldsymbol{Y}$ using a perfect and sufficient training dataset $\mathbf{D}$ consisting of pairs $\mathbf{D} = \{(\boldsymbol{X}_i, \boldsymbol{Y}_i)|i = 1, ..., N\}$, where $N$ representing the number of images in dataset, $\boldsymbol{X} \in \mathbb{R}^{h \times w}$ representing a 2D image. The training process is to update parameters $\theta$ of network $f$ to minimize the loss with back propagation. The supervised segmentation loss is illustrated as $L(\theta) = \sum_{i=1}^{N} L(y_i, y_{pred_i})$, where $L(y_i, y_{pred_i})$ is the difference (loss) between ground truth and predicted segmentation map. A brief supervised learning pipeline for image segmentation is sketched in Figure 2.8.

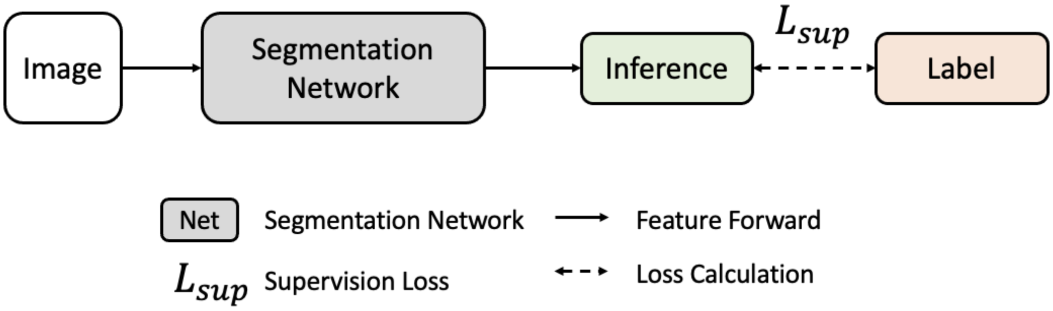


**Figure 2.8:** Overview of Supervised Learning for Image Segmentation

### 2.2.2 Semi-Supervised Learning

Semi-supervised learning is a machine learning strategy that leverages both labeled and unlabeled data during training. This is particularly useful in medical image segmentation, where obtaining labeled data can be expensive and time-consuming and where unlabeled data comes from in great quantity. The goal is to train a deep learning network $f_\theta : \boldsymbol{X} \mapsto \boldsymbol{Y}$ that maps input $\boldsymbol{X}$ to output $\boldsymbol{Y}$ by utilizing the information from the available labeled data $(\boldsymbol{X}, \boldsymbol{Y}_{\mathrm{gt}}) \in \mathbf{D_l}$ and the unlabeled data $(\boldsymbol{X}) \in \mathbf{D_u}$. $\mathbf{D_l} = \{(x_i, y_i) | i = 1, ..., N_l\}$, $\mathbf{D_u} = \{x_j | j = 1, ..., N_u\}$ where $N_l, N_u$ representing the number of images in labeled set and unlabeled set. The semi-supervised segmentation loss is illustrated as $L(\theta) = \mathcal{L}_{\mathrm{sup}} + \lambda \mathcal{L}_{\mathrm{semi}} = \sum_{i=1}^{N_l} L(y_i, y_{pred_i}) + \lambda \sum_{j=1}^{N_u} L(x_j)$, where $\lambda$ is a hyperparameter controlling the trade-off between the supervised loss $\mathcal{L}_{\mathrm{sup}}$ and the consistency loss $\mathcal{L}_{\mathrm{semi}}$, which encourages similar predictions for perturbed versions of the same unlabeled data. A brief semi-supervised learning for image segmentation is sketched in Figure 2.9.

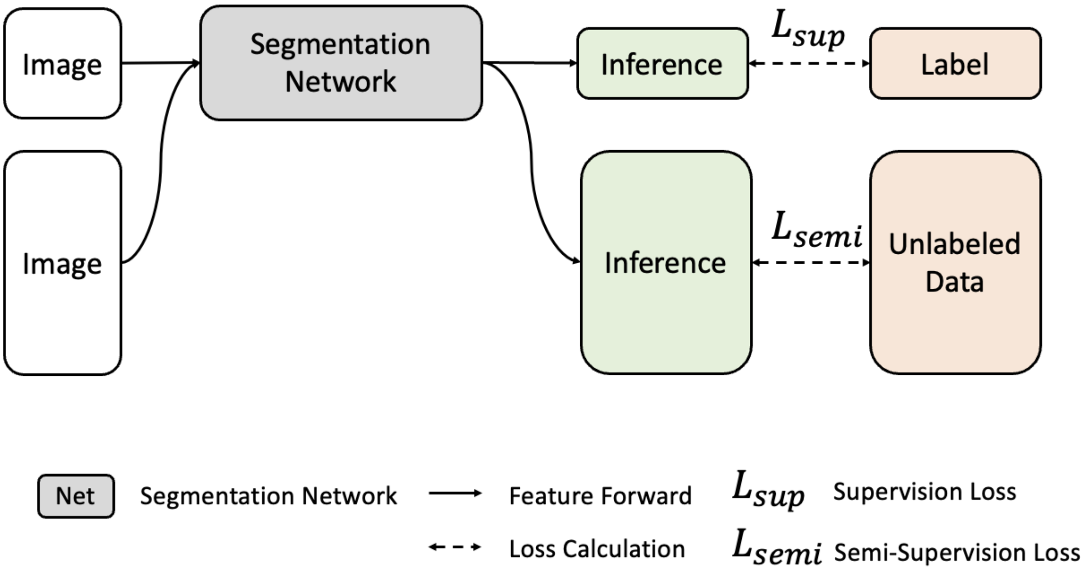


**Figure 2.9:** Overview of Semi-Supervised Learning for Image Segmentation.

### 2.2.3 Noise-Robust Learning

Noise robust learning aims to train a network that is resilient to label noise in the training data. This is also particularly relevant in medical image segmentation, where ground truth annotations might be subject to errors or inconsistencies. The goal is to train a deep learning network $f_\theta : \boldsymbol{X} \mapsto \boldsymbol{Y}$ that maps input $\boldsymbol{X}$ to output $\boldsymbol{Y}$ by utilizing the information from the available labeled data with noise $(\boldsymbol{X}, \boldsymbol{Y}_{\text{gt}} + \mathit{noise}) \in \mathbf{D_n}$. $\mathbf{D_n} = \{(x_i, y_i + \mathit{noise}) | i = 1, ..., N\}$, with $y_i$ is with potentially noisy. The training process of noise-robust learning is developed to minimize the influence of noise. A brief noise-robust learning for image segmentation is sketched in Figure 2.10.

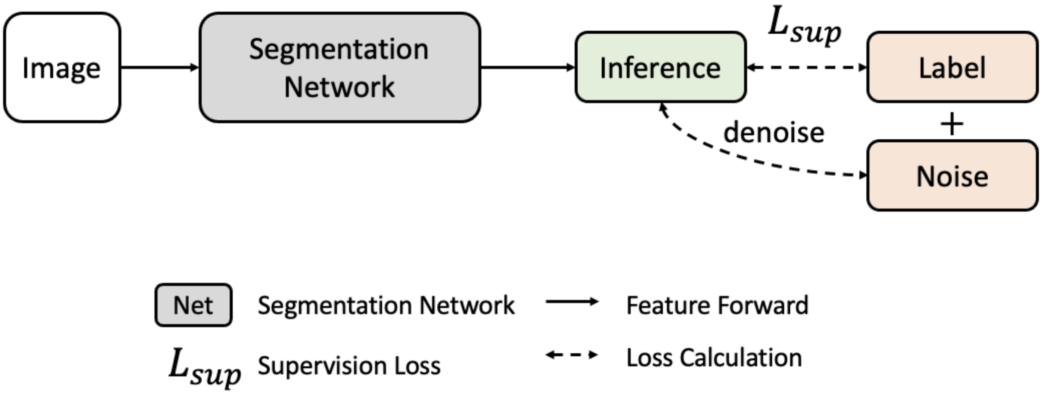


**Figure 2.10:** Overview of Noise-Robust Learning for Image Segmentation.

### 2.2.4 Weakly-Supervised Learning

Weakly-supervised learning utilizes with sparse or coarse annotations instead of pixel-level annotated segmentation masks. For medical image segmentation, the network can be trained with image-level labels, bounding boxes, points, and scribbles. The goal is to train a deep learning network $f_\theta : \boldsymbol{X} \mapsto \boldsymbol{Y}$ that maps input $\boldsymbol{X}$ to output $\boldsymbol{Y}$ by utilizing the information from the available sparse labeled data $(\boldsymbol{X}, \boldsymbol{Y}_{\text{weak}}) \in \mathbf{D_w}$. $\boldsymbol{Y}_{\text{weak}}$ is with limited signal compared with $\boldsymbol{Y}_{\text{gt}}$ in supervised or semi-supervised learning. Loss design: $L(\theta) = \mathcal{L}_{\text{weak}} = \sum_{i=1}^{N} L(y_{weaki}, y_{pred_i})$, where $L(a_i, y_{pred_i})$ is the loss function adapted to accommodate weak annotations. A brief weakly-supervised learning for image segmentation is sketched in Figure 2.11.

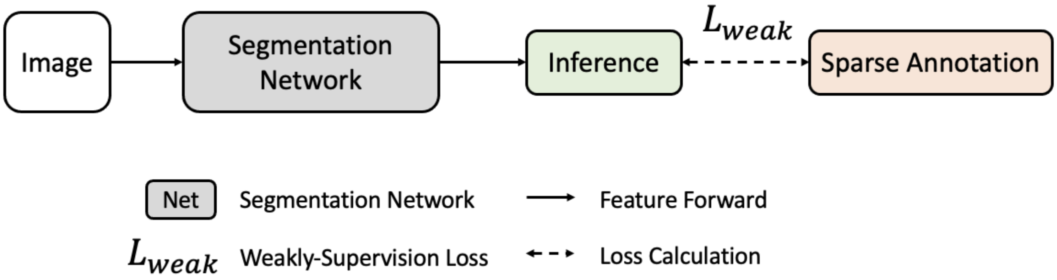


**Figure 2.11:** Overview of Weakly-Supervised Learning for Image Segmentation.

### 2.2.5 Further Details

Each of the strategies described in Section 2.2 have been studied in detail and are described in appropriate contexts in Chapters 4 to 7.

# 3

# Medical Image Segmentation Triathlon

## Contents

3.1 **Motivation** . . . . . . . . . . . . . . . . . . . . . . . . . . . 33
3.2 **Datasets** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 36
3.3 **Computational Platform** . . . . . . . . . . . . . . . . . . . . 41
3.4 **Practical Segmentation Annotations Simulation Scenarios** . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43
  3.4.1 Comprehensive Data Annotation with Full Dense Masks 43
  3.4.2 Partial Data Annotation with Full Dense Masks . . . . 44
  3.4.3 Comprehensive Data Annotation with Noisy Dense Masks 44
  3.4.4 Comprehensive Data Annotation with Sketchy Contours 45
3.5 **Evaluation Metrics** . . . . . . . . . . . . . . . . . . . . . . 46
  3.5.1 Qualitative Analysis . . . . . . . . . . . . . . . . . . . . . 46
  3.5.2 Region-based Evaluation Metrics . . . . . . . . . . . . . 47
  3.5.3 Boundary-based Evaluation Metrics . . . . . . . . . . . 49

## 3.1 Motivation

In recent years, there has been a significant increase in the number of research studies focusing on medical image segmentation using deep learning techniques [3, 4, 7, 29, 32–34, 163, 175, 176, 178, 180, 205, 207, 211, 213–217]. These studies present various deep learning-based networks with different architectures, validated across diverse datasets, experimental settings, and evaluation metrics, each claiming

SOTA performance (seen in Table 2.2 and Table 2.3). This diversity makes it challenging to directly compare their relative performance. The name of Triathlon motivated by 'Medical Image Segmentation Decathlon' [218] is proposed, and the main motivation behind the Medical Image Segmentation Triathlon is to establish a standardized framework that facilitates a solid and comprehensive evaluation of different deep learning methods for medical image segmentation. This framework aims to address the following concerns:

1. Diverse Medical Imaging Modalities: Many studies focus on a single modality, such as CT or MRI, which limits the generalizability of the findings. The Medical Image Segmentation Triathlon incorporates multiple imaging modalities (CT, MRI, ultrasound, and histology images) to ensure that the proposed methods are robust and applicable across different types of medical images.

2. Consistent Experimental Settings: Variations in data preprocessing, network architectures, and training strategies can significantly affect the performance of segmentation networks. The Triathlon provides a standardized approach to data preprocessing, network implementation, and training strategies, ensuring that all methods are evaluated under fair and comparable conditions.

3. Comprehensive Evaluation Metrics: Different studies often use various evaluation metrics, complicating comparisons. The Medical Image Segmentation Triathlon simultaneously employs multiple evaluation metrics, such as region-based metrics: Dice Coefficient, Jaccard Index, Sensitivity, Specificity, and boundary-based metrics: Hausdorff Distance, and etc providing a comprehensive assessment of the performance of various methods.

4. Real-world Challenges: Many existing studies focus on scenarios with perfect or near-perfect annotations, which is often not the case in real-world clinical settings. The Triathlon addresses this by simulating various challenging situations with unreliable annotations, including semi-supervised learning, noise-robust learning, and weakly-supervised learning. This approach enables

a more realistic evaluation of the proposed techniques and their potential applicability in real-world clinical scenarios.

In summary, the Medical Image Segmentation Triathlon aims to establish a rigorous and standardized framework for evaluating deep learning techniques in medical image segmentation. By addressing the diversity of imaging modalities, ensuring consistent experimental settings, using comprehensive evaluation metrics, and simulating real-world challenges, the Triathlon framework can help researchers and clinicians better understand and compare the performance of various segmentation methods, ultimately contributing to the development of more effective and robust solutions for medical image analysis.

## 3.2 Datasets

To conduct a comprehensive evaluation of medical image segmentation methods, we utilize a diverse collection of medical imaging datasets. These datasets cover a wide range of medical imaging modalities, including CT, MRI, ultrasound, and histology images. Each dataset contains images of different organs and pathologies, providing a triathlon test bed for assessing the performance of segmentation techniques.

The datasets are introduced respectively as follows:

1. CT Spine [219]: A dataset containing CT spine dataset published from California and NIH, consisting of CT scans from 10 patients aged from 16 to 35 years with up to 600 slices per scan, at a resolution of $512 \times 512$, and 1mm inter-slice spacing. Example images are briefly sketched in Figure 3.1.

2. CT COVID-19 [220]: A dataset containing CT scans of patients affected by COVID-19 from MedSeg, a commercial AI medical company. The dataset consists of 20 CT scans with up to 630 slices per scan and is aimed at improving the detection and analysis of COVID-19 related lung abnormalities. Example images are briefly sketched in Figure 3.2.

3. MRI Cardiac [8]: A dataset comprises cardiac MRI scans that include annotations for various cardiac structures. The dataset consists of data collected from 100 patients, totaling nearly 6,000 images. These images represent diverse feature information distributions across five subgroups: normal, myocardial infarction, dilated cardiomyopathy, hypertrophic cardiomyopathy, and abnormal right ventricle. Example images are briefly sketched in Figure 3.3.

4. MRI Brain Tumor [9]: A dataset of multi-modal MRI scans of brain tumor patients, containing annotations for different types of brain tumors. The dataset is aimed at enhancing the understanding and treatment of brain tumors. It contains routine clinically-acquired 3T multimodal MRI scans, with

accompanying ground-truth masks annotated by neuro-radiologists. Example images are briefly sketched in Figure 3.4.

5. Ultrasound Nerve [221]: A dataset of ultrasound images with annotations for peripheral nerves. Identifying nerve structures in ultrasound images is critical for inserting a patient's pain management catheter. Example images are briefly sketched in Figure 3.5.
6. Histology Nuclei [169]: A dataset of digitized histological sections with annotations for cell nuclei from the University of Warwick. The dataset is aimed at enhancing the understanding and analysis of various cancer types. Example images are briefly sketched in Figure 3.6.

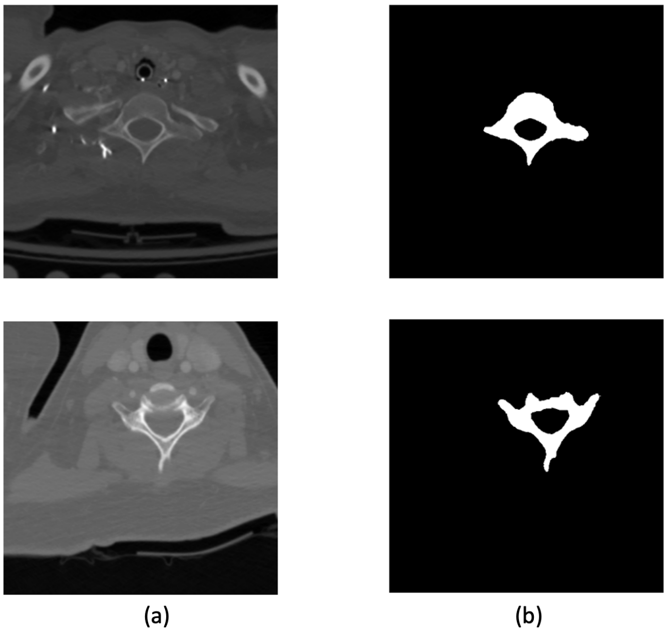


**Figure 3.1:** Example CT Spine Images with Segmentation Ground Truth. The White Pixels Representing the Spine, and the Black Pixels Representing Background.

These diverse datasets provide a comprehensive platform to evaluate and compare the performance of various medical image segmentation methods. By analyzing the performance of the techniques on these datasets, we can better understand their strengths and weaknesses and determine their suitability for specific clinical applications.

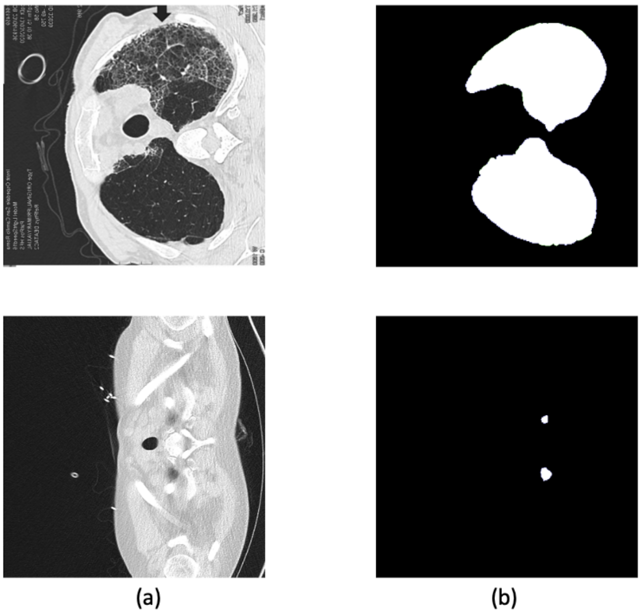


**Figure 3.2:** Example CT COVID-19 Images with Segmentation Ground Truth. The White Pixels Representing the Lung, and the Black Pixels Representing Background.

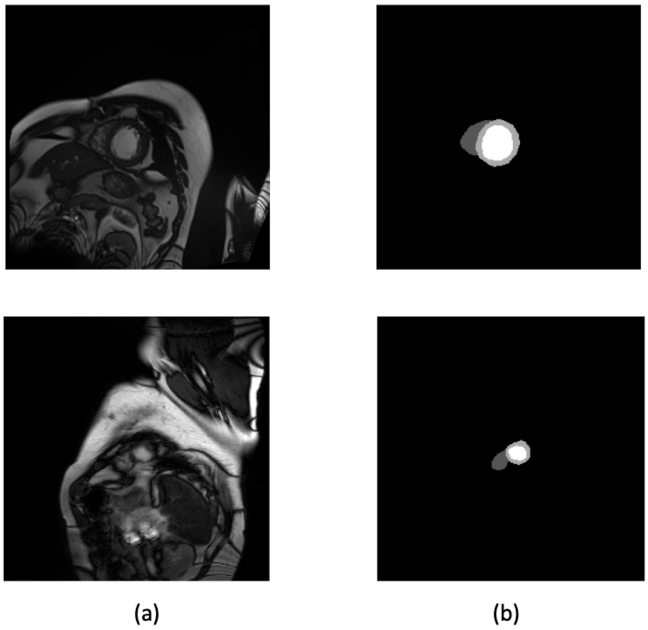


**Figure 3.3:** Example MRI Cardiac Images with Segmentation Ground Truth. The White Pixels Representing the Right Ventricle (RV), Myocardium (Myo), and Left Ventricle (LV), and the Black Pixels Representing Background.

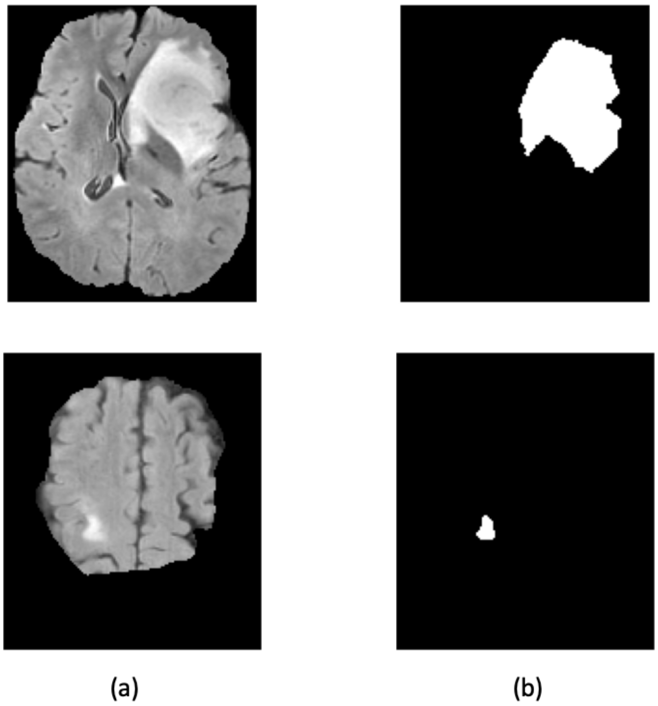


**Figure 3.4:** Example MRI Brain Tumor Images with Segmentation Ground Truth. The White Pixels Representing the Brain Tumour, and the Black Pixels Representing Background.

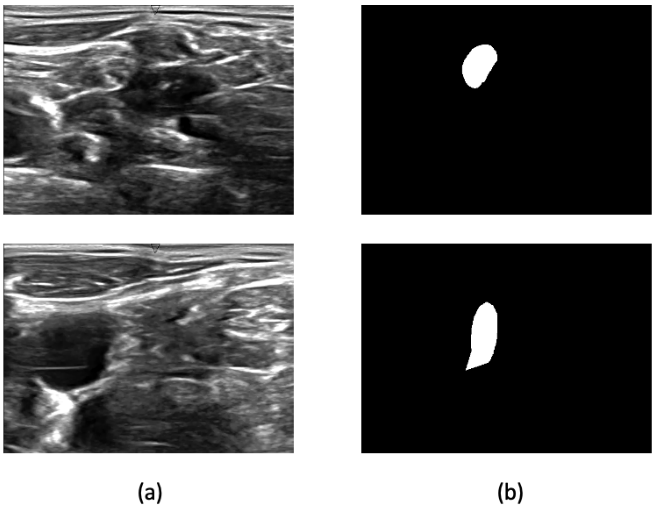


**Figure 3.5:** Example Ultrasound Nerve Images with Segmentation Ground Truth.

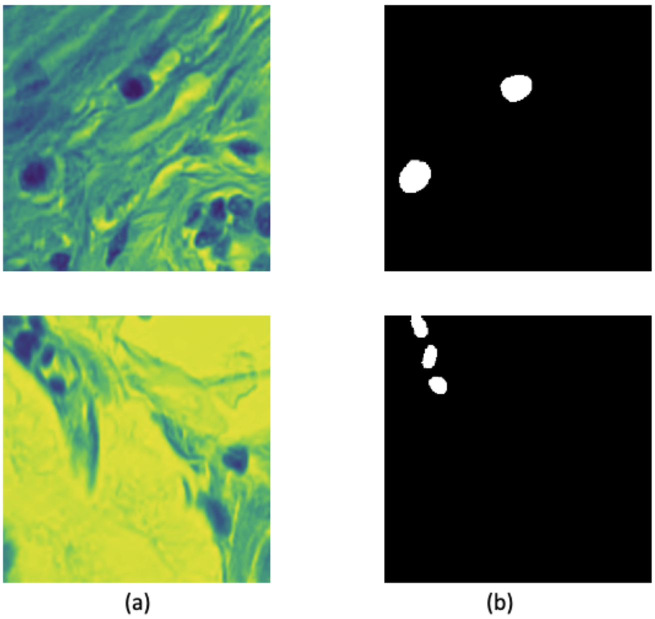


**Figure 3.6:** Example Histology Nuclei Images with Segmentation Ground Truth.

## 3.3 Computational Platform

To conduct the experiments presented in this thesis, we utilized a self-built dedicated deep learning workstation (Seen in Figure 3.7) specifically designed for efficient execution of medical image segmentation tasks. The hardware components of this workstation include:

**CPU**: An Intel Core i9-10900K processor with 10 cores and 20 threads, providing a base clock speed of 3.7 GHz and a turbo boost frequency of 5.3 GHz. This high-performance processor ensures smooth execution of pre-processing, post-processing, and other non-GPU tasks.

**GPU**: Four NVIDIA GeForce RTX 3090 graphics cards, each featuring 24 GB of GDDR6X memory and 10,496 CUDA cores. These powerful GPUs enable the efficient training and evaluation of deep learning networks for medical image segmentation.

**RAM**: 64 GB of DDR4 memory with a speed of 3200 MHz, ensuring sufficient memory capacity for handling large medical datasets and facilitating efficient data loading during network training and evaluation.

**Storage**: A 2 TB NVMe SSD for the operating system and essential software, as well as a 4 TB SATA SSD for storing medical imaging datasets and network checkpoints. This storage configuration ensures rapid data access and smooth system operation.

**Cooling**: A high-performance liquid cooling system to maintain optimal temperatures for the CPU and GPUs, preventing thermal throttling and ensuring the stability and longevity of the components.

**Power Supply**: One 2000W and one 750W 80 PLUS Platinum certified Power Supply Unit(PSU), providing stable and efficient power delivery to all components of the workstation.

The deep learning workstation's hardware configuration is selected to ensure the efficient execution of medical image segmentation tasks, allowing for rapid experimentation and comprehensive evaluations. By using this powerful computational platform, we are able to test various deep learning techniques on diverse medical imaging datasets, enabling a thorough assessment of their performance in real clinical scenarios. The running times relevant to each experiment are reported in each Chapter.

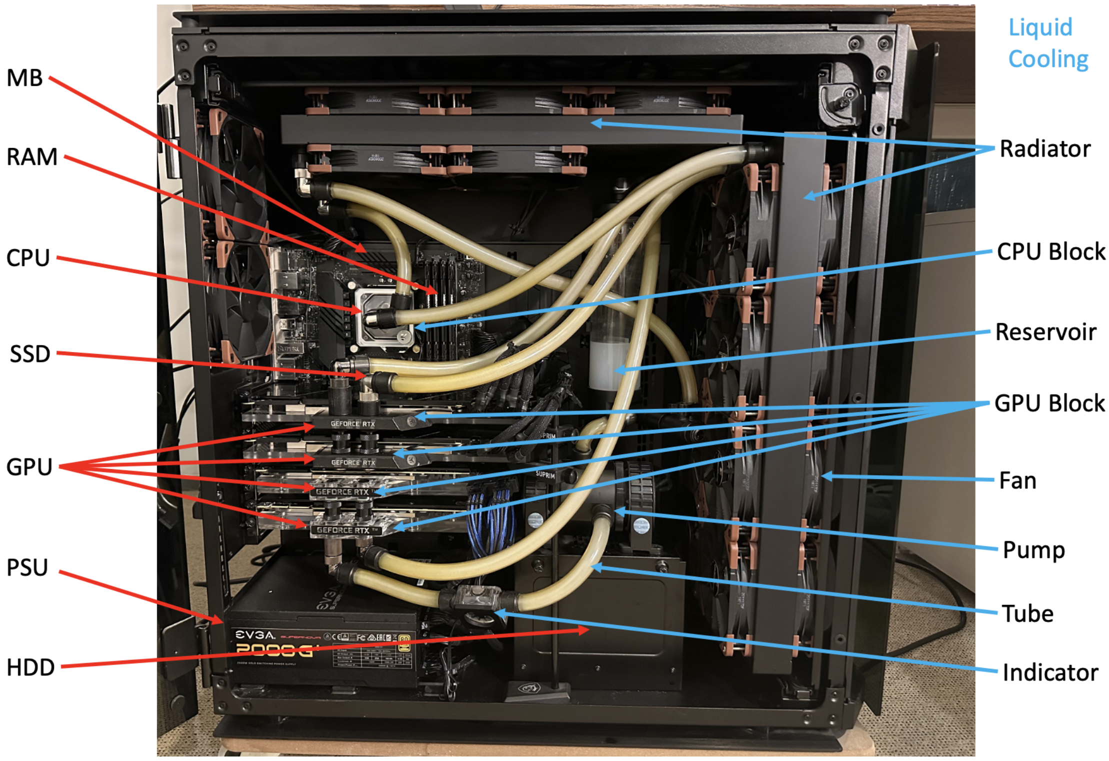


**Figure 3.7:** GPU Workstation as Computational Resources for Medical Image Segmentation Triathlon.

## 3.4 Practical Segmentation Annotations Simulation Scenarios

In this section, we introduce the data preprocessing approaches to adapt various medical imaging datasets to the different data scenarios addressed in this thesis. These scenarios includes comprehensive data annotation with full dense masks, partial data annotation with full dense masks, comprehensive data annotation with noisy dense masks, and comprehensive data annotation with noisy dense masks. Notably, all these data scenarios are simulated once and then used to validate all baseline methods and proposed method to ensure a fair comparison. The simulation algorithms are developed to align with real-world clinical settings.

### 3.4.1 Comprehensive Data Annotation with Full Dense Masks

For conventional sufficient annotated data, the preprocessing steps focus on ensuring that the data is clean, well-organized, and ready for use in supervised learning tasks. The following preprocessing steps are performed:

**Image normalization**: The intensities of the medical images are normalized to a common range (e.g., [0, 1]), ensuring that the values can be effectively processed by the deep learning network.

**Data augmentation**: To improve the network's generalization capabilities, various data augmentation techniques, such as rotation, scaling, and flipping, are randomly applied to the original images and their corresponding annotations. All images are resized to 256 × 256 for CNN, and resized to 224 × 224 to align with ViT fashion.

**Splitting the dataset**: The dataset is split into training, validation, and testing subsets to ensure that the network can be effectively trained, fine-tuned, and evaluated on separate data. Images are randomly selected and there is no overlap between the training, validation, and testing subsets. The randomly selection is conducted only once when comparing the proposed methods with all baseline methods for a fair comparison.

### 3.4.2 Partial Data Annotation with Full Dense Masks

For the scenario involving massively unannotated data, the preprocessing steps aim to prepare both the limited annotated data and the large volume of unannotated data for semi-supervised learning tasks as labeled data. In addition to the steps performed for conventional sufficient annotated data, the following preprocessing steps are performed:

**Splitting the dataset**: We randomly select different ratios of the training set to split as labeled and unlabeled subsets such as 5%, 10% or 20% of training data as labeled data. This allows us to explore the performance of the semi-supervised learning methods under various proportions of labeled and unlabeled data.

### 3.4.3 Comprehensive Data Annotation with Noisy Dense Masks

For the scenario with noisy annotated data, the preprocessing steps focus on introducing noise to the dataset in a controlled manner. In addition to the steps performed for conventional sufficient annotated data, the following preprocessing steps are performed:

**Noise introduction**: We manually add noise to the ground truth annotations by applying morphological operations such as erosion and dilation, as well as elastic transformation. This simulates the presence of annotation noise in real-world clinical scenarios, following our previous work [17, 31]. Example annotated data with noise is shown in Figure 1.1.

**Splitting the dataset**: In order to maintain the authenticity of our evaluation, the dataset is partitioned such that the noisy annotations are restricted to the training and validation sets, while the testing set retains the original, noise-free annotations. This allows for a fair comparison of our proposed noise-robust learning strategies against other methods under realistic circumstances.

### 3.4.4 Comprehensive Data Annotation with Sketchy Contours

For the scenario with scribble-based annotated data, the preprocessing steps focus on generating sparse scribble annotations from the provided perfect ground truth dense annotations. In addition to the steps performed for conventional sufficient annotated data, the following preprocessing steps are performed:

**Scribble generation**: We manually generate scribbles by using an automatic software that creates sparse annotations based on the provided perfect ground truth dense annotations. These scribbles serve as the input for weakly-supervised learning tasks. Example annotated data with scribble is sketched in Figure 1.2. The scribble generation is following our previous work [222].

**Splitting the dataset**: The dataset is divided such that the sparse scribble annotations are allocated to the training and validation sets, while the testing set retains the original, dense annotations. This setup ensures an unbiased evaluation of our proposed weakly-supervised learning strategies, as it mirrors the conditions faced in real-world applications.

## 3.5 Evaluation Metrics

In the general evaluation of deep learning-based methods for medical image segmentation tasks, $\mathbf{D}_{train}, \mathbf{D}_{test}$ normally denote as a training set, and a test set. A batch of labeled training set is denoted as $(\boldsymbol{X}_{\mathrm{l}}, \boldsymbol{Y}_{\mathrm{gt}}) \in \mathbf{D}_{train}$, a batch of testing set as $(\boldsymbol{X}_{\mathrm{t}}, \boldsymbol{Y}_{\mathrm{gt}}) \in \mathbf{D}_{test}$, where $\boldsymbol{X}_{\mathrm{l}}, \boldsymbol{X}_{\mathrm{t}} \in \mathbb{R}^{h \times w}$, and $\boldsymbol{Y}_{\mathrm{gt}} \in [0,1]^{h \times w}$ represent 2D grey-scale images, and their corresponding ground-truth annotations, respectively. A prediction $\boldsymbol{Y}_{\mathrm{p}} \in [0,1]^{h \times w}$ is generated by a segmentation network $f(\theta) : \boldsymbol{X} \mapsto \boldsymbol{Y}_{\mathrm{p}}$ using the parameters $\theta$ of the network $f$. The pair of $(\boldsymbol{Y}_{\mathrm{gt}}, \boldsymbol{Y}_{\mathrm{p}})$ on $\mathbf{D}_{train}$ can be used to update the parameter $\theta$ as network $f$ training, and the evaluation metrics is to validate the network by calculating the difference of pair $(\boldsymbol{Y}_{\mathrm{gt}}, \boldsymbol{Y}_{\mathrm{p}})$ on $\mathbf{D}_{test}$. This section divides evaluation methods into qualitative analysis, region-based metrics, and boundary-based metrics.

### 3.5.1 Qualitative Analysis

The qualitative analysis allows us to visually assess the performance of the methods and better understand their strengths and weaknesses.

The qualitative analysis involves the visualization of example images, ground truth annotations, and segmentation results generated by the deep learning networks. In these visualizations, we use the following color scheme to facilitate the comparison between the inference and the ground truth:

- **Yellow**: True positive (TP) – correctly segmented pixels that belong to the target region.
- **Red**: False positive (FP) – segmented pixels that do not belong to the target region.
- **Green**: False negative (FN) – missed pixels that should have been segmented as part of the target region.
- **Black**: True negative (TN) – correctly identified background pixels.

An example of qualitative analysis is illustrated in Figure 3.8 where yellow, red, green, and black demonstrates the quality of inference of network. The prediction is multi-class classification, i.e. glass, consolidation, lung other, and background, respectively. By visualizing the segmentation results using this color scheme, we can easily identify the areas where the network performs well and the regions where it struggles. Such qualitative analysis provides valuable insights into the behavior of the segmentation methods, complementing the quantitative metrics, and helping to guide the improvement of the networks for better performance in real clinical scenarios.

### 3.5.2 Region-based Evaluation Metrics

Region-based evaluation metrics focus on the similarity between the predicted and ground truth segmented regions. To simplify the expression of different evaluation metrics, several parameters and notations are defined in detail. $k+1$ classes of target Organ $O$ areas (e.g. $O_0$ refers to a lung, $O_1$ refers to a kidney, $O_k$ refers to background and etc.), and the total of Pixels $P$ ($P_{ij}$ refers to the number of pixels organ $O_i$ are predicted to belong to the class of organ $O_j$) are considered. In other words, $P_{ii}$ refers to the TP number of pixels. $P_{ij}$ refers to the FP number of pixels. $P_{ji}$ refers to the FN number of pixels. $P_{jj}$ refers to the TN number of pixels.

Pixel Accuracy (PA) is a ratio between the number of correctly classified pixels and the total number of pixels.

$$PA = \frac{\sum_{i=0}^{k} P_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} P_{ij}} \tag{3.1}$$

Mean Pixel Accuracy (MPA) is a average ratio for all classes of organs between the number of correctly classified pixels and the total number of pixels.

$$MPA = \frac{1}{k+1}\sum_{i=0}^{k}\frac{P_{ii}}{\sum_{j=0}^{k} P_{ij}} \tag{3.2}$$

Pixel Precision (PP) is a ratio between the number of correctly classified positive pixels and the total number of positive predicted pixels.

$$PP = \frac{\sum_{i=0}^{k} P_{ii}}{\sum_{i=0}^{k}(P_{ii} + P_{ij})} \tag{3.3}$$

Mean Pixel Precision (MPP) is a average ratio for all classes of organs between the number of correctly classified positive pixels and the total number of positive predicted pixels.

$$MPP = \frac{1}{k+1}\sum_{i=0}^{k}\frac{P_{ii}}{P_{ii} + P_{ij}} \tag{3.4}$$

Pixel Recall (PR) is the proportion of boundary pixels in the ground truth that are correctly classified by the segmentation.

$$PR = \frac{\sum_{i=0}^{k} P_{ii}}{\sum_{i=0}^{k} P_{ii} + \sum_{j=0}^{k} P_{jj}} \tag{3.5}$$

Mean Pixel Recall (MPR) is the average ratio for all classes of organs between the number of true positive pixels and the sum of true positive and false negative pixels.

$$MPR = \frac{1}{k+1}\sum_{i=0}^{k}\frac{P_{ii}}{P_{ii} + P_{jj}} \tag{3.6}$$

Pixel Specificity (PS) is the ratio between the number of true negative pixels and the sum of true negative and false positive pixels.

$$PS = \frac{\sum_{i=0}^{k} P_{jj}}{\sum_{i=0}^{k}(P_{jj} + P_{ij})} \tag{3.7}$$

Mean Pixel Specificity (MPS) is the average ratio for all classes of organs between the number of true negative pixels and the sum of true negative and false positive pixels.

$$MPS = \frac{1}{k+1}\sum_{i=0}^{k}\frac{P_{jj}}{P_{jj} + P_{ij}} \tag{3.8}$$

Dice Coefficient (DSC) also known as Sørensen–Dice index, is a statistic used to gauge the similarity of two boundary.

$$DSC = \frac{2 * \sum_{i=0}^{k} P_{ii}}{2 * \sum_{i=0}^{k} P_{ii} + \sum_{i=0}^{k}\sum_{j=0}^{k}(P_{ij} + P_{ji})} \tag{3.9}$$

Mean Dice Coefficient (MDSC) is the average Dice Coefficient for all classes of organs, providing a measure of the overall similarity between the predicted and ground truth segmentation across all organ classes.

$$MDSC = \frac{1}{k+1}\sum_{i=0}^{k}\frac{2P_{ii}}{2P_{ii}+\sum_{j=0}^{k}(P_{ij}+P_{ji})} \tag{3.10}$$

Intersection Over Union (IoU) also known as Jaccard index, is the percent overlap between the target mask and the prediction output.

$$IoU = \frac{\sum_{i=0}^{k} P_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} P_{ij} - \sum_{j=0}^{k} P_{jj}} \tag{3.11}$$

Mean Intersection Over Union (MIoU) is the average percent overlap for each class between the target mask and the prediction output.

$$MIoU = \frac{1}{k+1}\sum_{i=0}^{k}\frac{P_{ii}}{\sum_{j=0}^{k} P_{ij} + \sum_{j=0}^{k} P_{ji} - P_{ii}} \tag{3.12}$$

### 3.5.3 Boundary-based Evaluation Metrics

Boundary-based evaluation metrics focus on the accuracy of the boundaries in the segmented regions. These metrics provide an assessment of how well the segmentation methods capture the precise contours of the target structures:

Hausdorff Distance (HD) is a metric that calculates the maximum distance between the boundaries of the predicted and ground truth segmentation. It measures the worst-case similarity between the boundaries, providing an indication of the maximum error in the segmentation.

$$HD = max(h(P, G), h(G, P)) \tag{3.13}$$

where $h(P, G)$ and $h(G, P)$ are the directed Hausdorff distances, computed as:

$$h(P, G) = \max_{p \in P} \min_{g \in G} d(p, g) \tag{3.14}$$

$$h(G, P) = \max_{g \in G} \min_{p \in P} d(g, p) \tag{3.15}$$

Here, $P$ and $G$ represent the boundaries of the predicted and ground truth segmentation, respectively, and $d(p, g)$ denotes the Euclidean distance between points $p$ and $g$.

Average Surface Distance (ASD) is the mean distance between the boundaries of the predicted and ground truth segmentation. This metric provides an estimate of the overall discrepancy between the segmented boundaries.

$$ASD = \frac{1}{2}(a(P, G) + a(G, P)) \tag{3.16}$$

where $a(P, G)$ and $a(G, P)$ are the directed average surface distances, computed as:

$$a(P, G) = \frac{1}{|P|} \sum_{p \in P} \min_{g \in G} d(p, g) \tag{3.17}$$

$$a(G, P) = \frac{1}{|G|} \sum_{g \in G} \min_{p \in P} d(g, p) \tag{3.18}$$

In this case, $|P|$ and $|G|$ represent the number of points on the boundaries of the predicted and ground truth segmentation, respectively.

