University of Texas at Arlington

MavMatrix

Computer Science and Engineering Dissertations

Computer Science and Engineering Department

2023

Towards Nuclei Segmentation with Limited Annotations

Mohammad Minhazul Haq

Follow this and additional works at: https://mavmatrix.uta.edu/cse_dissertations

Part of the Computer Sciences Commons

Recommended Citation

Haq, Mohammad Minhazul, "Towards Nuclei Segmentation with Limited Annotations" (2023). *Computer Science and Engineering Dissertations*. 382.
https://mavmatrix.uta.edu/cse_dissertations/382

This Dissertation is brought to you for free and open access by the Computer Science and Engineering Department at MavMatrix. It has been accepted for inclusion in Computer Science and Engineering Dissertations by an authorized administrator of MavMatrix. For more information, please contact leah.mccurdy@uta.edu, erica.rousseau@uta.edu, vanessa.garrett@uta.edu.

TOWARDS NUCLEI SEGMENTATION WITH LIMITED ANNOTATIONS

by

MOHAMMAD MINHAZUL HAQ

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

August 2023

Copyright © by Mohammad Minhazul Haq 2023

All Rights Reserved

To my mother, my father, and my wife.

ACKNOWLEDGEMENTS

I would like to thank my supervising Professor Dr. Junzhou Huang for his invaluable advice and guidance during my doctoral studies, and also for constantly motivating and encouraging me. None of the work in this thesis would be possible without him. I would also like to thank my Ph.D. committee members Dr. Jean Gao, Dr. Jia Rao, and Dr. Dajiang Zhu for their interest in my research and for their valuable insights regarding this thesis. I want to thank all my colleagues from the SMILE Lab for making my time at UTA enjoyable and productive. It is my pleasure to meet such a concentration of brilliant, helpful and kind people in the lab.

Finally, I am deeply grateful to my mother, my father, and my wife for their continuous and unconditional support, sacrifice and encouragement throughout my Ph.D journey. I am also thankful to each of my family members, friends, and colleagues in Bangladesh and USA.

August 24, 2023

ABSTRACT

TOWARDS NUCLEI SEGMENTATION WITH LIMITED ANNOTATIONS

Mohammad Minhazul Haq, Ph.D.

The University of Texas at Arlington, 2023

Supervising Professor: Dr. Junzhou Huang

Nuclei segmentation is a fundamental but challenging task in histopathology image analysis. For semantic segmentation of nuclei, Convolutional Neural Network (CNN), and Vision Transformer (VT) models give very promising results. However, to successfully train fully-supervised CNN and VT models we need significant amount of annotated data which is highly rare in biomedical domain. Also, collecting an unannotated histopathology dataset first, and then manually doing pixel-level labeling is expensive, time-consuming and tedious process. Therefore, we require to discover a way for training nuclei segmentation models with unlabeled datasets. In this thesis, I present my work towards solving this critical problem by utilizing Adversarial Learning, Self-Supervised Learning (SSL), and Diffusion Models. Thus, my approaches can be summarized into three directions: 1) employing adversarial learning based unsupervised and semi-supervised domain adaptation techniques to solve nuclei segmentation problem for unannotated datasets; 2) proposing SSL based approaches for pre-training VT models with unannotated image dataset; 3) introducing Denoising Diffusion Probabilistic Model (DDPM) based approach for pre-training nuclei segmentation model with large-scale histology image dataset. In the first approach, I

apply Unsupervised Domain Adaptation (UDA) and Semi-Supervised Domain Adaptation (SSDA) with the help of another labeled dataset that may come from another organs or sources. Later, I extend the model by utilizing an adversarial learning incorporated reconstruction network to translate the source-domain images to the target domain for further training. Then, in my second approach, I introduce a novel region-level SSL based framework for pre-training semantic nuclei segmentation model with a large-scale unannotated histopathology image dataset extracted from Whole Slide Images (WSI). Additionally, I propose hierarchical, scale, and transformation equivariance loss to reduce the disagreements among predictions. Finally, in the third approach, I utilize DDPM for extracting discriminative and powerful features. Then, I combine a generation module, a discriminator, and scale loss with DDPM for effective label-efficient SSL based pre-training. Extensive and comprehensive experiments demonstrate the superiority of the proposed methods over the baseline models.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iv

ABSTRACT v

LIST OF ILLUSTRATIONS x

LIST OF TABLES xii

Chapter Page

1. INTRODUCTION 1

1.1 Introduction to Nuclei Segmentation 1

1.2 Challenges and Proposed approaches 1

1.3 Dataset 3

1.3.1 Pre-training dataset 3

1.3.2 Fine-tuning datasets 4

1.4 Dissertation Structure 4

2. ADVERSARIAL DOMAIN ADAPTATION FOR CELL SEGMENTATION 7

2.1 Introduction 7

2.2 Related Work 10

2.3 Methodology 10

2.3.1 CellSegUDA 11

2.3.2 CellSegSSDA 14

2.3.3 Implementations 14

2.4 Experiments 15

2.4.1 Dataset 15

2.4.2 Experimental results 15

2.5 Conclusion 19
3. NuSegDA: DOMAIN ADAPTATION FOR NUCLEI SEGMENTATION . 20
3.1 Introduction 21
3.2 Related Work 25
3.3 Methodology 27
3.3.1 Problem Definition 27
3.3.2 Unsupervised Domain Adaptation 28
3.3.3 Semi-Supervised Domain Adaptation 35
3.4 Experiments 36
3.4.1 Dataset 36
3.4.2 Implementations 36
3.4.3 Experimental results 37
3.5 Conclusion 44
4. SELF-SUPERVISED PRE-TRAINING FOR NUCLEI SEGMENTATION 45
4.1 Introduction 45
4.2 Related Work 48
4.3 Methodology 48
4.3.1 Self-Supervised Pre-Training with unannotated dataset 49
4.3.2 Fine-Tuning with annotated dataset 54
4.3.3 Implementations 54
4.4 Experiments 54
4.4.1 Dataset 54
4.4.2 Experimental results 54
4.5 Conclusion 57
5. TranSSCon: CONSISTENT SELF-SUPERVISED PRE-TRAINING FOR NUCLEI SEGMENTATION 59