Relative Volume Difference (RVD) is a metric that measures the discrepancy in the segmented region volumes between the predicted and ground truth segmentation. It provides an assessment of the overall volume error in the segmented regions and is useful for evaluating the segmentation methods in terms of volumetric consistency.

$$RVD = \frac{V_P - V_G}{V_G} \tag{3.19}$$

Here, $V_P$ and $V_G$ represent the volumes of the predicted and ground truth segmentation, respectively. A lower RVD value indicates better volume agreement between the predicted and ground truth segmentation. Negative RVD values signify that the predicted segmentation has a smaller volume compared to the ground truth, while positive RVD values indicate a larger predicted segmentation volume.

Directed Boundary-based method as novel evaluation metrics from previous work [223] have also been utilized and reported including Directed Boundary Dice relative to GT (DBD$_G$), Directed Boundary Dice relative to MS (DBD$_M$) and Symmetric Boundary Dice (SBD) In a von Neumann neighbourhood $N_x$ of each pixel $x$ on the boundary $\partial G$ of the ground truth or machine segmentation $\partial M$,

$$DBD_G = \frac{\sum\limits_{x \in \partial G} \mathrm{Dice}(N_x)}{|\partial G|} \tag{3.20}$$

$$DBD_M = \frac{\sum\limits_{x \in \partial M} \mathrm{Dice}(N_x)}{|\partial M|} \tag{3.21}$$

$$SBD = \frac{\sum\limits_{x \in \partial G} Dice(N_x) + \sum\limits_{y \in \partial M} Dice(N_y)}{|\partial G| + |\partial M|} \tag{3.22}$$

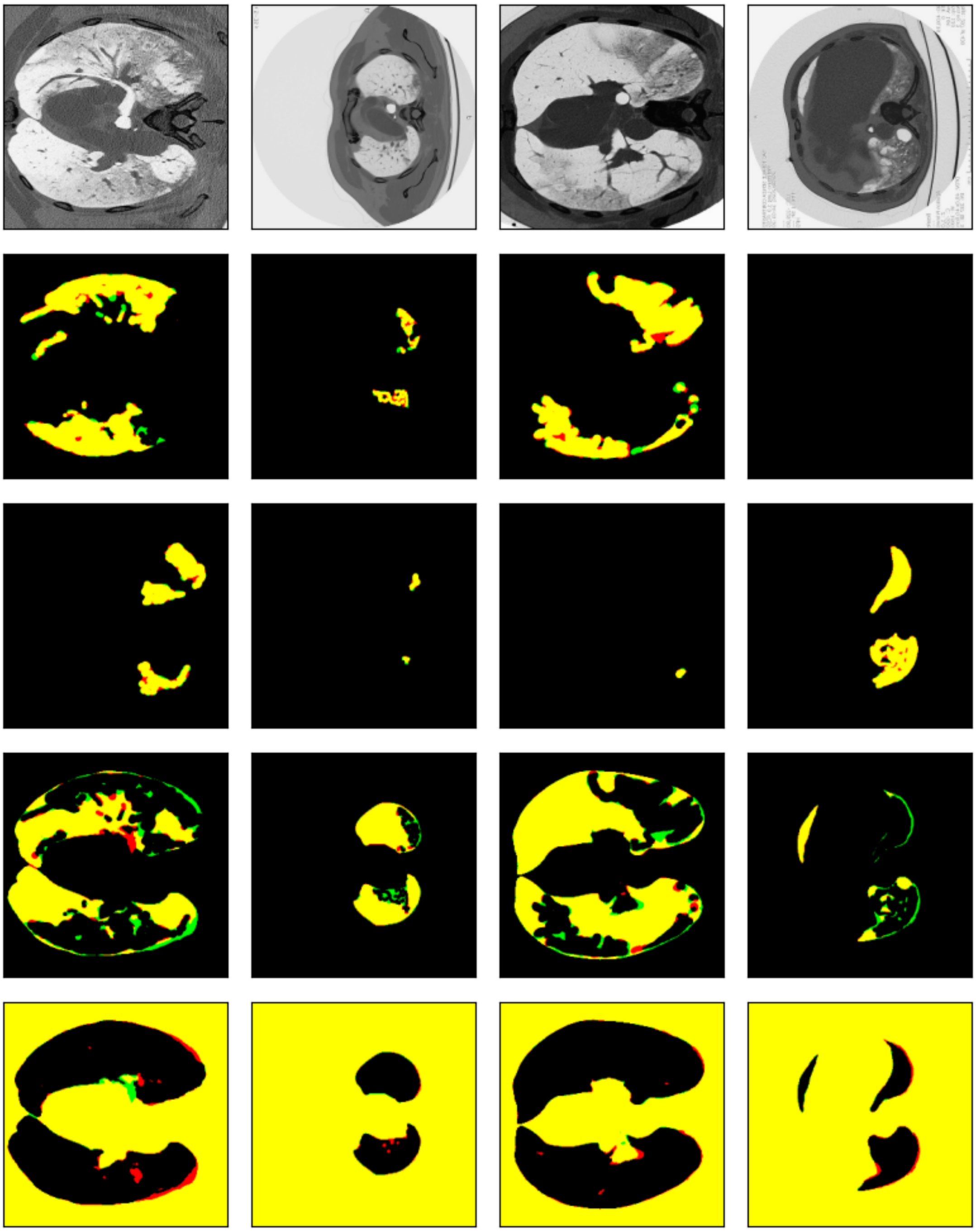


**Figure 3.8:** Example CT COVID-19 Segmentation Multi-Class Inference of a Network Against Ground Truth with TP, TN, FP, and FN Pixels on Consolidation, Glass, Lung Other, and Background.

# 4

# Supervised Learning: Comprehensive Data Annotation with Full Dense Masks

## Contents

**4.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . 53**
**4.2 Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55**
  4.2.1 Supervised Learning Framework Setup . . . . . . . . . . 55
  4.2.2 Segmentation Backbone Network . . . . . . . . . . . . . 55
  4.2.3 Network Block . . . . . . . . . . . . . . . . . . . . . . . 57
  4.2.4 Study of Supervised Learning Segmentation . . . . . . . 66
**4.3 Experiments and Results . . . . . . . . . . . . . . . . . . . 72**
  4.3.1 Implementation Details . . . . . . . . . . . . . . . . . . 72
  4.3.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . 72
  4.3.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . 72
  4.3.4 Ablation Study . . . . . . . . . . . . . . . . . . . . . . . 73
**4.4 Contribution and Discussion . . . . . . . . . . . . . . . . . 79**

## 4.1 Motivation

Supervised learning is one of the most popular studies that leverage deep learning in the medical image segmentation. These techniques rely on a large number of medical images with corresponding high-quality precise annotated data. Supervised learning strategy can make network learn from data during the training process and apply the learned knowledge to unseen testing data, facilitating accurate and precise segmentation.

This thesis, however, claims that obtaining high-quality annotations for medical images can be a labor-intensive and costly process. It involves expert radiologists and physicians manually label the ROI on pixel level, which can often be time-consuming given the complexity and volume of the medical images. Moreover, the process can be prone to inter- and intra-observer variability, introducing noise into the labeled data. Despite these challenges, the reliance on supervised learning approaches continues due to their effectiveness and the richness of the information that labeled data provide.

In this chapter, we study on the supervised learning for medical image segmentation. We aim to explore and enhance the capabilities of these techniques of neural network architecture engineering in this conventional scenario. We reproduce various advanced methods including backbone networks, network blocks, and training strategies, evaluate their performance, and propose new modified networks to improve segmentation performance.

## 4.2 Methods

### 4.2.1 Supervised Learning Framework Setup

In the task of supervised learning, $\mathbf{D}_{train}$, $\mathbf{D}_{test}$ normally denote precise and perfect labeled training set, and testing set. We denote a batch of labeled data as $(\boldsymbol{X}, \boldsymbol{Y}_{\mathrm{gt}}) \in \mathbf{D}_{train}$, $(\boldsymbol{X}, \boldsymbol{Y}_{\mathrm{gt}}) \in \mathbf{D}_{test}$, where $\boldsymbol{X} \in \mathbb{R}^{h \times w}$ representing a 2D image with the size $h \times w$, and $\boldsymbol{Y}_{\mathrm{gt}} \in [0,1]^{h \times w \times c}$ representing the annotation on each pixel whether 0 is background and 1 is ROI, and $c$ is the number of classes of ROI. $\boldsymbol{Y}_{\mathrm{p}}$ is the dense map predicted by the segmentation network $f(\theta) : \boldsymbol{X} \mapsto \boldsymbol{Y}_{\mathrm{p}}$. $\mathcal{L}_{\mathrm{sup}} : (\boldsymbol{Y}_{\mathrm{p}}, \boldsymbol{Y}_{\mathrm{gt}}) \mapsto \mathbb{R}$ represents supervised segmentation loss. In general, the training is to update the parameter $\theta$ of segmentation network $f(\theta)$ aiming to minimize the loss $\mathcal{L}_{\mathrm{sup}}$ on training set $\mathbf{D}_{train}$. The final evaluation is to calculate the difference of $(\boldsymbol{Y}_{\mathrm{p}}, \boldsymbol{Y}_{\mathrm{gt}}) \mapsto \mathbb{R}$ on testing set.

### 4.2.2 Segmentation Backbone Network

Following the literature review of neural network architecture engineering in Chapter 2, the supervised learning with segmentation networks in this chapter are primarily revolve around Convolution Neural Network (CNN) [3, 4, 17, 194] and Vision Transformer (ViT) [39, 79, 182, 186] based U-shape Encoder-Decoder style networks with the exploration of various advanced network blocks to further improve performance. Both of CNN and ViT layers with Encoder-Decoder style network have demonstrated remarkable success in image segmentation due to their ability to learn hierarchical representations and capture long-range dependencies in the data. The related U-shaped Encoder-Decoder style segmentation networks are illustrated in Figure 4.1.

**CNN-based Segmentation Network**

CNN, one of the fundamental neural network architecture, has been dominated in medical image segmentation. Our research leverage the power of these networks, specifically employing U-shaped encoder-decoder style segmentation networks. Inspired by the CNN-based UNet [4] and its modifications [175, 180–182],

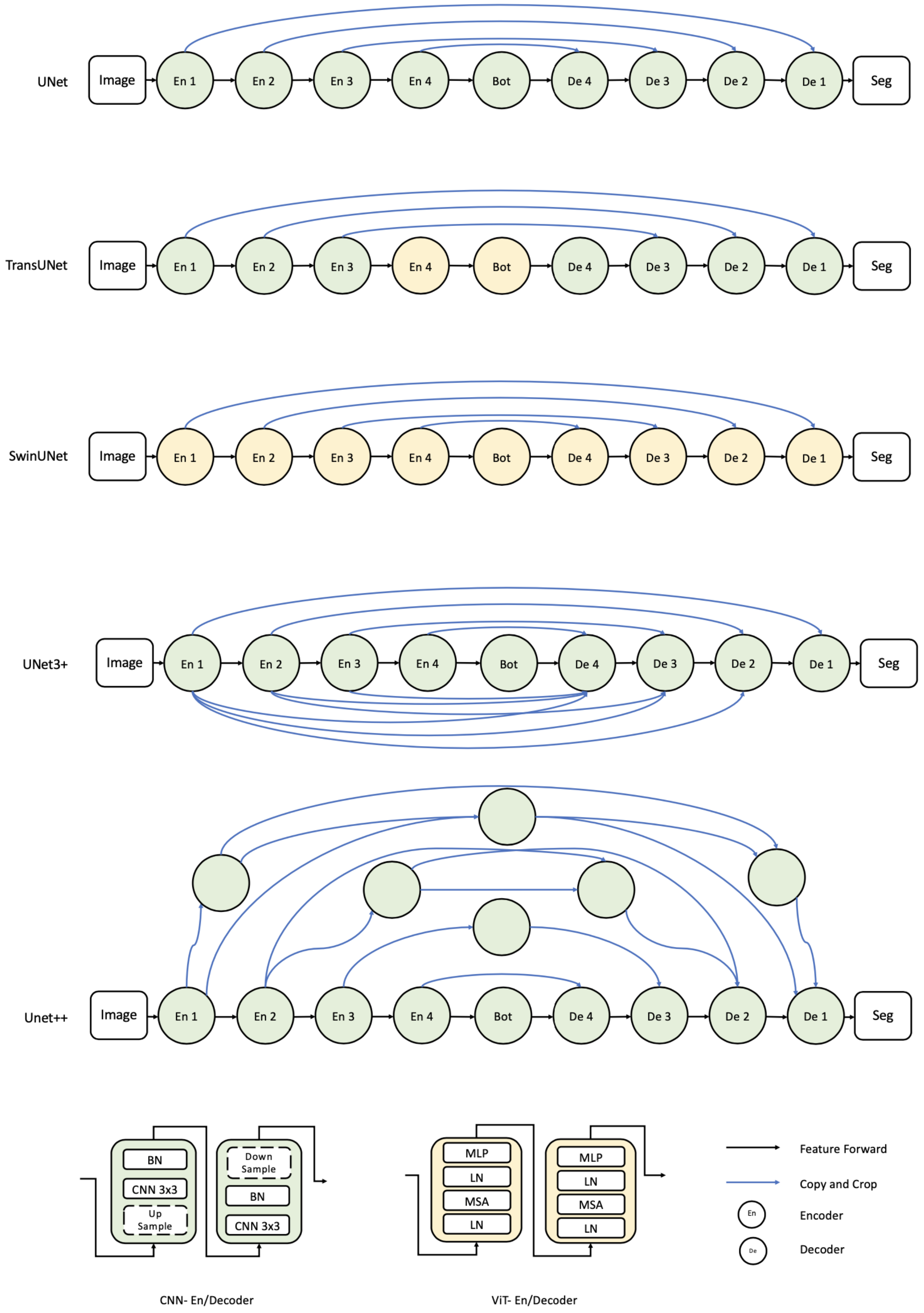


**Figure 4.1:** The Conventional U-shaped Encoder-Decoder Segmentation Backbone Network with Various Modified Networks developed by CNN or ViT Network Blocks, including UNet, TransUNet, SwinUNet, Unet3+, and UNet++. The green CNN-based block consists of two successive CNN layers. The yellow ViT-based block consists of two successive ViT layers.

these architectures have demonstrated promising performance in medical image segmentation tasks. An example of 2 successive CNN layers including convolutional operations, batch normalization, and dropout for each level of Encoder and Decoder is sketched in the bottom of Figure 4.1.

**ViT-based Segmentation Network**

Recently, ViT has gradually been explored and outperforms CNN in computer vision and medical image segmentation [186]. Different with CNN leverage local spatial correlations within images, ViT is originally designed to model long-range dependencies for sequence-to-sequence tasks [39, 79, 186]. In ViT network study, the image can be considered as a sequence of patches, capturing global dependencies through ViT's self-attention scheme. We expand ViT by exploring the application of ViT-based networks with same U-shape Encoder-Decoder network for medical image segmentation. An example of 2 successive ViT layers including shift-window-based multi-head self-attention, layer normalization, and multi-layer perception for a Encoders and Decoders is sketched in the bottom of Figure 4.1. All of the experimental design in this chapter incorporates the prevalent U-shape style Encoder-Decoder segmentation network architecture, as illustrated in Figure 4.1. This is done to ensure a fair comparison across experiments with the same architecture of backbone network. Each level of the encoder or decoder is constructed using either 2 successive CNN layers comprising of 3 × 3 convolutional operations, batch normalization, and dropout, or 2 successive ViT layers, which include layer normalization, multi-head self-attention, and multi-layer perception.

Figure 4.1 provides a brief sketch of example UNet [4] and with several modified UNets which either based on CNN or ViT including TransUNet [181], SwinUNet [182], UNet++ [175], and UNet3+ [180].

### 4.2.3 Network Block

Network block which is an architectural engineering strategies that can be integrated into the segmentation backbone network, leading to a substantial enhancement of feature learning performance. These blocks normally includes residual learning

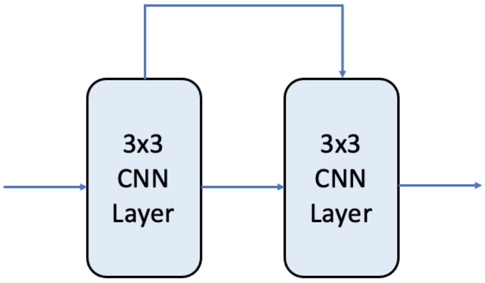


**Figure 4.2:** A Residual Connection Between 2 CNN Layers.

[56, 155, 161, 224], densely connections [75, 147, 214, 225], attention mechanisms [38, 82, 85, 178], pyramid pooling [44, 194] and etc. This section explores classical network blocks and introduces our proposed novel network blocks, which is from the part of our past work [17, 31, 167, 168, 194].

### Residual Connections

Inspired by residual learning from ResNet [56], we incorporate a concatenation function within each 2 successive CNN layer block of the segmentation network [17]. This architectural modification, as illustrated in Figure 4.2, aims to augment the network's capability to express features and facilitate gradient information propagation. The features from a previous layer are obtained and subsequently processed through a unique feature extraction sequence. This sequence is characterized by the successive application of two $3 \times 3$ unpadded convolutions, each followed by a Rectified Linear Unit (ReLU) [226]. Post the first $3 \times 3$ convolution operation, the number of feature channels is doubled in comparison to the preceding layer. The final step of this process involves concatenating the input feature with the output emanating from the second $3 \times 3$ convolution operation. This operation is tailored to establish intricate interconnections across different layers, thereby enabling the network to sufficient transfer feature information and alleviate the challenge of vanishing gradients. This is part of past work in residual encoder [17].

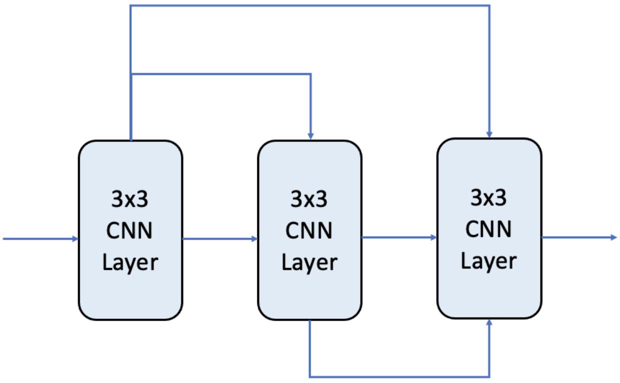


**Figure 4.3:** A Dense Connection Among 3 CNN Layers.

### Densely Connections

Densely connected network (DenseNet) [147] represents an innovative design strategy that has been proven to improve the performance of segmentation networks [162, 227]. Inspired by DenseNet [147], dense connections, as illustrated in Figure 4.3, can be incorporated within multi-layer-based network block as an enhanced feature information flow throughout the network. Within a densely connected block, each layer connects to every other layer in a feed-forward fashion, thus facilitating the direct propagation of features and gradients across layers. Consequently, the network is beneficial with sufficient feature information with deep architectures without being subjected to the common pitfalls, such as the phenomena of vanishing gradients and feature degradation.

### Attention Mechanism

To improve the performance of the pixel-wise decoder classification, an attention mechanism is explored. This is part of past work in attention encoder [17]. In contrast to a traditional attention gate that filters features from skip connections [178], an attention module can be typically incorporated with convolutional layers to enable the CNN focus on key feature information of the feature map [38]. The proposed attention module for the convolutional layer of the decoder is illustrated in Figure 4.4. The module contains two components related to channel

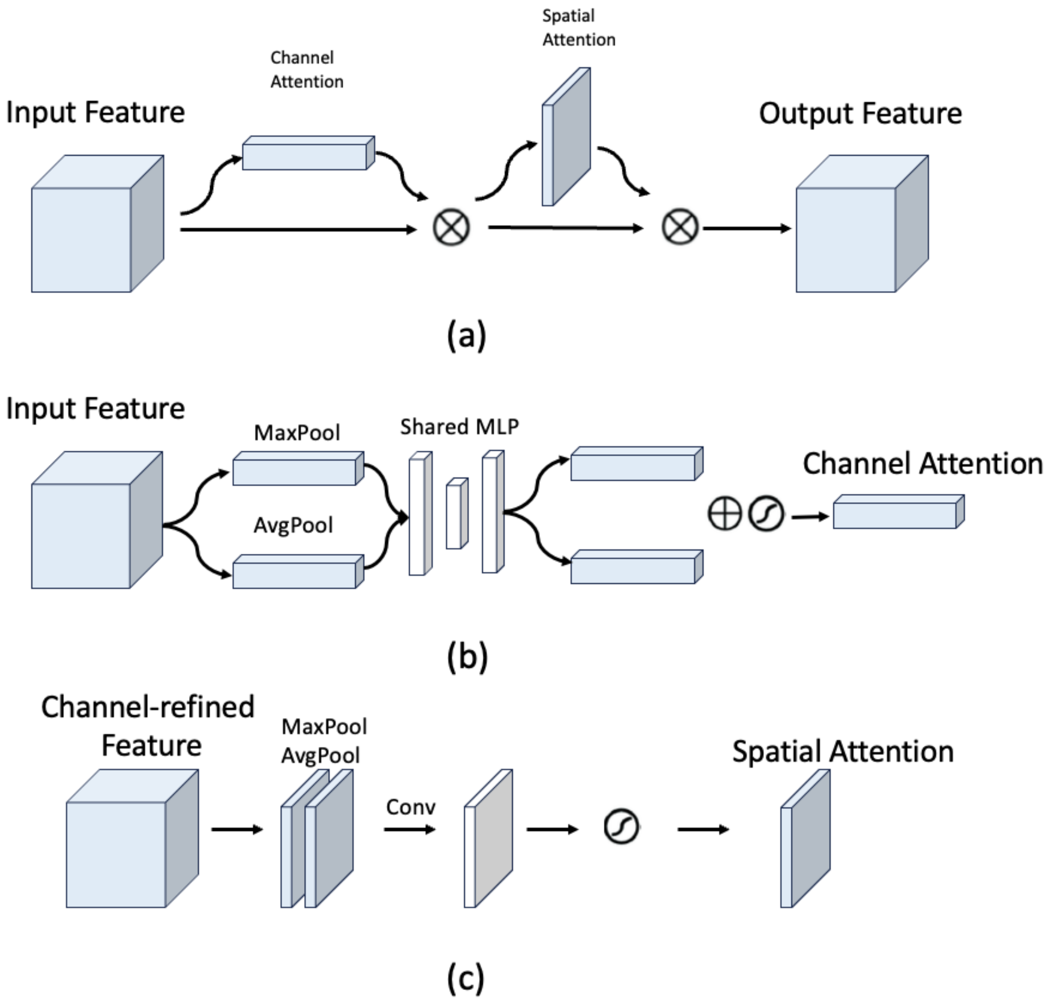


**Figure 4.4:** (a) The Convolutional Attention Network Block. (b) Channel Attention Module. (c) Spatial Attention Module.

and spatial attention of different feature maps. Both components are developed by pooling layers and Sigmoid activation. Average and max pooling layers mitigate the influence of noisy label gradients to maintain the integrity of trunk parameters [17]. The Sigmoid function, concurrently, generates an attention weight value for each pixel location and channel.

As demonstrated in Figure 4.4, a feature map $F \in R^{W \times H \times C}$ with the size of $W \times H \times C$ from a preceding CNN is dispatched to the attention module pipeline. The feature maps derived from average and max pooling layers along the spatial dimensions $W \times H$ are represented as $F^{Avg}_{Spatial}$ and $F^{Max}_{Spatial} \in R^{W \times H \times 1}$. Similarly, $F^{Avg}_{Channel}$ and $F^{Max}_{Channel} \in R^{1 \times 1 \times C}$ denote the feature maps derived from average and

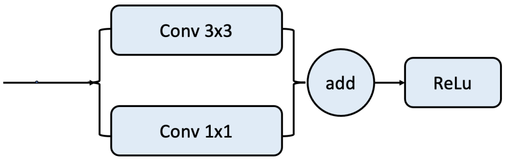


**Figure 4.5:** The Residual Block Path.

max pooling layers across the channel dimension $C$.

Both the spatial attention value $W_{Spa}$ and channel attention value $W_{Channel}$ are computed via Sigmoid activation $\sigma$. The final output feature map $F_{out}$ is adaptively refined from feature map $F$ through successive application of a spatial attention layer and a channel attention layer, thereby capturing essential information as illustrated in Equation 4.1, where $\otimes$ denotes as element-wise multiplication.

$$\begin{aligned} W_{Spa} &= \sigma(F^{Avg}_{Spa} + F^{Max}_{Spa}) \\ W_{Channel} &= \sigma(F^{Avg}_{Channel} + F^{Max}_{Channel}) \\ F_{out} &= W_{Spa}(W_{Channel}(F) \otimes F) \otimes (W_{Channel}(F) \otimes F) \end{aligned} \tag{4.1}$$

### Residual Block Path

This is part of past work in residual connections [17]. To address the significant disparity between encoders and decoders in UNet [4], which could potentially impair segmentation performance, we modify the skip connections by implementing residual block paths. We incorporate residual learning to bridge each layer of the encoder and decoder.

The network block of the residual block path is illustrated in Figure 4.5, which is based on InceptionNet [149], and MultiResUNet [177]. Formally, let $x$ and $y$ represent the input and output vectors of the layers, respectively. The relationship is expressed as:

$$y = F(x, \{W_i\}) + x \tag{4.2}$$

where $F(x, \{W_i\})$ denotes the residual mapping to be learned. For instance, in Figure 4.5, for a layer with a $3 \times 3$ filter, $F$ is represented as $\sigma(W_1 x)$, with $\sigma$

and $W$ indicating the ReLU activation function and weight matrix, respectively. Biases are omitted for simplicity. The operation $F + x$ is executed via a shortcut connection and element-wise addition. Additionally, a $1 \times 1$ convolution is applied to align the channel dimensions.

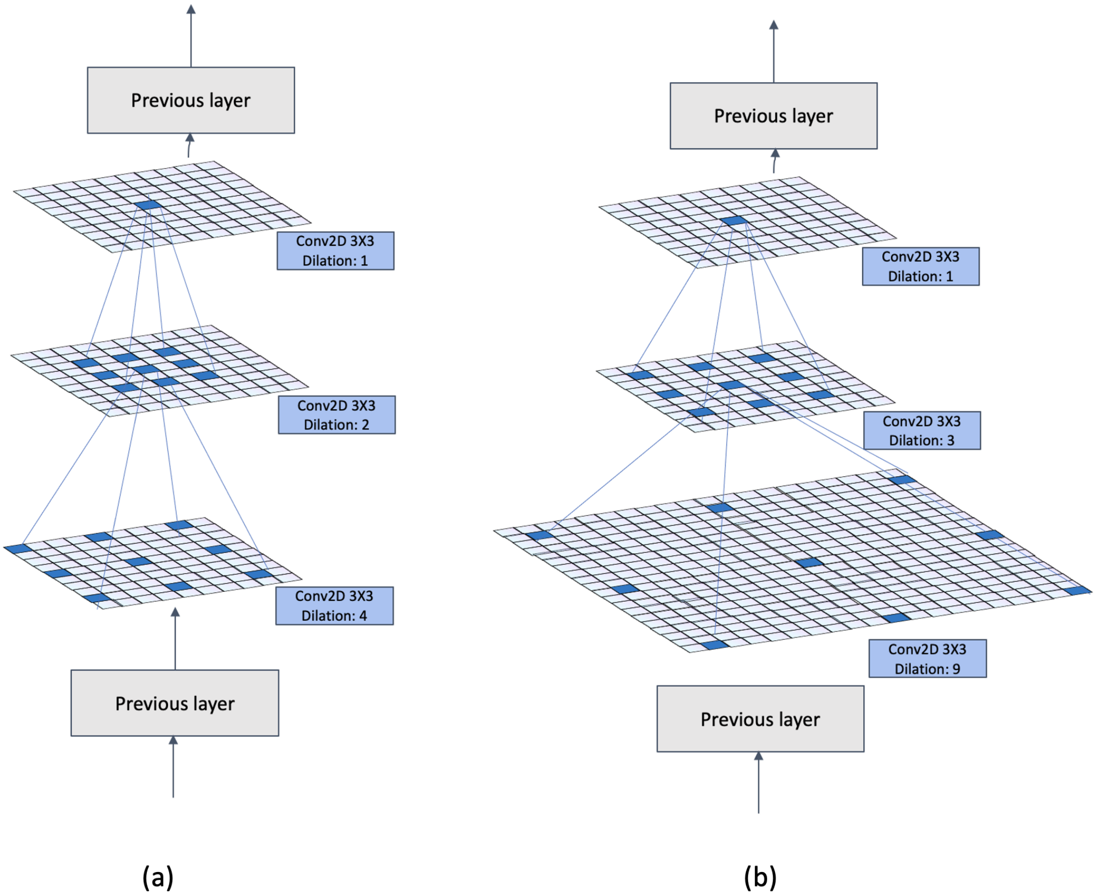


**Figure 4.6:** The Proposed Network Blocks in Pyramid CNN.

## Pyramid CNN

This is part of past work in [194]. Convolutional layers have significantly advanced computer vision tasks. Unlike densely connected neural networks that extract features from all input nodes, CNN selectively extract a limited number of nodes from an input image, known as the Receptive Field (RF). However, multi-layer CNN encounter challenges such as vanishing or exploding gradients, which can be detrimental to semantic segmentation tasks. This is because pixel-level features may not be effectively transferred through multiple CNN layers and downsampling processes. To address this issue and capture features of varying sizes, an alternative approach is the use of Atrous CNN [33, 228]. Atrous CNN increases RF by inserting zeros between non-zeros of filters. These networks increase the RF without additional computational costs by introducing spaces within convolutional filters. Figure 4.6 illustrates an example of multi atrous CNN layers, where the dilation rate is set to [1,2,4] and [1,3,9]. We consider a CNN filter whose size is $f * f$ with setting of Dilation Rate $dr$, the RF can be increased without additional computational cost.

The size of receptive field denoted as $RF * RF$ can be calculated by Equation 4.3.

$$RF = f + (dr - 1)(f - 1) \tag{4.3}$$

The dilation rate setting, however, has not been clearly studied, and Atrous CNN lead to gridding effect [229]. To simplify the relationship between RF and dr study, a one-dimensional (1D) feature map size is selected without considering the use of non-linear modules like ReLU or Sigmoid. In this setup, $F^n$ represents the feature map calculated by the $n^{th}$ Atrous CNN. Here, $F^0$ is the input feature map, while $F^n$ is the output. The RF for the final layer is determined using the formula outlined in Equation 4.4.

$$RF^n = f + (dr^0 - 1)(f - 1) + \sum_{1}^{n} dr^n \tag{4.4}$$

Although the use of Atrous CNN in encoders and decoders can expand the RF without additional computational costs, there is a potential drawback. The insertion of non-trainable zeros into the filter can lead to certain feature nodes not being captured. We can calculate the number of these uncollected nodes, denoted as $UN^n$, after the $n^{th}$ Atrous CNN. This calculation is detailed in Equation 4.5.

$$UN^n = (f - 1) * (dr^0 - n * \sum_{1}^{n} dr^n) \tag{4.5}$$

If the value of $UN^n$ is negative, it implies that no nodes are left uncollected and that some nodes have been processed multiple times.

To assess the effectiveness of the dilation rate setting, an Evaluation Ratio (ER) is established. This ratio, calculated using Equation 4.6, compares the Receptive Field ($RF$) to the number of uncollected nodes ($UN^n$).

$$ER = UN^n / RF^n \tag{4.6}$$

For an optimal configuration of atrous CNN layers, the ER must be minimized while maintaining a fixed number of layers, $n$. This is achieved by integrating Equation 4.4 and Equation 4.5 into Equation 4.6. Consequently, the dilation rate setting, $dr$, should adhere to a geometric progression as outlined in Equation 4.7. This approach aims to maximize the $RF$ size while minimizing the number of $UN^n$, thereby ensuring efficient feature capture with the available atrous CNN layers.

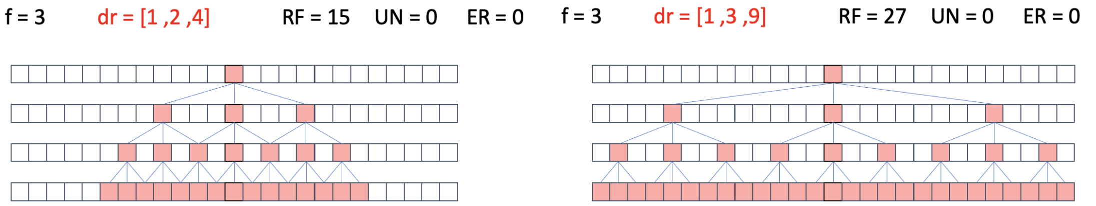
$$dr = [1, ..., (dr)^{n-1}] \tag{4.7}$$

An intuitive example is illustrated in Figure 4.7. The setting of $DR$ with [1,2,4] and [1,3,9] is visualized and compared against the setting of $DR$ with [1,3,4] which potentially results in glidding effects or [1,2,9] which leads to several uncollected feature nodes.

Drawing inspiration from the Inception module [149], we develop an atrous CNN module rooted in a dual-layered atrous CNN pyramid structure. This module aims to bridge the gap between the encoder and decoder components of a network [177], enhancing the transfer of feature information to the decoder. To accommodate various levels of detail in feature extraction, the module incorporates a range of atrous CNN blocks, specifically arranged in a descending order from four to one across the network paths. This configuration allows for a balanced processing of both down-sampled and up-sampled features, ensuring a comprehensive feature representation in the final output.

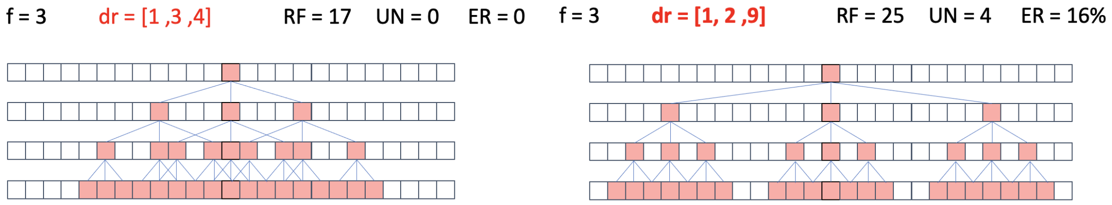
### Pyramid Pooling

Pooling layers are commonly utilized with CNN in various computer vision tasks. The main functions of pooling layers include: (1) providing translation, rotation,



Proper set follow by geometric progression

Proper set follow by geometric progression



Glidding effect

Several feature nodes uncollected by Atrous CNN

**Figure 4.7:** Example Sequence of 1D Atrous CNN Layers with Different Dilation Rate Setting.

and scale invariance; (2) reducing computational costs through down-sampling; (3) preventing overfitting; and (4) enhancing the generalization ability of the network.

Max pooling is favored for its proficiency in extracting prominent features like boundaries and textures by focusing on the maximum pixel values. This approach is especially effective under conditions like noisy labels, varied scene contexts, and strong contrast variations. Conversely, average pooling is adept at minimizing the deviation of the estimated mean, offering better performance in specific scenarios compared to max pooling. Therefore, the proposed pyramid pooling integrates both max and average pooling to capture a comprehensive range of features. The pooling operations vary in size, ranging from $4 \times 4$ to $10 \times 10$, with each output resized by interpolation (except for the $4 \times 4$ layer) to ensure uniformity in feature concatenation along the channel axis. This design as illustrated in Figure 4.8 results in a feature map size reduced to a quarter of the input, effectively capturing diverse feature sizes while maintaining computational efficiency.

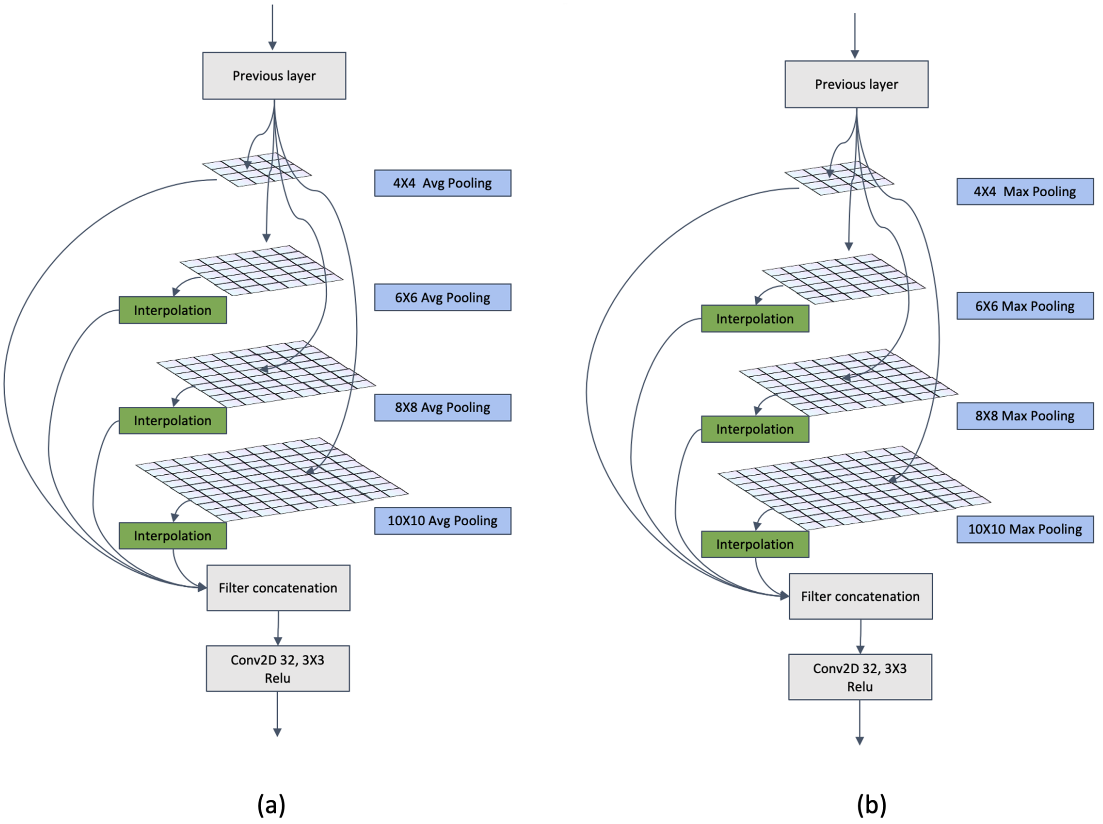


**Figure 4.8:** The Proposed Network Blocks in Pyramid Pooling.

### 4.2.4 Study of Supervised Learning Segmentation

Considering the above segmentation backbone network architecture achievements, and advanced network blocks development, we explore and develop several novel networks for medical image segmentation including RARUNet [17], QAPNet [194], SwinUNet [167, 168, 193], and NRUNet [31, 181].

#### RARUNet

This is part of past work in RARUNet [17]: we propose a **R**esidual encoder to **A**ttention decoder by **R**esidual connections network for medical image segmentation, which we explore several advanced network blocks for an Encoder-Decoder network. Its main novelty consists of: (1) skip interconnections on the four down-sampling blocks as residual encoders, to enhance gradient information transfer. (2) residual-block-based concatenation to mitigate the disparity between encoders and decoders.

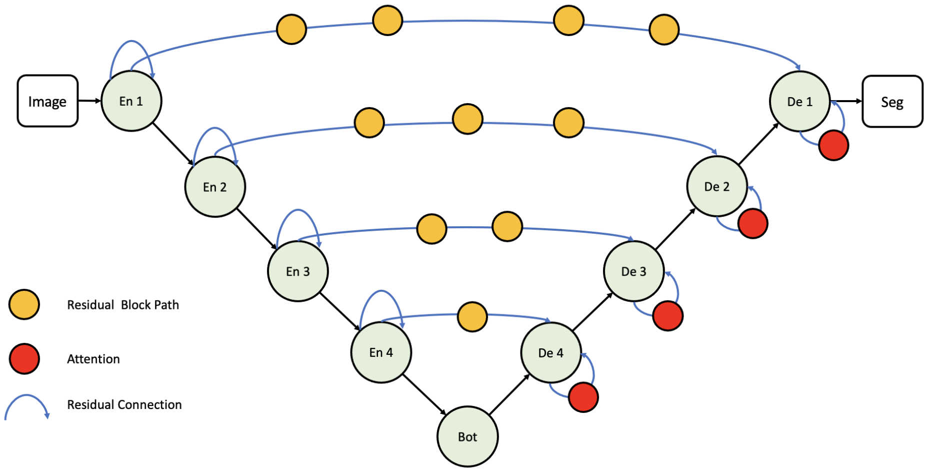


**Figure 4.9:** The Residual Encoder to Attention Decoder by Residual Block Path for Medical Image Segmentation.

(3) convolutional attention module on four up-sampling blocks to capture essential information. The architecture of RARUNet is illustrated in Figure 4.9.

### QAPNet

This is part of past work in QAPNet [194]. We introduce the innovative **Q**uadruple **A**ugmented **P**yramid Network (QAPNet), detailed in [194], designed for multi-class medical image segmentation. This network addresses the limitations of conventional multi-CNN layer pipelines in capturing and transferring image features effectively. QAPNet's uniqueness lies in its integration of pyramid CNN and pyramid pooling within an Encoder-Decoder framework. The network features four pyramid network blocks, each based on various sizes of pooling layers and atrous CNN with differing dilation rates. This augmented pyramid network is further diversified by experimenting with different pooling layers and dilation rate settings. The atrous CNN-based pyramid network, arranged in parallel, serves as a skip connection, ensuring the transfer of both global and local information for precise pixel-level segmentation. Additionally, the pooling-based pyramid network, incorporating both average and max pooling, enhances the network's robustness and efficiency. The QAPNet architecture, as illustrated in Figure 4.10, demonstrates a sophisticated balance of feature extraction, computational efficiency, and adaptability to variations in image size and orientation.