5.1 Introduction 60
5.2 Related Work 63
5.3 Methodology 64
5.3.1 Pre-Training with large-scale unannotated dataset 65
5.3.2 Fine-Tuning with annotated dataset 74
5.3.3 Implementations 75
5.4 Experiments 75
5.4.1 Dataset 75
5.4.2 Experimental results 75
5.5 Conclusion 78
6. DiffNuSS: DIFFUSION MODEL BASED SELF-SUPERVISED PRE-TRAINING FOR NUCLEI SEGMENTATION 79
6.1 Introduction 80
6.2 Related Work 83
6.3 Methodology 84
6.3.1 Pre-Training with large-scale unannotated dataset 85
6.3.2 Fine-Tuning with annotated dataset 90
6.3.3 Implementations 90
6.4 Experiments 90
6.4.1 Dataset 90
6.4.2 Experimental results 91
6.5 Conclusion 93
7. CONCLUSIONS 95
Bibliography 97
BIOGRAPHICAL STATEMENT 115

LIST OF ILLUSTRATIONS

Figure Page

1.1 The output of semantic nuclei segmentation 2

2.1 Observation and motivations for CellSegDA 8

2.2 Complete architecture of CellSegUDA 11

2.3 Visualization of segmentation for KIRC→TNBC 17

2.4 Visualization of segmentation for TNBC→KIRC 18

3.1 The semantic segmentation of nuclei 21

3.2 Observation and motivation for NuSegUDA 23

3.3 Complete architecture of NuSegUDA 29

3.4 Visualization of the target-translated source domain images 31

3.5 Visualizations of Unsupervised Domain Adaptation 38

3.6 Visualizations of Semi-Supervised Domain Adaptation 40

3.7 Visualizations of the effectiveness of proposed losses on NuSegUDA . 42

4.1 Observation and motivation for TransNuSS 47

4.2 Complete architecture of TransNuSS 50

4.3 Visualization of the nuclei segmentation outputs 57

4.4 Visualizations of $HardN'$ matrices 57

5.1 Observation and motivation for TranSSCon 62

5.2 Complete architecture of TranSSCon. 65

5.3 Visualization of the nuclei segmentation outputs 77

6.1 Denoising Diffusion Probabilistic Model (DDPM) 81

6.2 Complete architecture of DiffNuSS. 85

6.3 Visualization of the nuclei segmentation outputs 93

LIST OF TABLES

Table Page

2.1 Segmentation results of CellSegUDA and CellSegSSDA 16

3.1 Unsupervised Domain Adaptation results of NuSegUDA 37

3.2 Semi-Supervised Domain Adaptation results of NuSegSSDA 39

3.3 Impacts of different losses on NuSegUDA 41

3.4 Impacts of different segmentation network backbones in NuSegUDA . 43

4.1 Nuclei segmentation results of TransNuSS 56

5.1 Nuclei segmentation results for Experiment-1 and Experiment-2. . . . 76

6.1 Nuclei segmentation results for Experiment-1 and Experiment-2. . . . 92

CHAPTER 1

INTRODUCTION

1.1 Introduction to Nuclei Segmentation

Nuclei are the fundamental organizational unit of life [71]. The accurate segmentation of nuclei is crucial for cancer diagnosis and further clinical treatments. Because of that, nuclei segmentation is considered as an essential task of digital histopathology image analysis [99, 29]. However, accurate nuclei segmentation is quite challenging due to the significant variations in the shape and appearance of nuclei, clustered and overlapped nuclei, blurred nuclei boundaries, inconsistent staining methods, scanning artifacts, etc. Also, histopathology of different organs or cancer types may exhibit different textures, color distributions, morphology and scales [95, 56].

1.2 Challenges and Proposed approaches

In semantic segmentation of nuclei, we want to segment the nuclei from its background (see Figure 1.1). For semantic nuclei segmentation, Convolutional Neural Network (CNN) based approaches give very promising results [54, 66, 107, 28]. Alternatively, Vision Transformers (VT) have the potentiality to outperform CNN based models due to their ability to model long-range dependencies (i.e., global context) [104]. However, to successfully train fully-supervised CNN models, we need at least a few amount of annotated data (i.e., images with their corresponding pixel-level ground-truth labels) [46, 101]. Furthermore, VT needs lot of data for training, usually more than what is necessary to standard CNNs [52]. Unfortunately, such well-annotated datasets, even if very small-sized, are highly rare in biomedical domain.

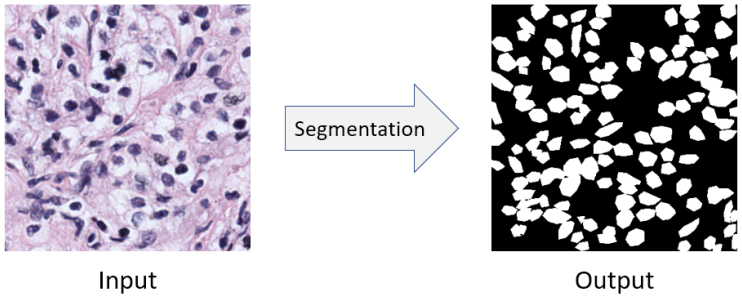

Figure 1.1. The output of semantic nuclei segmentation.

Also, collecting an unannotated dataset first, and then doing the manual labeling with the help of experts is also an expensive, time-consuming and tedious process [94, 13, 99]. For example, annotating even a small nuclei segmentation dataset consisting of 50 image paches takes 120-130 hours of an expert pathologist's time [34]. Therefore, we require to discover a way for training the nuclei segmentation network with unlabeled dataset.

To solve the aforementioned problem, we may think of applying conventional solutions like Transfer Learning, Pre-training with generic image dataset, etc. However, simply applying Transfer Learning (i.e., models trained with one organ or cancer type, and then evaluated with different organ or cancer types) unfortunately leads to poor performance due to the domain shift problem [71]. This domain shift problem happens due to different scanners, scanning protocols, tissue types, etc. [71]. As an alternative solution, CNN and VT models can be pre-trained with large-scale natural image dataset (i.e., ImageNet [19]) in fully-supervised manner, and then fine-tuned to downstream tasks [23]. However, pre-training nuclei segmentation models with ImageNet is not much helpful because of morphological and textural differences between natural image domain and medical image domain. Also, ImageNet-like large-scale annotated histology dataset rarely exists in medical image domain.