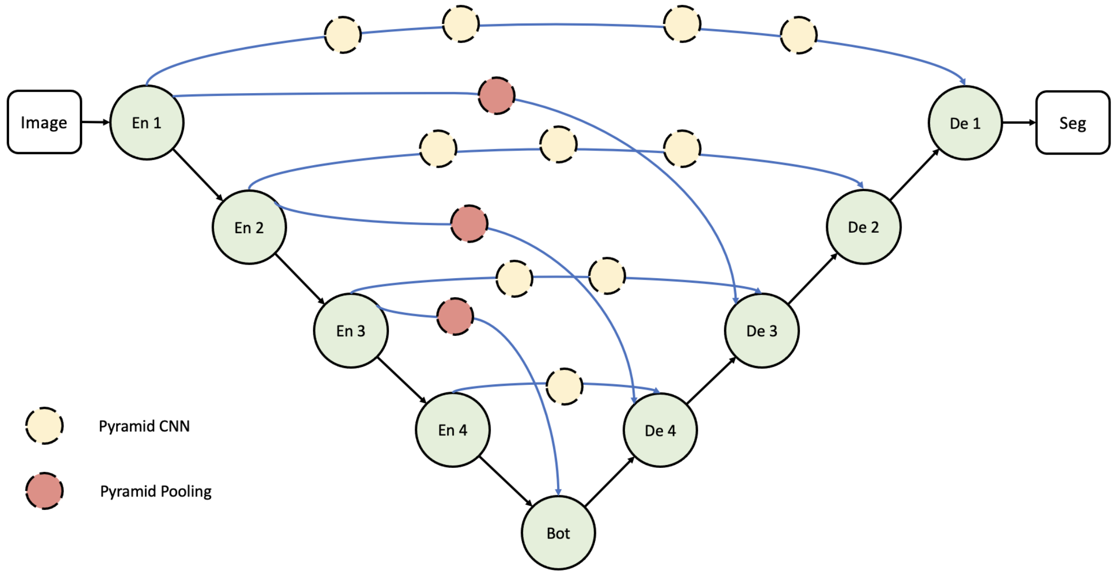


**Figure 4.10:** Quadruple Augmented Pyramid Network for Medical Image Segmentation.

### NRUNet

This is part of past work in NRUNet [31]. We introduce a novel network architecture for medical image segmentation, integrating ViT enhanced encoders with CNN-based decoders. The NRUNet, as shown in Figure 4.11, features a symmetrical design with multiple encoder and decoder levels, symbolized as $En_i$ and $De_i$ where $i \in [1, 2, 3, 4]$ indicates the level of Encoders and Decoders. This design facilitates effective transfer of feature maps across corresponding levels, utilizing a methodology akin to that in UNet. The CNN components in the encoders/decoders consist of dual-layer CNN combined with Batch Normalization and sampling operations. The ViT components are intricately woven into the encoders, particularly at the $En_4$ level, incorporating a bottleneck structure to optimize feature extraction. The ViT layers encompass Layer Normalization (LN), Multi-Head Self-Attention (MSA), and Multi-Layer Perceptron (MLP), drawing on the foundational concepts from the original Transformer network. This design ensures efficient processing of input features, transformed into a series of non-overlapping patches and then linearly embedded.

Positional encoding enriches these embedded tokens with spatial context before they undergo sequential processing through the ViT layers, which is essential in capturing long-range dependencies achieving high-quality medical image segmentation.

(1) Tokenization is performed by reshaping the input image $x$ which width and length of image is $H \times W$ into a sequence of flattened 2D patches $x_i^p \in \mathbb{R}^{P^2 \cdot C} | i = 1, \ldots, N$, where each patch has a size of $P \times P$ and $N = \frac{H \times W}{P^2}$ represents the number of image patches (i.e. the input sequence length).

(2) The patches are then mapped to vectors $x^p$ in a latent $D$-dimensional embedding space using a trainable linear projection. The network feature learning capability normally benefit with the high value of $D$, but high value of $D$ also leads to increase in computational cost. To encode spatial information of the patches, learnable position embeddings are added to the patch embeddings, as follows:

$$z^0 = [x_1^p E; x_2^p E; \ldots; x_N^p E] + E_{\text{pos}} \tag{4.8}$$

where $E \in \mathbb{R}^{(P^2 \cdot C) \times D}$ represents the patch embedding projection, and $E_{\text{pos}} \in \mathbb{R}^{N \times D}$ denotes the position embedding, and $z^0$ is the feature map which is feed to the first ViT layer.

(3) The Transformer encoder consists of $L$ layers, each containing an MSA and an MLP. Consequently, the output of the $l^{th}$ layer can be expressed as:

$$z_l^{'} = \text{MSA}(\text{LN}(z_{l-1})) + z_{l-1} \tag{4.9}$$

$$z_l = \text{MLP}(\text{LN}(z_l^{'})) + z_l^{'} \tag{4.10}$$

where $\text{LN}(\cdot)$ represents the layer normalization operator, and $z_l$ is the encoded image representation.

(4) Specifically, the MLP is a fully connected feed forward neural network that consists of multiple layers of nodes, also known as neurons. In proposed NRUNet, the MLP is employed to further refine the features extracted by the

MSA mechanism. The MLP consists of two linear layers with a GELU activation function [99] applied in between:

$$\mathrm{MLP}(z_l^{'}) = \mathrm{Linear}_2(\mathrm{GELU}(\mathrm{Linear}_1(z_l^{'}))) \tag{4.11}$$

where $\mathrm{Linear}_1$ and $\mathrm{Linear}_2$ represent the first and second linear layers, respectively. The following MSA consists of multiple self-attention heads that operate in parallel to capture different aspects of the input tokens. Each self-attention head computes the attention scores using Query ($Q$), Key ($K$), and Value ($V$) matrices, which are derived from the input tokens through linear transformations:

$$Q = z_l^{'} W_Q, \quad K = z_l^{'} W_K, \quad V = z_l^{'} W_V \tag{4.12}$$

where $W_Q$, $W_K$, and $W_V$ are learnable weight matrices [39].

The attention scores are computed by taking the dot product of the Query and Key matrices, followed by a scaling operation and a softmax normalization:

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V \tag{4.13}$$

where $d_k$ is the dimension of the Key vectors.

The output of the individual self-attention heads is then concatenated and linearly transformed to produce the final MSA output:

$$\mathrm{MSA}(z_l^{'}) = \mathrm{Concat}(\mathrm{Head}_1, \ldots, \mathrm{Head}_H) W_O \tag{4.14}$$

where $\mathrm{Head}_i$ represents the output of the $i$-th self-attention head, $H$ is the number of heads, and $W_O$ is a learnable weight matrix.

### SwinUNet

This is part of past work in CESSViT, UAMTViT [167, 168]. We introduce SwinUNet, a pure Swin ViT-based encoder-decoder U-Shape network, as a novel approach to medical image segmentation. SwinUNet addresses the limitation of conventional CNN-based networks, which tend to blur image features after multiple encoding layers. While UNet uses copy and crop techniques to transfer semantic

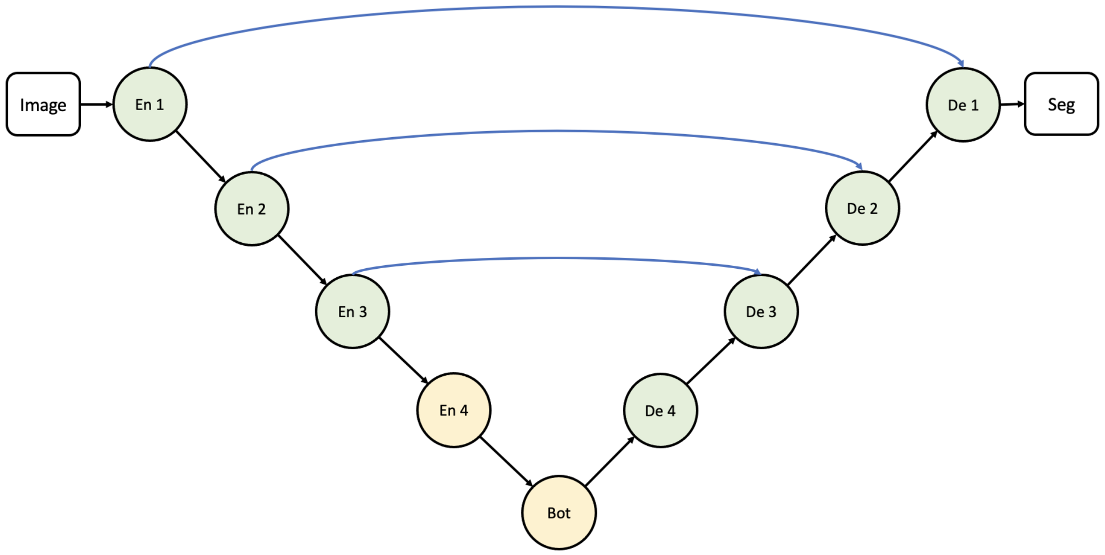


**Figure 4.11:** ViT-Improved-Encoder to CNN-based Decoder Network for Medical Image Segmentation.

features effectively, the fine details, particularly at the edges of regions of interest, can be lost. SwinUNet, by employing a purely self-attention-based mechanism, aims to capture the global context of images more effectively, preserving crucial boundary information. This architecture, as shown in Figure 4.12, replaces the conventional encoders and decoders with successive Swin ViT layers, thus harnessing the potential of ViT for enhanced segmentation performance.

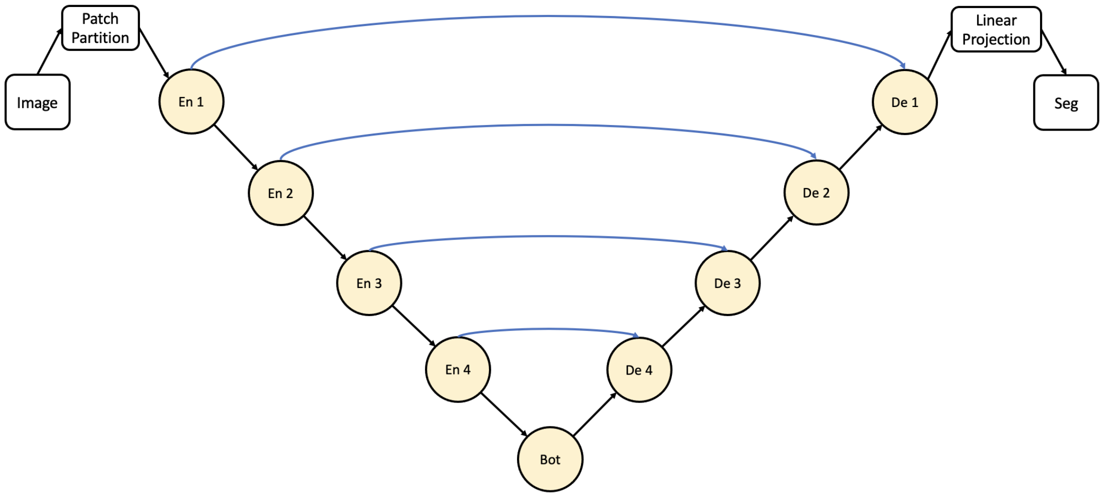


**Figure 4.12:** Pure Swin ViT-based U-Shape Network for Medical Image Segmentation.

## 4.3 Experiments and Results

### 4.3.1 Implementation Details

We implemented supervised learning studies using Python and TensorFlow. Training all networks for 50 epochs. The training batch size is set to 4. The Adam optimizer [48] is employed, with a learning rate of $10^{-5}$. Our loss function is based on the Dice coefficient, a commonly used metric for evaluating overlap in semantic segmentation, particularly suited to tackling the unbalance between ROI and background. Two datasets including Spine CT dataset [219], and COVID-19 CT dataset [220] are utilized for evaluation.

### 4.3.2 Qualitative Results

Figure 4.14 and Figure 4.13 illustrates eight examples on CT Spine and eight examples on CT COVID-19 of raw images, ground truth, and the predicted result of a number of different networks: UNet [4], Residual UNet [214], Dense UNet [214], MultiResUnet [177], LinkNet [205], FPN [22], UNet++ [175], UNet3+ [180], VNet [5], RARUNet [17], and QAPNet [194]. Figure 4.13 further illustrates example raw images and the predicted results of four classes including consolidation area, glass area, and other lung area(which is not infected by diseases), and background against ground truth for selected networks: UNet [4], FPN [22], and QAPNet [194].

### 4.3.3 Quantitative Results

The quantitative comparison of proposed U-Shape Encoder-Decoder segmentation networks, i.e. RARUNet [17], QAPNet [194], and NRUNet [31] with other baseline methods including FPN [22], VNet [5], LinkNet [205], UNet [4], Residual UNet [214], Dense UNet [214], MultiRes UNet [177], and UNet++ [175] are given in Table 4.1, and Table 4.2 respectively depending on CT Spine test set and CT COVID-19 test set. The average performance of Dice-coefficient, IoU, accuracy, precision, sensitivity, and specificity are reported. Our proposed methods are highlighted with **Bold**. The best performance are with **Bold**, and the second best performance with the proposed methods are with <u>Underline</u>.

**Table 4.1:** The Direct Comparison of Existing Methods on CT Spine Test Set.

| Network | Dice | IoU | Acc | Pre | Sen | Spe |
|---|---|---|---|---|---|---|
| UNet [4] | 0.9580 | 0.9193 | 0.9963 | 0.9619 | 0.9541 | 0.9983 |
| VNet [5] | 0.9446 | 0.8950 | 0.9950 | 0.9202 | **0.9703** | 0.9961 |
| LinkNet [205] | 0.9524 | 0.9091 | 0.9959 | 0.9662 | 0.9390 | 0.9985 |
| Residual UNet [214] | 0.9416 | 0.8897 | 0.9949 | 0.9481 | 0.9353 | 0.9976 |
| Dense UNet [214] | 0.9612 | 0.9252 | 0.9966 | 0.9600 | 0.9624 | 0.9982 |
| MultiRes UNet [177] | 0.9644 | 0.9312 | 0.9969 | 0.9633 | 0.9655 | 0.9983 |
| UNet++ [175] | 0.9659 | 0.9340 | 0.9970 | 0.9676 | 0.9642 | 0.9985 |
| SwinUNet [193] | 0.9601 | 0.9233 | 0.9948 | 0.9640 | 0.9563 | 0.9975 |
| **RARUNet** [17] | 0.9674 | 0.9369 | 0.9972 | <u>0.9721</u> | 0.9629 | <u>0.9987</u> |
| **QAPNet** [194] | <u>0.9690</u> | <u>0.9399</u> | <u>0.9973</u> | 0.9715 | 0.9666 | <u>0.9987</u> |
| **NRUNet** [31] | **0.9703** | **0.9424** | **0.9974** | **0.9740** | <u>0.9667</u> | **0.9988** |

Furthermore, each of image on test set is also validated individually. The distribution of Dice Coefficients is visually represented in Figure 4.15. This graphical depiction offers insights into the performance variability among networks. Networks with a higher median and a more compact box plot generally indicate superior and more consistent performance across various image slices. Additionally, we conducted pairwise $t$-tests to statistically compare the performance differences in Dice coefficients between NRUNet [31] and UNet [4]. The resulting $t$-statistic is 2.64, with a significant $p$-value of 0.0085. This low $p$-value, typically considered significant if below 0.05, underscores a statistically significant performance disparity between the two networks, reinforcing the empirical observations from the box plot analysis. In addition, we conduct $t$-statistic for all baseline methods with NRUNet, and the $p$-value is always below 0.05. This rigorous statistical approach provides a robust basis for comparing and validating the performance of different segmentation networks.

### 4.3.4 Ablation Study

To evaluate the impact of various components and their combinations on the U-Shape Encoder-Decoder segmentation network, a detailed ablation study is introduced.

**Ablation Study on RARUNet**

The ablation study of RARUNet is indicated in Table 4.3 revealing that omitting certain elements leads to a noticeable drop in performance. We explored

**Table 4.2:** The Direct Comparison of Existing Methods on CT COVID-19 Test Set.

| Network | Dice | IoU | Acc | Pre | Sen | Spe |
|---|---|---|---|---|---|---|
| UNet [4] | 0.8489 | 0.7374 | 0.9894 | 0.8133 | 0.8792 | 0.9924 |
| VNet [5] | 0.7869 | 0.6487 | 0.9908 | 0.8212 | 0.7554 | 0.9962 |
| LinkNet [205] | 0.7598 | 0.6126 | 0.9833 | 0.7181 | 0.7783 | 0.9883 |
| Attention UNet [178] | 0.7167 | 0.5585 | 0.8234 | 0.5871 | 0.9157 | 0.8251 |
| Residual UNet [214] | 0.8649 | 0.7620 | 0.9895 | 0.8287 | 0.9005 | 0.9923 |
| Dense UNet [214] | 0.7133 | 0.5544 | 0.7772 | 0.5871 | 0.8921 | 0.7788 |
| MultiRes UNet [177] | 0.5979 | 0.4264 | 0.9294 | 0.4328 | **0.9757** | 0.9276 |
| SwinUNet [193] | 0.8243 | 0.7011 | 0.7350 | **0.9952** | 0.7035 | 0.9740 |
| **RARUNet** [17] | 0.8817 | 0.7884 | 0.9969 | 0.8536 | 0.9120 | **0.9982** |
| **QAPNet** [194] | **0.8989** | **0.8163** | **0.9976** | 0.8460 | <u>0.9580</u> | <u>0.9980</u> |
| **NRUNet** [31] | <u>0.8911</u> | <u>0.8035</u> | <u>0.9912</u> | <u>0.9217</u> | 0.8623 | 0.9970 |

**Table 4.3:** The Ablation Study on Contributions of RARUNet.

| Residual Encoders | Residual Connections | Attention Decoders | IoU |
|---|---|---|---|
| | | | 0.7182 |
| ✓ | | | 0.7873 |
| | ✓ | | 0.9119 |
| | | ✓ | 0.8927 |
| ✓ | | ✓ | 0.9070 |
| ✓ | ✓ | | 0.9126 |
| ✓ | ✓ | ✓ | **0.9369** |

combinations involving residual encoders, residual block paths, and attention decoders, assessing the corresponding performance. This ablation study is conducted on CT Spine dataset.

**Ablation Study on QAPNet**

Similarly with RARUNet, for QAPNet, as presented in Table 4.4, we observed that the exclusion of specific components adversely affects the overall effectiveness of the network. The different combinations of dual pyramid CNN networks and dual pyramid pooling networks are explored, again measuring both their influence of network performance. This ablation study is conducted on CT COVID-19 dataset.

Table 4.4: The Ablation Study on Contributions of QAPNet.

| CNN Pyramid Net | | Pooling Pyramid Net | | MIoU |
|---|---|---|---|---|
| 1 3 9 | 1 2 4 | Max | Avg | |
| | | | | 0.5331 |
| ✓ | | | | 0.7469 |
| | ✓ | | | 0.7966 |
| ✓ | ✓ | | | 0.7996 |
| | | ✓ | | 0.7419 |
| | | | ✓ | 0.7872 |
| | | ✓ | ✓ | 0.7373 |
| ✓ | ✓ | | ✓ | 0.8072 |
| ✓ | ✓ | ✓ | | 0.8023 |
| ✓ | ✓ | ✓ | ✓ | **0.8163** |

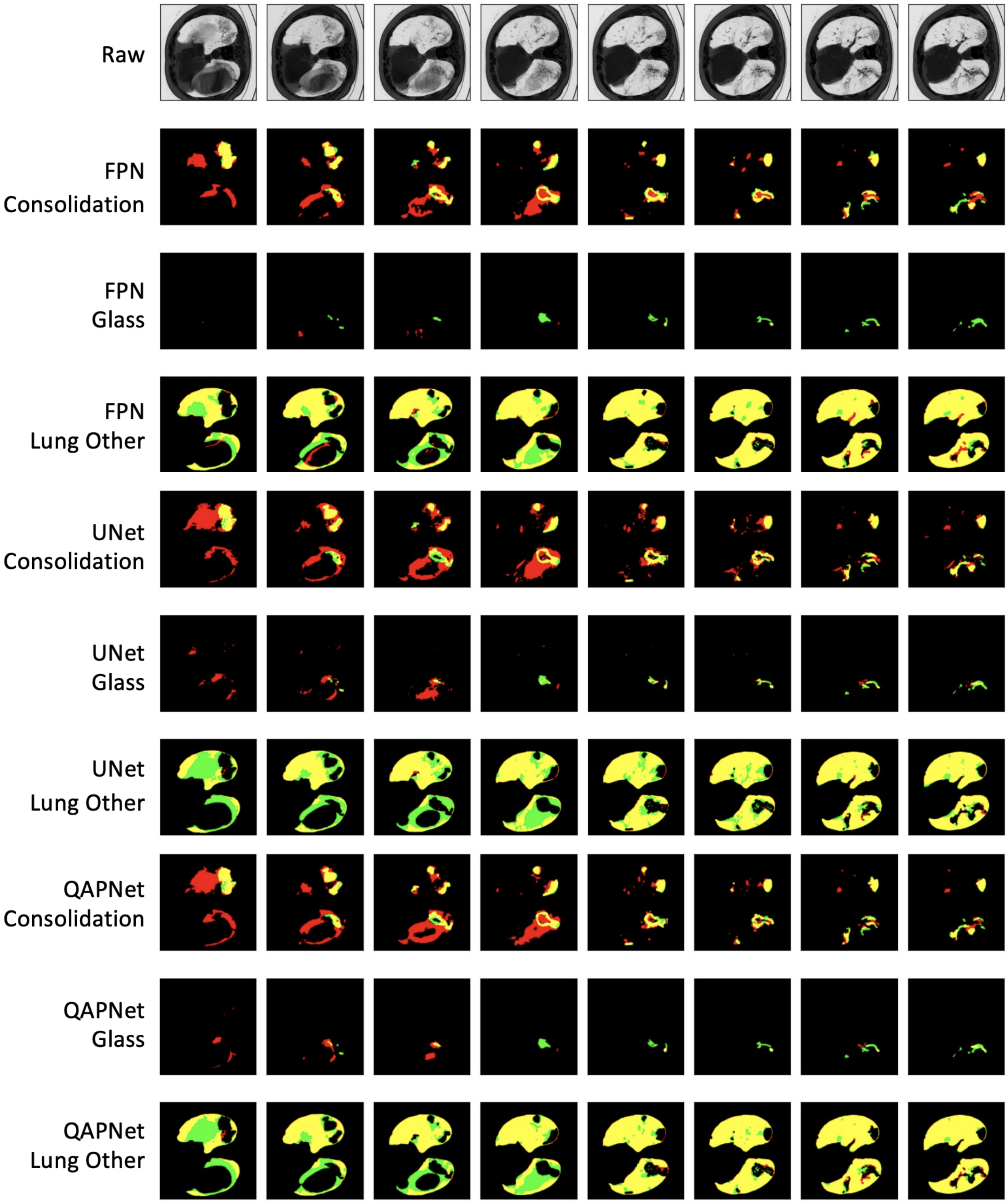


**Figure 4.13:** Example Results on CT COVID-19 Test Set. Given a batch of CT images, each of FPN, UNet, and the proposed QAPNet provides the segmentation inference of four classes of ROI including consolidation area, glass area, other part of lung, and background.

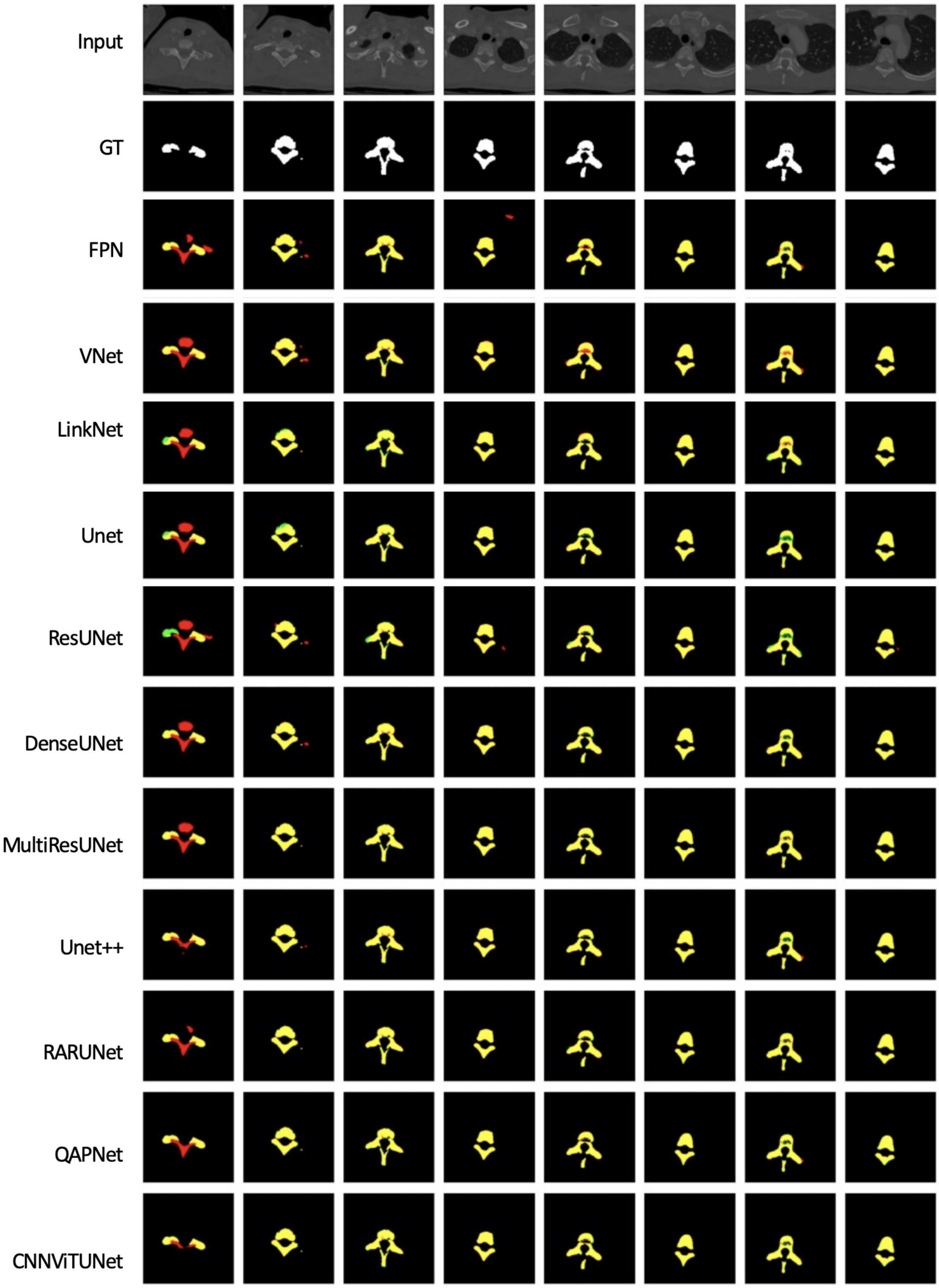


**Figure 4.14:** Example Results on CT Spine Test Set. Given a batch of CT images, each of baseline networks and proposed networks provides the segmentation inference of binary classification of spine.

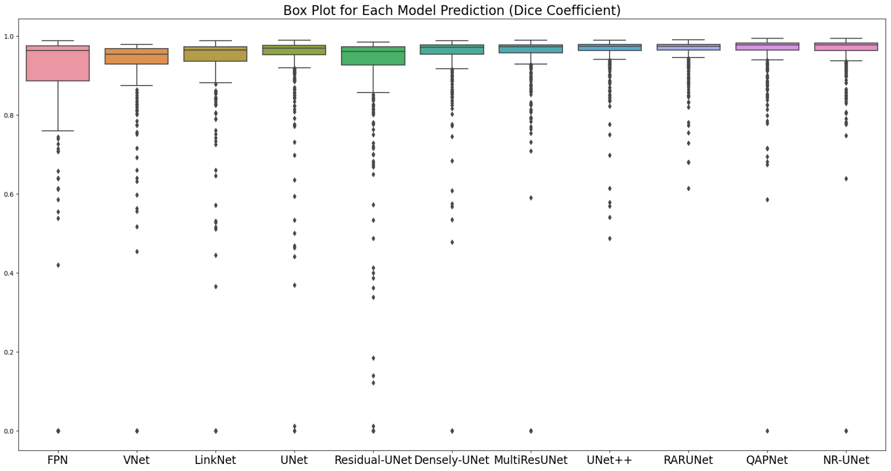


**Figure 4.15:** Box Plot for the Dice-Coefficient Distribution of Prediction by Each Network on CT Spine Test Set. Our proposed modified RARUNet, QAPNet, and NRUNet are more likely to predict segmentation results with high dice score.

## 4.4 Contribution and Discussion

This chapter provides a comprehensive exploration of utilizing various network blocks within segmentation backbone networks, resulting in several novel segmentation networks with competitive performance against existing methods. Specifically, we introduce residual learning, densely connected layers, attention mechanisms, and other critical network blocks that significantly improve the performance of CNN- or ViT-based segmentation networks. Our exploration provides examples into the development and application of integrating network blocks with the segmentation backbone network.

The important point of this thesis, however, is to consider that the effectiveness of a certain combination of network blocks is inherently dependent on the specific dataset used. We observed that some combinations excel in specific contexts, while may also perform worse under different conditions. This highlights the complex interplay between network architecture and data characteristics, reminding us that there is no single network architecture design for different segmentation tasks.

Another concern is the quality of the datasets we use to train our networks in practical scenarios. In supervised learning studies, we assume that the data provided is accurate and correctly annotated. The reality of clinical scenarios, however, is that data can often be noisy, incomplete, or mislabeled. We need to question 'how good is good enough'? A network that achieves a high score on a given dataset is not necessarily one that will perform well in real-world scenarios. And also, 'how good' does the annotation need to be? The gold standard of success should not be the ability to overfit to a particular dataset but the ability to generalize and perform well on unseen data.

Finally, it is essential to recognize that supervised learning networks, despite their impressive performance, still have their limitations. Many of the previous works in this domain could be seen as an 'overfitting game', where networks are designed to perform exceptionally well on a specific task but fail to generalize to real-world scenarios.

In conclusion, while our exploration of various network blocks for the segmentation backbone network shows promising results, it also highlights the need for research between network architecture engineering and data situations in real world.

# 5

# Semi-Supervised Learning: Partial Data Annotation with Full Dense Masks

## Contents

**5.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . 81**
**5.2 Literature . . . . . . . . . . . . . . . . . . . . . . . . . . . . 82**
    5.2.1 Consistency-Aware Learning . . . . . . . . . . . . . . . . . 82
    5.2.2 Adversarial Learning . . . . . . . . . . . . . . . . . . . . . 88
    5.2.3 Proposed Advanced Schemes . . . . . . . . . . . . . . . . 90
**5.3 Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 95**
    5.3.1 Semi-Supervised Learning Framework Setup . . . . . . . 95
    5.3.2 Triple-View Learning . . . . . . . . . . . . . . . . . . . . . 96
    5.3.3 Examiner-Student-Teacher Learning . . . . . . . . . . . 96
    5.3.4 Uncertainty-Aware Mean Teacher ViT . . . . . . . . . . 99
    5.3.5 Computational-Efficient Cross Supervision ViT . . . . . 100
**5.4 Experiments and Results . . . . . . . . . . . . . . . . . . . 102**
    5.4.1 Implementation Details . . . . . . . . . . . . . . . . . . 102
    5.4.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . 102
    5.4.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . 104
    5.4.4 Ablation Study . . . . . . . . . . . . . . . . . . . . . . . 107
**5.5 Contribution and Discussion . . . . . . . . . . . . . . . . 118**

## 5.1 Motivation

While supervised learning has seen significant advancements in medical image segmentation, the practical situation is often with a substantial obstacle of utilizing developed methods into real-world clinical settings - the limitation of fully annotated medical images. The process of obtaining annotated medical images is costly, time-consuming, and requires the expertise of trained medical professionals. Consequently, the majority of medical images in real-world scenarios remain unlabeled, thereby limiting the potential of purely supervised learning techniques.

Semi-Supervised Learning (SSL) leverages both labeled and unlabeled data for network training. The application of SSL in medical imaging tasks has the potential to significantly improve the performance of the networks and decrease the annotation cost of clinicians.

By successfully navigating these challenges through SSL, this chapter propose a series of strategies to train more robust and reliable medical image segmentation networks by exploring the full potential of medical data.

## 5.2 Literature

### 5.2.1 Consistency-Aware Learning

Consistency-aware is currently the most important approach in the study of semantic segmentation with SSL. Consistency regularization operates under the premise that when slight changes, known as perturbations, are made to unlabeled data, the predictions of the network should not vary significantly. It is about training a network to give consistent outputs, even when minor variations are introduced to its inputs.

The network training process is designed to enable consistency output under various perturbation normally consisting of data perturbation and network perturbation. In the context of data perturbation, various methods have been explored. These include data perturbations such as Ouali et al. [230] and Chen et al. [231], where the idea is to introduce small, controlled changes in the input data of the network to test and enhance the robustness of the network. Data augmentation techniques such as CutMix [232] creatively blends parts of images. MixMatch [233] takes this further by assigning low-entropy high-confidence labels to augment unlabeled examples and then combines labeled and unlabeled data using a technique called MixUp [234]. FixMatch [235] leverages pseudo labels on mildly augmented data to guide the network's behavior on more strongly augmented data.

Network perturbation focuses on varying the neural network architecture itself. This includes employing different network designs like dual-students [236], which introduces an extra 'student' network alongside the primary one to add a stabilization constraint, reminiscent of the Student-Teacher SSL style [237]. TriNet [238] explores a shared encoder with three distinct decoders, each processing different types of data for classification. Triple-view learning [170] extends this concept to multi-view learning for image semantic segmentation.

Some popular methods in consistency-aware learning include network-ensembling learning [237], cross pseudo learning [239]. Various schemes aiming at general improve SSL performance have also been developed, such as dynamic pseudo label ensembling [240], which dynamically adjusts the labels used for training, uncertainty-aware strategies [241], which factor in the network's confidence in its predictions, and transformation-consistency [213], which ensures the network's output remains stable even when input transformations are applied.

### Network-Ensembling Learning

A self-ensembling SSL method named Mean Teacher [237], which is an extension of temporal ensembling [242], has been widely adopted in SSL for medical image segmentation [167, 243, 244]. The semi-supervised framework of mean teacher for medical image segmentation is briefly sketched in Figure 5.1. This method typically consists of a student network and a teacher network, with the same architecture. The student network learns from annotated data with feature perturbation. Meanwhile, the teacher network, generally more robust, is continuously updated based on the student network's weights. It provides guidance to the student network via pseudo-labels, emphasizing consistency in the learning process.

The architecture of the Teacher $f_T(\overline{\theta})$ is similar to $f_S(\theta)$, except the Teacher does not learn from data directly [237]. It is updated from the exponential moving average network weights (illustrated in Eq. 5.1) of $f_S(\theta)$, and it is more likely than the Student to infer the correct value. The Teacher's predictions are considered as the pseudo labels to supervise the Student.

$$\overline{\theta} = \alpha\theta_{t-1} + (1-\alpha)\theta_t \tag{5.1}$$

where $\overline{\theta}$ is updated based on the Student parameter $\theta_t$ from the previous training step $t$; weight factor $\alpha = 1 - \frac{1}{t+1}$. The pseudo labels are generated by the Teacher without noise as:

$$\boldsymbol{Y}_{\mathrm{p}} = f_{\mathrm{T}}(X_{\mathrm{u}}; \overline{\theta}) \tag{5.2}$$

Thus the unlabeled training set $X_u$ can be utilized to train the Student with $(\mathbf{X}_{\mathrm{u}}, \boldsymbol{Y}_{\mathrm{p}})$.

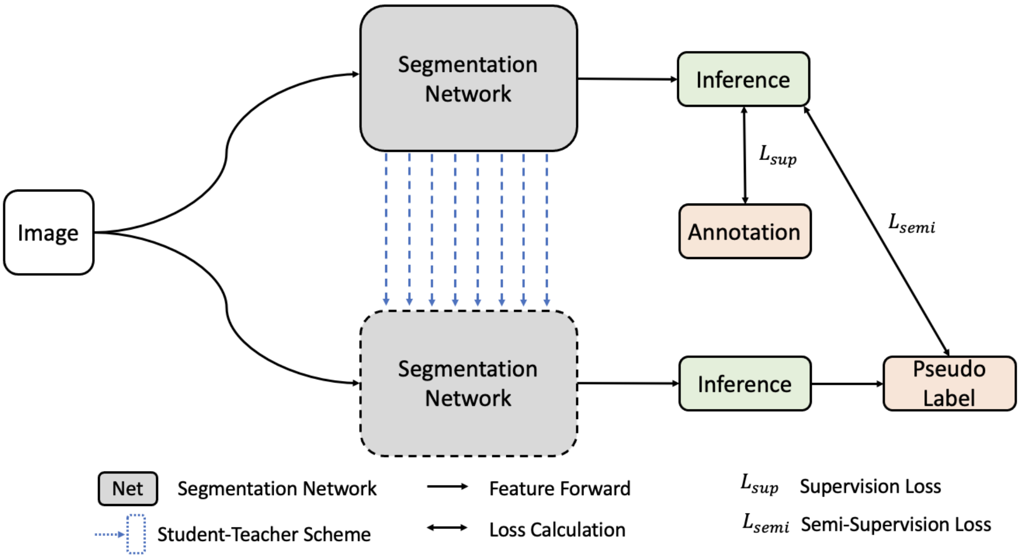


**Figure 5.1:** The Framework of Mean Teacher for Medical Image Segmentation, Which Consisting of a Student Network and a Teacher Network. The parameters of Teacher network is updated by Student network, and the prediction of Teacher network can supervise Student network thus expanding unlabeled dataset.

### Cross Pseudo Label Learning

Cross pseudo label learning is firstly proposed by [239], and has been widely adopted in SSL for medical image segmentation [14, 168]. The aim of cross pseudo label learning is encouraging the consistency of predictions by adding perturbations on networks. Cross pseudo label learning consists of two segmentation networks $f(\theta)$ with the same architecture initialized with different weight parameters as network perturbation. The output from the same input image $\boldsymbol{X}$ can be illustrated as $\boldsymbol{Y}_{\mathrm{p1}} = f_{\mathrm{t}}(\boldsymbol{X};\theta_1)$ and $\boldsymbol{Y}_{\mathrm{p2}} = f_{\mathrm{t}}(\boldsymbol{X};\theta_2)$ where $\theta$ is the weight set of each network. The inference from one network is considered as pseudo labels into the training of the other network. The semi-supervised framework of cross pseudo label learning for medical image segmentation is sketched in Figure 5.2.

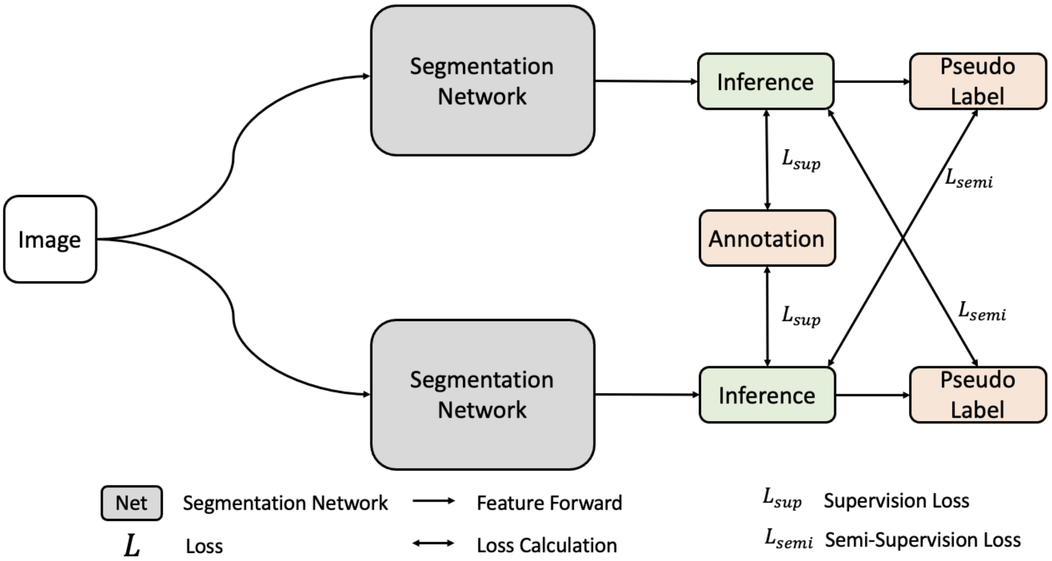


**Figure 5.2:** The Framework of Cross Pseudo Supervision for Medical Image Segmentation, Which Consisting of Two Networks with Same Architecture but Initialize Separately. Two networks collaborate and beneficial each other with their inference as pseudo label.

### FixMatch

FixMatch, an SSL method proposed by [235] boosts the data perturbation using of unlabeled data by combining weak and strong data augmentations. The framework of FixMatch for medical image segmentation is illustrated in Figure 5.3. In this method, two versions of the same unlabeled input data are created. The first version is a weakly augmented version $\boldsymbol{X}_{\mathrm{w}}$, and the second is a strongly augmented version $\boldsymbol{X}_{\mathrm{s}}$. The weakly augmented version is processed through the network $f_{\mathrm{S}}(\theta)$ to obtain a set of soft predictions which are then converted to hard pseudo-labels:

$$\boldsymbol{Y}_{\mathrm{p}} = f_{\mathrm{S}}(\boldsymbol{X}_{\mathrm{w}}; \theta) \tag{5.3}$$

These pseudo-labels are then used to supervise the network's predictions on the strongly augmented version of the same input data:

$$\boldsymbol{Y}_{\mathrm{s}} = f_{\mathrm{S}}(\boldsymbol{X}_{\mathrm{s}}; \theta) \tag{5.4}$$

To ensure the robustness of the generated pseudo-labels, a confidence threshold $\tau$ is introduced. Only the pseudo-labels with prediction confidence above $\tau$ are used for training. In this way, FixMatch leverages the unlabeled data by incorporating it into the training process with $\boldsymbol{X}_{\mathrm{s}}$ and $\boldsymbol{Y}_{\mathrm{p}}$.

The key insight in FixMatch is to use the network's predictions on the weakly augmented data as 'pseudo labels' for the strongly augmented versions of the same data. Since weakly augmented data maintains a closer resemblance to the original, the network's predictions here are assumed to be more reliable. By enforcing data consistency, it is ensured that the network produces similar predictions for both weakly and strongly augmented versions of the same input. FixMatch effectively leverages unlabeled data for training. The network learns not only to recognize features in slightly altered images but also to maintain its understanding even when those images are substantially transformed. This FixMatch is sketched in Figure 5.3. The framework captures the process where inferences from weakly augmented data guide the network in learning from the more drastically altered, strongly augmented data.