In this thesis, I tackle these obstacle by proposing: 1) adversarial learning based Unsupervised Domain Adaptation (UDA) and Semi-Supervised Domain Adaptation (SSDA) approaches to solve nuclei segmentation problem for unannotated datasets; 2) reconstruction network incorporated feature-space and output-space domain adaptation model so that the source domain images can be translated to the target domain for further training of the nuclei segmentation network; 3) Self-Supervised Learning (SSL) based region-level triplet learning for pre-training VT models with unannotated histology image dataset; 4) disagreement loss (i.e., hierarchical, scale, and transformation equivariance loss) incorporated SSL based pre-training framework for nuclei segmentation; and 5) Denoising Diffusion Probabilistic Model (DDPM) based approach for pre-training nuclei segmentation model with large-scale histology image dataset.

1.3 Dataset

In this thesis, I use a large-scale unannotated dataset for pre-training purposes, and four annotated datasets for fine-tuning the model. We discuss the details of these pre-training and fine-tuning datasets in the following.

1.3.1 Pre-training dataset

MoNuSegWSI MoNuSeg [45, 47] training dataset contains thirty 1000×1000 annotated image patches extracted from thirty Whole Slide Images (WSI) of different patients collected from The Cancer Genomic Atlas (TCGA). Similar to train-split of AttnSSL [68], we select 19 patients, and download corresponding 19 H&E stained WSIs from which we extract patches of size 512×512 at 40x magnification. Following AttnSSL [68], we perform a simple thresholding in HSV color space for each extracted patch to determine whether the patch contains tissue or not. Patches with

less than 70% tissue cover are not used. Thus, a total of 178217 patches are selected for pre-training. In our experiments, we denote this unannotated pre-training dataset as MoNuSegWSI.

1.3.2 Fine-tuning datasets

Dataset-1 (TNBC) The images of TNBC dataset [58] are collected at 40x magnification. This dataset consists of 50 H&E stained histology images of size 512×512. Labeling of this dataset is performed by expert pathologist and research fellows. In our experiments, we randomly split TNBC into 80% for training, 10% for validation, and 10% for testing.

Dataset-2 (MoNuSeg) We split thirty 1000×1000 annotated images of MoNuSeg [45, 47] training data into 80% for training, and 20% for validation. MoNuSeg-test consists of 14 images of MoNuSeg testing data. We refer this dataset as MoNuSeg in our experiments.

Dataset-3 (KIRC) The images of this dataset are extracted at 40x magnification from Whole Slide Images (WSI) of Kidney Renal Clear cell carcinoma (KIRC). We take this dataset from [37]. This dataset, referred as KIRC, has of 486 H&E stained histology images of 400×400 pixel size. The ground-truth labels are annotated by expert pathologists and research fellows. We follow the same data splitting as TNBC for this dataset.

1.4 Dissertation Structure

In this thesis, I will present how each of the proposed approaches solve nuclei segmentation problem with limited annotations as follows:

Chapter 2 presents a network named CellSegUDA for nuclei segmentation on the unlabelled dataset (target domain). High unavailability of annotated nuclei seg-

mentation dataset, and tedious labeling process enforce us to discover a way (i.e., CellSegUDA) for training with unlabeled dataset. In CellSegUDA, we apply Unsupervised Domain Adaptation (UDA) technique with the help of another labeled dataset (source domain) that may come from other organs or sources. Then, considering the scenario when we have a small number of annotations available from the target domain, we extend our work to CellSegSSDA, a Semi-Supervised Domain Adaptation (SSDA) based approach. Extensive and comprehensive experiments on two public nuclei segmentation datasets demonstrate the superiority of our proposed CellSegUDA and CellSegSSDA models.

Chapter 3 introduces another nuclei segmentation model, NuSegUDA, in which we apply UDA technique at both of feature space and output space. We additionally utilize a reconstruction network and incorporate adversarial learning into it so that the source-domain images can be accurately translated to the target-domain for further training of the segmentation network. Then, assuming we have a few annotation available from target domain, we extend our work to SSDA. We validate our proposed UDA and SSDA frameworks on two public nuclei segmentation datasets, and obtain significant improvement as compared with the baseline models.

Chapter 4 presents a novel region-level Self-Supervised Learning (SSL) approach and corresponding triplet loss for pre-training semantic nuclei segmentation model using a large-scale unannotated histology image dataset extracted from Whole Slide Images (WSI). Due to this triplet loss, our pre-trained SSL model learns to separate nuclei features from the background features in the embedding space. Additionally, our SSL approach involves the image-level sub-task of predicting the scale of image, which enables the segmentation network to implicitly acquire further knowledge of nuclei size and shape. In the end we empirically demonstrate the superiority of

our proposed SSL incorporated Vision Transformer (VT) model, TransNuSS, on two public nuclei segmentation datasets.

Chapter 5 introduces region-level, image-level and clustering-based SSL approach for pre-training semantic nuclei segmentation model with unannotated histology images extracted from WSIs. Unfortunately, due to the lack of annotations, SSL alone can not guarantee the consistency of the model while pre-training. To reduce disagreements among the predictions, we propose hierarchical, scale and transformation equivariance consistency losses. Thus, we introduce a simple yet effective combination of SSL approaches and consistency losses for pre-training semantic nuclei segmentation model. We empirically demonstrate the superiority of our proposed consistency-preserving SSL incorporated VT model on two public nuclei segmentation datasets.

Chapter 6 presents Denoising Diffusion Probabilistic Model (DDPM) based SSL approach for pre-training semantic nuclei segmentation model with unannotated histology images extracted from WSIs. We feed-forward the DDPM outputs (i.e., estimated noise) to a generation module for predicting the segmentation mask. Since DDPM are capable of extracting powerful and discriminative features via generative pre-training for dense prediction tasks, we combine SSL with DDPM. To pre-train the model for generating realistic segmentation masks and acquiring knowledge of nuclei, we employ a discriminator and scale loss, respectively. Thus, we introduce a simple yet effective combination of DDPM, generation module, discriminator, and scale loss for label-efficient pre-training of semantic nuclei segmentation model. We empirically demonstrate the superiority of our proposed VT incorporated DDPM based SSL approach on two public nuclei segmentation datasets.

Finally, Chapter 7 provides a summary of this research, and concludes the dissertation.

CHAPTER 2

ADVERSARIAL DOMAIN ADAPTATION FOR CELL SEGMENTATION

To successfully train a cell segmentation network in fully-supervised manner for a particular type of organ or cancer, we need the dataset with ground-truth annotations. However, high unavailability of such annotated dataset and tedious labeling process enforce us to discover a way for training with unlabeled dataset. In this chapter, we introduce a network named CellSegUDA for cell/nuclei segmentation on the unlabeled dataset (target domain). It is achieved by applying Unsupervised Domain Adaptation (UDA) technique with the help of another labeled dataset (source domain) that may come from other organs or sources. We validate our proposed CellSegUDA on two public cell segmentation datasets and obtain significant improvement as compared with the baseline methods. Finally, considering the scenario when we have a small number of annotations available from the target domain, we extend our work to CellSegSSDA, a Semi-Supervised Domain Adaptation (SSDA) based approach. Our SSDA model also gives excellent results which are quite close to the fully-supervised upper bound in target domain.

2.1 Introduction

Convolutional Neural Network (CNN) based approaches like Fully Convolutional Network (FCN) [53], U-Net [67], UNet++ [106] give very promising results in biomedical image segmentation tasks as well as in cell/nuclei segmentation problems [73]. However, to successfully train these fully-supervised methods, we need at least a few amount of annotated data (i.e., images with their corresponding pixel-level

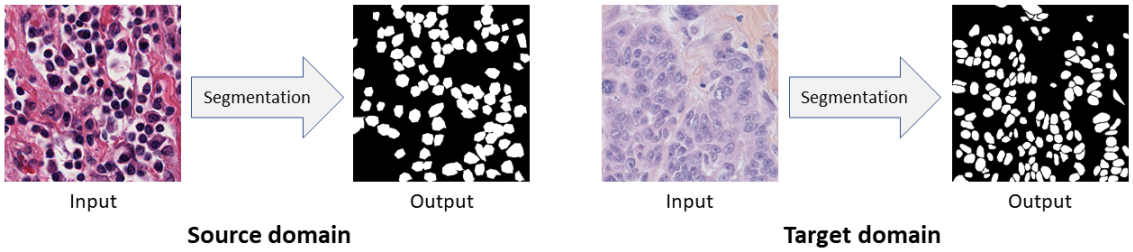

Figure 2.1. Images from different domains look dissimilar while their pixel-level segmentation outputs are similar. In this figure, source domain and target domain images come from Kidney Renal Clear cell carcinoma (KIRC) and Triple Negative Breast Cancer (TNBC) respectively.

ground-truth labels) [46, 101]. Unfortunately, such well-annotated datasets, even if very small-sized, are highly rare in biomedical domain. Also, collecting an unannotated dataset first, and then doing the manual labeling with the help of experts is also an expensive, time-consuming and tedious process [94, 13]. How if we could train a deep CNN model for nuclei segmentation without any further needs for the annotations? Domain Adaptation, a subclass of Transfer Learning, provides solution in such scenarios.

Here, we consider the unannotated dataset (i.e., for which we want to predict the labels) as target domain. Then, with the help of another related but different annotated dataset, referred as source domain, we apply adversarial learning [25] based domain adaptation technique for nuclei segmentation problem. Thus, our proposed framework, learns from labeled source domain and adapts to the unlabeled target domain. We very carefully observed that, images from different nuclei datasets, even if collected from different organs or cancer types, exhibit dissimilarity although their corresponding segmentation ground-truth labels are quite similar (see Figure 2.1). In summary, ground-truth labels for nuclei segmentation are domain-invariant.

In this work, we first propose a unsupervised domain adaptation model for nuclei segmentation. Because of our aforementioned observation, we apply our domain adaptation in the output space rather than in the feature space. With the help of adversarial learning, we train a robust biomedical image segmentation network to generate source-domain look-alike outputs for target images. Additionally, we use a decoder network to make target images and target predictions correlated to each other as much as possible. Finally, we extend our Unsupervised Domain Adaptation (UDA) technique to Semi-Supervised Domain Adaptation (SSDA) considering that we have some annotations available from the target domain.

Conducting extensive experiments on two nuclei segmentation datasets we conclude that, our proposed UDA method, CellSegUDA, outperforms both of a fully-supervised model [67] trained on source domain and evaluated on target domain, and a baseline UDA model [22]. Experimental result (see Section 2.4) also shows that, accuracy of our SSDA strategy appears very close to the upper bound of fully-supervised model trained in target domain.

Thus, the main contributions of this paper are: **1)** We propose an adversarial learning based Unsupervised Domain Adaptation (UDA) approach to solve nuclei segmentation problem for unannotated datasets. **2)** Our proposed method is simple as it does not depend on any data synthesization or data augmentation. **3)** Our proposed UDA framework can be easily extended to Semi-Supervised Domain Adaptation (SSDA) in the scenario where a small portion of the target domain is labeled. **4)** Extensive and comprehensive experiments on two datasets have demonstrated the superiority of the proposed methods.

2.2 Related Work

A multi-level adversarial network based domain adaptation approach for semantic segmentation was proposed by Tsai et al. [79]. Hoffman et al. [33] proposed an unsupervised domain adaptation model utilizing both of pixel-level and feature-level adaptation. Isola et al. [38] applied conditional GAN [57] for image-to-image translation problems. Chen et al. [17] proposed a cross-domain consistency loss based pixel-wise adversarial domain adaptation algorithm. Zhang et al. [103] proposed a fully convolutional adaptation network for semantic segmentation.

For different types of biomedical image segmentation, several adversarial network based approaches also have been proposed. A multi-connected domain discriminator based UDA model for brain lesion segmentation was proposed by Kamnitsas et al. [41]. Dong et al. [22] introduced another UDA framework for cardiothoracic ratio estimation through chest organ segmentation. Mahmood et al. [55] proposed a cell segmentation approach in which a large dataset is generated using synthesization. Hou et al. [34] also synthesized annotated training data for histopathology image segmentation. Huo et al. [36] proposed an end-to-end CycleGAN [108] based whole abdomen MRI to CT image synthesis and CT splegonmegaly segmentation network.

2.3 Methodology

Formally, in our nuclei segmentation problem, we have nuclei histology patches as input X of size $H \times W \times 3$. Then, we want to predict the segmentation output $\hat{Y}$ of size $H \times W \times 1$. Depending on the domain, we may also have pixel-wise ground-truth label Y of size $H \times W \times 1$ which is basically a binary mask.