### Interpolation Consistency Learning

Interpolation Consistency Training (ICT), as proposed by [234]. The central premise of ICT is the incorporation of interpolation into data consistency regularization. The ICT for medical image segmentation is sketched in Figure 5.4. In this method, pairs of images are interpolated both in the input space and in the output space to generate new training examples. In the input space, two images $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are mixed to create an interpolated image $\boldsymbol{X}_{\mathrm{mix}}$:

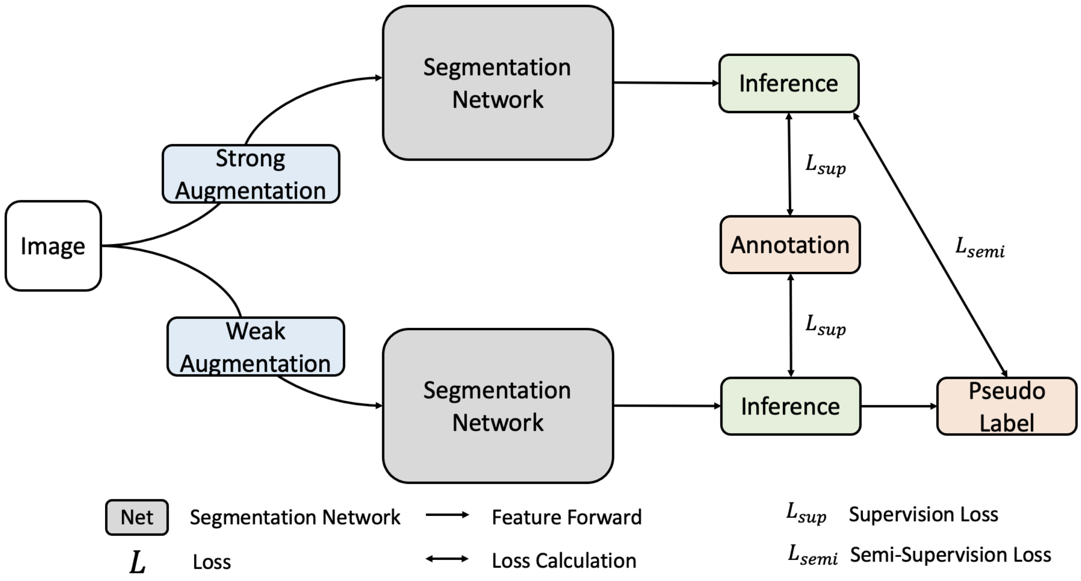


**Figure 5.3:** The Framework of FixMatch for Medical Image Segmentation, Which Consisting of Strong and Weak Augmentation. The inference from weak augmentation is more likely to be precise than inference from strong augmentation, and can be considered as pseudo label to train network.

$$\boldsymbol{X}_{\text{mix}} = \lambda \boldsymbol{X}_1 + (1 - \lambda) \boldsymbol{X}_2 \tag{5.5}$$

where $\lambda$ is a mixing coefficient randomly sampled from a Beta distribution. A similar operation is performed in the output space, where the network's predictions on the two original images, $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$, are mixed to generate the interpolated label $\boldsymbol{Y}_{\text{mix}}$:

$$\boldsymbol{Y}_{\text{mix}} = \lambda \boldsymbol{Y}_1 + (1 - \lambda) \boldsymbol{Y}_2 \tag{5.6}$$

The network is then trained to predict $\boldsymbol{Y}_{\text{mix}}$ when given $\boldsymbol{X}_{\text{mix}}$ as input. By encouraging the network's predictions to be consistent across interpolated inputs and outputs, ICT introduces an additional form of regularization. This technique helps the network generalize from the labeled data to the unlabeled data, thus effectively leveraging unlabeled data in SSL scenarios.

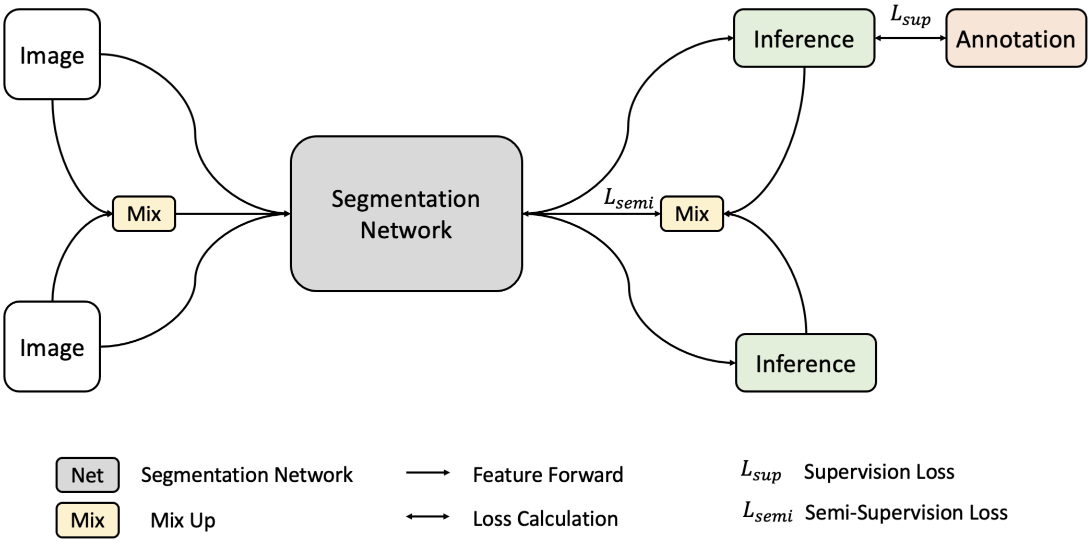


**Figure 5.4:** The Framework of Interpolation Consistency Training Data with MixUp for Medical Image Segmentation. The Inference of the Combination of Two Input Images is Similar with the Combination of Inference of Two Input Image Separately.

### 5.2.2 Adversarial Learning

Alongside consistency training, another common SSL technique is adversarial training [245–247]. It normally involves developing an additional discriminator network to extract statistical features which aim to distinguish the quality of inferences of the network. The core principle of adversarial training is to develop a competitive scenario between the segmentation network and an auxiliary discriminator network.

The semi-supervised adversarial training for medical image segmentation is sketched in Figure 5.5. This framework comprises of a segmentation network $f_S(\theta)$ and a discriminator network $f_D(\phi)$. The segmentation network $f_S(\theta)$ is trained to generate accurate segmentation of both labeled and unlabeled data, denoted as $\boldsymbol{Y}_{\mathrm{p}}$ and $\boldsymbol{Y}_{\mathrm{u}}$, respectively:

$$\boldsymbol{Y}_{\mathrm{p}} = f_S(\boldsymbol{X}_{\mathrm{p}}; \theta) \tag{5.7}$$

$$\boldsymbol{Y}_{\mathrm{u}} = f_S(\boldsymbol{X}_{\mathrm{u}}; \theta) \tag{5.8}$$

Simultaneously, the discriminator network $f_D(\phi)$ is trained to differentiate between the segmentation network's predictions on labeled and unlabeled data:

$$D_{\mathrm{p}} = f_D(\boldsymbol{Y}_{\mathrm{p}}; \phi) \tag{5.9}$$

$$D_{\mathrm{u}} = f_D(\boldsymbol{Y}_{\mathrm{u}}; \phi) \tag{5.10}$$

During training, the segmentation network is optimized to enforce the discriminator network by making its predictions on labeled and unlabeled data indistinguishable. Conversely, the discriminator network is trained to become better at distinguishing between the two types of predictions.

This adversarial dynamic compels the segmentation network to produce consistent and high-quality predictions across both labeled and unlabeled datasets, thus capitalizing on the abundance of unlabeled data to enhance its performance. The essence of robustness in this context arises from the network's ability to maintain prediction accuracy despite variations in data quality and labeling. The inherent challenge posed by the discriminator network ensures that the segmentation network does not simply memorize or overfit to the labeled data but rather learns to generalize its predictive capabilities effectively.

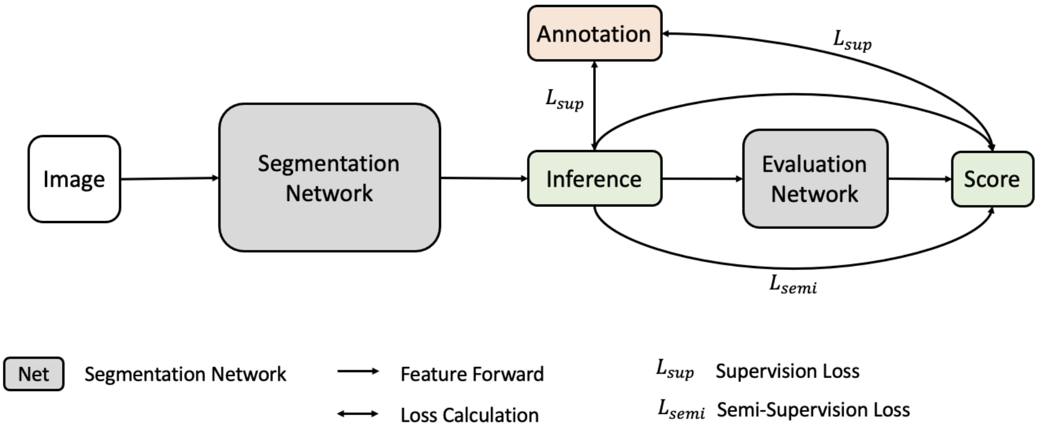


**Figure 5.5:** The Framework of Adversarial Learning for Medical Image Segmentation, Consisting of a Segmentation Network and a Evaluation Network. Two networks are trained separately (segmentation network is to provide high-quality inference, and evaluation network is to classify the quality of inference is good enough) and against each other as adversarial learning.

### 5.2.3 Proposed Advanced Schemes

#### Dynamic Pseudo Label Scheme

The Dynamic Pseudo Label Scheme introduces a novel pseudo label ensembling approach in the training process. It can be considered as an enhanced data perturbation scheme, which is sketched in Figure 5.6. In this approach, for each input, two pseudo labels are generated, and a composite of these labels, weighted by a random factor $\alpha$, is used for training the network. $\alpha$, which ranges from 0 to 1, is assigned by a uniform distribution. It determines the weight each pseudo label contributes to the composite label used for SSL. $\alpha$ serves as an additional data perturbation mechanism because it is unlearnable and dynamic. The formulation of the dynamic pseudo label blending based on two distinct pseudo labels can be described as follows:

$$\boldsymbol{Y}_{\text{pseudo}} = \alpha \, \boldsymbol{Y}_{\text{pseudo1}} + (1-\alpha) \, \boldsymbol{Y}_{\text{pseudo2}} \tag{5.11}$$

where $\boldsymbol{Y}_{\text{pseudo}}$ represents the final composite pseudo label used for training, with $\boldsymbol{Y}_{\text{pseudo1}}$ and $\boldsymbol{Y}_{\text{pseudo2}}$ as the initially generated pseudo labels. Incorporating this variable weighting into the label generation process effectively introduces a controlled form of data perturbation. This randomness compels the network to adapt to a broader spectrum of scenarios, thereby enhancing its capacity for generalization.

The rationale behind this enhancement in generalization capability lies in the diversity introduced by the varying combinations of pseudo labels. Each training epoch presents the network with a slightly different version of the labels, preventing it from becoming overly tuned to specific patterns or features of the data. This dynamic and varied learning environment mimics the diversity of real-world data more closely than static training methods, equipping the network with the ability to handle unseen and varied data effectively in SSL scenarios. As a result, the network not only learns from the limited labeled data but also effectively utilizes the abundant unlabeled data, leading to improved performance in SSL tasks.

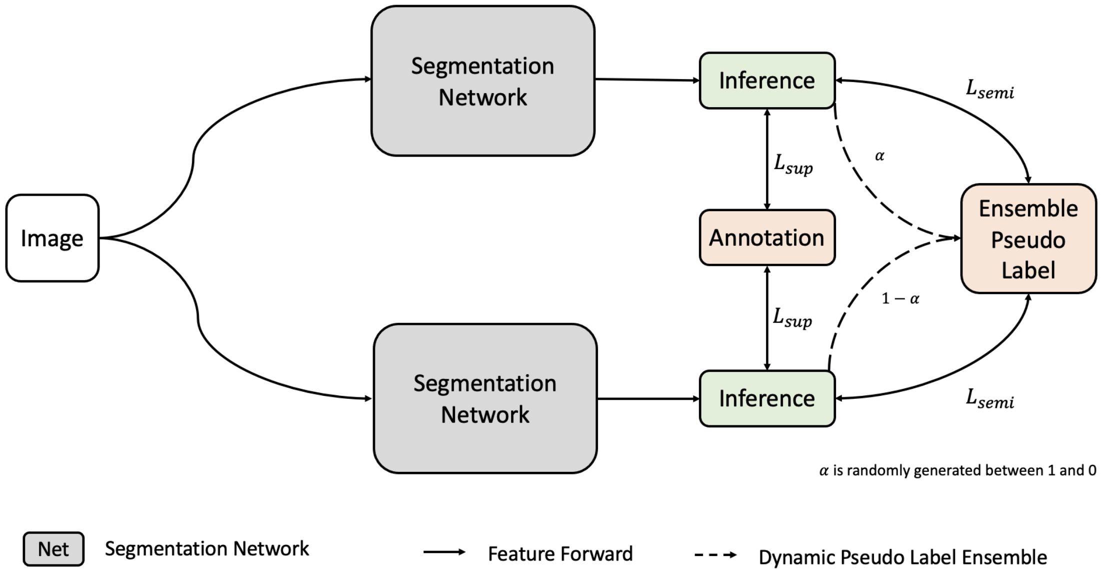


**Figure 5.6:** The Example Dynamic Pseudo Label Scheme with Cross Pseudo Label Learning for Medical Image Segmentation, Which Consisting of Two Networks. The pseudo label is generated by both of inference.

### Uncertainty-Aware Scheme

The Uncertainty-Aware Scheme is designed to reduce the impact of potentially inaccurate pseudo labels during the training process. An illustrative example of uncertainty-aware learning using the mean teacher network is sketched in Figure 5.11. This scheme is predicated on the understanding that a network's level of uncertainty in its predictions can be a crucial indicator of those predictions' reliability.

Within this framework, an uncertainty map is generated by analyzing the network's prediction probabilities. This is achieved by computing the entropy of these probabilities, which is considered as a measure of unpredictability or uncertainty following past studies [241]. The equation used to calculate the uncertainty map is as follows:

$$\boldsymbol{U} = -\sum_{c} \boldsymbol{P}_c \log(\boldsymbol{P}_c) \tag{5.12}$$

Here, $\boldsymbol{U}$ represents the uncertainty map, while $\boldsymbol{P}_c$ denotes the predicted probability for class c. The summation extends over all classes. Essentially, this equation calculates the entropy for each class prediction across the network's output, thus generating a map that highlights areas of high and low uncertainty.

The interpretation of this uncertainty map is crucial: probabilities near 1 or 0 indicate high certainty, suggesting that the network is confident in its prediction being either the background or a region of interest. Conversely, probabilities around 0.5 signify a higher degree of uncertainty, implying that the network is less sure about its prediction in these regions. The threshold hereby is utilized to filter the network's inferences, categorizing them as 'certain' or 'uncertain'. For instance, if the prediction probability for a specific pixel is above a high threshold (close to 1) or below a low threshold (close to 0), it is considered 'certain.' Predictions that fall between these thresholds, typically around 0.5, are tagged as 'uncertain'. Setting the uncertainty threshold too low or too high can lead to poor convergence or the risk of confirmation bias. To effectively utilize the estimated uncertainty map, we explore varies of threshold setting strategy in this study.

During training, supervision is applied primarily to those parts of the pseudo labels classified as 'certain,' based on the uncertainty map. By focusing on these more reliable areas, the strategy effectively minimizes the influence of potentially incorrect pseudo labels. This selective approach to supervision enhances the overall robustness and accuracy of the SSL process by ensuring that the network is trained more on data points where its predictions are deemed reliable.

### Transformation-Consistency Scheme

Transformation-Consistency Schemes, such as rotation-based SSL, introduce a new paradigm of enhanced data perturbation in SSL [213]. This approach leverages the invariance of underlying structure to specific transformations in data, like rotations, to provide additional supervision for unlabeled data.

The transformation-consistency framework for SSL methods, such as Mean Teacher, is briefly described in Figure 5.7. Here, a specific transformation $R$ (e.g. 36 degrees) is applied to the unlabeled input data $\boldsymbol{X}_{\mathrm{u}}$ for the student network:

$$\boldsymbol{X}_{\mathrm{ur}} = R(\boldsymbol{X}_{\mathrm{u}}) \tag{5.13}$$

The transformed data $\boldsymbol{X}_{\mathrm{ur}}$ is then processed through the student network $f_S(\theta)$, and the resulting prediction is transformed back:

$$\boldsymbol{Y}_{\mathrm{ur}} = R^{-1}(f_S(\boldsymbol{X}_{\mathrm{ur}};\theta)) \tag{5.14}$$

In this case, $R^{-1}$ is the inverse transformation (e.g. - 36 degrees) rotation.

The teacher network $f_T(\overline{\theta})$ then generates a prediction for the original, untransformed data:

$$\boldsymbol{Y}_{\mathrm{u}} = f_T(\boldsymbol{X}_{\mathrm{u}};\overline{\theta}) \tag{5.15}$$

The consistency loss can then be computed between $\boldsymbol{Y}_{\mathrm{ur}}$ and $\boldsymbol{Y}_{\mathrm{u}}$:

$$L_{\text{consistency}} = L(\boldsymbol{Y}_{\mathrm{ur}}, \boldsymbol{Y}_{\mathrm{u}}) \tag{5.16}$$

This scheme capitalizes on the idea that applying a transformation to the input data and the corresponding inverse transformation to the network's output should not change the underlying segmentation. By enforcing consistency between the network's predictions on transformed and original data, the network can effectively learn from unlabeled data, thereby improving the performance of semi-supervised learning.

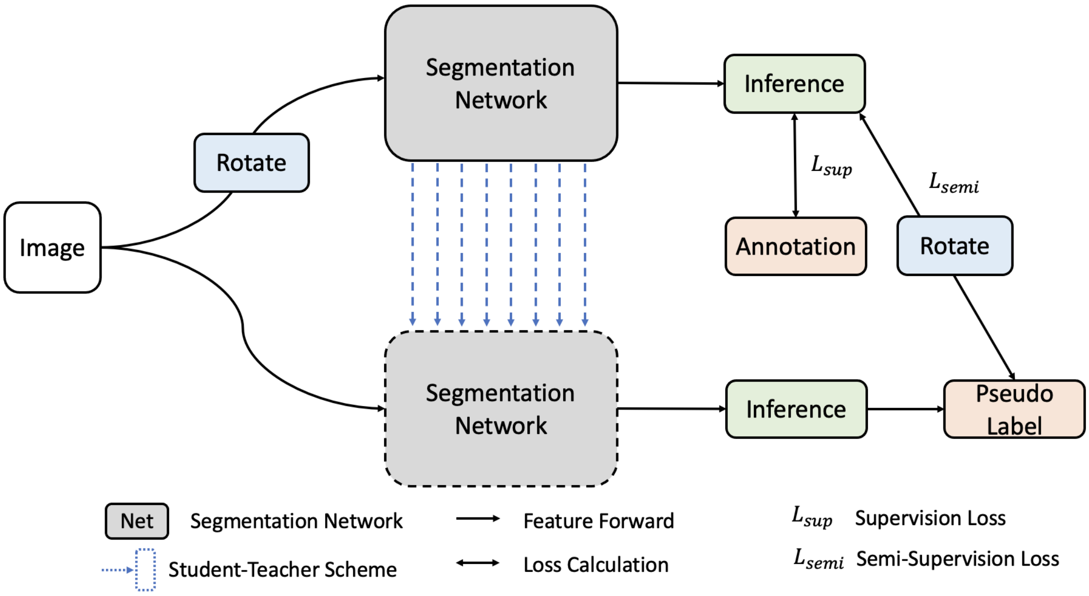


**Figure 5.7:** The Example Transformation Consistency Scheme with Mean Teacher for Medical Image Segmentation. The inference from rotated input image is similar with the rotated inference from input image.

## 5.3 Methods

### 5.3.1 Semi-Supervised Learning Framework Setup

In the task of semi-supervised learning, $\mathbf{D}_{train}$, $\mathbf{D}_{unlabel}$, $\mathbf{D}_{test}$ normally denote labeled training dataset, unlabeled training dataset, and testing set. We denote a batch of labeled data as $(X, Y_{\mathrm{gt}}) \in \mathbf{D}_{train}$, $(X, Y_{\mathrm{gt}}) \in \mathbf{D}_{test}$, and a batch of only raw data as $(X) \in \mathbf{D}_{unlabel}$ in unlabeled dataset, where $X \in \mathbb{R}^{h \times w}$ representing a 2D image with the size $h \times w$. $Y_{\mathrm{p}}$ is the dense map predicted by several segmentation networks such as $f_1(\theta) : \boldsymbol{X} \mapsto Y_1$, and $f_2(\theta) : X \mapsto Y_2$. $\mathcal{L}_{\mathrm{sup}} : (Y, Y_{\mathrm{gt}}) \mapsto \mathbb{R}, \mathcal{L}_{\mathrm{semi}} : (Y_1, Y_2) \mapsto \mathbb{R}$ represent supervised segmentation loss, and semi-supervised consistency loss on labeled training set and unlabeled training set. In general, the training updates the parameter $\theta$ of segmentation networks $f_1(\theta), f_2(\theta)$ aiming to minimize the combined loss $\mathcal{L}$. The final evaluation is to measure the difference of $(Y_{\mathrm{p}}, Y_{\mathrm{gt}}) \mapsto \mathbb{R}$ on the test set.

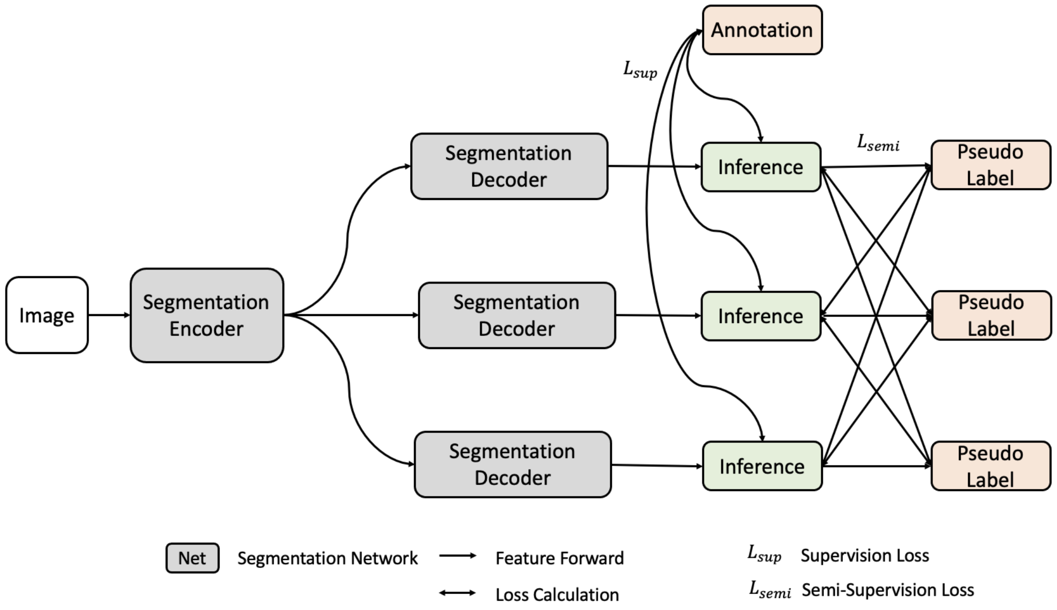


**Figure 5.8:** The Framework of Multi-View Learning for Medical Image Segmentation, Which Consisting of Multi Networks to Achieve Multi Cross Pseudo Label Supervision.

### 5.3.2 Triple-View Learning

This is part of our past work on triple-view learning [170]. The architecture of our Triple-View Learning (TVL) for medical image semantic segmentation network is illustrated in Figure 5.8. TVL is motivated by the concept of Cross Pseudo Supervision (CPS) [239], wherein multiple unique perspectives (three in our study) are simultaneously developed in separate networks. This approach allows each network to complement the others, enhancing the overall learning process. Distinctively, each network processes separate subsets of the data, learning from these segmented portions to build a comprehensive understanding.

While multi-view co-training methods have been proposed for classification tasks [238], semantic segmentation is much challenging. Our method innovatively generates pseudo-labels at various stages, leveraging the output from two networks to train the other network. This process is distinct from uncertainty-aware schemes [25, 167], focusing on propagating high-confidence pseudo-labels, thereby enhancing the overall framework's certainty. TVL utilizes pseudo-labels generated by networks with high confidence (as indicated by a high dice-coefficient). The confidence threshold for pseudo-label propagation is adjusted based on the training stage, progressively increasing as the training advances. Consequently, the quantity of training data is dynamically expanded. TVL lies a shared low-level feature learning module, a pre-trained ResNet [56], which is integral to the high-level feature learning classifiers denoted as $f_{\mathrm{A}}$, $f_{\mathrm{B}}$, and $f_{\mathrm{C}}$. Each classifier, based on different architectures—FPN [22], LinkNet [205], and UNet [4]—focuses on varied aspects of feature learning, including long-range dependencies, feature size variation, and multi-scale feature processing. The shared ResNet module aids in extracting universal low-level features, while the classifiers collaborate in feature extraction, voting, and pseudo-label generation, mutually enriching the SSL process.

### 5.3.3 Examiner-Student-Teacher Learning

This is part of our past work on Examiner-Student-Teacher [248], and the architecture of our exigent examiner and mean teacher is sketched in Figure 5.10. we explore the consistency-training-based SSL with further adversarial training scheme via extending the Student-Teacher style prototype with an Examiner paradigm, creating an Examiner-Student-Teacher SSL framework. The proposed framework consists of three 3D CNN-based networks that can make the most of a training set which includes some annotated images as well as some unannotated raw data. Adversarial training and consistency regularization are proposed via Examiner ↔ Student, and Teacher ↔ Student respectively during the training process.

**Student Network** In order to exploit the 3D nature of some MRI scan volumes, a 3D UNet is used as the Student network $f_S(\theta)$ [173]. Each network block of this UNet is based on 3D convolutional operations, Batch Normalization and DropOut shown in Figure 7.2. Like in [237], $f_S(\theta)$ learns directly from data with annotations ($\boldsymbol{X}_{\mathrm{l}}$, $\boldsymbol{Y}_{\mathrm{gt}}$), and supervised by the Teacher via pseudo labels ($\boldsymbol{X}_{\mathrm{u}}$, $\boldsymbol{Y}_{\mathrm{p}}$). The crucial difference is that, at the same time, $f_S(\theta)$ is also validated against the Examiner via adversarial training. The inference of student network $f_S(\theta)$ is given in Eq. 5.17, where Gaussian noise is applied to all input data (both labeled and unlabeled) $\mathbf{X} = \boldsymbol{X}_{\mathrm{l}} \cup \boldsymbol{X}_{\mathrm{u}}$ during training.

$$\boldsymbol{Y}_{\mathrm{p}} = f_{\mathrm{s}}(\boldsymbol{X} + \mathit{Noise}; \theta_{\mathrm{s}}) \tag{5.17}$$

**Teacher Network** The architecture of the Teacher $f_T(\overline{\theta})$ is similar to $f_S(\theta)$, except the Teacher does not learn from data directly. It is updated from the exponential moving average network weights (illustrated in Eq. 5.18) of $f_S(\theta)$, and it is more likely than the Student to predict the correct inference. The Teacher's predictions are considered as the pseudo labels to supervise the Student [237].

$$\overline{\theta} = \alpha\theta_{t-1} + (1 - \alpha)\theta_t \tag{5.18}$$

where $\overline{\theta}$ is updated based on the Student parameter $\theta_t$ from the previous training step $t$; weight factor $\alpha = 1 - \frac{1}{t+1}$. The pseudo labels are generated by the Teacher without noise as:

$$\boldsymbol{Y}_{\mathrm{p}} = f_{\mathrm{T}}(X_{\mathrm{u}}; \overline{\theta}) \tag{5.19}$$

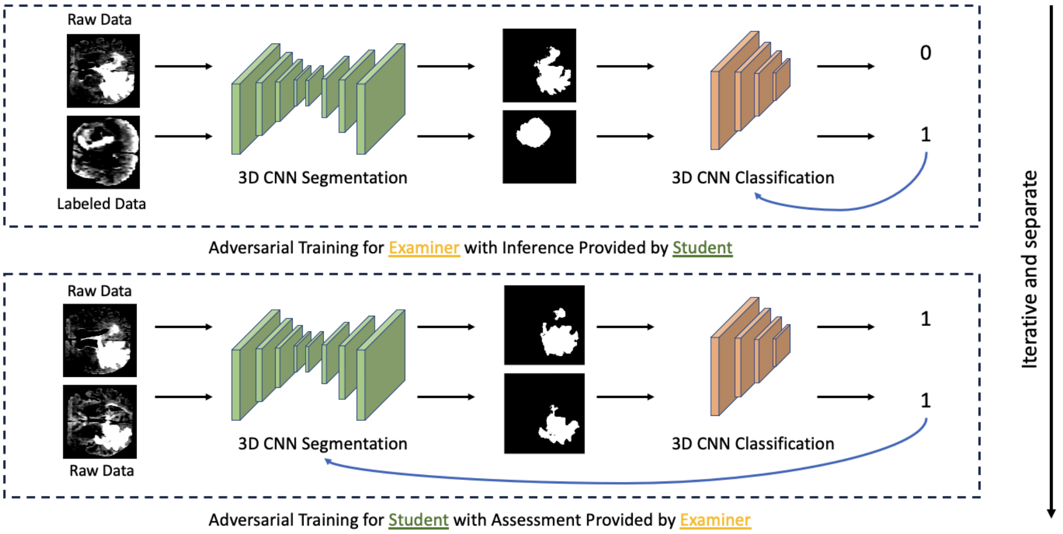


**Figure 5.9:** The Illustration of Adversarial Training Between Examiner and Student. In two training stages, Examiner is trained to classify the inference from Student as to whether from labeled or unlabeled data. Student is trained to provide high-quality inference from unlabeled data that make examiner to classify as labeled data.

Thus the unlabeled training set $X_u$ can be utilized to train the Student with $(\mathbf{X}_{\mathrm{u}}, \boldsymbol{Y}_{\mathrm{p}})$.

**Examiner Network** In order to capitalize on adversarial learning [249], a 3D CNN-based discriminator is adopted to assess the quality of the Student's inference. This matches the metaphor for an Examiner which checks the quality of the learning. The Examiner consists of four 3D CNN layers, a down-sampling operation, and multi-linear layers shown in Figure 5.10. Its architecture is following the classical VGGNet [148]. The Examiner and Student are trained against each other repeatedly for the duration of the training. The Examiner classifies the quality of the inference from student network (seen in Figure 5.9).

$$\boldsymbol{Y}_{\mathrm{e}} = f_{\mathrm{E}}(\boldsymbol{Y}_{\mathrm{p}}; \theta_{\mathrm{e}}) \tag{5.20}$$

Here, a segmentation mask predicted by the Student, $\boldsymbol{Y}_{\mathrm{p}}$, originating from a ground-truth label $\boldsymbol{Y}_{\mathrm{gt}}$, is marked as a *pass*, whereas an inference from a pseudo label is marked as a *fail* ($Y_{\mathrm{e}} \in [pass, fail]$). As the adversarial training progresses, the dynamics between the Student and Examiner evolve significantly. The Student, constantly challenged by the Examiner's evaluations, is encouraged to refine its inferences, striving to make them indistinguishable from high-quality, ground-truth derived predictions. In response, the Examiner, adapting to the Student's improving performance, heightens its evaluative criteria, becoming more discerning in distinguishing between inferences derived from labeled and unlabeled data. This iterative process of mutual adaptation drives the Student to generate increasingly sophisticated and reliable inferences. The Examiner, in turn, evolves to provide a more stringent and nuanced assessment, ensuring that only the most accurate predictions are classified as *pass*. This escalating cycle of improvement and challenge underpins the essence of adversarial training, fostering a robust learning environment that progressively enhances the quality of SSL segmentation.

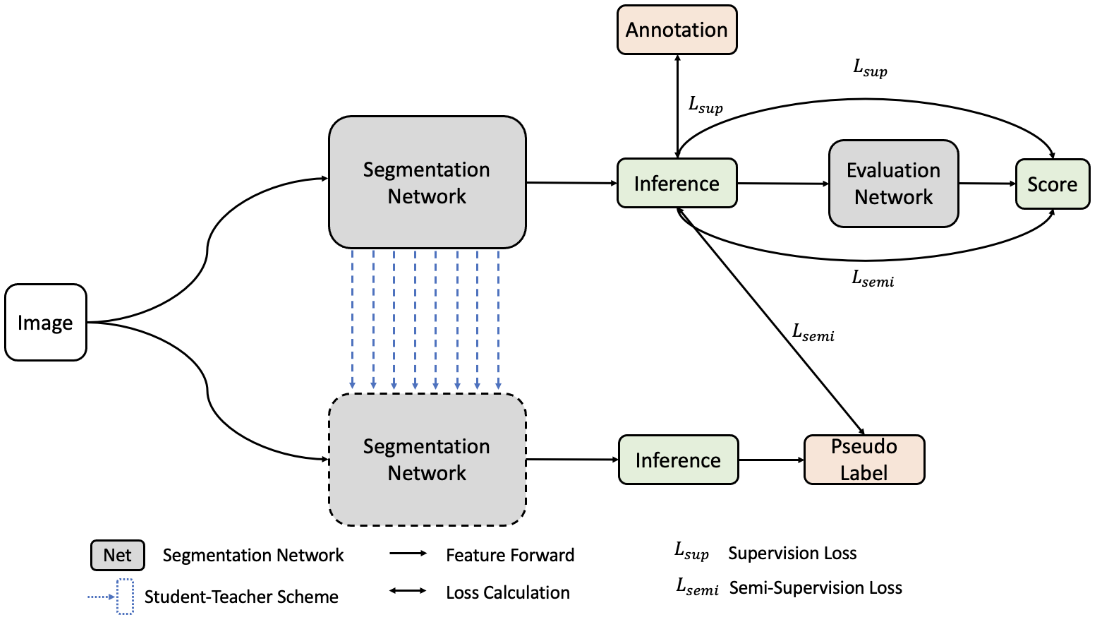


**Figure 5.10:** The Proposed Examiner-Student-Teacher Framework for Medical Image Segmentation.

### 5.3.4 Uncertainty-Aware Mean Teacher ViT

This is part of our past work on UAMTViT [167], and the architecture of our UAMTViT is sketched in Figure 5.11. To further enhance the teacher's supervision, an awareness of the uncertainty can be utilized in the training student network stage [241]. These schemes revolve around the concept of uncertainty estimation, which can be considered as a measure of the network's confidence in its predictions. In this scheme, alongside the typical training process, an uncertainty measure is computed for each prediction made by the network. This uncertainty measure can be derived from various sources, such as the network's output probabilities or the disagreement between ensemble Networks.

In the context of Mean Teacher [237], the network's uncertainty can be computed based on the disagreement between the Student and the Teacher Networks:

$$U = |f_S(\boldsymbol{X}\mathrm{u}; \theta) - f_T(\boldsymbol{X}_\mathrm{u}; \overline{\theta})| \tag{5.21}$$

where $U$ represents the uncertainty measure, and $|.|$ denotes the absolute difference operation.

The uncertainty measure is then used to modulate the learning process. For instance, predictions with high uncertainty can be given less weight during the loss calculation, or they can be excluded from the training process altogether.

$$Loss = U \cdot L(\boldsymbol{Y}\mathrm{u}, \boldsymbol{Y}\mathrm{p}) \tag{5.22}$$

where $L$ represents the loss function, and $Loss$ is the final, uncertainty-weighted loss.

By incorporating uncertainty measures into the learning process, the student network is forced to only learn the pseudo label where the teacher is confident about, while treating uncertain predictions with caution. This can lead to more robust learning, as the network becomes less prone to overfitting on uncertain or noisy predictions. As such, uncertainty-aware schemes can greatly enhance the performance of SSL methods, making them more reliable and robust in practice.

### 5.3.5 Computational-Efficient Cross Supervision ViT

This is part of our past work on CESSViT [167], and the architecture of our CESSViT is sketched in Figure 5.12. The key parameters are carefully optimized: the patch size is set to 16 × 16, the multi-head count in the self-attention sub-layer is fixed at 6, and the encoder featured 12 identical layers. Additionally, the decoder is designed with

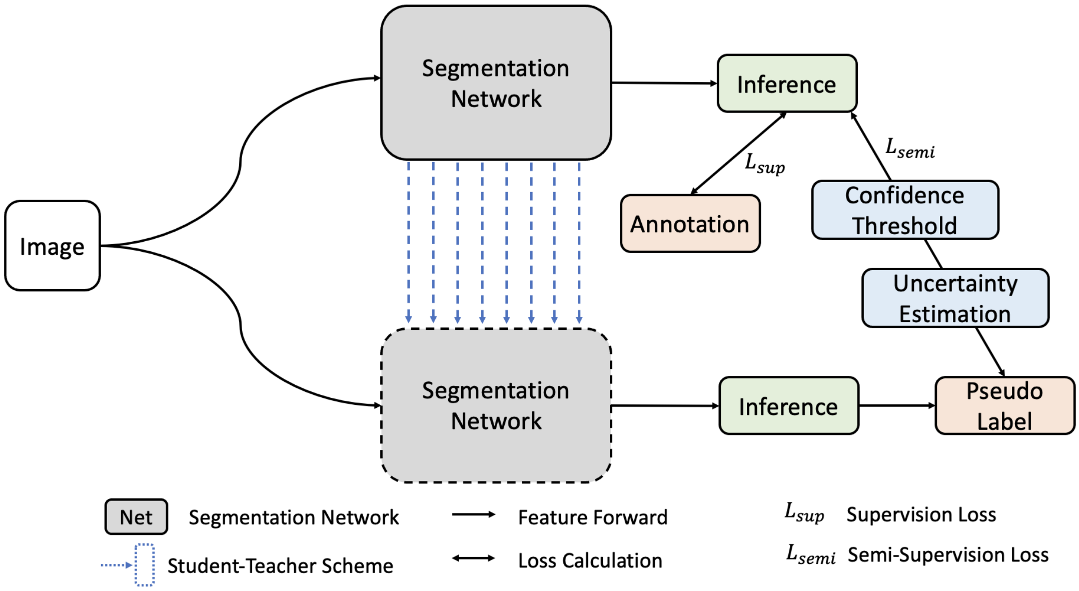


**Figure 5.11:** The Example Uncertainty-Aware Scheme with Mean Teacher for Medical Image Segmentation.

2 identical layers, each producing an output dimension of 384 from the self-attention layer. This architecture is formulated to balance performance with computational efficiency, making it well-suited for semi-supervised training in segmentation tasks.

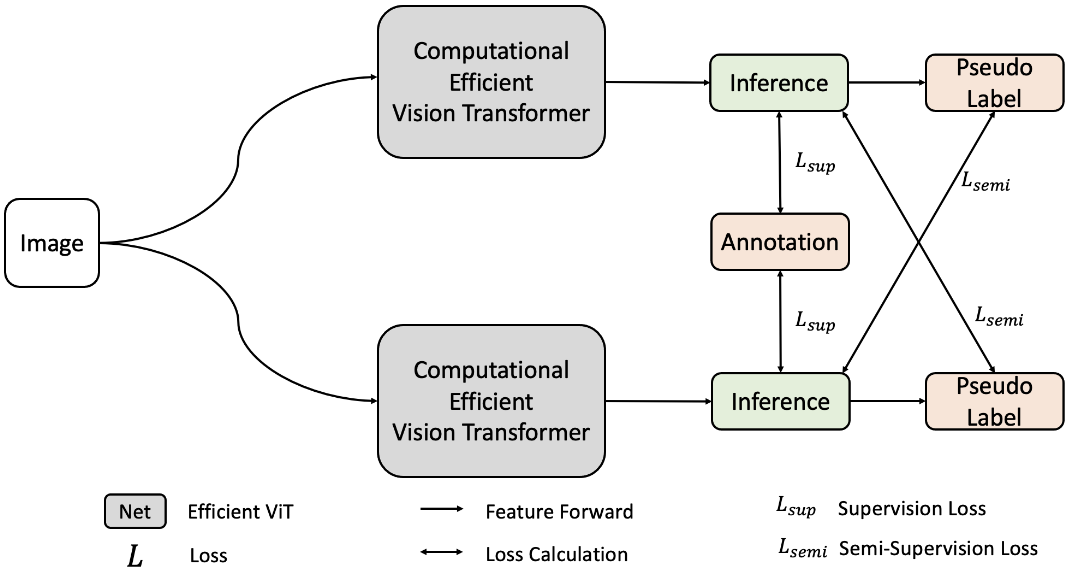


**Figure 5.12:** The Example Computational-Efficient Vision Transformer for Semi-Supervised Medical Image Segmentation.

## 5.4 Experiments and Results

### 5.4.1 Implementation Details

All the proposed SSL methods, along with baseline methods, are trained with the same hyper-parameters setting. This includes a training duration of 30,000 iterations, a batch size of 24, and the use of the SGD optimizer (learning rate: 0.01, momentum: 0.9, weight decay: 0.0001). Network performance is evaluated on the validation set every 200 iterations, saving the network weights only if there is an improvement in performance compared to the previous best. The segmentation backbone networks are either CNN-based Encoder-Decoder network UNet [4], ViT-based Encoder-Decoder network SwinUNet [182], or computational efficient SegFormer [7]. We evaluate Deep Adversarial Network (DAN) [249], ADVENT [250], ICT [234], Mean Teacher (MT), Uncertainty-Aware Mean Teacher (UAMT) [241], CPS [239], FixMatch [235], Triple-View Learning (TVL) [170], Uncertainty-Aware ViT via Mean Teacher (UAMViT) [167], Computational-Efficient Segmentation ViT (CESSViT) [168], and Exigent Examiner and Mean Teacher (EEMT) [248] on two public available benchmark dataset including MRI Cardiac [8], and MRI Brain Tumour dataset [9]. Ultrasound nerve [221], histology nuclei [169], and CT spine datasets [219] are further selected to validate three CNN-based FPN [22], LinkNet [205], UNet [4] and their combination for SSL in TVL [170]. We split the dataset 80% as training set, and 20% as testing set.

### 5.4.2 Qualitative Results

Figure 5.13, Figure 5.14, and Figure 5.15, illustrates eight examples on MRI Cardiac segmentation test set with various SSL strategies and 2D CNN-based segmentation network when 10%, 30%, and 50% of training set as labeled training set. Figure 5.16, Figure 5.17, and Figure 5.18, illustrates eight examples on MRI Cardiac segmentation test set with various SSL strategies and 2D ViT-based segmentation network when 10%, 30%, and 50% of training set as labeled training set. Figure 5.19 illustrates eight examples on MRI Cardiac segmentation test set with 2D CNN-based and

2D ViT-based segmentation network when all the training set as labeled training set, i.e. fully supervised learning.

Figure 5.20, Figure 5.21, and Figure 5.22, illustrates eight examples on MRI brain tumour segmentation test set with various SSL strategies and 3D CNN-based segmentation network when 10%, 30%, and 50% of training set as labeled training set. Figure 5.23 illustrates eight examples on MRI brain tumour segmentation test set with 3D CNN-based segmentation network when all the training set as labeled training set, i.e. fully supervised learning.

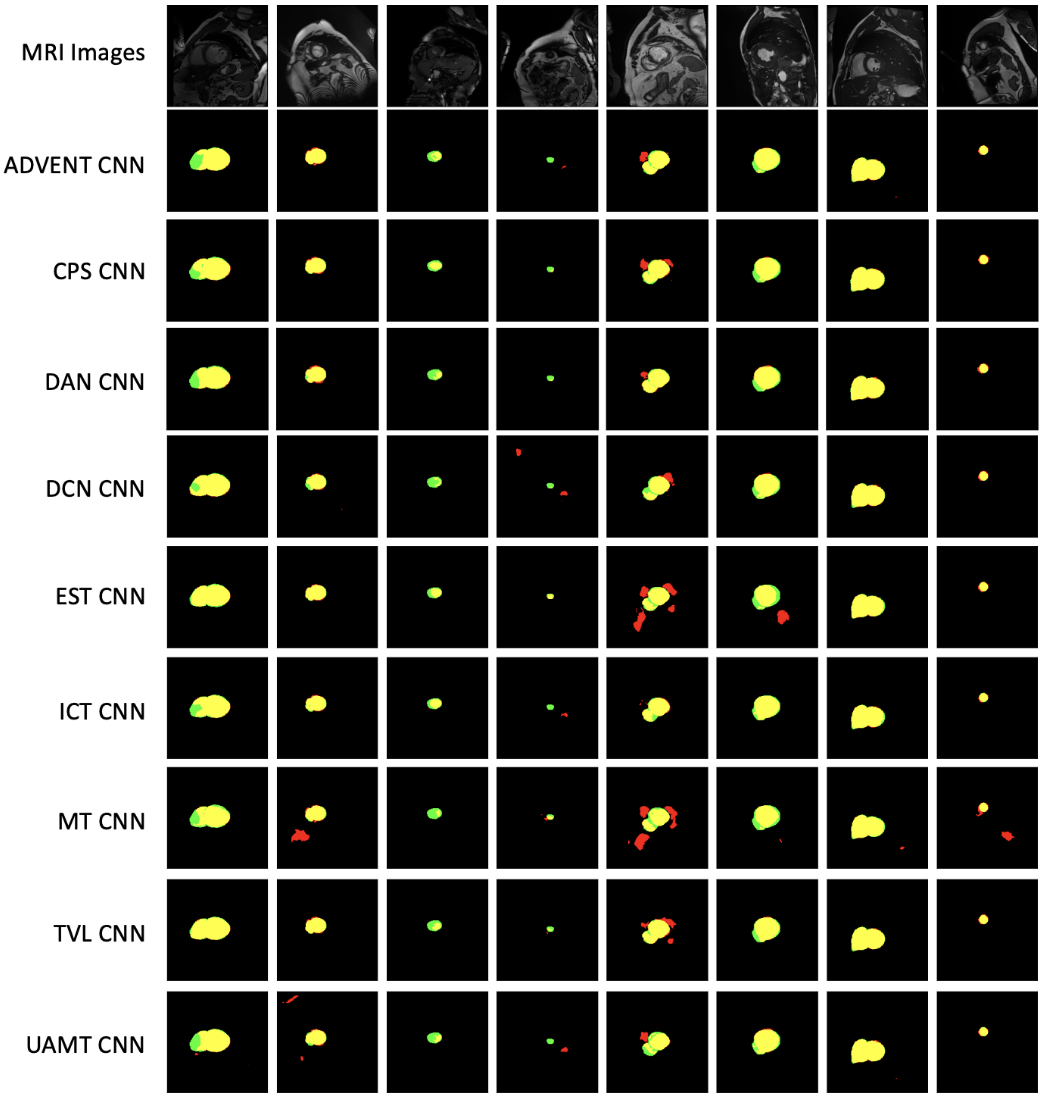


**Figure 5.13:** The Example MRI Cardiac Segmentation Inference with 2D CNN network and All SSL Strategies when 10% of Training Set as Labeled Data.

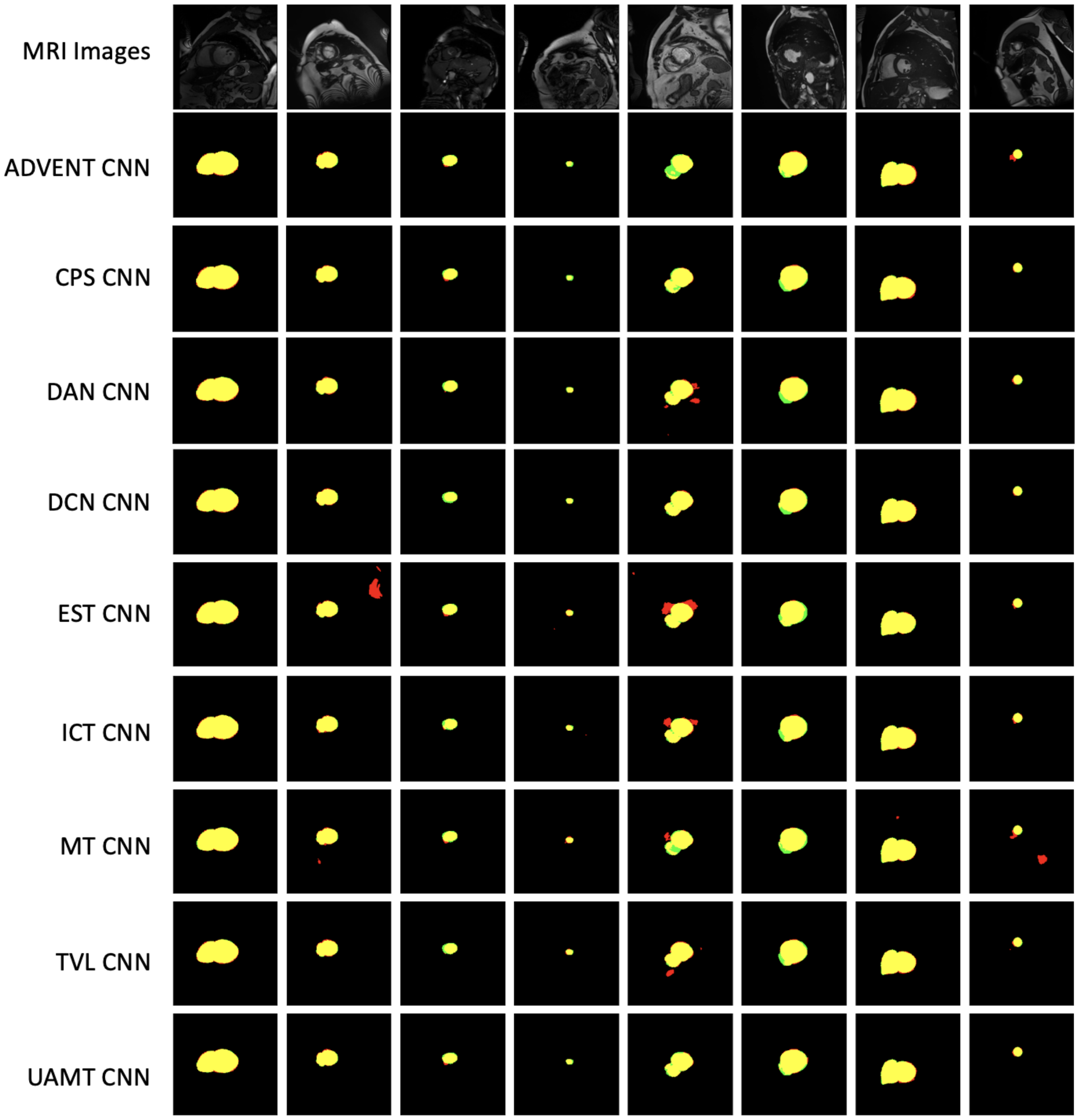


**Figure 5.14:** The Example MRI Cardiac Segmentation Inference with 2D CNN network and All SSL Strategies when 30% of Training Set as Labeled Data.

### 5.4.3 Quantitative Results

Table 5.1 and Table 5.2 reports the direct comparison of all SSL methods including similarity measures and difference measures when the ratio of assumed labelled/total data is 10%. Table 5.3, and Table 5.4 further reports all the SSL methods performance under different assumptions of ratio of labeled/total data. Our proposed methods are highlighted with **Bold**. The best performance are with **Bold**, and the second best performance with the proposed methods are with Underline.

**Table 5.1:** The Performance of All SSL Methods and FSL Methods on MRI Cardiac Test Set.

| Strategy | Net | Dice | Acc | Pre | Sen | Spe | HD | ASD |
|---|---|---|---|---|---|---|---|---|
| DAN [249] | 2D CNN | 0.8522 | 0.9904 | 0.8887 | 0.8186 | **0.9964** | 10.3342 | 2.2456 |
| ADVENT [250] | 2D CNN | 0.8785 | 0.9949 | 0.8919 | 0.8666 | 0.9664 | 7.8870 | 2.4178 |
| ICT [234] | 2D CNN | 0.8907 | 0.9956 | **0.9041** | 0.8791 | 0.9710 | **7.8132** | 2.2884 |
| DCN [251] | 2D CNN | 0.8786 | 0.9949 | 0.8890 | 0.8694 | 0.9674 | 10.8204 | 3.3456 |
| MT [237] | 2D CNN | 0.8597 | 0.9940 | 0.8579 | 0.8617 | 0.9683 | 18.4509 | 5.1873 |
| UAMT [241] | 2D CNN | 0.8752 | 0.9948 | 0.8888 | 0.8646 | 0.9656 | 11.3137 | 2.9892 |
| CPS [239] | 2D CNN | 0.8911 | 0.9956 | 0.8935 | 0.8900 | 0.9752 | 8.6877 | 2.5044 |
| ADVENT [250] | 2D ViT | 0.8654 | 0.9949 | 0.8689 | 0.8625 | 0.9743 | 20.4530 | 1.9021 |
| ICT [234] | 2D ViT | 0.8626 | 0.9947 | 0.8577 | 0.8684 | 0.9766 | 24.4194 | 2.5312 |
| DCN [251] | 2D ViT | 0.8705 | 0.9950 | 0.8691 | 0.8733 | 0.9766 | 21.7381 | 2.0275 |
| DAN [249] | 2D ViT | 0.8232 | 0.9932 | 0.8243 | 0.8222 | 0.9686 | 23.5825 | 2.8410 |
| MT [237] | 2D ViT | 0.8597 | 0.9947 | 0.8683 | 0.8516 | 0.9717 | 22.8486 | 2.2084 |
| **TVL** [170] | **2D CNN** | **0.8965** | **0.9957** | 0.8978 | 0.8967 | 0.9764 | 8.9242 | 2.3171 |
| **UAMT** [241] | **2D ViT** [167] | 0.8639 | 0.9949 | 0.8635 | 0.8656 | 0.9765 | 23.6093 | 1.8273 |
| **CPS** [239] | **2D ViT** [168] | 0.8907 | 0.9955 | 0.8793 | **0.9035** | 0.9826 | 19.5130 | **1.7960** |
| **EEMT** [248] | **2D CNN** | 0.8944 | 0.9869 | 0.8924 | 0.8964 | 0.9929 | 6.7235 | 2.3468 |
| FSL | 2D CNN | 0.9362 | 0.9974 | 0.9285 | 0.9445 | 0.9902 | 2.8905 | 0.6798 |
| FSL | 2D ViT | 0.9213 | 0.9969 | 0.9311 | 0.9124 | 0.9810 | 4.9625 | 0.4542 |

**Table 5.2:** The Performance of All SSL Methods and FSL Methods on MRI Brain Test Set.

| Strategy | Net | Dice | Acc | Pre | Sen | Spe | HD | ASD |
|---|---|---|---|---|---|---|---|---|
| DAN [249] | 3D CNN | 0.8522 | 0.9904 | 0.8887 | 0.8186 | 0.9964 | **10.3342** | 2.2456 |
| ADVENT [250] | 3D CNN | 0.7784 | 0.9868 | 0.8933 | 0.6897 | 0.9971 | 19.2713 | 3.9854 |
| ICT [234] | 3D CNN | 0.8444 | 0.9901 | 0.8926 | 0.8012 | 0.9966 | 12.0808 | 2.3638 |
| MT [237] | 3D CNN | 0.8554 | 0.9907 | 0.8945 | 0.8197 | 0.9966 | 14.1178 | 2.3128 |
| UAMT [241] | 3D CNN | 0.8427 | 0.9899 | 0.8840 | 0.8051 | 0.9963 | 12.0487 | 2.3872 |
| CPS [239] | 3D CNN | 0.8581 | 0.9908 | 0.8882 | **0.8299** | 0.9964 | 12.9194 | **2.0330** |
| **TVL** [170] | **3D CNN** | 0.8286 | 0.9894 | **0.9057** | 0.7635 | **0.9972** | 20.0065 | 3.3834 |
| **EEMT** [248] | **3D CNN** | **0.8606** | **0.9910** | 0.9003 | 0.8242 | 0.9968 | 11.8949 | 2.3691 |
| FSL | 3D CNN | 0.8804 | 0.9921 | 0.9027 | 0.8591 | 0.9968 | 9.0964 | 1.8919 |

**Table 5.3:** The Direct Comparison Between Each SSL Method on MRI Cardiac Test Set Under Various Data Situations.

| Strategy | Net | 10% | | | 30% | | | 50% | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Dice | HD | ASD | Dice | HD | ASD | Dice | HD | ASD |
| DAN [249] | 2D CNN | 0.8522 | 10.3342 | 2.2456 | 0.9032 | 8.8791 | 2.3715 | 0.9221 | 3.2110 | 0.9172 |
| ADVENT [250] | 2D CNN | 0.8785 | 7.8870 | 2.4178 | 0.9162 | 5.0521 | 1.6228 | 0.9215 | 10.3330 | 2.4280 |
| ICT [234] | 2D CNN | 0.8907 | 7.8132 | 2.2884 | 0.9151 | 7.7996 | 2.0972 | 0.9222 | 9.3451 | 2.2805 |
| DCN [251] | 2D CNN | 0.8786 | 10.8204 | 3.3456 | 0.9159 | 4.6247 | 1.6323 | 0.9276 | 2.6540 | **0.6889** |
| MT [237] | 2D CNN | 0.8597 | 18.4509 | 5.1873 | 0.9115 | 10.7320 | 2.9546 | 0.9256 | 4.6768 | 1.3181 |
| UAMT [241] | 2D CNN | 0.8752 | 11.3137 | 2.9892 | 0.9172 | 5.2173 | 1.4913 | 0.9219 | 4.3713 | 1.2422 |
| CPS [239] | 2D CNN | 0.8911 | 8.6877 | 2.5044 | 0.9178 | 6.1870 | 1.6476 | 0.9256 | 5.0754 | 1.4251 |
| ADVENT [250] | 2D ViT | 0.8654 | 20.4530 | 1.9021 | 0.9044 | 14.5383 | 1.6169 | 0.9162 | 16.9461 | 1.0097 |
| ICT [234] | 2D ViT | 0.8626 | 24.4194 | 2.5312 | 0.9061 | 19.5836 | 1.6599 | 0.9169 | 12.8991 | 0.9572 |
| DCN [251] | 2D ViT | 0.8705 | 21.7381 | 2.0275 | 0.9075 | 21.3305 | 1.7129 | 0.9135 | 15.4894 | 1.1424 |
| DAN [249] | 2D ViT | 0.8232 | 23.5825 | 2.8410 | 0.8933 | 22.2748 | 1.7836 | 0.9021 | 17.3750 | 1.3412 |
| MT [237] | 2D ViT | 0.8597 | 22.8486 | 2.2084 | 0.9004 | 23.3482 | 2.0657 | 0.9154 | 15.0237 | 1.0204 |
| **TVL** [170] | **2D CNN** | **0.8965** | 8.9242 | 2.3171 | 0.9175 | 5.0569 | 1.5493 | 0.9261 | 4.6308 | 1.4368 |
| **UAMT** [241] | **2D ViT** [167] | 0.8639 | 23.6093 | 1.8273 | 0.9043 | 15.5615 | **1.3896** | 0.9155 | 15.8339 | 1.0489 |
| **CPS** [239] | **2D ViT** [168] | 0.8907 | 19.5130 | **1.7960** | 0.9095 | 15.8668 | 1.4273 | 0.9186 | 12.8070 | 0.8510 |
| **EEMT** [248] | **2D CNN** | 0.8944 | **6.7235** | 2.3468 | **0.9258** | **4.3497** | 2.3993 | **0.9357** | **2.3478** | 1.2437 |

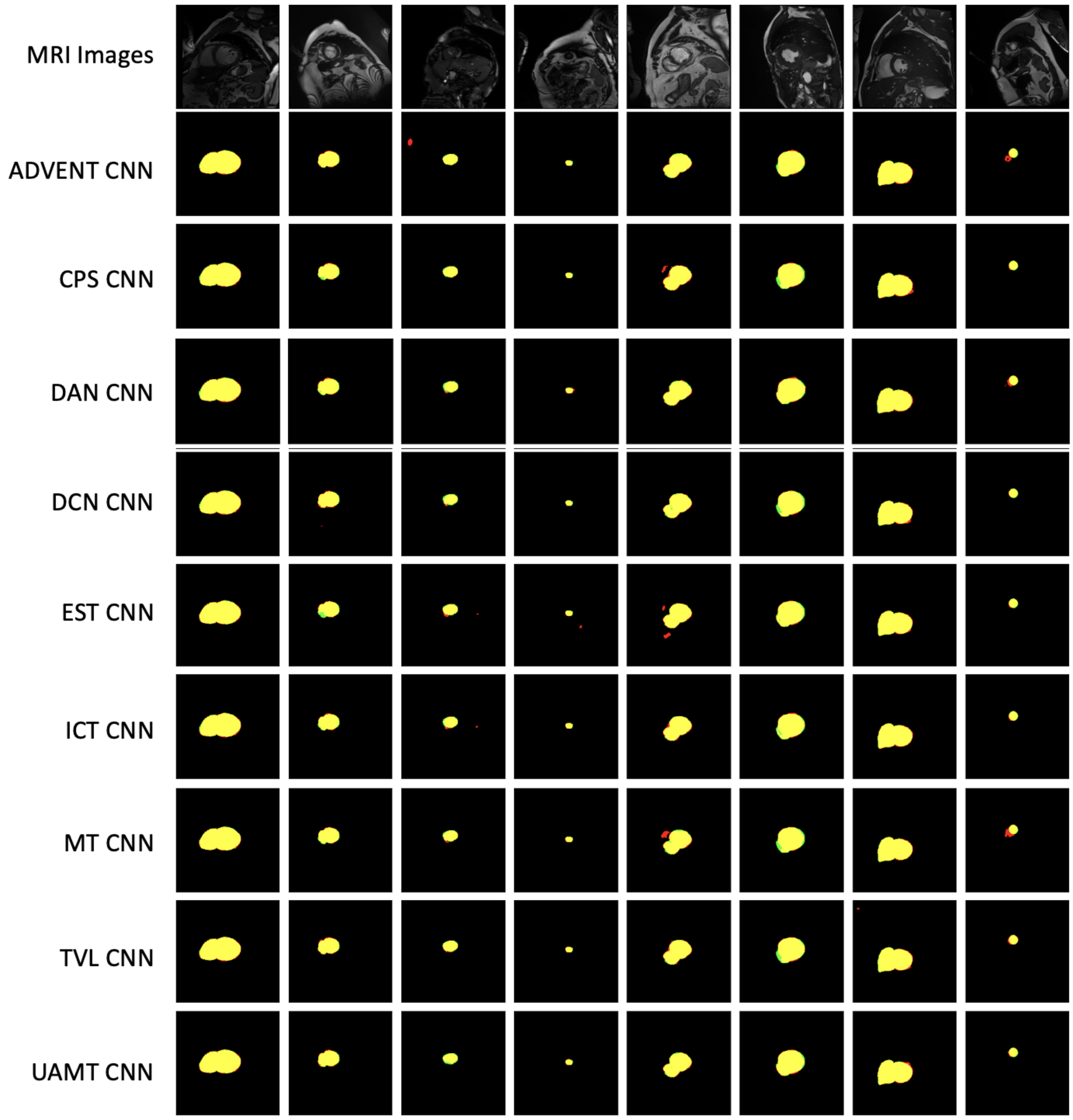


**Figure 5.15:** The Example MRI Cardiac Segmentation Inference with 2D CNN network and All SSL Strategies when 50% of Training Set as Labeled Data.

**Table 5.4:** The Direct Comparison Between Each SSL Method on MRI Brain Test Set Under Various Data Situations.

| Strategy | Net | 10% | | | 30% | | | 50% | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Dice | HD | ASD | Dice | HD | ASD | Dice | HD | ASD |
| DAN [249] | 3D CNN | 0.8522 | **10.3342** | 2.2456 | 0.8711 | 9.6285 | 2.0558 | 0.8765 | 7.8296 | 2.0352 |
| ADVENT [250] | 3D CNN | 0.7784 | 19.2713 | 3.9854 | 0.8156 | 13.7736 | 2.9624 | 0.7688 | 29.3865 | 3.5888 |
| ICT [234] | 3D CNN | 0.8444 | 12.0808 | 2.3638 | 0.8757 | 7.2862 | 2.0705 | 0.8861 | 6.7221 | 1.8699 |
| MT [237] | 3D CNN | 0.8554 | 14.1178 | 2.3128 | 0.8811 | 9.2195 | 1.9452 | 0.8889 | 7.5318 | 1.8078 |
| UAMT [241] | 3D CNN | 0.8427 | 12.0487 | 2.3872 | 0.8743 | 7.5782 | 1.8391 | 0.8844 | 7.7278 | 1.8040 |
| CPS [239] | 3D CNN | 0.8581 | 12.9194 | **2.0330** | 0.8832 | 8.2775 | **1.7738** | 0.8842 | 8.3446 | 1.8787 |
| **TVL** [170] | **3D CNN** | 0.8286 | 20.0065 | 3.3834 | 0.8741 | 10.9023 | 1.9004 | 0.8858 | 6.7873 | **1.7230** |
| **EEMT** [248] | **3D CNN** | **0.8606** | 11.8949 | 2.3691 | **0.8886** | **6.4793** | 1.8074 | **0.9046** | **6.4368** | 1.7467 |

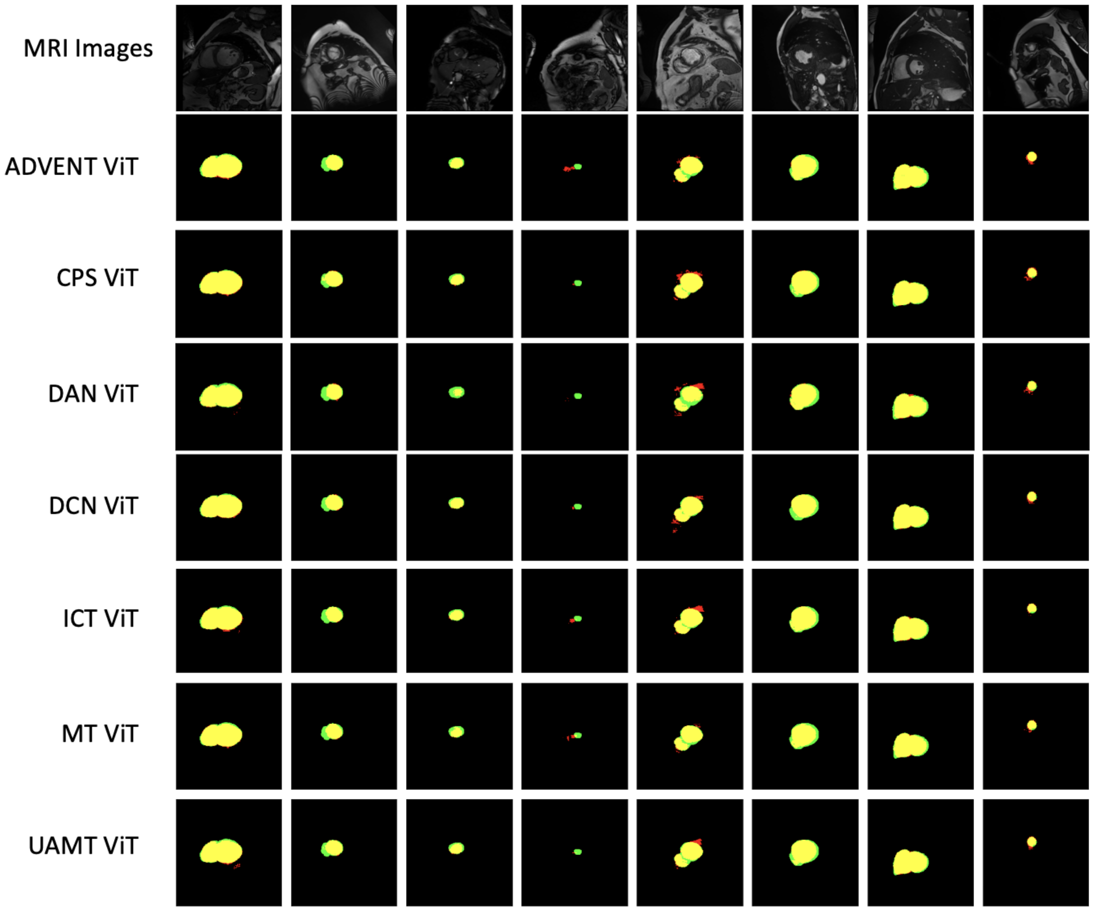


**Figure 5.16:** The Example MRI Cardiac Segmentation Inference with 2D ViT network when 10% of Training Set as Labeled Data.

### 5.4.4 Ablation Study

To assess the effects of each of the proposed contributions and their different combinations, extensive ablation study have been conducted and reported.

#### Ablation Study on UAMTViT

In our ablation study of UAMTViT, showcased in Table 5.5, the implementation of the Mean Teacher, denoted by ✓, is compulsory for SSL study purpose. We observe that omitting the uncertainty aware scheme leads to a decline in performance. The study involved testing with diverse network architectures, including UNet [4], E-Net [252], and our specifically designed segmentation ViT. Additional experiments conducted under fully supervised learning conditions are indicated by ✗in Table 5.5.

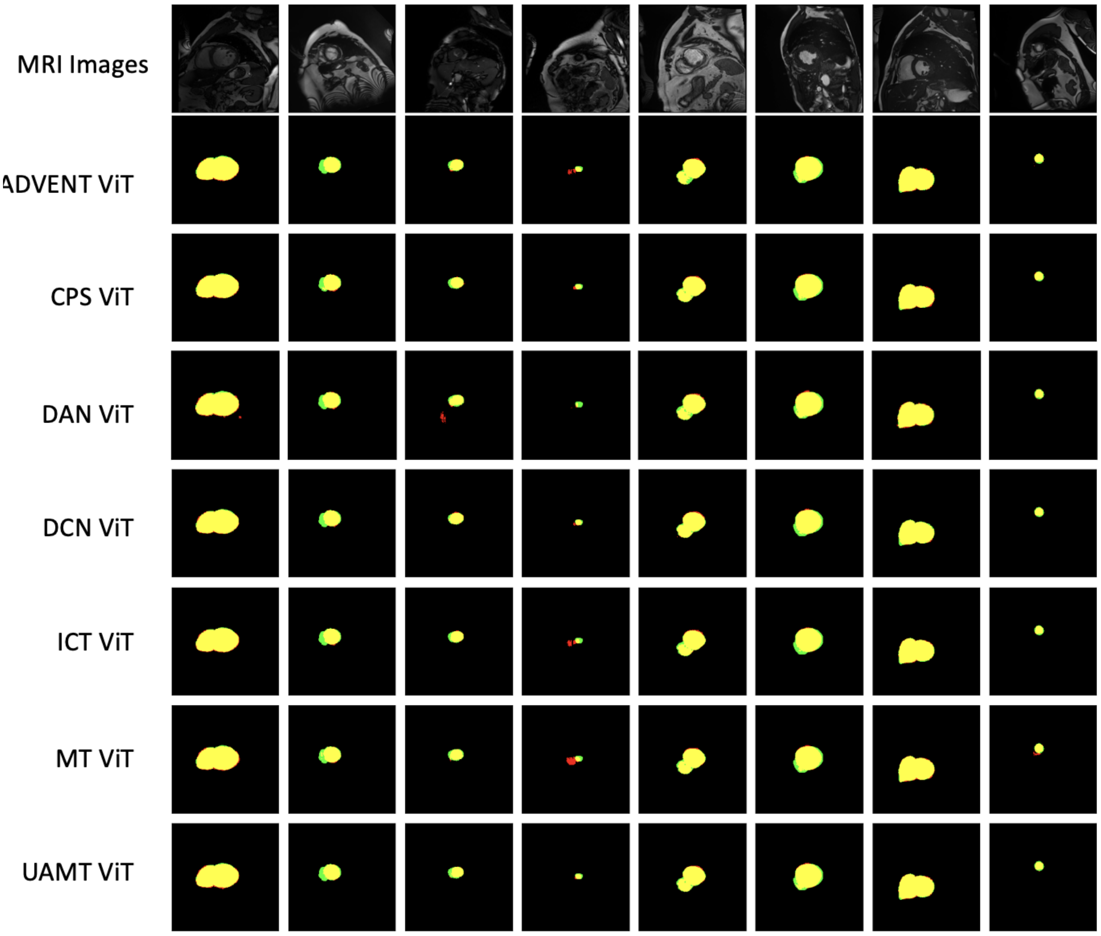


**Figure 5.17:** The Example MRI Cardiac Segmentation Inference with 2D ViT network when 30% of Training Set as Labeled Data.

These experiments reveal that our ViT, equipped with an uncertainty estimation scheme, demonstrates particularly promising performance, notably in terms of IoU and sensitivity, in both semi-supervised and fully-supervised settings.

Tables 5.6 and 5.7 present the outcomes of different settings for the threshold $\tau$ and the weight $\lambda$ of the consistency loss $\mathcal{L}_{\mathrm{c}}$ during each training iteration. Our exploration encompasses fixed values as well as dynamic approaches like exponential ramp-up, linear ramp-up, and cosine ramp-down, along with their variants. The specific formulas for these dynamic approaches are detailed in Equations 5.23, 5.24, and 5.25.

For each experiment, we apply varied approaches to update $\tau$ and $\lambda$, keeping one parameter fixed to an exponential ramp-up as a control. Our results indicate that

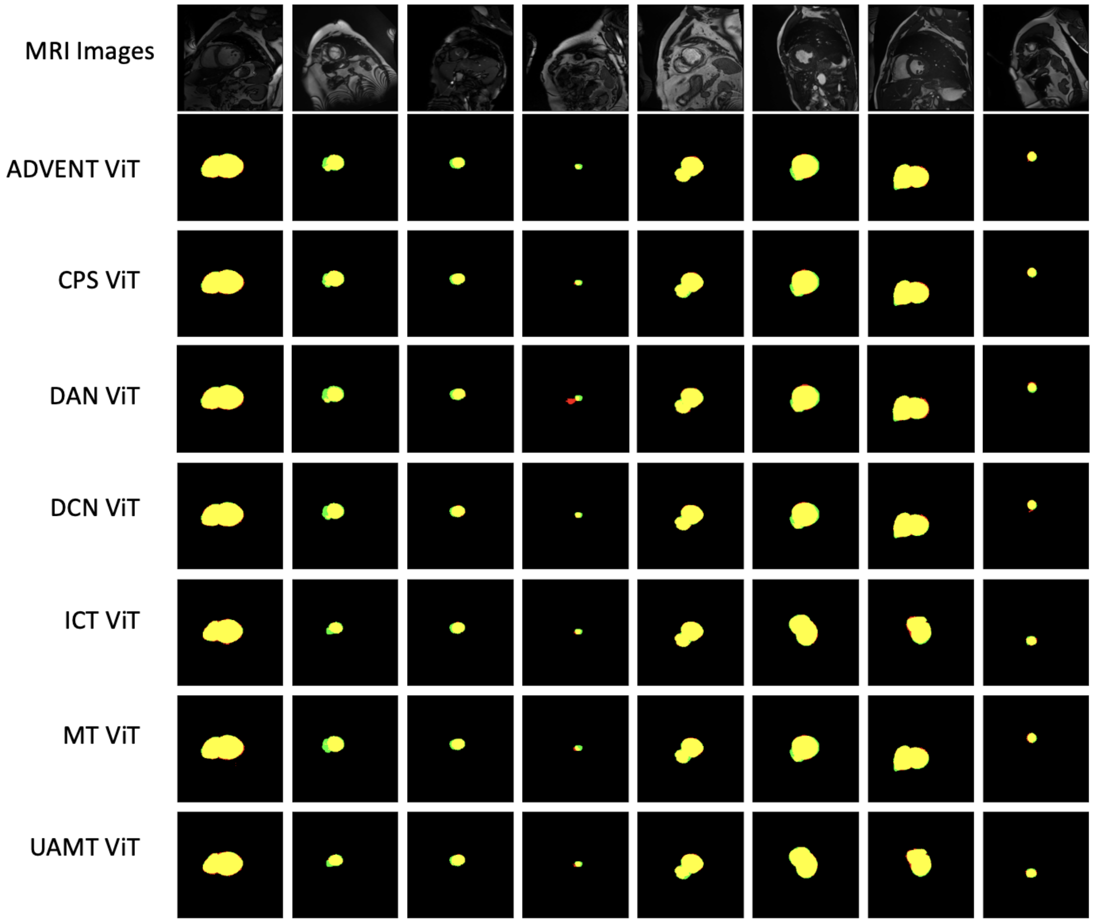


**Figure 5.18:** The Example MRI Cardiac Segmentation Inference with 2D ViT network when 50% of Training Set as Labeled Data.

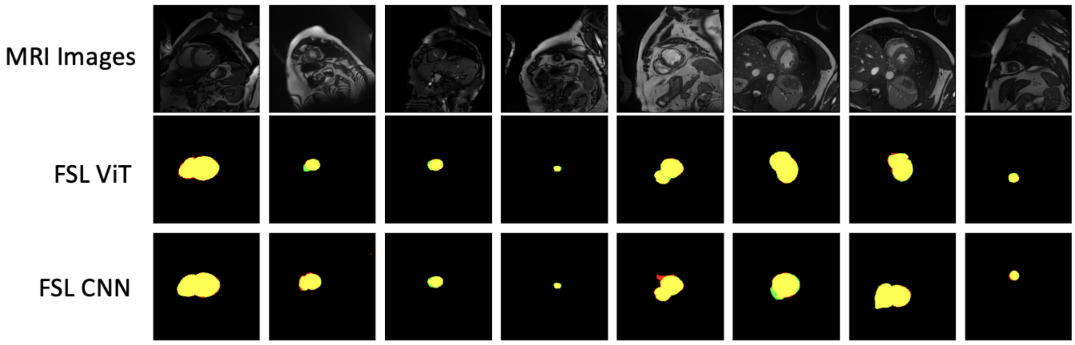


**Figure 5.19:** The Example MRI Cardiac Segmentation Inference with Fully Supervised 2D CNN- and 2D ViT-based Segmentation Network.

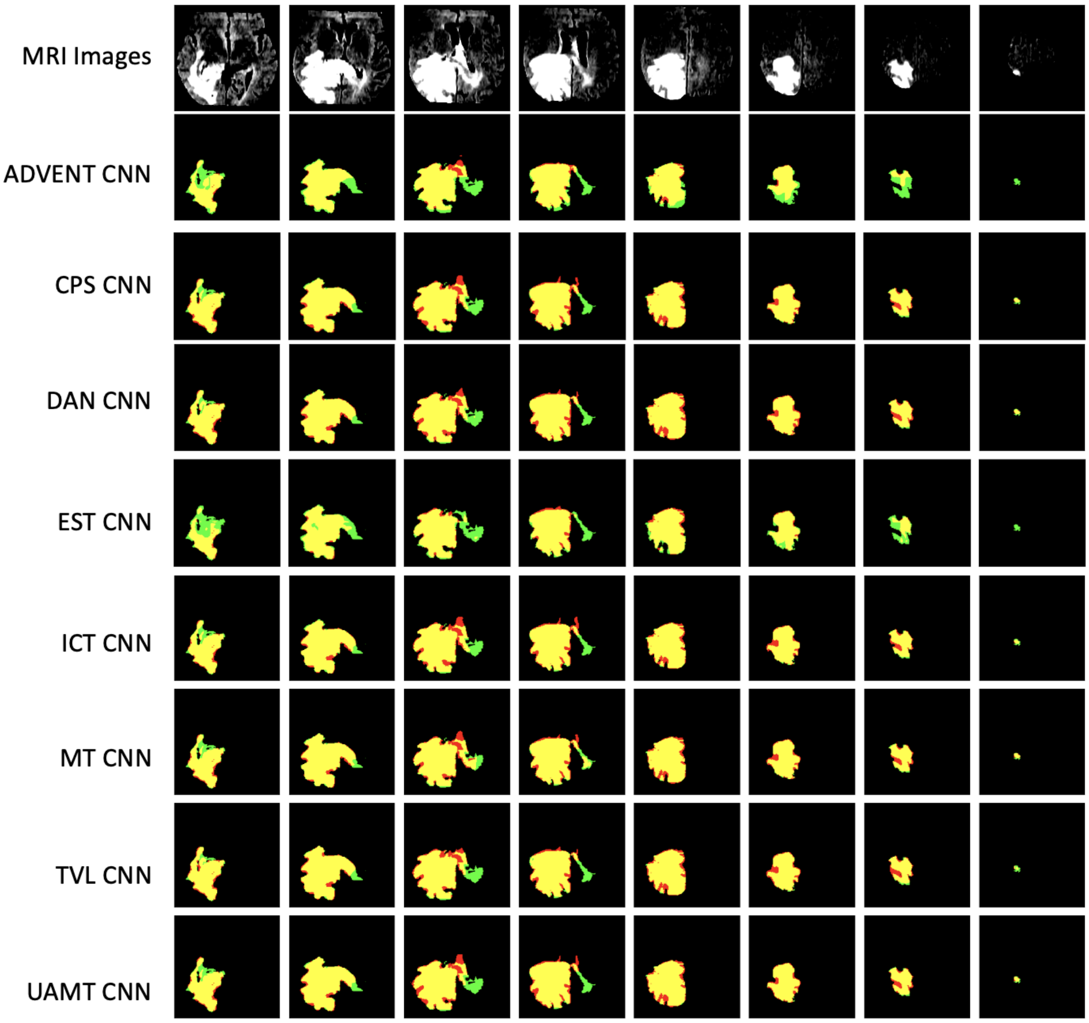


**Figure 5.20:** The Example MRI Brain Tumour Segmentation Inference with 3D CNN Network when 10% of Training Set as Labeled Data.

the different methodologies for updating $\tau$ and $\lambda$ each iteration do not markedly enhance the performance of our proposed method. Consequently, for all further experiments involving $\tau$ and $\lambda$, we maintained an exponential ramp-up setting.

$$\tau or \lambda = e^{-5\times(1-t_{\text{iteration}}/t_{\text{maxiteration}})^2} \tag{5.23}$$

$$\tau or \lambda = t_{\text{iteration}}/t_{\text{maxiteration})} \tag{5.24}$$

$$\tau or \lambda = 0.5 \times (cosine(\pi \times t_{\text{iteration}}/t_{\text{maxiteration}}) + 1) \tag{5.25}$$

Figure 5.24 indicates a selection of input MRI raw images with their corresponding uncertainty maps and generated masks at various stages of the training process

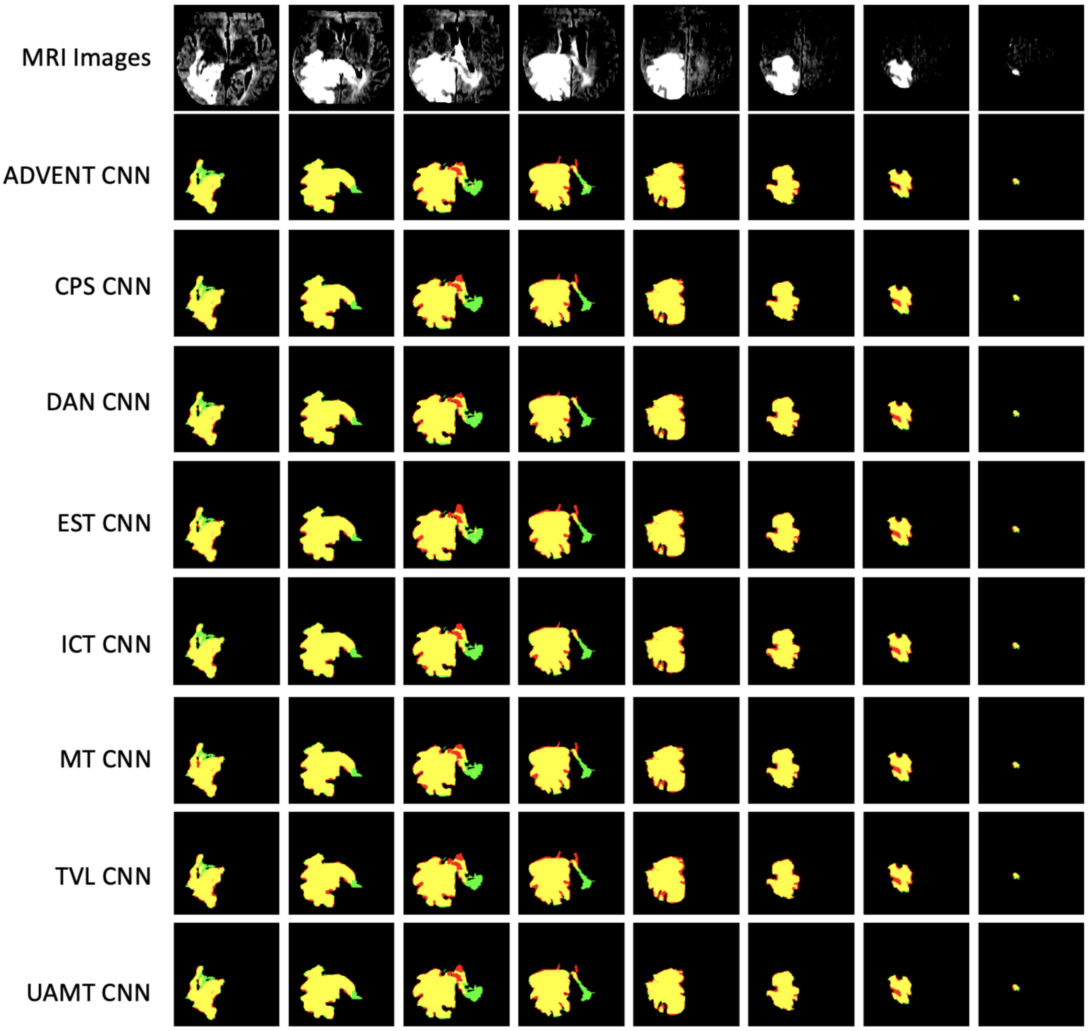


**Figure 5.21:** The Example MRI Brain Tumour Segmentation Inference with 3D CNN Network when 30% of Training Set as Labeled Data.

by segmentation networks. In these uncertainty maps, areas of high uncertainty in the teacher ViT's ($f_t$) predictions are highlighted in yellow, while regions of high certainty are marked in blue.