Then, in unsupervised domain adaptation problem, we have a source domain with N_s annotated images $\{(X_s, Y_s)\}$, and a target domain which has N_t unannotated

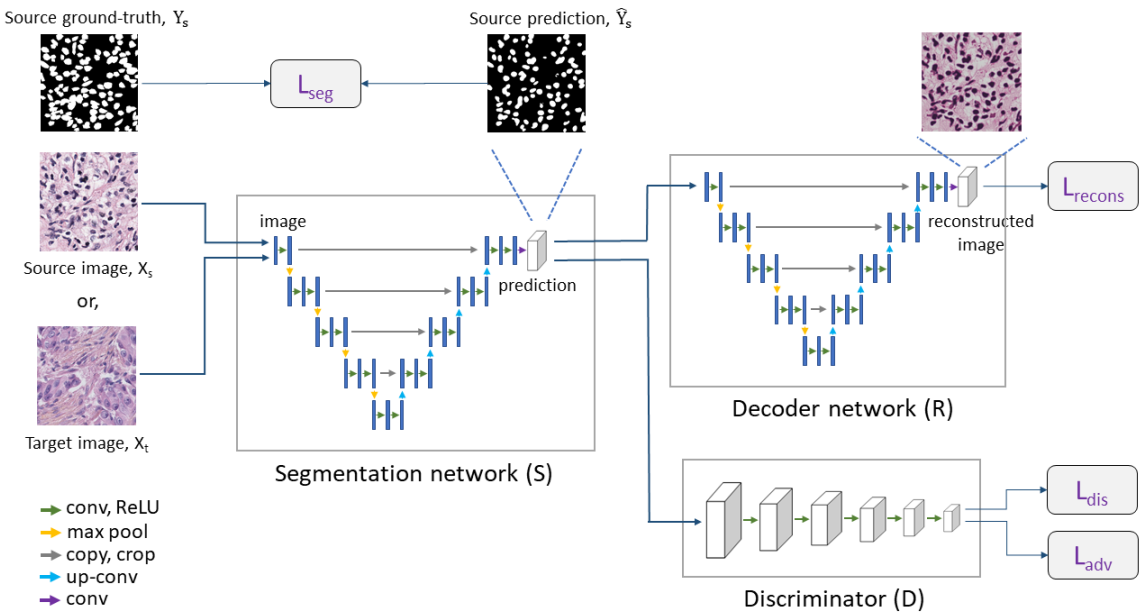

Figure 2.2. Complete architecture of CellSegUDA. Segmentation network generates segmentation outputs, from which decoder reconstructs input images. Discriminator distinguishes between source domain outputs and target domain outputs.

images $\{(X_t)\}$. In the case of semi-supervised domain adaptation problem, we assume that our target domain consists of N_t^l images with annotations $\{(X_t^l, Y_t)\}$, and N_t^u unannotated images $\{(X_t^u)\}$. Our ultimate goal is to learn a nuclei segmentation model that accurately produces the segmentation output in the target domain.

2.3.1 CellSegUDA

We refer our nuclei segmentation Unsupervised Domain Adaptation (UDA) model as CellSegUDA which is shown in Figure 2.2. CellSegUDA consists of three modules: Segmentation network (S), Decoder (R), and Discriminator (D).

Segmentation network (S) Our segmentation network S takes images X as input and produces the segmentation prediction $\hat{Y}$ of the same size as input, hence $\hat{Y} =$

$S(X)$. This segmentation network can be thought as the generator module of a GAN [25] framework.

We train S to generate the segmentation predictions $\hat{Y}_s$ similar to the ground-truth labels Y_s in source domain. We can not compute any pixel-level loss for target predictions since ground-truth labels are not available for target images in UDA. In practice, we found dice-coefficient loss to be more effective than binary cross-entropy loss for nuclei segmentation tasks. Therefore, we choose dice-coefficient loss as our segmentation loss:

$$L_{seg}(X_s) = 1 - \frac{2.Y_s'.\hat{Y}_s'}{Y_s' + \hat{Y}_s'} \tag{2.1}$$

where Y_s' and $\hat{Y}_s'$ are flatten Y_s and $\hat{Y}_s$ respectively.

Training S with only the annotated source data teaches S to make accurate predictions for source images. However, this segmentation network will generate incorrect outputs for target images as there are visual discrepancies between source images and target images. Because of our observation that cell segmentation outputs are domain-invariant, we require S to produce target domain predictions as much as close to the source domain predictions. In other words, we want to make the distribution of target predictions $\hat{Y}_t$ closer to source predictions $\hat{Y}_s$. Thus, we define adversarial loss as:

$$L_{adv}(X_t) = -\frac{1}{H' \times W'} \sum_{h',w'} \log\left(D(\hat{Y}_t)\right) \tag{2.2}$$

where $\hat{Y}_t = S(X_t)$, and H' and W' are height and width of discriminator output $D(\hat{Y}_t)$. This adverserial loss helps S to fool the discriminator so that it considers $\hat{Y}_t$ as source domain segmentation outputs.

Segmentation loss and adversarial loss altogether guides S to generate target domain predictions $\hat{Y}_t$ which look similar to source domain ground-truths. However, it is highly probable that these target predictions are not well-correlated with correspond-

ing target input images. The ability of reconstructing images from the predictions with similar visual appearance as input images will ensure that there is a correlation between the input image and segmentation output.

Decoder (R) To ensure that our target domain predictions spatially correspond to the target domain images, we use a decoder network R in CellSegUDA. In a similar way to [89], we consider our segmentation network S as an encoder. Then, decoder R reconstructs target images from the corresponding predictions. Thus, S and R altogether works as an autoencoder.

Using our decoder network R, we first reconstruct target input images X_t from $\hat{Y}_t$. Then, we calculate the reconstruction loss as:

$$L_{recons}(X_t) = \frac{1}{H \times W \times C} \sum_{h,w,c} \left(X_t - R(\hat{Y}_t)\right)^2 \tag{2.3}$$

where, $R(\hat{Y}_t))$ is the output of decoder for $\hat{Y}_t$, and C is the number of channels of input image X.