As training progresses, the uncertainty map shows a transition from yellow to green. This shift signifies a gradual increase in prediction certainty. The application of a certainty threshold to the uncertainty map results in the generation of masks, where white areas indicate sufficient certainty in the teacher ViT's ($f_t$) predictions to guide the student ViT ($f_s$) in calculating the consistency loss $\mathcal{L}_s$. Conversely, black areas denote pixels too uncertain to be considered in the consistency semi-supervision loss calculation.

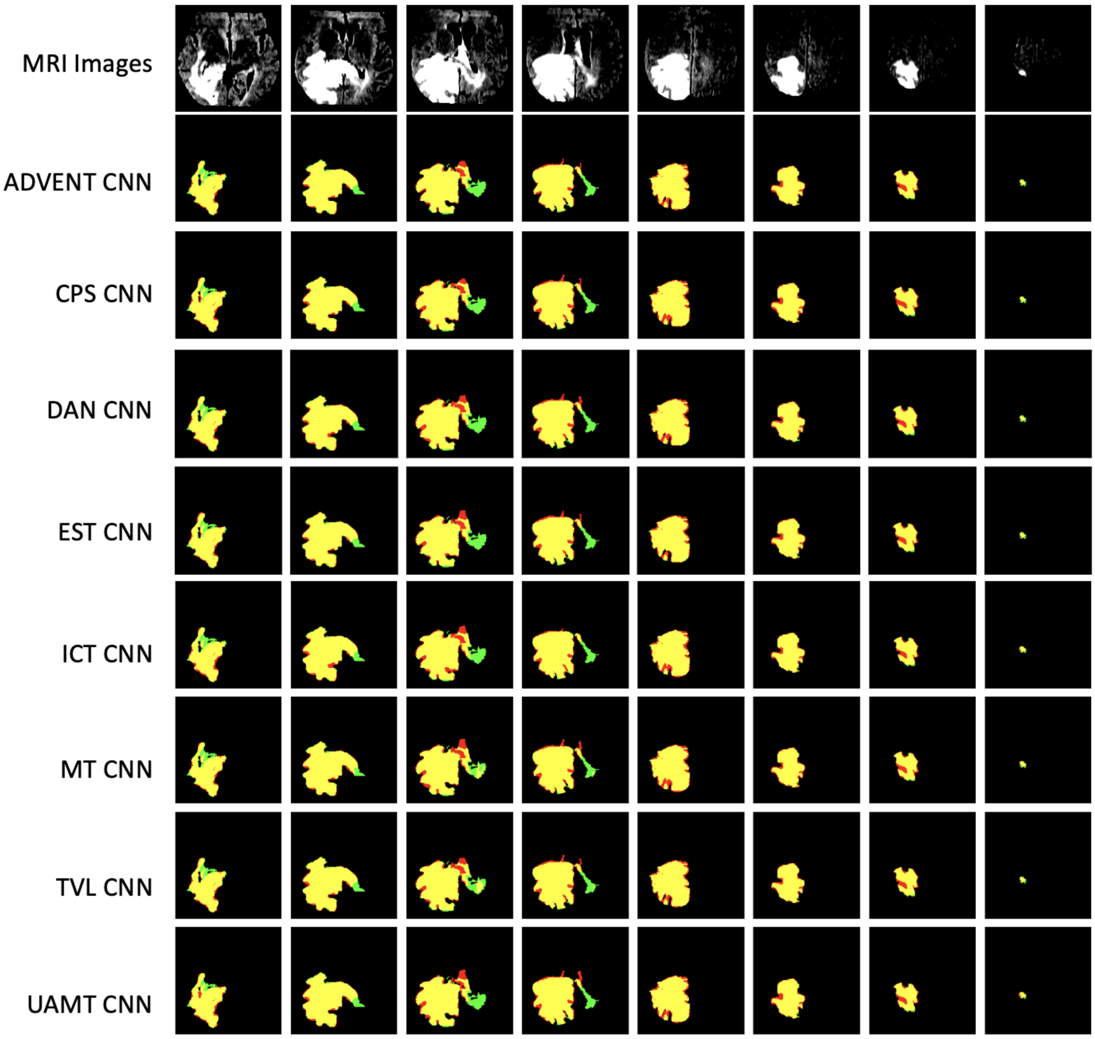


**Figure 5.22:** The Example MRI Brain Tumour Segmentation Inference with 3D CNN Network when 50% of Training Set as Labeled Data.

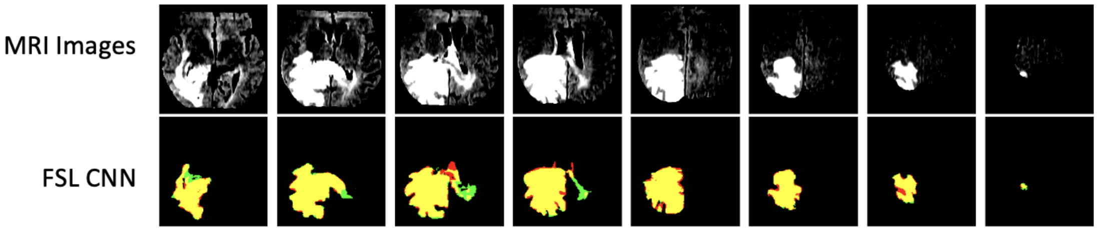


**Figure 5.23:** The Example MRI Cardiac Segmentation Inference with Fully Supervised 3D CNN-based Segmentation Network.

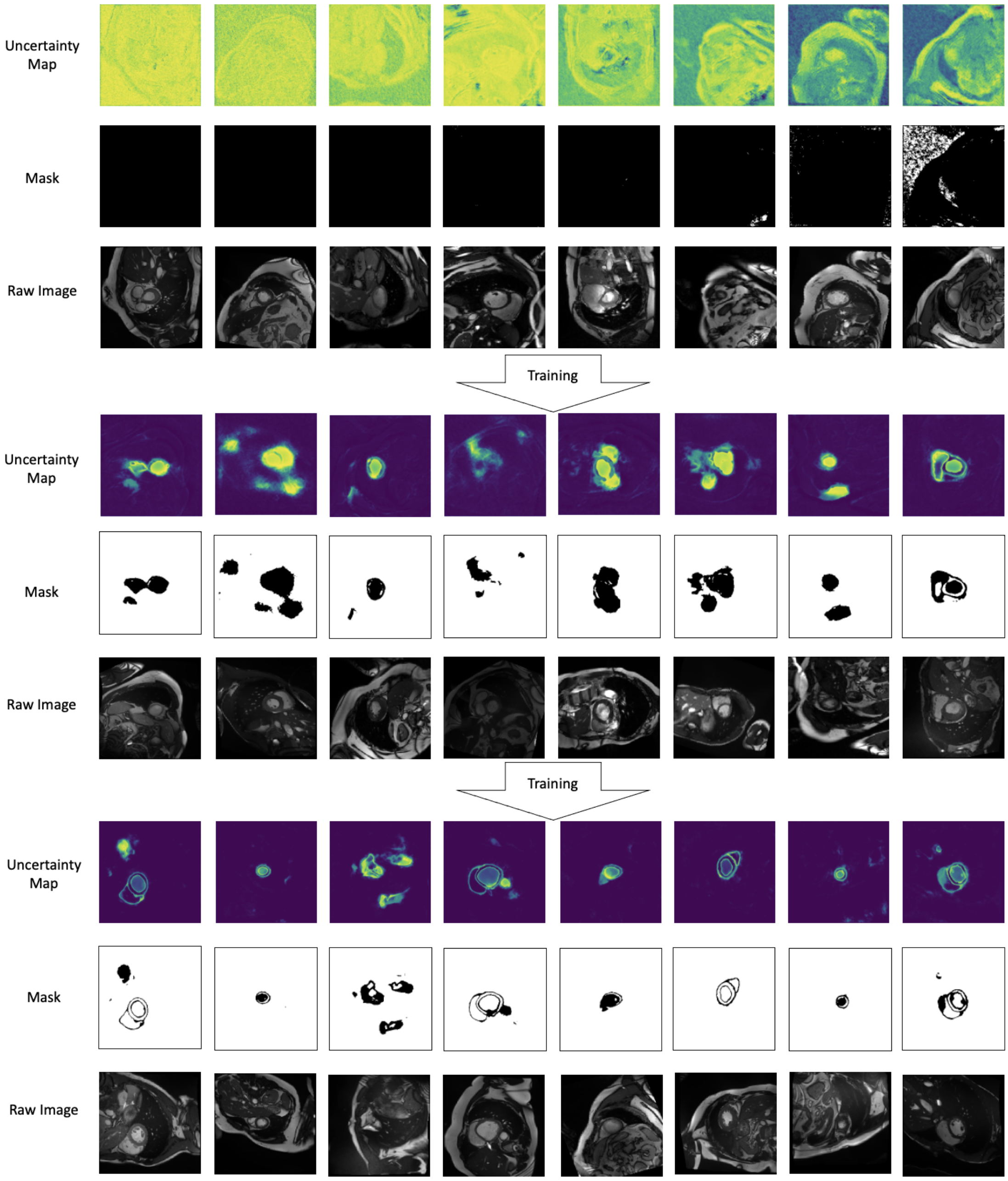


**Figure 5.24:** Sample Uncertainty Maps, Masks, and Raw Images during the Training Process. In uncertainty map, the yellow denotes the uncertainty area, and blue denotes certainty area. In mask, white denotes as the eligible area of pseudo label to supervise, and black denotes the uncertain area of pseudo label to be filtered. During training process from early stage to end stage, we can find the certain area is growing larger and eligible area of pseudo label to supervise is growing as well, and the only uncertain area is the boundary of region of interest.

**Table 5.5:** The Ablation Study on Proposed Contributions of SSL Architecture with Different Segmentation Networks.

| Mean Teacher | Uncertainty Aware | Net | IoU | Sen | Spe |
|---|---|---|---|---|---|
| ✓ | | UNet [4] | 0.7494 | 0.7903 | **0.9977** |
| ✓ | ✓ | UNet [4] | 0.7164 | 0.8037 | 0.9949 |
| ✓ | | ENet [252] | 0.7549 | 0.8314 | 0.9958 |
| ✓ | ✓ | ENet [252] | 0.7460 | 0.8529 | 0.9941 |
| ✓ | | UAMTViT [167] | 0.7840 | 0.8405 | 0.9970 |
| ✓ | ✓ | **UAMTViT** [167] | **0.7891** | **0.8398** | 0.9973 |
| ✗ | ✗ | UNet [4] | 0.7924 | 0.8409 | **0.9975** |
| ✗ | ✗ | ENet [252] | 0.7549 | 0.8696 | 0.9937 |
| ✗ | ✗ | **UAMTViT** [167] | **0.8173** | **0.9137** | 0.9951 |

**Table 5.6:** The Exploration of the Proposed Setting of Threshold to Filtering Uncertainty Region.

| Threshold | IoU | Acc | Pre | Sen | Spe |
|---|---|---|---|---|---|
| Threshold 0.2 | 0.7465 | 0.9889 | 0.8895 | 0.8229 | 0.9958 |
| Threshold 0.5 | 0.7480 | 0.9891 | **0.9048** | 0.8119 | **0.9965** |
| Threshold 0.8 | 0.7042 | 0.9862 | 0.8299 | 0.8231 | 0.9930 |
| Exponential Ramp Up | 0.7543 | **0.9892** | 0.8895 | 0.8324 | 0.9957 |
| Linear Ramp Up | 0.7179 | 0.9866 | 0.8189 | **0.8534** | 0.9922 |
| Cosine Ramp Down | 0.7046 | 0.9861 | 0.8230 | 0.8305 | 0.9926 |
| 0.6 * Exponential Ramp Up | 0.7321 | 0.9879 | 0.8588 | 0.8324 | 0.9943 |
| 0.6 * Linear Ramp Up | 0.7354 | 0.9883 | 0.8852 | 0.8130 | 0.9956 |
| 0.6 * Cosine Ramp Down | 0.8552 | 0.9889 | 0.8931 | 0.8205 | 0.9959 |
| 0.8 * Exponential Ramp Up | 0.7240 | 0.9874 | 0.8528 | 0.8275 | 0.9941 |
| 0.8 * Linear Ramp Up | 0.7326 | 0.9882 | 0.8836 | 0.8109 | 0.9956 |
| 0.8 * Cosine Ramp Down | 0.7674 | 0.9899 | 0.9017 | 0.8374 | 0.9962 |
| 1.2 * Exponential Ramp Up | 0.7326 | 0.9882 | 0.8834 | 0.8109 | 0.9956 |
| 1.2 * Linear Ramp Up | 0.7304 | 0.9876 | 0.8458 | 0.8426 | 0.9936 |
| 1.2 * Cosine Ramp Down | 0.7493 | 0.9889 | 0.8807 | 0.8340 | 0.9953 |
| 1.4 * Exponential Ramp Up | **0.8359** | 0.9874 | 0.8724 | 0.8024 | 0.9951 |
| 1.4 * Linear Ramp Up | 0.8167 | 0.9856 | 0.8305 | 0.8034 | 0.9932 |
| 1.4 * Cosine Ramp Down | 0.7427 | 0.9884 | 0.8638 | 0.8412 | 0.9945 |

Notably, both the background and ROI can simultaneously exhibit high certainty, indicated by white in the masks. Typical examples of these masks show uncertainty primarily at the ROI boundaries. Ideally, with appropriate threshold settings, the network can achieve high certainty across the entire image. Towards the end of the training process, as the uncertainty map becomes predominantly blue, the

**Table 5.7:** The Ablation Study of the Setting of Weight Factor on Consistency Loss.

| Weight | IoU | Acc | Pre | Sen | Spe |
|---|---|---|---|---|---|
| Threshold 0.2 | 0.5243 | 0.9723 | 0.6238 | 0.7667 | 0.9808 |
| Threshold 0.5 | 0.3956 | 0.9567 | 0.4719 | 0.7101 | 0.9670 |
| Threshold 0.8 | 0.4052 | 0.9703 | 0.6667 | 0.5082 | 0.9894 |
| Exponential Ramp Up | 0.7105 | 0.9870 | 0.8613 | 0.8023 | 0.9946 |
| Linear Ramp Up | 0.7149 | 0.9868 | 0.8357 | 0.8319 | 0.9932 |
| Cosine Ramp Down | 0.7547 | 0.9894 | 0.9044 | 0.8201 | 0.9964 |
| 0.6 * Exponential Ramp Up | 0.7723 | 0.9900 | 0.8978 | 0.8467 | 0.9960 |
| 0.6 * Linear Ramp Up | 0.7586 | 0.9896 | **0.9069** | 0.8227 | **0.9965** |
| 0.6 * Cosine Ramp Down | **0.7742** | 0.9900 | 0.8908 | **0.8554** | 0.9956 |
| 0.8 * Exponential Ramp Up | 0.7110 | 0.9864 | 0.8216 | 0.8408 | 0.9924 |
| 0.8 * Linear Ramp Up | 0.7248 | 0.9875 | 0.8559 | 0.8256 | 0.9942 |
| 0.8 * Cosine Ramp Down | 0.7178 | 0.9869 | 0.8376 | 0.8338 | 0.9933 |
| 1.2 * Exponential Ramp Up | 0.7432 | 0.9887 | 0.8854 | 0.8223 | 0.9956 |
| 1.2 * Linear Ramp Up | 0.5596 | 0.9742 | 0.6363 | 0.8227 | 0.9805 |
| 1.2 * Cosine Ramp Down | 0.7509 | 0.9891 | 0.8955 | 0.8230 | 0.9960 |
| 1.4 * Exponential Ramp Up | 0.6968 | 0.9864 | 0.8621 | 0.7482 | 0.9948 |
| 1.4 * Linear Ramp Up | 0.6557 | 0.9832 | 0.7807 | 0.8037 | 0.9906 |
| 1.4 * Cosine Ramp Down | 0.7550 | **0.9893** | 0.8979 | 0.8259 | 0.9961 |

corresponding mask becomes predominantly white, indicating overall certainty in the network's predictions.

### Ablation Study on EST

To analyze the individual effects of each proposed contribution, as well as their combined effects, we conducted extensive ablation experiments, which are detailed in Table 5.8. All supervision schemes are with marks ✓on the mandatory Student network, because it is the only feature learning network from a limited annotation set directly. ✓with Teacher or Examiner indicates the only Teacher-Student consistency training or Examiner-Student adversarial training SSL scheme, which are both able to help the Student network learn from unannotated medical data. Further experiments of only fully supervised learning of the student network with 10% annotated data and 100% annotated data are also conducted as the lower-bound and upper-bound performance, respectively. Table 7.3 presents the promising improvement for Student with the help of Teacher and Examiner network.

**Table 5.8:** The Ablation Study of Examiner-Student-Teacher on Brain Tumor MRI Testing Set.

| Supervision | | | Performance | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Student | Teacher | Examiner | Dice | Acc | Pre | Sen | Spe | HD | ASD |
| ✓ | (10% Full) | | 0.8405 | 0.9898 | 0.8889 | 0.7970 | 0.9965 | 12.2822 | 2.2771 |
| ✓ | ✓ | | 0.8546 | 0.9906 | 0.8995 | 0.8139 | 0.9968 | 12.4331 | **2.1487** |
| ✓ | | ✓ | 0.8555 | 0.9906 | 0.8917 | **0.8222** | 0.9965 | 12.3674 | 2.5333 |
| ✓ | ✓ | ✓ | **0.8605** | **0.9911** | **0.9135** | 0.8134 | **0.9973** | **8.7455** | 2.1574 |
| (100% Full) | | | 0.8804 | 0.9921 | 0.9027 | 0.8591 | 0.9968 | 9.0964 | 1.8919 |

### Extension Study on TVL

Table 5.9 reports the quantitative results on four dataset when using 2%–20% of data are assumed as labeled data. Considering TVL consists of three classifiers, one of the best classifiers on validation set is utilized for testing against on other baseline methods which with only a single classifier. Notably, the networks on CT spine dataset demonstrate significantly higher IoU and Sen scores, attributable to the consistent feature distribution of spinal structures within CT images, which typically with similarly sized entities centrally located, thereby leading to high scores for all networks.

**Table 5.9:** The Direct Comparison of TVL Against Existing Segmentation Networks under Various Data Situation.

| | Experimental Results when **2%** Data is Assumed as Labeled Data | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Ultrasound Nerve | | CT Spine | | MRI Cardiac | | Histology Nuclei | |
| | IoU | Sen | IoU | Sen | IoU | Sen | IoU | Sen |
| UNet[4] | 0.1628 | 0.2020 | 0.7875 | 0.8078 | 0.3888 | 0.8351 | 0.6574 | 0.7814 |
| LinkNet[205] | 0.0919 | 0.1280 | 0.8232 | 0.8164 | 0.1498 | **0.9329** | 0.6905 | 0.8219 |
| FPN[22] | 0.1227 | 0.1320 | 0.8099 | 0.8029 | 0.4802 | 0.5143 | 0.6942 | 0.8284 |
| **TVL** | **0.2800** | **0.3678** | **0.8451** | **0.8433** | **0.9923** | 0.8411 | **0.6946** | **0.8293** |
| | Experimental Results when **5%** Data is Assumed as Labeled Data | | | | | | | |
| UNet[4] | 0.2762 | 0.3234 | 0.8301 | 0.8385 | 0.6805 | 0.8053 | 0.7075 | 0.5953 |
| LinkNet[205] | 0.2505 | 0.2885 | 0.8264 | 0.8373 | 0.6762 | 0.7988 | 0.7552 | 0.8645 |
| FPN[22] | 0.2703 | 0.3093 | 0.8010 | 0.8360 | 0.7721 | 0.8351 | 0.7031 | 0.8333 |
| **TVL** | **0.3765** | **0.5090** | **0.8354** | **0.8633** | **0.8094** | **0.8880** | **0.7691** | **0.8757** |
| | Experimental Results when **10%** Data is Assumed as Labeled Data | | | | | | | |
| UNet[4] | 0.3554 | 0.4090 | 0.8398 | 0.8287 | 0.8492 | **0.9253** | 0.8012 | 0.8880 |
| LinkNet[205] | 0.3464 | 0.3991 | 0.8505 | 0.9160 | 0.7832 | 0.8641 | 0.7957 | 0.8862 |
| FPN[22] | 0.2416 | 0.4866 | 0.8556 | 0.7937 | 0.8078 | 0.8591 | 0.8034 | 0.8973 |
| **TVL** | **0.4260** | **0.5789** | **0.8728** | **0.9365** | **0.8545** | 0.9159 | **0.8114** | **0.8981** |
| | Experimental Results when **20%** Data is Assumed as Labeled Data | | | | | | | |
| UNet[4] | 0.4352 | 0.5254 | 0.9160 | 0.8374 | 0.8984 | 0.9448 | 0.8104 | 0.9019 |
| LinkNet[205] | 0.4333 | 0.5237 | 0.9365 | 0.9248 | 0.8712 | 0.9258 | 0.8027 | 0.8971 |
| FPN[22] | 0.3956 | 0.5953 | 0.9411 | 0.8573 | 0.8857 | 0.9307 | 0.8176 | 0.9094 |
| **TVL** | **0.4981** | **0.6528** | **0.9628** | **0.9272** | **0.9020** | **0.9459** | **0.8530** | **0.9244** |

**Table 5.10:** The Ablation Study on Contributions of Architecture and Modules.

| Label Process | Dual Loss Design | Classifier A | Classifier B | Classifier C | IoU |
|---|---|---|---|---|---|
| | | ✓× 2 | ✓ | | 0.8724 |
| | | | ✓× 2 | ✓ | 0.8739 |
| | | ✓ | | ✓× 2 | 0.8641 |
| ✓ | | ✓× 3 | | | 0.8666 |
| | ✓ | ✓× 3 | | | 0.8579 |
| | | | ✓× 3 | | 0.8598 |
| ✓ | | | ✓× 3 | | 0.8605 |
| | | | | ✓× 3 | 0.8619 |
| ✓ | | | | ✓× 3 | 0.8739 |
| ✓ | | ✓ | ✓ | ✓ | 0.8787 |
| | ✓ | ✓ | ✓ | ✓ | 0.8841 |
| ✓ | ✓ | ✓ | ✓ | ✓ | **0.9020** |

### Ablation Study on TVL

To determine the individual contributions of the components in proposed TVL framework, ablation study is conducted and reported in Table 5.10. These experiments are designed to evaluate the significance of having distinct networks (Networks A, B, and C) in our setup. Instead of using three different networks, we experimented with configurations employing multiple instances of the same network (e.g., $A{\times}2$, $A{\times}3$, etc.). As shown in Table 5.10, our findings reveal that the IoU is adversely affected in these scenarios. The best performance is observed when Networks A, B, and C are each uniquely represented in the framework, underscoring their individual importance to the overall efficacy of TVL.

## 5.5 Contribution and Discussion

In this chapter, we have explored the utility and efficacy of SSL strategies within the context of medical image segmentation. The key principle of the SSL framework is to utilize perturbation-based consistency as a regularization to leverage unlabelled data. Specifically, we propose various strategies for data and network perturbation, utilizing both CNN- and ViT-based segmentation network. We reproduce numerous classical semi-supervised framework including MT, CPS, FixMatch, adversarial training, ICT, and various advanced approaches to improve performance such as dynamic ensembling pseudo labels, uncertainty estimation, multi-view learning. The evaluation results demonstrate the strengths and weaknesses of each architecture when incorporated within a SSL paradigm. We also conducted extensive experiments to validate these methods on public datasets, and all methods are with the same hyper-parameter settings and ratio of labeled of total training set to ensure the fairness of our comparisons.

The proposed SSL strategies demonstrate nearly similar performance with that of fully supervised learning baseline methods, yet reduce annotation costs. The situation of limited labeled data, supplemented by large amount of raw data is common in clinical settings, and is particularly essential in reducing clinicians' workload. Moreover, the principles of SSL is foundational essential for exploring weakly-supervised learning domains, where scenarios such as sparse labeling present similar to the SSL of combining both labeled and unlabeled information in medical data.

# 6

# Noise-Robust Learning: Comprehensive Data Annotation with Noisy Dense Masks

## Contents

**6.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . 119**
**6.2 Noise-Robust Learning Framework Setup . . . . . . . . 121**
**6.3 Noisy Label Generation . . . . . . . . . . . . . . . . . . . . 121**
**6.4 Study of Noise-Robust Segmentation . . . . . . . . . . . 122**
**6.5 Experiments and Results . . . . . . . . . . . . . . . . . . . 126**
  6.5.1 Implementation Details . . . . . . . . . . . . . . . . . . . 126
  6.5.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . . 126
  6.5.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . . 127
**6.6 Contribution and Discussion . . . . . . . . . . . . . . . . 130**

## 6.1 Motivation

The exceptional performance of both ViT- and CNN-based networks significantly relies on access to sufficient amounts of high-quality annotated data. However, there is a barrier to implementing advanced deep learning networks within clinical settings when data is not 100% accurate and precise. Although SSL [167, 170, 253] (introduced in Chapter 5) and WSL [29, 213, 254] (introduced in Chapter 7) have been utilized to address the costly labeling process in the context of ViT and

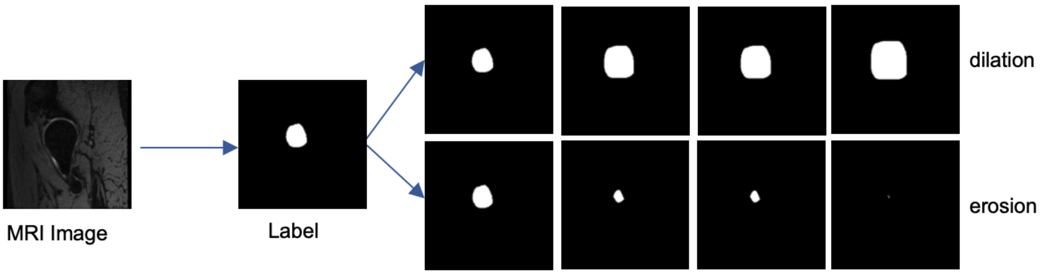


**Figure 6.1:** The Example MRI Bone Images with Corresponding Ground Truth Segmentation. We generate noisy labels by erosion and dilation, and the level of noise can be higher or lower.

CNN, their application to medical image segmentation remains under explored, particularly regarding noisy labels.

The 'noisy labels' pertains to inconsistencies or inaccuracies in annotation as ground truth for machine leanring purpose, presenting significant challenges in network training and generalization [11, 17]. In the realm of medical image segmentation, the occurrence of noisy labels is often inevitable, primarily due to factors including: inter-observer variability, the inexperience of junior clinicians, or the inherent complexity of anatomical structures [255–258]. As shown in Figure 6.1, where the annotation process may deviate from the ideal standard, leading to labeled features that display erosion or dilation of contours, along with various elastic deformations. We categorize these deviations as noisy labels, recognizing their potential to impact network performance.

The integration of denoising techniques into segmentation networks enables the development of noise-robust networks. These networks are adept at segmenting medical images even when faced with imprecise annotations. This capability improves segmentation performance which is essential in enhancing the clinical applicability of these deep learning-based networks. Therefore, this contributes to advancements of bringing deep learning techniques of medical image analysis to practical clinical diagnosis.

## 6.2 Noise-Robust Learning Framework Setup

In the task of noise-robust learning, $\mathbf{D}_{train}$ and $\mathbf{D}_{test}$ typically represent the training set (assumed to be labeled with noise) and the testing set (assumed to be noise-free), respectively. We denote a batch of labeled data as $(\boldsymbol{X}, \boldsymbol{Y}_{\text{gt}} + noise) \in \mathbf{L}$ and $(\boldsymbol{X}, \boldsymbol{Y}_{\text{gt}}) \in \mathbf{D}_{test}$, where $\boldsymbol{X} \in \mathbb{R}^{h \times w}$ represents a 2D image with dimensions $h \times w$, and $\boldsymbol{Y}_{\text{gt}} \in [0, 1]^{h \times w \times c}$ denotes the annotation for each pixel (with 0 representing the background and 1 indicating the ROI).

We manually introduce noise, denoted as *noise*, to a pre-defined proportion $\beta \in (0\%, 100\%)$ of $\boldsymbol{Y}_{\text{gt}}$ in the training set to simulate a realistic clinical data scenario during the training process. $\boldsymbol{Y}_{\text{p}}$ is the dense map predicted by the segmentation network, $f(\theta) : \boldsymbol{X} \mapsto \boldsymbol{Y}_{\text{p}}$. $\mathcal{L}_{\text{s}} : (\boldsymbol{Y}_{\text{p}}, \boldsymbol{Y}_{\text{gt}} + noise) \mapsto \mathbb{R}$ represents the supervised segmentation loss.

The general training goal is to update the parameter $\theta$ of the segmentation network $f(\theta)$ to minimize the loss $\mathcal{L}$ on the training set $\mathbf{D}_{train}$ while mitigating the detrimental influence of noisy labels. The final evaluation calculates the difference of $(\boldsymbol{Y}_{\text{p}}, \boldsymbol{Y}_{\text{gt}}) \mapsto \mathbb{R}$ on the testing set.

## 6.3 Noisy Label Generation

To assess proposed strategy resilience against noisy labels, we employ artificial noise into the dataset that is initially considered to be accurately annotated. This process allows to simulate real-world scenarios of annotation imperfections and validation. The noisy label simulation process comprises the following steps:

(*i*) From a dataset with perfect annotations, we randomly select a subset for alteration. The ratio of noisy labels to the entire dataset is denoted as $\beta \in [0, 1]$.

(*ii*) We employ erosion, dilation, and elastic transformation to generate noisy labels for the randomly selected ground truth. Erosion effectively simulates under-segmentation where critical features may be missed due to conservative annotation, common when clinicians are unsure about the boundary limits of ROI. Conversely, dilation represents over-segmentation, reflecting scenarios where clinicians might annotate beyond the actual boundaries of an ROI, a frequent occurrence due to the overlapping nature of medical images. These simulation approaches, alongside elastic transformations that introduce realistic deformations reflecting human error in manual labeling are introduced for evaluating segmentation networks under clinical data situations.

Erosion and dilation are applied as follows:

$$(A \ominus B)(x, y) = \min_{(i,j)\in B} A(x - i, y - j) \tag{6.1}$$

$$(A \oplus B)(x, y) = \max_{(i,j)\in B} A(x + i, y + j) \tag{6.2}$$

where $A$ is the binary annotation mask, $B$ the structuring element, and $(x, y)$ the pixel coordinates.

Elastic transformation is a non-linear deformation technique simulating local shape warping. For an image $I(x, y)$ and displacement fields $\Delta x(x, y)$ and $\Delta y(x, y)$, generated by the Gaussian smoothing of random fields, the transformation is defined as:

$$I_{\text{transformed}}(x, y) = I(x + \gamma\Delta x(x, y), y + \gamma\Delta y(x, y)) \tag{6.3}$$

where $\gamma$ represents the deformation scale factor.

(*iii*) The manipulated annotations replace their original annotations in the dataset, generating a new dataset containing a mixture of perfect and noisy labels. By creating noisy labels in this manner (examples shown in Figure 6.1), we design a challenging dataset for experimental purposes, facilitating the assessment of our proposed method's noise-robustness and the comparison of its performance with existing advanced techniques.

## 6.4 Study of Noise-Robust Segmentation

This is the part of our past work from RARUNet [17] and NRUNet [31]. To mimic the challenges of noisy (or imprecise) labels emerging in practical settings, we introduce an Adaptive Denoising Learning (ADL) strategy during the training process. To mirror this scenario, a certain proportion $\beta$ of masks in the training data has been replaced with synthetically generated noisy labels featuring erosion, dilation, or elastic transformation. In our study, we explored three situations with $\beta \in [75\%, 50\%, 25\%]$, and randomly select and replace the original label to noisy label. The noisy level $\alpha$ thus can be calculated by the difference between the original label and generated noisy label via Dice-Coefficient.

Different with developing a noise-robust network from noise distribution, such as noise-robust loss design [256, 259, 260], we propose to actively detect and remove noisy label. Motivated by O2UNet [30], the proposed noise-robust learning strategy (seen in Figure 6.2) involves a meticulous analysis of prediction-label discrepancies at every epoch of the training process. We compute and record the loss for each prediction compared to its corresponding label, with a key premise: labels that consistently yield higher losses are more likely to be noisy or incorrect. This premise is rooted in the understanding of the learning dynamics of neural networks. During the initial phase of training (underfitting), a network is still learning to capture the fundamental patterns in the data, and high losses are more prevalent. As training progresses, the network starts fitting more closely to the training data, entering an overfitting phase where it might start capturing noise as patterns. Hence, consistently high losses even in later stages of training can indicate problematic labels. The number $N(t)$ of labels identified and removed in each epoch is determined by the current training epoch $t$, the noisy level $\alpha$, the proportion $\beta$ of noise-infused items in the training dataset, the total number of training epochs $x$, and the total number of masks $y$.

Our strategy, therefore, is to identify and eliminate these high-loss labels throughout the training. At the start of the training, where underfitting is more prevalent, a larger number of noisy labels are identified and removed. As the network's fit improves, the focus shifts to fine-tuning by removing fewer labels, reflecting the reduced incidence of high-loss labels as the network starts overfitting [17].

The process we proposed in [31] is quantified as follows:

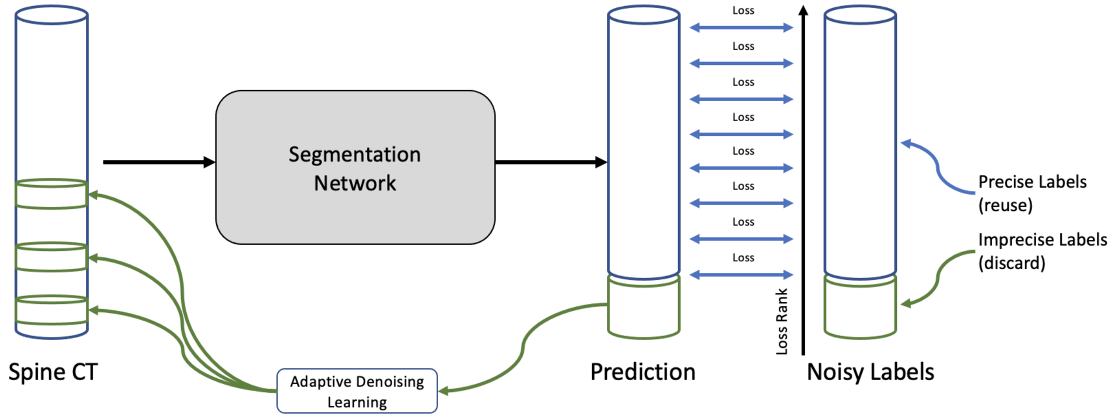


**Figure 6.2:** The Training Process of Adaptive Denoising Learning Strategy. Predictions are generated through segmentation network with a conventional training manner. Labels with higher losses against of network prediction are more likely to be considered as noisy labels. ADL iteratively removes a specific number of labels deemed as noisy labels in each training epoch. Precise (in blue) and imprecise (in green) labels get classified and, if imprecise, discarded.

$$N(t) = \frac{\beta y}{x^2}(x - t) \tag{6.4}$$

Here, $N(t)$ denotes the number of labels identified and removed in epoch $t$, with $\beta$ representing the proportion of noise-infused items in the training set, $x$ the total number of training epochs, and $y$ the total number of masks. This equation reflects a deliberate reduction in the number of labels removed as training progresses, aligning with the network's transition from underfitting towards overfitting. Additionally, once we consider the ratio of noisy level as $\alpha$ which is calculated by Dice-Coefficient, the process we proposed in [17] is quantified as follows:

$$N(t) = \begin{cases} 0.5(1-\alpha)\beta y, & 0 < t < 0.1(1-\alpha)\beta x \\ \frac{y}{x}t, & 0.1(1-\alpha)\beta x \leq t \leq 0.5(1-\alpha)\beta x \\ 0.1(1-\alpha)\beta y, & 0.5(1-\alpha)\beta x < t \leq x \end{cases} \tag{6.5}$$

where the hyper-parameters 0.1 and 0.5 of Equation 6.5 are obtained through systematic search. By integrating ADL into the training regime, we aim to bolster the segmentation network's robustness against noisy labels. This strategy is not only about discarding potentially misleading information but also about refining the network's focus on reliable data. The anticipated outcome is a network that not only performs with higher accuracy but also exhibits enhanced reliability and applicability in clinical settings, where the quality of data can be highly variable.

## 6.5 Experiments and Results

### 6.5.1 Implementation Details

The dataset employed in this chapter is the CT spine dataset [219]. Accurate ground truth masks are available for each image, and a subset of these masks is modified to introduce noise, simulating real-world annotation challenges. The segmentation network utilized includes RARUNet [17] and NRUNet [31], with input feature maps sized at $256 \times 256 \times 1$. In each encoder and decoder layer of these U-shape networks, segmentation networks featured 64, 128, 256, and 512 CNN, respectively, and incorporated two successive CNN layers. With regards to the ViT-related design elements, we assigned an image patch-embedding dimension of 768, an MLP count of 1024, and 12 heads for the MSA. The NR-UNet's bottleneck is composed of 6 ViT layers, with the final layer being a $1 \times 1$ CNN layer, suited for 2D image binary semantic segmentation. The training, spanning 50 epochs, varied between 500 and 800 minutes, including data transfer, with a batch size of 4 and Adam optimizer at a learning rate of $10^{-5}$. The segmentation loss is based on the Dice coefficient.

### 6.5.2 Qualitative Results

The experiments focus on assessing the robustness of the proposed algorithms to annotation noise, i.e. the impact of the proposed ADL strategy. Table 6.1 particularly evaluates the effect of the ADL strategy. The term 'Proportion' indicates the percentage $\beta$ of noisy labels introduced into the training set, with other proportions also tested but omitted here to prevent overloading the table. The Network as 'net' clarifies that different networks are trained independently. The inclusion of a module implementing the ADL strategy (denoted by ✓) consistently improve performance.

Figure 6.3 illustrates example predictions by NRUNet under various data conditions (i.e., different ratios $\beta$ of noisy labels). By comparing the results of different algorithms and adjusting the proportion of noisy labels injected into the training dataset, we glean valuable insights into the robustness and adaptability of

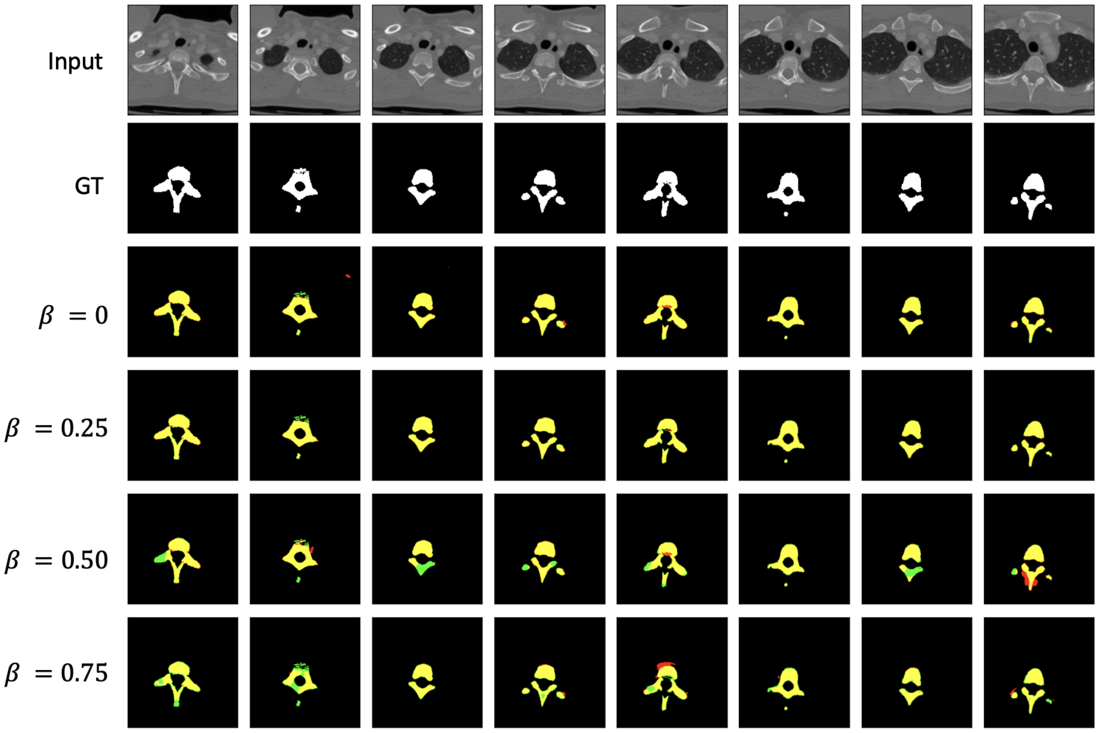


**Figure 6.3:** The Example Input CT Spine Images, and Prediction of NRUNet Against Ground Truth Under Various Data Situations.

our method in tackling the challenges that noisy annotations present to medical image segmentation tasks.