Thus, we minimize the following total loss while training our segmentation network:

$$L_s(X_s, X_t) = L_{seg}(X_s) + \lambda_{adv} L_{adv}(X_t) + \lambda_{recons} L_{recons}(X_t) \tag{2.4}$$

where, λ_{adv} and λ_{recons} are the weights to balance corresponding losses.

Discriminator (D) Since we want to generate similar predictions for both of source images and target images, we incorporate a discriminator D in CellSegUDA. This discriminator takes source domain prediction or target domain prediction as input,

and then distinguishes whether the input, i.e. prediction, comes from source domain or target domain. To train D, we use following cross-entropy loss:

$$L_{dis}(\hat{Y}) = -\frac{1}{H' \times W'} \sum_{h',w'} z.\log\left(D(\hat{Y})\right) + (1-z).\log\left(1 - D(\hat{Y})\right) \tag{2.5}$$

where z=0 when D takes target domain prediction as its input, and z=1 when input comes from source domain prediction.

2.3.2 CellSegSSDA

In Semi-Supervised Domain Adaptation (SSDA) problem, we must make sure the best usages of available target domain annotations Y_t while training our segmentation network S. In such scenarios, we extend our CellSegUDA framework to CellSegSSDA, a cell segmentation semi-supervised domain adaptation model.

In CellSegSSDA, for unannotated target images we do the same as CellSegUDA. However, when we encounter an annotated target data (X_t^l, Y_t) while training, we additionally compute the segmentation loss $L_{seg}(X_t^l)$ in the similar manner to Eq. (2.1). Then, while computing the total loss we incorporate $L_{seg}(X_t^l)$ so that the segmentation network learns to generate the predictions closer to target ground-truths. Therefore, Eq. (2.4) is now modified as below:

$$L_s(X_s, X_t^l) = L_{seg}(X_s) + L_{seg}(X_t^l) + \lambda_{adv} L_{adv}(X_t^l) + \lambda_{recons} L_{recons}(X_t^l) \tag{2.6}$$

2.3.3 Implementations

In our work, we use U-Net [67] as both of our segmentation network and decoder. We choose U-Net so that our proposed segmentation framework can be directly applied in other biomedical domains. We preferred U-Net over UNet++ [106] because of the less number of parameters. Following DCGAN [64], we designed our discriminator consisting of five convolutional layers. To train CellSegUDA and CellSegSSDA,

we followed the training strategy from GAN [25]. Adam optimizer [44] with learning rate 0.0001, 0.001 and 0.001 are used in segmentation network, discriminator and decoder respectively. We empirically choose 0.001 and 0.01 as λ_{adv} and λ_{recons} respectively. We do not use any data augmentation in our experiments.

2.4 Experiments

2.4.1 Dataset

In this paper, we use two datasets: 1) KIRC, and 2) TNBC. Although both datasets consist of H&E stained histopathology images, they are collected from two different organs and different institutions. KIRC images are collected from TCGA portal (image acquiring tools are unknown to us), whereas TNBC images were acquired at Curie Institute using Philips Ultra Fast Scanner 1.6RA. Organ difference, institutional difference, and using different imaging tools and protocols cause the visual difference among the images from these two datasets. See Figure 2.1, where TNBC image looks dimmer than KIRC image.

2.4.2 Experimental results

Experiment-1 (KIRC → TNBC) In our first experiment, we choose KIRC as source domain and TNBC as target domain, denoted by KIRC → TNBC. We start with our unsupervised domain adaptation (UDA) model CellSegUDA which gives much better accuracies than a UDA baseline DA-ADV [22]. We also choose a fully-supervised model U-Net [67] to get an idea how it performs when directly applying transfer learning (i.e., training with only KIRC and then test it on TNBC without any modifications) which is also considered as the lower-bound of experimental performance. This poor performance of transfer learning (see the first row of Table 2.1) happens because of the visual domain gap between source training images and target

	Experiment-1 KIRC → TNBC		**Experiment-2** TNBC → KIRC	
Method	**IoU%**	**Dice score**	**IoU%**	**Dice score**
U-Net (source-trained) [67]	52.66	0.6875	54.82	0.7056
DA-ADV [22]	54.93	0.7079	55.43	0.7107
CellSegUDA w/o recons	56.56	0.72	56.91	0.7224
CellSegUDA	59.02	0.7394	57.09	0.7242
U-Net (source 100% + target 10%)	60.74	0.7534	56.89	0.7194
CellSegSSDA (source 100% + target 10%)	60.96	0.7557	58.81	0.7377
U-Net (source 100% + target 25%)	61.67	0.7607	59.32	0.7405
CellSegSSDA (source 100% + target 25%)	62.94	0.771	59.73	0.7443
U-Net (source 100% + target 50%)	56.73	0.7208	59.95	0.7464
CellSegSSDA (source 100% + target 50%)	63.59	0.7748	60.32	0.7494
U-Net (source 100% + target 75%)	59.06	0.7394	61.63	0.7592
CellSegSSDA (source 100% + target 75%)	64.96	0.7862	61.01	0.7541
U-Net (target-trained)	66.57	0.7985	62.04	0.7621

Table 2.1. Segmentation results for Experiment-1 and Experiment-2. IoU denotes intersection over union. Here, unsupervised domain adaptation (UDA) baseline is denoted as DA-ADV. CellSegUDA w/o recons, CellSegUDA and CellSegSSDA refer to our proposed UDA model without reconstruction loss, proposed UDA with reconstruction loss, and proposed semi-supervised domain adaptation method respectively. CellSegSSDA(source 100% + target n%) denotes n% annotations available in TNBC-train and KIRC-train for experiment-1 and experiment-2 respectively. Results are from testing on TNBC-test and KIRC-test for experiment-1 and experiment-2 respectively.

test images, also known as domain shift problem. Figure 2.3(c) shows the visualization result of applying transfer learning in which many of the nuclei are missed out when comparing to the ground-truth. Then, training U-Net with TNBC-train and testing it on TNBC-test gives us the upper-bound (last row of Table 2.1). Table 2.1 shows that, CellSegUDA gives 6.36 higher IoU% than source-trained U-Net model. We see that, CellSegUDA also has 4.09 higher IoU% than UDA baseline DA-ADV. We check the effect of our decoder network R by training CellSegUDA without reconstruction loss, denoted as CellSegUDA w/o recons in Table 2.1. We find that, reconstruction