### 6.5.3 Quantitative Results

While the performance of various networks in conventional supervised learning is similar (as shown in Table 4.1), the ADL strategy's effectiveness becomes apparent in Table 6.1 and 6.2. This strategy significantly mitigates the impact of noisy labels, achieving segmentation performance that can be considered practically acceptable in many clinical applications.

**Table 6.1:** The Ablation Study of Adaptive Denoising Learning under Different Proportion of Noisy Labels on Training Set with Various Segmentation Networks.

| Proportion ($\beta$) | Net | ADL | Dice | IoU |
|---|---|---|---|---|
| 75% | UNet | ✗ | 0.8004 | 0.6672 |
| | | ✓ | 0.8337 | 0.7148 |
| 75% | Residual UNet | ✗ | 0.7962 | 0.6614 |
| | | ✓ | 0.8210 | 0.6964 |
| 75% | NR-UNet | ✗ | 0.8196 | 0.6943 |
| | | ✓ | **0.8466** | **0.7340** |
| 50% | UNet | ✗ | 0.8188 | 0.6932 |
| | | ✓ | 0.8564 | 0.7489 |
| 50% | Residual UNet | ✗ | 0.8179 | 0.6919 |
| | | ✓ | 0.8453 | 0.7321 |
| 50% | NR-UNet | ✗ | 0.8362 | 0.7185 |
| | | ✓ | **0.8832** | **0.7908** |
| 25% | Dense UNet | ✗ | 0.9096 | 0.8342 |
| | | ✓ | 0.9284 | 0.8664 |
| 25% | UNet | ✗ | 0.9084 | 0.8322 |
| | | ✓ | 0.9303 | 0.8697 |
| 25% | Residual UNet | ✗ | 0.9002 | 0.8185 |
| | | ✓ | 0.9213 | 0.8541 |
| 25% | NR-UNet | ✗ | 0.9101 | 0.8350 |
| | | ✓ | **0.9532** | **0.9106** |

**Table 6.2:** The Ablation Study of Adaptive Denoising Learning under Different Proportion of Noisy Labels and Different Noisy Level when Generating Noisy Labels on Training Set with Various Segmentation Networks.

| Proportion ($\beta$) | Level ($\alpha$) | Net | ADL | IoU | Recall |
|---|---|---|---|---|---|
| 75% | 0.68 | U-Net | ✗ | 0.6445 | 0.7303 |
| 75% | 0.68 | U-Net | ✓ | **0.6742** | **0.8072** |
| 75% | 0.68 | Residual UNet | ✗ | 0.7732 | 0.9097 |
| 75% | 0.68 | Residual UNet | ✓ | **0.8138** | **0.9462** |
| 75% | 0.68 | Attention UNet | ✗ | 0.7809 | 0.8823 |
| 75% | 0.68 | Attention UNet | ✓ | **0.8087** | **0.9142** |
| 50% | 0.77 | UNet | ✗ | 0.7523 | 0.8420 |
| 50% | 0.77 | UNet | ✓ | **0.8522** | **0.9295** |
| 50% | 0.77 | Attention UNet | ✗ | 0.8464 | 0.9201 |
| 50% | 0.77 | Attention UNet | ✓ | **0.8561** | **0.9283** |
| 25% | 0.85 | Residual UNet | ✗ | 0.8615 | 0.9051 |
| 25% | 0.85 | Residual UNet | ✓ | **0.8868** | **0.9433** |
| 25% | 0.85 | Dense UNet | ✗ | 0.8443 | 0.9378 |
| 25% | 0.85 | Dense UNet | ✓ | **0.8864** | **0.9424** |
| 25% | 0.55 | U-Net | ✗ | 0.8024 | 0.8698 |
| 25% | 0.55 | U-Net | ✓ | **0.8304** | **0.9176** |
| 25% | 0.55 | Residual UNet | ✗ | 0.8230 | 0.8956 |
| 25% | 0.55 | Residual UNet | ✓ | **0.8495** | **0.9126** |

## 6.6 Contribution and Discussion

This chapter's primary contribution is the development of the ADL strategy to tackle the prevalent problem of noisy labels in medical image segmentation. We demonstrate the effectiveness of ADL into a hybrid CNN-ViT encoder-decoder U-shape segmentation network, and a modified CNN-based U-shape segmentation network, demonstrating that the ADL strategy effectively improve the segmentation network's robustness against noise in annotation compared against with traditional training strategy.

By generating noisy labels through image pre-processing including erosion, dilation, and elastic transformation, we are able to evaluate our proposed method in a controlled, reproducible scenario, and accurately assess its noise robustness. The denoising strategy is valuable in handling the real-world challenges of imperfect annotations that are prevalent in the medical imaging field, especially in the clinical context.

Different with developing a noise-robust network under noise distribution such as a network with an additional noise-robust loss or network blocks [256, 259, 260], we propose ADL strategy to actively detect and remove noisy label to reduce the impact of unreliable data during training. Compare with some similar strategies such as sample selection works [261–263], ADL is much simple to be deployed and efficient with any types of segmentation network. More specifically, ADL is a novel strategy that based on a fresh perspective on understanding how training dynamics change when encountering noisy labels, transitioning from an underfitting to overfitting state. Our proposed strategy adapts to this change by initially detecting a large number of noisy labels and gradually reducing the number as training progresses.

The current framework, however, requires manual setting of the proportion of noisy label in the dataset $\beta$ and optional noisy level of noisy label $\alpha$. In a real-world setting, these values are not easily quantifiable or known a priors. In addition, once the noisy label been removed with ADL, the feature information of corresponding images is still valuable for network training with SSL strategy [264–267]. Multi-rater learning is also a potential strategy to deal with unreliable data [268–270].

In conclusion, this chapter presents a novel strategy to improve the robustness of medical image segmentation networks against noisy labels, bringing deep learning closer to practical healthcare applications.

# 7

# Weakly-Supervised Learning: Comprehensive Data Annotation with Sketchy Contours

### Contents

**7.1 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . 133**
**7.2 Weakly-Supervised Learning Framework Setup . . . . 135**
**7.3 Study of Weakly-Supervised Segmentation . . . . . . . 136**
7.3.1 Scribble Supervision Loss . . . . . . . . . . . . . . . . . . 136
7.3.2 Collaborative Hybrid Networks . . . . . . . . . . . . . . . 136
**7.4 Experiments and Results . . . . . . . . . . . . . . . . . . . 140**
7.4.1 Implementation Details . . . . . . . . . . . . . . . . . . . 140
7.4.2 Qualitative Results . . . . . . . . . . . . . . . . . . . . . . 140
7.4.3 Quantitative Results . . . . . . . . . . . . . . . . . . . . . 141
7.4.4 Ablation Study . . . . . . . . . . . . . . . . . . . . . . . . 141
**7.5 Conclusion and Discussion . . . . . . . . . . . . . . . . . . 144**

## 7.1 Motivation

Whilst recent network architecture engineering of CNN and ViT have demonstrated exceptional performance in medical image segmentation [4, 39, 186], many of these studies are validated on large benchmark datasets with detailed pixel-level annotations [79, 186, 271]. To address the often prohibitive cost of such detailed

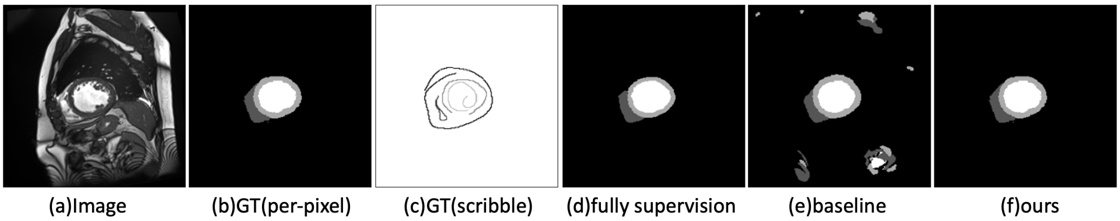


**Figure 7.1:** The Illustration of a Multi-Class Scribble-Supervised Segmentation. (a) Input sagittal left-facing MRI, (b) dense ground-truth annotations, (c) sparse annotations via scribble, (d) segmentation inference by dense-label-supervised UNet, (e) segmentation inference by scribble-label-supervised UNet, (f) segmentation inference by proposed scribble-label-supervised CHNets.

annotations in image semantic segmentation, SSL (discussed in Chapter 5) has been explored as a method to train with limited densely labeled data supplemented by a larger volume of raw data. Additionally, Weakly-Supervised Learning (WSL) such as scribble-based labeling, presents an alternative way for medical data annotation by clinicians. This chapter introduces employing scribble annotations to train CNN- and ViT-based segmentation networks in a simultaneous and collaborative manner.

WSL for segmentation typically leverages sparse annotations like bounding boxes, points, text, and scribbles to train networks [272–274]. Scribble annotation, as illustrated in Figure 7.1, stands out as a practical and convenient form of clinician labeling. However, the limited supervision signal from such sparse annotations poses significant challenges for image semantic segmentation in medical imaging, especialy for accurately classifying pixels at the boundaries of ROI. Current SSL and WSL methods often employ partial-supervision losses for network initialization, utilizing prior assumptions to expand data. This strategy allows network inferences to extend scribbles into dense pseudo labels. Notably, ScribbleSup proposes a graph-based method to spread feature information from scribbles to unlabeled pixels with a unique training loss [273]. Conditional Random Fields have been utilized for segmentation refinement [33], while Scribble2Label introduces an efficient pseudo-labeling mechanism, enhancing label reliability during training [275]. Other approaches, like CycleMix [274], integrate mixing augmentations with consistency regularization for scribble-supervised segmentation, and adversarial training networks have been developed to promote high-quality pseudo-labels [276].

The key study in recent SSL and WSL research is the pseudo label consistency under various data- and network-perturbation, as consistency-aware training. Triple-view learning, for instance, uses three different networks to iteratively generate pseudo labels, enhancing multi-view learning [170]. Additionally, techniques like cross teaching [14] and mix pseudo supervision [240] have been explored for their effectiveness in scribble-supervised segmentation, with network and data perturbation technique for pseudo label generation achieving SOTA performance in MRI cardiac segmentation.

## 7.2 Weakly-Supervised Learning Framework Setup

In the WSL task, $\mathbf{D}_{train}$, $\mathbf{D}_{test}$ normally denote labeled training dataset with low quality, and fully labeled testing set. A batch of labeled low quality data is denoted as $(X, Y_{weak}) \in \mathbf{D}_{train}, (X, Y_{gt}) \in \mathbf{D}_{test}$, where $X \in \mathbb{R}^{h \times w}$ representing a 2D image with the size $h \times w$, $Y_{weak} \in [0, 1, None]^{h \times w \times c}$, $Y_{gt} \in [0, 1]^{h \times w \times c}$ representing the annotation on each pixel whether 0 is background and 1 is ROI. ($None$ indicates no annotation information on some of corresponding pixels.) $Y_p$ is the prediction densely by several segmentation networks such as $f(\theta_1) : X \mapsto Y_1$, and $f(\theta_2) : X \mapsto Y_2$. $\mathcal{L}_{weak} : (Y, Y_{\text{weak}}) \mapsto \mathbb{R}, \mathcal{L}_{semi} : (Y_1, Y_2) \mapsto \mathbb{R}$ represent WSL segmentation loss, and consistency loss on the training set. In general, the training process is to update the network parameter $\theta$ of some segmentation networks $f(\theta_1), f(\theta_2)$ aiming to minimize the sum of losses $\mathcal{L}$. The final evaluation is to calculate the difference of $(Y_p, Y_{gt}) \mapsto \mathbb{R}$ on testing set.

## 7.3 Study of Weakly-Supervised Segmentation

### 7.3.1 Scribble Supervision Loss

To tackle the challenges posed by scribble availability in WSL, the CrossEntropy *CE* function is applied exclusively on annotated pixels, while ignoring the unlabeled pixels as partial supervised segmentation loss (seen in Equation 7.1). By adopting this Partial Cross-Entropy *pCE*, the network training is enforced only on scribble signal [273].

$$\mathcal{L}_{\mathrm{pCE}}(y_{pred}, y_{scrib}) = -\sum_{i \in \omega_L} \sum_{k} y_{scrib}[i,k] log(y_{pred}[i,k]) \tag{7.1}$$

Here, $i$ represents the $i$-th pixel, and $\omega_L$ denotes the set of pixels labeled with scribble. $k$ refers to the $k$-th class, with $[i,k]$ indicating the probability that the $i$-th pixel belongs to the $k$-th class.

### 7.3.2 Collaborative Hybrid Networks

This is part of our past work in [277], where we introduce Collaborative Hybrid Networks (CHNets). CHNets comprises a CNN-based UNet [4], and a Swin Transformer-based UNet-style network [193]. Our methodology enables simultaneous and collaborative learning between these networks, employing an iterative labeling ensemble scheme for dense pseudo-label generation and retraining through external-consistency supervision. Additionally, each network benefits from self-ensemble techniques under internal-consistency supervision, further enhancing their performance. This dual consistency supervision allows CHNets to fully leverage both segmentation networks, yielding detailed pixel-level inferences. We evaluate CHNets on a pre-processed scribble-supervised set based on MRI cardiac dataset [8], and our experimental results demonstrate that the proposed WSL strategy surpasses other existing WSL methods [4, 29, 193, 213, 273, 278–280] in various evaluation metrics.

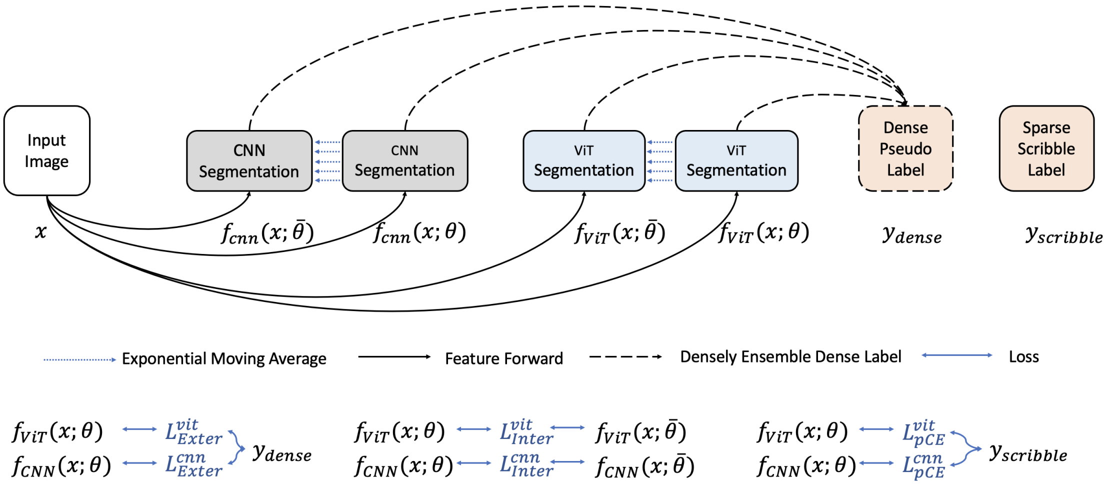


**Figure 7.2:** The Framework of Scribble-Supervised Collaborative Hybrid Networks for Medical Image Segmentation.

## Training Objective

The training of CHNets, as illustrated in Figure 7.2, is conducted end-to-end, focusing on optimizing a group of networks through the sum of different categories of segmentation loss functions, which is formulated in Equation 7.2.

$$\mathcal{L} = \underbrace{\mathcal{L}^{cnn}_{\text{pCE}} + \mathcal{L}^{vit}_{\text{pCE}}}_{Scribble-Supervision} + \underbrace{\lambda_1 \mathcal{L}^{cnn}_{\text{inter}} + \lambda_2 \mathcal{L}^{vit}_{\text{inter}}}_{Internal-Consistency} + \underbrace{\lambda_3 \mathcal{L}^{cnn}_{\text{exter}} + \lambda_4 \mathcal{L}^{vit}_{\text{exter}}}_{External-Consistency} \tag{7.2}$$

where loss includes scribble-supervision loss ($\mathcal{L}_{\text{pCE}}$), internal-consistency loss ($\mathcal{L}_{\text{inter}}$), and external-consistency loss ($\mathcal{L}_{\text{exter}}$). These losses are adapted for two pairs of segmentation networks, specifically as $\mathcal{L}^{cnn}$ for CNN-based networks and $\mathcal{L}^{vit}$ for ViT-based networks.

## Hybrid Networks

Motivated by the success of the UNet, ViT, and network perturbation, we propose group of hybrid networks as multi-view for a single feature segmentation task. This design aligns from the strategy in training multi-networks under limited signal supervision such as multi-view learning feeding with different augmented data [170], Co-Teaching handling with uncertainty labels [264], and cross supervision with varied parameter initialization [239], all of which enforce consistency in inference while diverse perturbations on networks. Our hybrid network not only introduces perturbations at the parameter level but also distinctively at the architectural level. Considering CNN and ViT are two different architecture, we utilize the CNN-based U-shape segmentation network, i.e. UNet and adopt two Swin-ViT layers as network blocks into the U-shape segmentation network, i.e. SwinUNet for a fair comparison.

### Internal Self-Ensembling Consistency Supervision

To boost the feature learning performance of each network, we employ the internal self-ensembling consistency supervision following Mean Teacher [237] from limited-signal SSL task to the similar WSL scribble-supervision task. The internal consistency supervision involves an additional network, named teacher, with the same architecture but updated by the other network, named student, (Equation 7.3). The student network directly learns from scribbles.

$$\theta_i^t = \alpha\theta_i^t + (1-\alpha)\theta_i^s \tag{7.3}$$

where $\theta^s, \overline{\theta}^t$ indicates as the parameters of the student network and teacher network, and the network can be either CNN-based $\theta_{cnn}$ or ViT-based network $\theta_{vit}$. The $\alpha \in [0,1]$ balances the weight of updating parameters. To achieve internal-consistency aware, the Gaussian perturbation is applied during training, and the inference by the Student is enforced to be similar to the Teacher from the same input with noise using internal-consistency loss $L_{inter}$ illustrated in Equation 7.4:

$$\mathcal{L}_{\text{inter}}(y_s, y_t) = CE(y_s, y_t) + Dice(y_s, y_t) \tag{7.4}$$

where $CE, Dice$ indicates as Cross-Entropy and Dice-Coefficient-based segmentation loss on the dense pseudo label provided by the Teacher.

### External Dynamic Cross-Consistency Supervision

To ensure both networks with different architecture collaborate and beneficial each other, external dynamic cross-consistency supervision is proposed. Inspired by MixUp [120], we ensemble the inference of two networks as a dense pseudo label

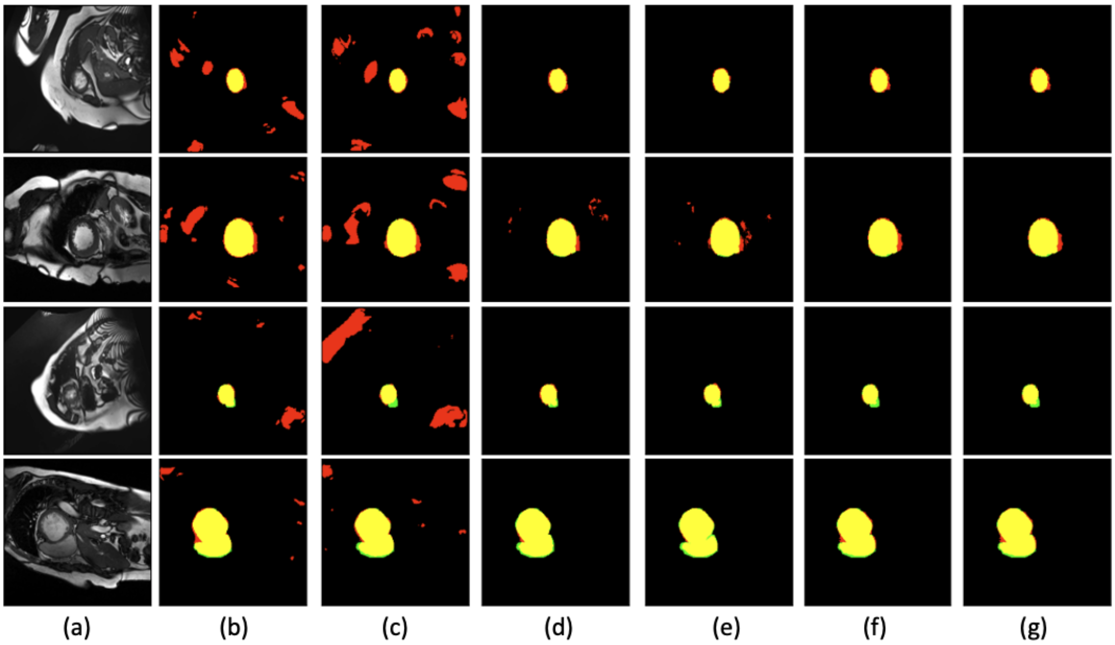


**Figure 7.3:** Example Raw Images with Corresponding Inference. (a) Input MRI image, and Inference by (b) pCE, (c) USTM, (d) Scribble2Label, (e) Mumford-Shah loss, (f) GatedCRFLoss and (g) CHNets.

to iteratively supervise each network [240]. The pseudo label, which provides a complete dense supervisory signal, is formulated as:

$$y_{pseudo} = argmax[\beta y_{\text{cnn}} + (1-\beta) y_{\text{vit}}] \tag{7.5}$$

where $y_{\text{cnn}}, y_{\text{vit}}$ refer to the inference by CNN and ViT, respectively. $y_{pseudo}$ is jointly generated dense pseudo label. $\beta \in [0,1]$ is randomly generated by a uniform distribution, and considered as a kind of 'dynamic' enhanced data perturbation. This process is iterative during training process, thus $y_{pseudo}$ is utilized for network training per iteration as external-consistency loss $L_{exter}$ (seen in Equation 7.6):

$$\mathcal{L}_{\text{exter}}(y_{pseudo}, y_{pred}) = CE(y_{pseudo}, y_{pred}) + Dice(y_{pseudo}, y_{pred}) \tag{7.6}$$

where $CE$ indicates Cross-Entropy-based segmentation loss and $Dice$ indicates Dice-Coefficient-based segmentation loss. Both of losses are all based on the differences between inference and dense pseudo label which is generated by the dynamic pseudo label ensembling.

## 7.4 Experiments and Results

### 7.4.1 Implementation Details

The original UNet [4] and SwinUNet [182] are utilized as the CNN- and ViT-based segmentation backbone respectively for all WSL methods. The MRI Cardiac dataset is selected for validation. The hyper-parameter setting consists of 60,000 training iterations with a batch size of 12. SGD is used as the optimizer, configured with a learning rate of 0.1, momentum of 0.9, and weight decay set to 0.0001. The memory requirement for these experiments is approximately 7 GB, with average runtimes around 4.5 hours. The scribble annotation is generated based on original full dense masks with a simulation algorithm in the data pre-processing stage [276].

### 7.4.2 Qualitative Results

CHNets is directly compared against with several existing WSL baseline methods including Partial Cross-Entropy (pCE) [273], Uncertainty-aware Self-ensembling and Transformation-consistent Mean Teacher (USTM) [213], Scribble2Label (S2L) [29], Mumford-Shah loss (Mumford) [278], and Gated Conditional Random Fields Loss (CRF) [280]. All baseline WSL methods and CHNets are trained with the same hyper-parameter setting, the same loss functions i.e. partial cross-entropy (introduced in Equation 7.1), and the same scribble-based annotation set. For a comprehensive evaluation, each of SSL baseline method is extended to be developed with either CNN- or ViT-based segmentation backbone network. The baseline WSL methods are not suitable with dual-network setting, but an ablation study further explore pure CNN or ViT setting for CHNets. The qualitative results are sketched in Figure 7.3 where the inference of each method is evaluated against original dense label at pixel level where yellow, black, red, and green indicating as true positive, true negative, false positive, false negative, and subfigure (a-g) represents input MRI image, and inference by pCE, USTM, Scribble2Label, Mumford-Shah loss, GatedCRFLoss and CHNets.

**Table 7.1:** The Direct Comparison of the Proposed CHNets Against All Weakly-Supervised Baseline Methods on the Test Set.

| Strategy | Net | Dice | IoU | Acc | Pre | Sen | Spe | HD | ASD |
|---|---|---|---|---|---|---|---|---|---|
| pCE [273] | ViT | 0.8459 | 0.7355 | 0.9954 | 0.8324 | 0.8709 | 0.9975 | 28.6010 | 7.3933 |
| USTM [213] | ViT | 0.8745 | 0.7802 | 0.9959 | 0.8648 | 0.8920 | 0.9977 | 13.4157 | 3.6616 |
| S2L [29] | ViT | 0.8641 | 0.7630 | 0.9960 | 0.8704 | 0.8655 | **0.9982** | 6.4881 | 1.7645 |
| Mumford [278] | ViT | 0.8632 | 0.7614 | 0.9960 | 0.8718 | 0.8620 | **0.9982** | 7.6870 | 2.2027 |
| CRF [280] | ViT | 0.8493 | 0.7405 | 0.9955 | 0.8475 | 0.8678 | 0.9978 | 8.3234 | 2.3858 |
| Scribblesup [273] | CNN | 0.6455 | 0.4918 | 0.9831 | 0.5318 | 0.8945 | 0.9848 | 163.5975 | 69.0296 |
| USTM [213] | CNN | 0.8588 | 0.6147 | 0.9904 | 0.6501 | **0.9203** | 0.9916 | 143.5347 | 44.8333 |
| S2L [29] | CNN | 0.8645 | 0.7644 | 0.9955 | 0.8449 | 0.8904 | 0.9973 | 28.4650 | 7.6293 |
| Mumford [278] | CNN | 0.8681 | 0.7709 | 0.9957 | 0.8518 | 0.8915 | 0.9975 | 23.6676 | 6.6040 |
| CRF [280] | CNN | 0.8709 | 0.7755 | 0.9957 | 0.8519 | 0.9030 | 0.9974 | 7.8396 | 1.8412 |
| **CHNets** | **Hybrid** | **0.8906** | **0.8058** | **0.9964** | **0.8698** | <u>0.9158</u> | <u>0.9978</u> | **5.4180** | **1.6484** |

### 7.4.3 Quantitative Results

The comprehensive quantitative results presenting the direct comparison of CHNets against all WSL baseline methods are reported in Table 7.1 with mean value for each metrics, and we further report the detailed performance of each ROI in Table 7.2. Our proposed methods are highlighted with **Bold**. The best performance are with **Bold**, and the second best performance with the proposed methods are with <u>Underline</u>.

**Table 7.2:** The Direct Comparison of the Proposed CHNets Against All Weakly-Supervised Baseline Methods on the Test Set of Each Segmented Feature.

| Strategy | Net | RV | | | Myo | | | LV | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Dice | HD | ASD | Dice | HD | ASD | Dice | HD | ASD |
| pCE [273] | ViT | 0.8587 | 9.3925 | 3.7748 | 0.7859 | 45.0363 | 10.0612 | 0.8929 | 31.3743 | 8.3438 |
| USTM [213] | ViT | 0.8639 | 9.4354 | 2.9105 | 0.8230 | 14.83338 | 4.3353 | 0.9366 | 15.9782 | 3.7390 |
| S2L [29] | ViT | 0.8727 | 6.7018 | 1.6205 | 0.8105 | 5.7516 | 1.5848 | 0.9091 | 7.0109 | 2.0971 |
| Mumford [278] | ViT | 0.8678 | 6.9280 | **1.6073** | 0.8137 | 7.1041 | 2.1839 | 0.9081 | 9.0291 | 2.8168 |
| CRF [280] | ViT | 0.8622 | 6.9086 | 1.7250 | 0.7904 | 8.0107 | 2.2518 | 0.8952 | 10.0510 | 3.1806 |
| Scribblesup [273] | CNN | 0.5806 | 182.2923 | 87.5389 | 0.5260 | 160.3049 | 68.6412 | 0.8300 | 163.5975 | 69.0296 |
| USTM [213] | CNN | 0.7304 | 138.4518 | 41.0612 | 0.7102 | 125.1634 | 31.8241 | 0.8360 | 166.9888 | 61.6147 |
| S2L [29] | CNN | 0.8502 | 11.2341 | 3.3072 | 0.8156 | 29.3005 | 8.0682 | 0.9276 | 44.8603 | 11.5125 |
| Mumford [278] | CNN | 0.8354 | 29.2791 | 8.0856 | 0.8260 | 24.3843 | 7.6606 | 0.9427 | 17.3394 | 4.0656 |
| CRF [280] | CNN | 0.8519 | 13.5882 | 3.1754 | 0.8164 | 3.8603 | 1.2166 | 0.9444 | 6.0701 | 1.1317 |
| **CHNets** | **Hybrid** | **0.8752** | **8.9538** | 2.3428 | **0.8445** | **3.6503** | **1.5336** | **0.9519** | **3.6499** | **1.0687** |

### 7.4.4 Ablation Study

In the ablation study, we investigate the impact of various contributions and configurations of the CHNets. We explore different combinations of both internal

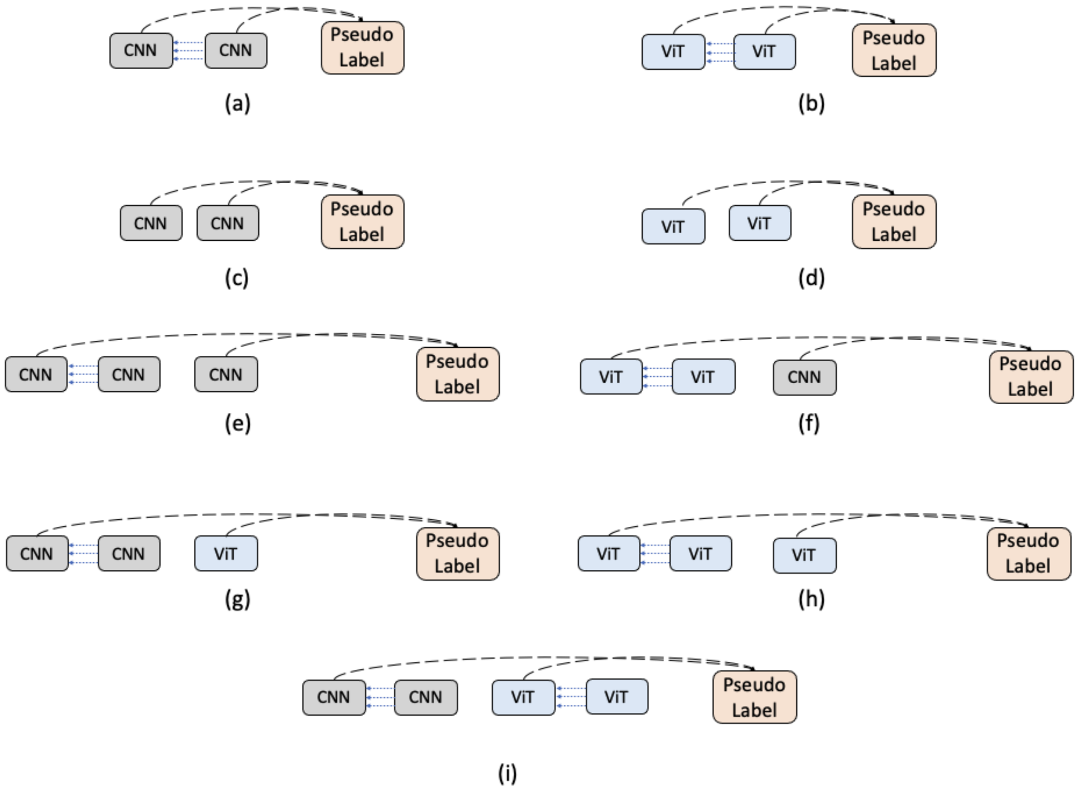


**Figure 7.4:** The Ablation Study of the Different Combinations of Pseudo Label Ensembling.

and external- consistency aware supervision with CNN or ViT backbone, which are sketched in Figure 7.4 and the results are reported in Table 7.3. The internal-consistency training are always with 2 × ✓to align the architecture requirement by the network self-ensembling. ✓filled in all internal-consistency and external-consistency simultaneously with CNN and ViT, refers to as CHNets, which achieves the best performance demonstrating the effectiveness of our proposed techniques. For a comprehensive analysis, we also include pixel-level fully supervised learning with original ground truth results with both CNN and ViT which can be considered as the upper-bound performance compare with weakly supervised learning. We also compare these results against pixel-level fully supervised learning (seen in the bottom of Table 7.3), indicating the maximum achievable performance, and find that CHNets closely match the high standard with only sparse annotations, indicating their practical applicability.

**Table 7.3:** The Ablation Study of Internal & External Consistency-Aware Supervision on Test Set and Pixel-Level Fully Supervision.

| Proposed Consistency-Aware Supervision | | | | | Results | | | |
|---|---|---|---|---|---|---|---|---|
| Strategy | Internal Consistency | | External Consistency | | Dice | IoU | HD | ASD |
| | CNN | ViT | CNN | ViT | | | | |
| a | ✓ × 2 | | | | 0.7083 | 0.5544 | 150.5851 | 50.3175 |
| b | | ✓ × 2 | | | 0.7612 | 0.6253 | 148.5577 | 43.7664 |
| c | | | ✓ × 2 | | 0.8837 | 0.7945 | 6.1310 | 4.9041 |
| d | | | | ✓ × 2 | 0.7392 | 0.6087 | 62.4700 | 24.7017 |
| e | ✓ × 2 | | ✓ | | 0.8846 | 0.7964 | 8.2995 | 2.8425 |
| f | | ✓ × 2 | ✓ | | 0.8880 | 0.8012 | 12.2475 | 3.2928 |
| g | ✓ × 2 | | | ✓ | 0.8815 | 0.7902 | 12.7286 | 3.7176 |
| h | | ✓ × 2 | | ✓ | 0.8633 | 0.7632 | 7.3206 | 2.4864 |
| i | ✓ × 2 | ✓ × 2 | ✓ | ✓ | **0.8906** | **0.8058** | **5.4180** | **1.6484** |
| Pixel-level Fully Supervision CNN | | | | | 0.9167 | 0.9120 | 3.7452 | 0.8615 |
| Pixel-level Fully Supervision ViT | | | | | 0.9049 | 0.8290 | 3.6233 | 0.8749 |

## 7.5 Conclusion and Discussion

In this chapter, the WSL collaborative hybrid networks which simultaneously train both CNN and ViT architectures is following from our insights from the semi-supervised learning chapter (Chapter 5), where the significance of network perturbation is highlighted. By engaging both CNN and ViT architectures, we aim to harness their distinct perspectives and strengths collaboratively, especially under limited-signal supervision situation. The simultaneous training of these two architectures is not just a strategy to achieve a improved feature learning of the imaging data but also a method to encourage a diverse range of 'views' or interpretations by the networks. This diversity is similar to having two experts with different areas of expertise collaborating on the same unknown problem, thereby enriching the analysis and leading to more robust and versatile networks. Such collaboration is particularly important in WSL medical image segmentation study.

WSL study is essential to bring deep learning into real clinical data situation. For example, in the MRI cardiac dataset, when clinicians are primarily concerned with assessing the robustness of the Myocardium, as indicated by the circularity of the Left Ventricle (LV). In such scenarios, the proposed WSL strategy underscores that a detailed segmentation of the LV may not be as crucial as accurately gauging its $xy$ aspect ratio. This crucial measurement can be effectively obtained through networks trained with scribble annotations, demonstrating that precision in medical imaging can be achieved without exhaustive annotation efforts.

In the future, scribble-based WSL is potentially extended to broaden the scope of other forms of limited-signal supervision, such as bounding box or point-based annotations.

In conclusion, a novel WSL study is explored, consisting of integrating internal and external-consistency training schemes. These schemes not only enhance the performance of each individual network but also allow each network benefits from the other. It can be considered as an extended study of SSL framework design with a novel limited data annotated situation. Our quantitative experiments conducted on a public benchmark MRI dataset have yielded promising results, showcasing the potential of our proposed scribble-supervised method in comparison to other similar techniques.

# 8
# Conclusion and Discussion

## Contents

**8.1 Contribution Summary . . . . . . . . . . . . . . . . . . . . 147**
**8.2 Discussion . . . . . . . . . . . . . . . . . . . . . . . . . . . 149**
  8.2.1 How Good is Good Enough? . . . . . . . . . . . . . . . . 149
  8.2.2 Dealing with Unreliable Annotations . . . . . . . . . . . 149
**8.3 Future Work . . . . . . . . . . . . . . . . . . . . . . . . . 150**

## 8.1 Contribution Summary

Throughout this thesis, we have thoroughly proposed and assessed a range of deep learning strategies for medical image segmentation, including supervised, semi-supervised, noise-robust, and weakly-supervised learning. This thesis motivated by many past works focusing on achieving SOTA performance on a specific dataset under a particular setting, without considering the unreliability situations that may arise with medical data in the real world. The extensive investigations of training strategies demonstrating the significant potential of developing advanced deep learning methods for practical clinical data situation. The contribution can be summarized to four fold:

1. **Supervised Learning for Medical Image Segmentation:** The study of supervised learning is to train a segmentation network with comprehensive data annotation with full dense masks. Firstly, we have a literature review of the past studies via 'Network Architecture Engineering', that we summarize the segmentation network development via backbone network, network block, and training strategy. Secondly, we introduce a 'Medical Image Segmentation Triathlon' aiming to have a comprehensive evaluation of segmentation network with CT, MRI, Ultrasound, Histology segmentation datasets with data pre-processing process to simulate different scenarios, evaluation metrics, and the computational platform. Finally, we have explored and proposed a variety of advanced network blocks with UNet resulting in RARUNet [17], QAPNet [194], and Hybrid UNet based on CNN & ViT [31].

2. **Semi-Supervised Learning for Medical Image Segmentation:** The study of SSL is to train a segmentation network with partial data annotation with full dense masks. The consistency regularization with several classical SSL frameworks are summarized and introduced. We further explored several advanced schemes such as uncertainty estimation, adversarial learning with SSL resulting in Triple-View Learning [170], Examiner-Student-Teacher [248], Uncertainty-Aware Vision Transformer [167], and Computationally-Efficient Cross-Supervised Vision Transformer [168].

3. **Noise-Robust Learning for Medical Image Segmentation:** The study of noise-robust learning is to train a segmentation network with comprehensive data annotation with noisy dense masks. The simulation algorithm of generating noisy label is developed. We propose Adaptive Denoising Learning strategy to effectively detect and remove noisy label during training process to reduce the impact of noisy label. The proposed strategy is validated with various segmentation backbone networks under different proportion and noise level data situation, resulting in noisy-robust networks [17, 31].

4. **Weakly-Supervised Learning for Medical Image Segmentation:** The study of WSL is to train a segmentation network with comprehensive data annotation with sketchy contours. The simulation algorithm of generating sparse label, i.e. scribble is introduced. Following the concept of SSL to train network with limited annotations, i.e. consistency regularization, we introduce a hybrid CNN- and ViT-based networks with internal and external consistency training, resulting in CHNets [277].

## 8.2 Discussion

### 8.2.1 How Good is Good Enough?

As the current studies mainly explores towards more complex network architecture with various combinations on different dataset, it is essential to assess the 'good enough' performance. The 'good performance' cannot be solely defined by how well a network overfitting on a single dataset under ideal conditions. Instead, this thesis argues to broaden current community perspective to consider how robust the proposed novel networks to unreliable annotations, how adaptable they are to various data situations, and how well they can learn with limited annotations. In essence, the 'good enough' needs to be focused from a narrow 'overfitting' game in idealistic academic view to a more practical, realistic clinical data situation.

### 8.2.2 Dealing with Unreliable Annotations

The problem of unreliable annotations is a significant challenge in deep learning-based medical image segmentation. To tackle the issue of networking training with unreliable data, we explore various semi-supervised, noise-robust, and weakly-supervised learning techniques throughout this thesis. These strategies can improve the robustness of deep learning networks to annotation and provide a foundation for future research in this area.

## 8.3 Future Work

In the future, there are several potential directions that still need to be explored.

1. **Further study in Semi-Supervised Learning:** Although our semi-supervised works have achieved competitive performance with a small amount of labeled data supplemented by a larger volume of unlabeled data, compared with fully supervised learning but with low annotation cost. In our study, we randomly select a proportion of data as labeled data (e.g. 10%) and the rest of data (e.g. 90%) is unlabeled data. An essential and expandable approach, however, is to study what is most effective way to specifically select a subset data (e.g. 10%) from entire dataset to enquiry clinicians' labeling that maximizes network training performance, rather than randomly selection. Active learning is a potential approach need to be studied further with SSL [281–284].

2. **Further study in Noise-Robust Learning:** Our current research in noise-robust learning is under the knowledge of the proportion of noisy labels in the dataset. A potential approach is to explore SSL with ADL Strategy once the noisy label been removed while training process. The future challenge can be considered to develop segmentation networks training strategy when the proportion of noisy label is unknown. This would align more closely with real-world 'good enough' scenarios where the quality of data annotations is often uncertain.

3. **Further study in Weakly-Supervised Learning:** Our exploration in scribble-supervised learning is one of the most practical and efficient labeling way for clinicians. Some of other forms of limited-supervision signals such as bounding boxes, check marks, and points, are also worth to study in the weakly-supervised learning. Moreover, there is potential for uniforming weakly-supervised and semi-supervised learning strategies together, given both of WSL and SSL focus on learning from limited signals.

4. **Domain Adaptation:** The challenge of applying network trained on one modality, such as CT scans, to another, like MRI, can also be a barrier in practical clinical scenario. Future work could focus on exploring our SSL strategies for domain adaptation to address shifts in data distributions, through we assume the labeled and unlabeled data are from two separate modalities.

5. **Leveraging Large Language Models:** The recent success of large language models present an opportunity for medical imaging analysis. The integration of language and vision networks is an unexplored study area to train network, potentially leading to breakthrough in network performance improvement with more modality data (e.g. medical prescriptions with corresponding medical images).

Overall, the aim of this thesis is to bring more reliable, robust, and practical deep learning networks of medical image segmentation to real clinical practice.

# Bibliography

[1] Dan Ciresan et al. “Deep neural networks segment neuronal membranes in electron microscopy images”. In: *Advances in neural information processing systems* 25 (2012).

[2] Ross Girshick et al. “Rich feature hierarchies for accurate object detection and semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.

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

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[5] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. “V-net: Fully convolutional neural networks for volumetric medical image segmentation”. In: *2016 fourth international conference on 3D vision (3DV)*. IEEE. 2016, pp. 565–571.