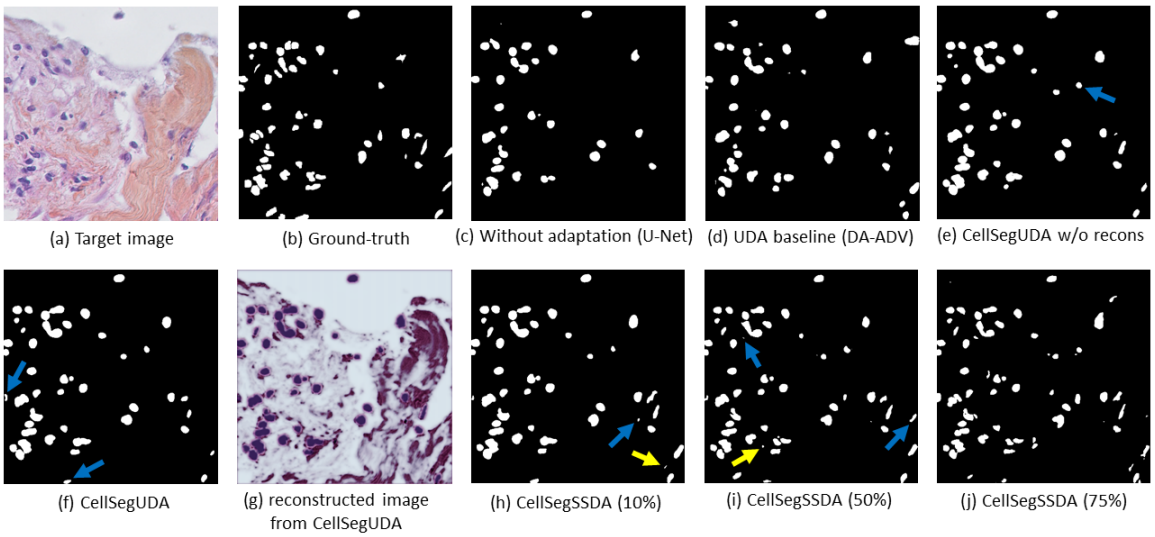

Figure 2.3. Visualization of segmentation for KIRC→TNBC. (g) shows that reconstructed target image (output from decoder) is quite similar to the input image which proves the efficacy of our proposed network. In (e)-(f) and (h)-(i), blue arrows indicate some missing nuclei of previous method. In (h) and (i), yellow arrows indicates false positives which are removed by following CellSegSSDA(50%) and CellSegSSDA(75%) respectively. Figure shows that, CellSegSSDA can identify more nuclei as the percentage of available annotations increases. This average-dense nuclei histopathology image in (a) is chosen so that the reader can easily find out the visual differences without further zooming-in.

loss really makes our segmentation network more accurate (see Figure 2.3(e)-(f) for visualization). Figure 2.3(g) also shows that we can reconstruct input images using our decoder from corresponding segmentation prediction, thus we believe that our prediction is well-correlated with its input.

Then, we assess our semi-supervised domain adaptation method CellSegSSDA for KIRC → TNBC. Source dataset, KIRC, is the same as UDA experiments. However, now we treat TNBC as partially labeled. We train CellSegSSDA considering 10%, 25%, 50% and 75% images from TNBC-train dataset has annotations available. Then, testing on TNBC-test gives us increasing IoUs and dice scores. This happens because more true positive nuclei can be identified and some false positive

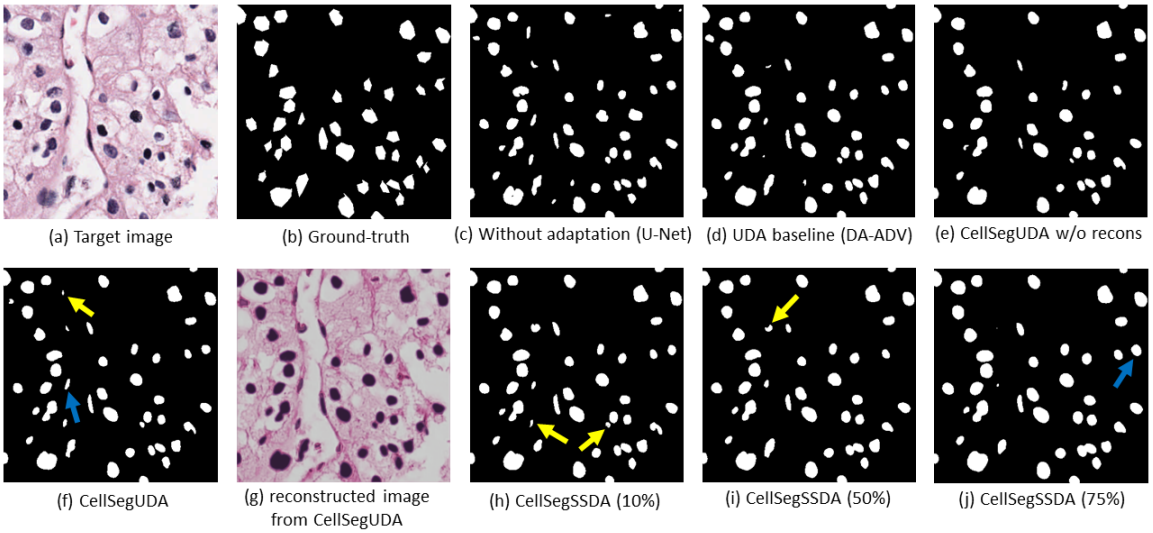

Figure 2.4. Visualization of segmentation for TNBC→KIRC. In (f) and (j), blue arrows indicate missing nuclei of previous method. In (f) and (h)-(i), yellow arrows indicate a false positive which is removed by following method. Similar to Figure 2.3, we chose this average-dense nuclei histopathology image for readability purposes.

cells can be removed by CellSegSSDA as we train it with more target annotations (see Figure 2.3(h)-(j)). We observe that, the accuracy of CellSegSSDA approaches to the upper-bound (only lower by 1.61 IoU%) as we train with more annotations from target domain. We also compare CellSegSSDA with fully-supervised model U-Net to demonstrate the superiority of our SSDA model. This time, to train U-Net, we combine full KIRC dataset with the same 10%, 25%, 50% and 75% of TNBC-train we chose to train CellSegSSDA. As domain adaptation helps to reduce the domain shift problem, we see that CellSegSSDA outperforms fully-supervised model in all of the cases.