[6] Fabian Isensee et al. “nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation”. In: *Nature methods* 18.2 (2021), pp. 203–211.

[7] Enze Xie et al. “SegFormer: Simple and efficient design for semantic segmentation with transformers”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 12077–12090.

[8] Olivier Bernard and etc. “Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: is the problem solved?” In: *IEEE transactions on medical imaging* 37.11 (2018), pp. 2514–2525.

[9] Bjoern H Menze and etc. “The multimodal brain tumor image segmentation benchmark (BRATS)”. In: *IEEE transactions on medical imaging* 34.10 (2014), pp. 1993–2024.

[10] Ophir Gozes et al. “Rapid ai development cycle for the coronavirus (covid-19) pandemic: Initial results for automated detection & patient monitoring using deep learning ct image analysis”. In: *arXiv preprint arXiv:2003.05037* (2020).

[11] Guotai Wang et al. “A noise-robust framework for automatic segmentation of COVID-19 pneumonia lesions from CT images”. In: *IEEE Transactions on Medical Imaging* 39.8 (2020), pp. 2653–2663.

[12] Ilker Hacihaliloglu. “Ultrasound imaging and segmentation of bone surfaces: A review”. In: *Technology* 5.02 (2017), pp. 74–80.

[13] J Alison Noble and Djamal Boukerroui. “Ultrasound image segmentation: a survey”. In: *IEEE Transactions on medical imaging* 25.8 (2006), pp. 987–1010.

[14] Xiangde Luo et al. “Semi-supervised medical image segmentation via cross teaching between cnn and transformer”. In: *International Conference on Medical Imaging with Deep Learning.* PMLR. 2022, pp. 820–833.

[15] Robin Strudel et al. “Segmenter: Transformer for semantic segmentation”. In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2021, pp. 7262–7272.

[16] Xiangde Luo et al. “Semi-supervised Medical Image Segmentation through Dual-task Consistency”. In: *AAAI Conference on Artificial Intelligence.* 2021, pp. 8801–8809.

[17] Ziyang Wang and etc. “Rar-u-net: a residual encoder to attention decoder by residual connections framework for spine segmentation under noisy labels”. In: *2021 IEEE International Conference on Image Processing (ICIP).* IEEE. 2021.

[18] Guha Balakrishnan et al. “Voxelmorph: a learning framework for deformable medical image registration”. In: *IEEE transactions on medical imaging* 38.8 (2019), pp. 1788–1800.

[19] Alessa Hering et al. “Learn2Reg: comprehensive multi-task medical image registration challenge, dataset and evaluation in the era of deep learning”. In: *IEEE Transactions on Medical Imaging* (2022).

[20] Mattias P Heinrich et al. “MRF-based deformable registration and ventilation estimation of lung CT”. In: *IEEE transactions on medical imaging* 32.7 (2013), pp. 1239–1248.

[21] Tsung-Yi Lin et al. “Feature pyramid networks for object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 2117–2125.

[22] Seung-Wook Kim et al. “Parallel feature pyramid network for object detection”. In: *Proceedings of the European Conference on Computer Vision (ECCV).* 2018, pp. 234–250.

[23] Eduardo Luz et al. “Towards an efficient deep learning model for covid-19 patterns detection in x-ray images”. In: *arXiv preprint arXiv:2004.05717* (2020).

[24] Mesut To gacar, Burhan Ergen, and Zafer Comert. “COVID-19 detection using deep learning models to exploit Social Mimic Optimization and structured chest X-ray images using fuzzy color and stacking approaches”. In: *Computers in biology and medicine* 121 (2020), p. 103805.

[25] Yingda Xia et al. “3d semi-supervised learning with uncertainty-aware multi-view co-training”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision.* 2020, pp. 3646–3655.

[26] Xiaomeng Li and etc. “Transformation-consistent self-ensembling model for semisupervised medical image segmentation”. In: *IEEE Transactions on Neural Networks and Learning Systems* 32.2 (2020), pp. 523–534.

[27] Xiaomeng Li et al. “Semi-supervised skin lesion segmentation via transformation consistent self-ensembling model”. In: *arXiv preprint arXiv:1808.03887* (2018).

[28] Jungbeom Lee, Eunji Kim, and Lee et al. “Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2019, pp. 5267–5276.

[29] Hyeonsoo Lee and Won-Ki Jeong. “Scribble2label: Scribble-supervised cell segmentation via self-generating pseudo-labels with consistency”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer. 2020, pp. 14–23.

[30] Jinchi Huang et al. “O2u-net: A simple noisy label detection approach for deep neural networks”. In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2019, pp. 3326–3334.

[31] Ziyang Wang and Irina Voiculescu. “Dealing with Unreliable Annotations: Noise-Robust Network for Semantic Segmentation through Transformer-Improved-Encoder and Convolution-Decoder”. In: *Applied Sciences* (2023).

[32] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. “Segnet: A deep convolutional encoder-decoder architecture for image segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence* 39.12 (2017), pp. 2481–2495.

[33] Liang-Chieh Chen et al. “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2017), pp. 834–848.

[34] Guosheng Lin et al. “Refinenet: Multi-path refinement networks for high-resolution semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 1925–1934.

[35] Xiaolong Wang et al. “Non-local neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2018, pp. 7794–7803.

[36] Xizhou Zhu et al. “An empirical study of spatial attention mechanisms in deep networks”. In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2019, pp. 6688–6697.

[37] Zilong Zhong et al. “Squeeze-and-attention networks for semantic segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2020, pp. 13065–13074.

[38] Sanghyun Woo et al. “Cbam: Convolutional block attention module”. In: *Proceedings of the European conference on computer vision (ECCV).* 2018, pp. 3–19.

[39] Ashish Vaswani and etc. “Attention is all you need”. In: *Advances in neural information processing systems.* 2017.

[40] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. “Layer normalization”. In: *arXiv preprint arXiv:1607.06450* (2016).

[41] Tim Cooijmans et al. “Recurrent batch normalization”. In: *arXiv preprint arXiv:1603.09025* (2016).

[42] Yuxin Wu and Kaiming He. “Group normalization”. In: *Proceedings of the European conference on computer vision (ECCV).* 2018, pp. 3–19.

[43] Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *International conference on machine learning*. PMLR. 2015, pp. 448–456.

[44] Hengshuang Zhao et al. “Pyramid scene parsing network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2881–2890.

[45] Varduhi Yeghiazaryan and Irina Voiculescu. “Boundary overlap for medical image segmentation evaluation”. In: *Medical Imaging 2017: Image-Guided Procedures, Robotic Interventions, and Modeling*. Vol. 10135. International Society for Optics and Photonics. 2017, 101351J.

[46] Tsung-Yi Lin et al. “Focal loss for dense object detection”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2980–2988.

[47] Suvrit Sra, Sebastian Nowozin, and Stephen J Wright. *Optimization for machine learning*. Mit Press, 2012.

[48] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[49] Sergey Zagoruyko and Nikos Komodakis. “Wide residual networks”. In: *arXiv preprint arXiv:1605.07146* (2016).

[50] Hang Zhang et al. “Resnest: Split-attention networks”. In: *arXiv preprint arXiv:2004.08955* (2020).

[51] Alejandro Newell, Kaiyu Yang, and Jia Deng. “Stacked hourglass networks for human pose estimation”. In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*. Springer. 2016, pp. 483–499.

[52] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. “Spatial transformer networks”. In: *Advances in neural information processing systems* 28 (2015).

[53] Taco Cohen and Max Welling. “Group equivariant convolutional networks”. In: *International conference on machine learning*. PMLR. 2016, pp. 2990–2999.

[54] Fisher Yu and Vladlen Koltun. “Multi-scale context aggregation by dilated convolutions”. In: *arXiv preprint arXiv:1511.07122* (2015).

[55] Yunpeng Chen et al. “Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 3435–3444.

[56] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[57] Francois Chollet. “Xception Deep learning with depthwise separable convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.

[58] Chen Qin et al. “Convolutional recurrent neural networks for dynamic MR image reconstruction”. In: *IEEE transactions on medical imaging* 38.1 (2018), pp. 280–290.

[59] Kai Han et al. “Ghostnet: More features from cheap operations”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2020, pp. 1580–1589.

[60] Tao Wang et al. “Pnp-detr: Towards efficient visual analysis with transformers”. In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2021, pp. 4661–4670.

[61] Yunpeng Chen et al. “Dual path networks”. In: *Advances in neural information processing systems* 30 (2017).

[62] Jie Hu, Li Shen, and Gang Sun. “Squeeze-and-excitation networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2018, pp. 7132–7141.

[63] Zilong Huang et al. “Ccnet: Criss-cross attention for semantic segmentation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2019, pp. 603–612.

[64] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. “Neural machine translation by jointly learning to align and translate”. In: *arXiv preprint arXiv:1409.0473* (2014).

[65] Rewon Child et al. “Generating long sequences with sparse transformers”. In: *arXiv preprint arXiv:1904.10509* (2019).

[66] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. “Effective approaches to attention-based neural machine translation”. In: *arXiv preprint arXiv:1508.04025* (2015).

[67] Han Zhang et al. “Self-attention generative adversarial networks”. In: *International conference on machine learning.* PMLR. 2019, pp. 7354–7363.

[68] Jun Fu et al. “Dual attention network for scene segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2019, pp. 3146–3154.

[69] Long Chen et al. “Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 5659–5667.

[70] Fei Wang et al. “Residual attention network for image classification”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 3156–3164.

[71] Moemmur Shahzad et al. “InferNER: An attentive model leveraging the sentence-level information for Named Entity Recognition in Microblogs”. In: *The International FLAIRS Conference Proceedings.* Vol. 34. 2021.

[72] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. “Recurrent models of visual attention”. In: *Advances in neural information processing systems* 27 (2014).

[73] Iz Beltagy, Matthew E Peters, and Arman Cohan. “Longformer: The long-document transformer”. In: *arXiv preprint arXiv:2004.05150* (2020).

[74] Kaiming He et al. “Spatial pyramid pooling in deep convolutional networks for visual recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 37.9 (2015), pp. 1904–1916.

[75] Maoke Yang et al. "Denseaspp for semantic segmentation in street scenes". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2018, pp. 3684–3692.

[76] Seokmin Han et al. "A deep learning framework for supporting the classification of breast lesions in ultrasound images". In: *Physics in Medicine & Biology* 62.19 (2017), p. 7714.

[77] Shang-Hua Gao et al. "Res2net: A new multi-scale backbone architecture". In: *IEEE transactions on pattern analysis and machine intelligence* 43.2 (2019), pp. 652–662.

[78] Mahyar Najibi, Bharat Singh, and Larry S Davis. "Autofocus: Efficient multi-scale inference". In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2019, pp. 9745–9755.

[79] Ze Liu et al. "Swin transformer: Hierarchical vision transformer using shifted windows". In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2021, pp. 10012–10022.

[80] Yi Li et al. "Data-driven neuron allocation for scale aggregation networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2019, pp. 11526–11534.

[81] Junjie Ke et al. "Musiq: Multi-scale image quality transformer". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2021, pp. 5148–5157.

[82] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. "Crossvit: Cross-attention multi-scale vision transformer for image classification". In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2021, pp. 357–366.

[83] Bo Chen et al. "Mnasfpn: Learning latency-aware pyramid architecture for object detection on mobile devices". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2020, pp. 13607–13616.

[84] Bharat Singh, Mahyar Najibi, and Larry S Davis. "Sniper: Efficient multi-scale training". In: *Advances in neural information processing systems* 31 (2018).

[85] Tongle Fan et al. "Ma-net: A multi-scale attention network for liver and tumor segmentation". In: *IEEE Access* 8 (2020), pp. 179656–179665.

[86] Baoyuan Wu et al. "Tencent ml-images: A large-scale multi-label image database for visual representation learning". In: *IEEE Access* 7 (2019), pp. 172683–172693.

[87] Chao Yang et al. "High-resolution image inpainting using multi-scale neural patch synthesis". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 6721–6729.

[88] Carole H Sudre et al. "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations". In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3.* Springer. 2017, pp. 240–248.

[89] Xingping Dong and Jianbing Shen. “Triplet loss in siamese network for object tracking”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 459–474.

[90] Jun-Yan Zhu et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2223–2232.

[91] Ishaan Gulrajani et al. “Improved training of wasserstein gans”. In: *Advances in neural information processing systems* 30 (2017).

[92] Xu Chen et al. “Learning active contour models for medical image segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 11632–11640.

[93] Kihyuk Sohn. “Improved deep metric learning with multi-class n-pair loss objective”. In: *Advances in neural information processing systems* 29 (2016).

[94] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. “Representation learning with contrastive predictive coding”. In: *arXiv preprint arXiv:1807.03748* (2018).

[95] Yaowei Zheng, Richong Zhang, and Yongyi Mao. “Regularizing neural networks via adversarial model perturbation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 8156–8165.

[96] Pierre Foret et al. “Sharpness-aware minimization for efficiently improving generalization”. In: *arXiv preprint arXiv:2010.01412* (2020).

[97] Jiankang Deng et al. “Arcface: Additive angular margin loss for deep face recognition”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 4690–4699.

[98] Benjamin Graham. “Fractional max-pooling”. In: *arXiv preprint arXiv:1412.6071* (2014).

[99] Dan Hendrycks and Kevin Gimpel. “Gaussian error linear units (gelus)”. In: *arXiv preprint arXiv:1606.08415* (2016).

[100] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. “Fast and accurate deep network learning by exponential linear units (elus)”. In: *arXiv preprint arXiv:1511.07289* (2015).

[101] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. “Binaryconnect: Training deep neural networks with binary weights during propagations”. In: *Advances in neural information processing systems* 28 (2015).

[102] Min Lin, Qiang Chen, and Shuicheng Yan. “Network in network”. In: *arXiv preprint arXiv:1312.4400* (2013).

[103] Mateusz Malinowski and Mario Fritz. “Learnable pooling regions for image classification”. In: *arXiv preprint arXiv:1301.3516* (2013).

[104] Prajit Ramachandran, Barret Zoph, and Quoc V Le. “Searching for activation functions”. In: *arXiv preprint arXiv:1710.05941* (2017).

[105] Kristof Schütt et al. “Schnet: A continuous-filter convolutional neural network for modeling quantum interactions”. In: *Advances in neural information processing systems* 30 (2017).

[106] Wenling Shang et al. “Understanding and improving convolutional neural networks via concatenated rectified linear units”. In: *international conference on machine learning*. PMLR. 2016, pp. 2217–2225.

[107] Xinyu Liu and Xiaoguang Di. “TanhExp: A smooth activation function with high convergence speed for lightweight neural networks”. In: *IET Computer Vision* 15.2 (2021), pp. 136–150.

[108] Matthew D Zeiler and Rob Fergus. “Stochastic pooling for regularization of deep convolutional neural networks”. In: *arXiv preprint arXiv:1301.3557* (2013).

[109] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. “Sigmoid-weighted linear units for neural network function approximation in reinforcement learning”. In: *Neural Networks* 107 (2018), pp. 3–11.

[110] David R So et al. “Primer: Searching for efficient transformers for language modeling”. In: *arXiv preprint arXiv:2109.08668* (2021).

[111] Simone Scardapane et al. “Kafnets: Kernel-based non-parametric activation functions for neural networks”. In: *Neural Networks* 110 (2019), pp. 19–32.

[112] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. “Instance normalization: The missing ingredient for fast stylization”. In: *arXiv preprint arXiv:1607.08022* (2016).

[113] Lei Huang et al. “Centered weight normalization in accelerating training of deep neural networks”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2803–2811.

[114] Lei Huang et al. “Iterative normalization: Beyond standardization towards efficient whitening”. In: *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*. 2019, pp. 4874–4883.

[115] Ping Luo et al. “Differentiable learning-to-normalize via switchable normalization”. In: *arXiv preprint arXiv:1806.10779* (2018).

[116] Anna C Reisetter et al. “Mixture model normalization for non-targeted gas chromatography/mass spectrometry metabolomics data”. In: *Bmc Bioinformatics* 18.1 (2017), pp. 1–17.

[117] Lei Huang et al. “Decorrelated batch normalization”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 791–800.

[118] Xiao-Yun Zhou et al. “Batch group normalization”. In: *arXiv preprint arXiv:2012.02782* (2020).

[119] Sergey Ioffe. “Batch renormalization: Towards reducing minibatch dependence in batch-normalized models”. In: *Advances in neural information processing systems* 30 (2017).

[120] Hongyi Zhang et al. “mixup: Beyond empirical risk minimization”. In: *arXiv preprint arXiv:1710.09412* (2017).

[121] Ekin D Cubuk et al. “Randaugment: Practical automated data augmentation with a reduced search space”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2020, pp. 702–703.

[122] Sangdoo Yun et al. “Cutmix: Regularization strategy to train strong classifiers with localizable features”. In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2019, pp. 6023–6032.

[123] Terrance DeVries and Graham W Taylor. “Improved regularization of convolutional neural networks with cutout”. In: *arXiv preprint arXiv:1708.04552* (2017).

[124] Sungbin Lim et al. “Fast autoaugment”. In: *Advances in Neural Information Processing Systems* 32 (2019).

[125] Ekin D Cubuk et al. “Autoaugment: Learning augmentation policies from data”. In: *arXiv preprint arXiv:1805.09501* (2018).

[126] Golnaz Ghiasi et al. “Simple copy-paste is a strong data augmentation method for instance segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2021, pp. 2918–2928.

[127] Dan Hendrycks et al. “Augmix: A simple data processing method to improve robustness and uncertainty”. In: *arXiv preprint arXiv:1912.02781* (2019).

[128] Zhun Zhong et al. “Random erasing data augmentation”. In: *Proceedings of the AAAI conference on artificial intelligence.* Vol. 34. 07. 2020, pp. 13001–13008.

[129] Shiqi Lin et al. “Local patch autoaugment with multi-agent collaboration”. In: *arXiv preprint arXiv:2103.11099* (2021).

[130] Herbert Robbins and Sutton Monro. “A stochastic approximation method”. In: *The annals of mathematical statistics* (1951), pp. 400–407.

[131] Ilya Loshchilov and Frank Hutter. “Decoupled weight decay regularization”. In: *arXiv preprint arXiv:1711.05101* (2017).

[132] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. “Neural networks for machine learning lecture 6a overview of mini-batch gradient descent”. In: *Cited on* 14.8 (2012), p. 2.

[133] Dariush Bahrami and Sadegh Pouriyan Zadeh. “Gravity optimizer: a kinematic approach on optimization in deep learning”. In: *arXiv preprint arXiv:2101.09192* (2021).

[134] Jianqiao Wangni et al. “Gradient sparsification for communication-efficient distributed optimization”. In: *Advances in Neural Information Processing Systems* 31 (2018).

[135] Juntang Zhuang et al. “Adabelief optimizer: Adapting stepsizes by the belief in observed gradients”. In: *Advances in neural information processing systems* 33 (2020), pp. 18795–18806.

[136] Yang You et al. “Large batch optimization for deep learning: Training bert in 76 minutes”. In: *arXiv preprint arXiv:1904.00962* (2019).

[137] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. “Simple and scalable predictive uncertainty estimation using deep ensembles”. In: *Advances in neural information processing systems* 30 (2017).

[138] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. “SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives”. In: *Advances in neural information processing systems* 27 (2014).

[139] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. "On the convergence of adam and beyond". In: *arXiv preprint arXiv:1904.09237* (2019).

[140] André Biedenkapp et al. "Dynamic algorithm configuration: Foundation of a new meta-algorithmic framework". In: *ECAI 2020*. IOS Press, 2020, pp. 427–434.

[141] Xuezhe Ma. "Apollo: An adaptive parameter-wise diagonal quasi-newton method for nonconvex stochastic optimization". In: *arXiv preprint arXiv:2009.13586* (2020).

[142] Noam Shazeer and Mitchell Stern. "Adafactor: Adaptive learning rates with sublinear memory cost". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 4596–4604.

[143] Nicola Landro, Ignazio Gallo, and Riccardo La Grassa. "Mixing Adam and SGD: a combined optimization method". In: *arXiv preprint arXiv:2011.08042* (2020).

[144] Sebastian U Stich. "Local SGD converges fast and communicates little". In: *arXiv preprint arXiv:1805.09767* (2018).

[145] Pavel Izmailov et al. "Averaging weights leads to wider optima and better generalization". In: *arXiv preprint arXiv:1803.05407* (2018).

[146] Yann LeCun et al. "Backpropagation applied to handwritten zip code recognition". In: *Neural computation* 1.4 (1989), pp. 541–551.

[147] Gao Huang et al. "Densely connected convolutional networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4700–4708.

[148] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *Int Conf Learning Representations(ICLR)* (2015).

[149] Christian Szegedy et al. "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[150] Joseph Redmon and Ali Farhadi. "Yolov3: An incremental improvement". In: *arXiv preprint arXiv:1804.02767* (2018).

[151] Ningning Ma et al. "Shufflenet v2: Practical guidelines for efficient cnn architecture design". In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 116–131.

[152] Forrest N Iandola et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size". In: *arXiv preprint arXiv:1602.07360* (2016).

[153] Xiangyu Zhang et al. "Shufflenet: An extremely efficient convolutional neural network for mobile devices". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 6848–6856.

[154] Andrew G Howard et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications". In: *arXiv preprint arXiv:1704.04861* (2017).

[155] Mark Sandler et al. "Mobilenetv2: Inverted residuals and linear bottlenecks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4510–4520.

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

[157] Mingxing Tan and Quoc Le. “Efficientnetv2: Smaller models and faster training”. In: *International conference on machine learning*. PMLR. 2021, pp. 10096–10106.

[158] Mingxing Tan and Quoc Le. “Efficientnet: Rethinking model scaling for convolutional neural networks”. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 6105–6114.

[159] Christian Szegedy et al. “Rethinking the inception architecture for computer vision”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2818–2826.

[160] Jingdong Wang et al. “Deep high-resolution representation learning for visual recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 43.10 (2020), pp. 3349–3364.

[161] Christian Szegedy et al. “Inception-v4, inception-resnet and the impact of residual connections on learning”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 31. 1. 2017.

[162] Xiaomeng Li et al. “H-DenseUNet: hybrid densely connected UNet for liver and tumor segmentation from CT volumes”. In: *IEEE transactions on medical imaging* 37.12 (2018), pp. 2663–2674.

[163] Runwen Hu et al. “Automated diagnosis of covid-19 using deep learning and data augmentation on chest ct”. In: *medRxiv* (2020).

[164] Shervin Minaee et al. “Deep-covid: Predicting covid-19 from chest x-ray images using deep transfer learning”. In: *Medical image analysis* 65 (2020), p. 101794.

[165] Ali Abbasian Ardakani et al. “Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks”. In: *Computers in Biology and Medicine* 121 (2020), p. 103795.

[166] Patrick Ferdinand Christ et al. “Automatic liver and tumor segmentation of CT and MRI volumes using cascaded fully convolutional neural networks”. In: *arXiv preprint arXiv:1702.05970* (2017).

[167] Ziyang Wang, Jian-Qing Zheng, and Irina Voiculescu. “An uncertainty-aware transformer for MRI cardiac semantic segmentation via mean teachers”. In: *Medical Image Understanding and Analysis: 26th Annual Conference, MIUA 2022, Cambridge, UK, July 27–29, 2022, Proceedings*. Springer. 2022, pp. 494–507.

[168] Ziyang Wang, Nanqing Dong, and Irina Voiculescu. “Computationally-efficient vision transformer for medical image semantic segmentation via dual pseudo-label supervision”. In: *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2022, pp. 1961–1965.

[169] Jevgenij Gamper and etc. “PanNuke: an open pan-cancer histology dataset for nuclei instance segmentation and classification”. In: *European Congress on Digital Pathology*. Springer. 2019.

[170] Ziyang Wang and Irina Voiculescu. “Triple-view feature learning for medical image segmentation”. In: *Resource-Efficient Medical Image Analysis: First MICCAI Workshop, REMIA 2022, Singapore, September 22, 2022, Proceedings.* Springer. 2022, pp. 42–54.

[171] Zhemin Zhuang, Alex Noel Joseph Raj, and et al. “Nipple Segmentation and Localization Using Modified U-Net on Breast Ultrasound Images”. In: *Journal of Medical Imaging and Health Informatics* 9.9 (2019), pp. 1827–1837.

[172] Hao Chen et al. “Iterative multi-domain regularized deep learning for anatomical structure detection and segmentation from ultrasound images”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19.* Springer. 2016, pp. 487–495.

[173] Özgün Çiçek et al. “3D U-Net: learning dense volumetric segmentation from sparse annotation”. In: *International conference on medical image computing and computer-assisted intervention.* Springer. 2016, pp. 424–432.

[174] Florian Dubost et al. “Gp-unet: Lesion detection from weak labels with a 3d regression network”. In: *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III.* Springer. 2017, pp. 214–221.

[175] Zongwei Zhou et al. “Unet++: A nested u-net architecture for medical image segmentation”. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support.* Springer, 2018, pp. 3–11.

[176] Qiangguo Jin et al. “RA-UNet: A hybrid deep attention-aware network to extract liver and tumor in CT scans”. In: *Frontiers in Bioengineering and Biotechnology* 8 (2020), p. 1471.

[177] Nabil Ibtehaz and M Sohel Rahman. “MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation”. In: *Neural Networks* 121 (2020), pp. 74–87.

[178] O Oktay et al. “Attention U-Net: Learning where to look for the pancreas”. In: *Int Conf Medical Imaging with Deep Learning* (2018).

[179] José Ignacio Orlando et al. “U2-net: A bayesian u-net model with epistemic uncertainty feedback for photoreceptor layer segmentation in pathological oct scans”. In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019).* IEEE. 2019, pp. 1441–1445.

[180] Huimin Huang et al. “UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation”. In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE. 2020, pp. 1055–1059.

[181] Jieneng Chen and etc. “Transunet: Transformers make strong encoders for medical image segmentation”. In: *arXiv preprint arXiv:2102.04306* (2021).

[182] Hu Cao et al. “Swin-unet: Unet-like pure transformer for medical image segmentation”. In: *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III.* Springer. 2023, pp. 205–218.

[183] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012), pp. 1097–1105.

[184] Liang-Chieh Chen et al. “Semantic image segmentation with deep convolutional nets and fully connected crfs”. In: *arXiv preprint arXiv:1412.7062* (2014).

[185] Liang-Chieh Chen et al. “Rethinking atrous convolution for semantic image segmentation”. In: *arXiv preprint arXiv:1706.05587* (2017).

[186] Alexey Dosovitskiy and etc. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *arXiv preprint arXiv:2010.11929* (2020).

[187] Hugo Touvron and etc. “Training data-efficient image transformers & distillation through attention”. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 10347–10357.

[188] Nicolas Carion and etc. “End-to-end object detection with transformers”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 213–229.

[189] Yanghao Li et al. “Exploring plain vision transformer backbones for object detection”. In: *arXiv preprint arXiv:2203.16527* (2022).

[190] Hwanjun Song et al. “Vidt: An efficient and effective fully transformer-based object detector”. In: *arXiv preprint arXiv:2110.03921* (2021).

[191] Sixiao Zheng et al. “Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 6881–6890.

[192] Ze Liu et al. “Video swin transformer”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 3202–3211.

[193] Hu Cao et al. “Swin-unet: Unet-like pure transformer for medical image segmentation”. In: (2023), pp. 205–218.

[194] Ziyang Wang and Irina Voiculescu. “Quadruple Augmented Pyramid Network for Multi-class COVID-19 Segmentation via CT”. In: *2021 43rd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*. 2021.

[195] Nitish Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[196] Junjie Yan et al. “Towards stabilizing batch statistics in backward propagation of batch normalization”. In: *arXiv preprint arXiv:2001.06838* (2020).

[197] Irving John Good. “Rational decisions”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 14.1 (1952), pp. 107–114.

[198] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. “Tversky loss function for image segmentation using 3D fully convolutional deep networks”. In: *Machine Learning in Medical Imaging: 8th International Workshop, MLMI 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, September 10, 2017, Proceedings 8*. Springer. 2017, pp. 379–387.

[199] Chi Wang et al. “Active boundary loss for semantic segmentation”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 2. 2022, pp. 2397–2405.

[200] John Duchi, Elad Hazan, and Yoram Singer. “Adaptive subgradient methods for online learning and stochastic optimization.” In: *Journal of machine learning research* 12.7 (2011).

[201] Chuansheng Zheng et al. “Deep learning-based detection for COVID-19 from chest CT using weak label”. In: *MedRxiv* (2020).

[202] Xiaolong Qi et al. “Machine learning-based CT radiomics model for predicting hospital stay in patients with pneumonia associated with SARS-CoV-2 infection: A multicenter study”. In: *Medrxiv* (2020).

[203] Shuo Jin et al. “AI-assisted CT imaging analysis for COVID-19 screening: Building and deploying a medical AI system in four weeks”. In: *MedRxiv* (2020).

[204] Halgurd S Maghdid et al. “Diagnosing COVID-19 pneumonia from X-ray and CT images using deep learning and transfer learning algorithms”. In: *arXiv preprint arXiv:2004.00038* (2020).

[205] Abhishek Chaurasia and Eugenio Culurciello. “Linknet: Exploiting encoder representations for efficient semantic segmentation”. In: *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE. 2017, pp. 1–4.

[206] Fei Shan et al. “Lung infection quantification of COVID-19 in CT images with deep learning”. In: *arXiv preprint arXiv:2003.04655* (2020).

[207] Tongxue Zhou, Stephane Canu, and Su Ruan. “An automatic COVID-19 CT segmentation network using spatial and channel attention mechanism”. In: *arXiv preprint arXiv:2004.06673* (2020).

[208] Muhammad Farooq and Abdul Hafeez. “Covid-resnet: A deep learning framework for screening of covid19 from radiographs”. In: *arXiv preprint arXiv:2003.14395* (2020).

[209] Ezz El-Din Hemdan, Marwa A Shouman, and Mohamed Esmail Karar. “Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images”. In: *arXiv preprint arXiv:2003.11055* (2020).

[210] Rodolfo M Pereira et al. “COVID-19 identification in chest X-ray images on flat and hierarchical classification scenarios”. In: *Computer Methods and Programs in Biomedicine* 194 (2020), p. 105532.

[211] Karim Hammoudi et al. “Deep learning on chest x-ray images to detect and evaluate pneumonia cases at the era of covid-19”. In: *arXiv preprint arXiv:2004.03399* (2020).

[212] Sampa Misra et al. “Multi-channel transfer learning of chest x-ray images for screening of covid-19”. In: *Electronics* 9.9 (2020), p. 1388.

[213] Xiaoming Liu et al. “Weakly supervised segmentation of COVID19 infection with scribble annotation on CT images”. In: *Pattern recognition* 122 (2022), p. 108341.

[214] M Kolařıék et al. “Optimized high resolution 3d dense-u-net network for brain and spine segmentation”. In: *Applied Sciences* 9.3 (2019), p. 404.

[215] Ran Zhou, Wei Ma, and et al. “U-Net based automatic carotid plaque segmentation from 3D ultrasound images”. In: *Medical Imaging 2019: Computer-Aided Diagnosis*. Vol. 10950. International Society for Optics and Photonics. 2019, 109504F.

[216] Pádraig Looney et al. "Automatic 3D ultrasound segmentation of the first trimester placenta using deep learning". In: *ISBI 2017*. IEEE. 2017, pp. 279–282.

[217] Ziyang Wang. "Deep learning in medical ultrasound image segmentation: a review". In: *arXiv preprint arXiv:2002.07703* (2020).

[218] Michela Antonelli et al. "The medical segmentation decathlon". In: *Nature communications* 13.1 (2022), p. 4128.

[219] Jianhua Yao, Joseph E Burns, and etc. "Detection of vertebral body fractures based on cortical shell unwrapping". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2012, pp. 509–516.

[220] J Hofmanninger. "Prayer F Pan J Röhrich S Prosch H Langs G Automatic lung segmentation in routine imaging is primarily a data diversity problem, not a methodology problem". In: *Eur. Radiol. Exp* 4.1 (2020), p. 1.

[221] Kaggle. *Ultrasound Nerve Segmentation.* `https://www.kaggle.com/c/ultrasound-nerve-segmentation`.

[222] Ziyang Wang and Irina Voiculescu. "Scribble-Supervised Collaborative Hybrid Networks for MRI Cardiac Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention- MICCAI 2023*. Springer. 2023.

[223] Varduhi Yeghiazaryan and Irina Voiculescu. "Family of boundary overlap metrics for the evaluation of medical image segmentation". In: *Journal of Medical Imaging* 5.1 (2018), pp. 015006–015006.

[224] Zhen-Liang Ni et al. "Raunet: Residual attention u-net for semantic segmentation of cataract surgical instruments". In: *International Conference on Neural Information Processing*. Springer. 2019, pp. 139–149.

[225] Steven Guan et al. "Fully Dense UNet for 2-D Sparse Photoacoustic Tomography Artifact Removal". In: *IEEE journal of biomedical and health informatics* 24.2 (2019), pp. 568–576.

[226] Abien Fred Agarap. "Deep learning using rectified linear units (relu)". In: *arXiv preprint arXiv:1803.08375* (2018).

[227] Jiawei Zhang et al. "Mdu-net: Multi-scale densely connected u-net for biomedical image segmentation". In: *arXiv preprint arXiv:1812.00352* (2018).

[228] Xiao-Yun Zhou et al. "Acnn: a full resolution dcnn for medical image segmentation". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 8455–8461.

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

[230] Yassine Ouali, Céline Hudelot, and Myriam Tami. "Semi-supervised semantic segmentation with cross-consistency training". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.

[231] Liang-Chieh Chen et al. "Naive-student: Leveraging semi-supervised learning in video sequences for urban scene segmentation". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*. Springer. 2020, pp. 695–714.

[232] Geoff French et al. “Semi-supervised semantic segmentation needs strong, varied perturbations”. In: *arXiv preprint arXiv:1906.01916* (2019).

[233] David Berthelot et al. “Mixmatch: A holistic approach to semi-supervised learning”. In: *Advances in neural information processing systems* 32 (2019).

[234] Vikas Verma and etc. “Interpolation consistency training for semi-supervised learning”. In: *International Joint Conference on Artificial Intelligence*. 2019.

[235] Kihyuk Sohn et al. “Fixmatch: Simplifying semi-supervised learning with consistency and confidence”. In: *Advances in neural information processing systems* 33 (2020), pp. 596–608.

[236] Zhanghan Ke and etc. “Dual student: Breaking the limits of the teacher in semi-supervised learning”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 6728–6736.

[237] Antti Tarvainen and Harri Valpola. “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results”. In: *Advances in neural information processing systems* 30 (2017).

[238] W Dong-DongChen and ZH WeiGao. “Tri-net for semi-supervised deep learning”. In: *Proceedings of twenty-seventh international joint conference on artificial intelligence*. 2018, pp. 2014–2020.

[239] Xiaokang Chen et al. “Semi-supervised semantic segmentation with cross pseudo supervision”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 2613–2622.

[240] Xiangde Luo et al. “Scribble-supervised medical image segmentation via dual-branch network and dynamically mixed pseudo labels supervision”. In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part I*. Springer. 2022, pp. 528–538.

[241] Lequan Yu et al. “Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation”. In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*. Springer. 2019, pp. 605–613.

[242] Samuli Laine and Timo Aila. “Temporal ensembling for semi-supervised learning”. In: *arXiv preprint arXiv:1610.02242* (2016).

[243] Wenhui Cui et al. “Semi-supervised brain lesion segmentation with an adapted mean teacher model”. In: *Information Processing in Medical Imaging: 26th International Conference, IPMI 2019, Hong Kong, China, June 2–7, 2019, Proceedings 26*. Springer. 2019, pp. 554–565.

[244] Kaiping Wang et al. “Semi-supervised medical image segmentation via a tripled-uncertainty guided mean teacher model with contrastive learning”. In: *Medical Image Analysis* 79 (2022), p. 102447.

[245] Nasim Souly, Concetto Spampinato, and Mubarak Shah. “Semi supervised semantic segmentation using generative adversarial network”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 5688–5696.

[246] Takeru Miyato et al. “Virtual adversarial training: a regularization method for supervised and semi-supervised learning”. In: *IEEE transactions on pattern analysis and machine intelligence* 41.8 (2018), pp. 1979–1993.

[247] Wei-Chih Hung et al. “Adversarial learning for semi-supervised semantic segmentation”. In: *arXiv preprint arXiv:1802.07934* (2018).

[248] Ziyang Wang and Irina Voiculescu. “Exigent Examiner and Mean Teacher: An Advanced 3D CNN-Based Semi-Supervised Brain Tumor Segmentation Framework”. In: *Workshop on Medical Image Learning with Limited and Noisy Data*. Springer. 2023, pp. 181–190.

[249] Yizhe Zhang and etc. “Deep adversarial networks for biomedical image segmentation utilizing unannotated images”. In: *International conference on medical image computing and computer-assisted intervention*. Springer. 2017.

[250] Tuan-Hung Vu et al. “Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 2517–2526.

[251] Siyuan Qiao et al. “Deep co-training for semi-supervised image recognition”. In: *Proceedings of the european conference on computer vision (eccv)*. 2018, pp. 135–152.

[252] Adam Paszke and etc. “Enet: A deep neural network architecture for real-time semantic segmentation”. In: *arXiv preprint arXiv:1606.02147* (2016).

[253] Gerda Bortsova et al. “Semi-supervised medical image segmentation via learning consistency under transformations”. In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part VI 22*. Springer. 2019, pp. 810–818.

[254] Hoel Kervadec et al. “Bounding boxes for weakly supervised segmentation: Global constraints get close to full supervision”. In: *Medical imaging with deep learning*. PMLR. 2020, pp. 365–381.

[255] Ryutaro Tanno et al. “Learning from noisy labels by regularized estimation of annotator confusion”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 11244–11253.

[256] Aritra Ghosh, Himanshu Kumar, and P Shanti Sastry. “Robust loss functions under label noise for deep neural networks”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 31. 1. 2017.

[257] Le Zhang et al. “Disentangling human error from ground truth in segmentation of medical images”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 15750–15762.

[258] Mou-Cheng Xu et al. “Learning to pay attention to mistakes”. In: *arXiv preprint arXiv:2007.15131* (2020).

[259] Yisen Wang et al. “Symmetric cross entropy for robust learning with noisy labels”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 322–330.

[260] Zhilu Zhang and Mert Sabuncu. “Generalized cross entropy loss for training deep neural networks with noisy labels”. In: *Advances in neural information processing systems* 31 (2018).

[261] Tongliang Liu and Dacheng Tao. "Classification with noisy labels by importance reweighting". In: *IEEE Transactions on pattern analysis and machine intelligence* 38.3 (2015), pp. 447–461.

[262] Aditya Menon et al. "Learning from corrupted binary labels via class-probability estimation". In: *International conference on machine learning.* PMLR. 2015, pp. 125–134.

[263] Giorgio Patrini et al. "Making deep neural networks robust to label noise: A loss correction approach". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 1944–1952.

[264] Bo Han et al. "Co-teaching: Robust training of deep neural networks with extremely noisy labels". In: *Advances in neural information processing systems* 31 (2018).

[265] Lu Jiang et al. "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels". In: *International conference on machine learning.* PMLR. 2018, pp. 2304–2313.

[266] Junnan Li, Richard Socher, and Steven CH Hoi. "DivideMix: Learning with Noisy Labels as Semi-supervised Learning". In: *International Conference on Learning Representations.* 2019.

[267] Ragav Sachdeva et al. "ScanMix: learning from severe label noise via semantic clustering and semi-supervised learning". In: *Pattern recognition* 134 (2023), p. 109121.

[268] Simon K Warfield, Kelly H Zou, and William M Wells. "Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation". In: *IEEE transactions on medical imaging* 23.7 (2004), pp. 903–921.

[269] Yicheng Wu et al. "Coactseg: Learning from heterogeneous data for new multiple sclerosis lesion segmentation". In: *International conference on medical image computing and computer-assisted intervention.* Springer. 2023, pp. 3–13.

[270] Yicheng Wu et al. "Diversified and Personalized Multi-rater Medical Image Segmentation". In: *arXiv preprint arXiv:2403.13417* (2024).

[271] Ze Liu et al. "Swin transformer v2: Scaling up capacity and resolution". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2022, pp. 12009–12019.

[272] Simon Reiß et al. "Every annotation counts: Multi-label deep supervision for medical image segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2021, pp. 9532–9542.

[273] Di Lin et al. "Scribblesup: Scribble-supervised convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2016, pp. 3159–3167.

[274] Ke Zhang and Xiahai Zhuang. "Cyclemix: A holistic strategy for medical image segmentation from scribble supervision". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2022, pp. 11656–11665.

[275] Yigit B Can et al. “Learning to segment medical images with scribble-supervision alone”. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*. Springer. 2018, pp. 236–244.

[276] Gabriele Valvano, Andrea Leo, and Sotirios A Tsaftaris. “Learning to segment from scribbles using multi-scale adversarial attention gates”. In: *IEEE Transactions on Medical Imaging* 40.8 (2021), pp. 1990–2001.

[277] Ziyang Wang and Irina Voiculescu. “Weakly Supervised Medical Image Segmentation Through Dense Combinations of Dense Pseudo-Labels”. In: *MICCAI Workshop on Data Engineering in Medical Imaging*. Springer. 2023, pp. 1–10.

[278] Boah Kim and Jong Chul Ye. “Mumford–Shah loss functional for image segmentation with deep learning”. In: *IEEE Transactions on Image Processing* 29 (2019), pp. 1856–1866.

[279] Yves Grandvalet and Yoshua Bengio. “Semi-supervised learning by entropy minimization”. In: *Advances in neural information processing systems* 17 (2004).

[280] Anton Obukhov et al. “Gated CRF loss for weakly supervised semantic image segmentation”. In: *arXiv preprint arXiv:1906.04651* (2019).

[281] Aneesh Rangnekar, Christopher Kanan, and Matthew Hoffman. “Semantic segmentation with active semi-supervised learning”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023, pp. 5966–5977.

[282] Yanchao Li et al. “Ascent: Active supervision for semi-supervised learning”. In: *IEEE Transactions on Knowledge and Data Engineering* 32.5 (2019), pp. 868–882.

[283] Mingfei Gao et al. “Consistency-based semi-supervised active learning: Towards minimizing labeling cost”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*. Springer. 2020, pp. 510–526.

[284] Shasvat Desai and Debasmita Ghose. “Active learning for improved semi-supervised semantic segmentation in satellite images”. In: *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2022, pp. 553–563.