Experiment-2 (TNBC → KIRC) We conduct another experiment in the similar way to Experiment-1 by selecting TNBC as source and KIRC as target domain. This experiment also reflects the excellence of CellSegUDA and CellSegSSDA com-

pared to other approaches in terms of segmentation accuracies (see last two columns of Table 2.1). Similar to experiment-1, we also see that segmentation accuracies of CellSegSSDA increase as more target images are annotated. Segmentation visualization from this experiment is shown in Figure 2.4. From this experiment, we once again observe that CellSegUDA performs better than CellSegUDA w/o recons which proves the validity of our decoder and the effectiveness of reconstruction loss (see reconstructed image in Figure 2.4(g)).

2.5 Conclusion

In this work, utilizing adversarial learning we propose a novel Unsupervised Domain Adaptation (UDA) framework for segmenting nuclei in unannotated datasets. Prominent experimental results validate the effectiveness of our UDA model. Finally, assuming we have a few annotations available, we extend our work to semi-supervised domain adaptation (SSDA). We expect our proposed UDA and SSDA approach to be very useful in other biomedical image segmentation tasks.

CHAPTER 3

NuSegDA: DOMAIN ADAPTATION FOR NUCLEI SEGMENTATION

The accurate segmentation of nuclei is crucial for cancer diagnosis and further clinical treatments. To successfully train a nuclei segmentation network in a fully-supervised manner for a particular type of organ or cancer, we need the dataset with ground-truth annotations. However, such well-annotated nuclei segmentation datasets are highly rare, and manually labeling an unannotated dataset is an expensive, time-consuming and tedious process. Consequently, we require to discover a way for training the nuclei segmentation network with unlabeled dataset. In this chapter, we propose a model named NuSegUDA for nuclei segmentation on the unlabeled dataset (target domain). It is achieved by applying Unsupervised Domain Adaptation (UDA) technique with the help of another labeled dataset (source domain) that may come from different type of organ, cancer or source. We apply UDA technique at both of feature space and output space. We additionally utilize a reconstruction network and incorporate adversarial learning into it so that the source-domain images can be accurately translated to the target-domain for further training of the segmentation network. We validate our proposed NuSegUDA on two public nuclei segmentation datasets, and obtain significant improvement as compared with the baseline methods. Extensive experiments also verify the contribution of newly proposed image reconstruction adversarial loss, and target-translated source supervised loss to the performance boost of NuSegUDA. Finally, considering the scenario when we have a small number of annotations available from the target domain, we extend

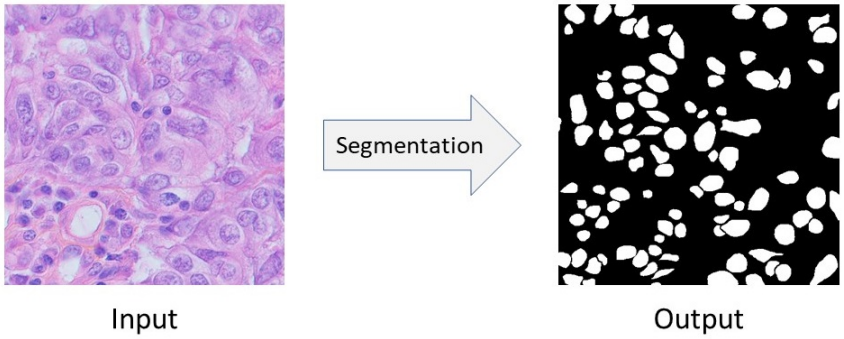

Figure 3.1. The semantic segmentation of nuclei. In this figure, the input image comes from Triple Negative Breast Cancer (TNBC).

our work and propose NuSegSSDA, a Semi-Supervised Domain Adaptation (SSDA) based approach.

3.1 Introduction

Nuclei are the fundamental organizational unit of life [71]. Nuclei segmentation, a subclass of biomedical image segmentation, is considered as an essential task of digital histopathology image analysis [99, 29]. However, accurate nuclei segmentation is quite challenging due to the significant variations in the shape and appearance of nuclei, clustered and overlapped nuclei, blurred nuclei boundaries, inconsistent staining methods, scanning artifacts, etc. (see Figure 3.1). Also, histopathology of different organs or cancer types may exhibit different textures, color distributions, morphology and scales [95, 56].

Nuclei segmentation problem can be seen as a semantic segmentation problem in which we want to segment the nuclei from it's background. Figure 3.1 shows the input image, and corresponding output of semantic segmentation of nuclei. Convolutional Neural Network (CNN) based approaches like Fully Convolutional Network (FCN) [53], U-Net [67], UNet++ [106], etc. give very promising results in biomedical

image segmentation tasks as well as in nuclei segmentation problems [73, 29, 71]. However, to successfully train these fully-supervised methods, we need at least a few amount of annotated data (i.e., images with their corresponding pixel-level ground-truth labels) [46, 101, 71]. Unfortunately, such well-annotated datasets, even if very small-sized, are highly rare in biomedical domain. Moreover, due to the heterogeneity of nuclei, it's even harder to learn good models under the scenario of lacking annotations and samples. Also, commonly used strategy which first collects an unannotated histopathology dataset and then do the manual pixel-level labeling with the help of experts is also an expensive, time-consuming and tedious process [94, 13, 99]. For example, annotating even a small nuclei segmentation dataset consisting of 50 image paches takes 120-130 hours of an expert pathologist's time [34]. Therefore, an urgent question is raised: how could we robustly train a deep CNN model for nuclei segmentation without any further need for annotations?

For nuclei segmentation problem, simply applying Transfer Learning (i.e., models trained with one organ or cancer type, and then evaluated with different organ or cancer types) unfortunately leads to poor performance due to the domain shift problem [71]. This domain shift problem happens due to different scanners, scanning protocols, tissue types, etc. [71]. In this paper, we propose Domain Adaptation, a subclass of Transfer Learning, based framework to solve the domain shift problem for nuclei segmentation. We consider the unannotated dataset (i.e., for which we want to predict the labels) as the target domain. Then, with the help of another related but different annotated dataset, referred as the source domain, we apply adversarial learning [25] based domain adaptation technique for nuclei segmentation problem. Thus, our proposed framework, learns from the labeled source domain and adapts to the unlabeled target domain.

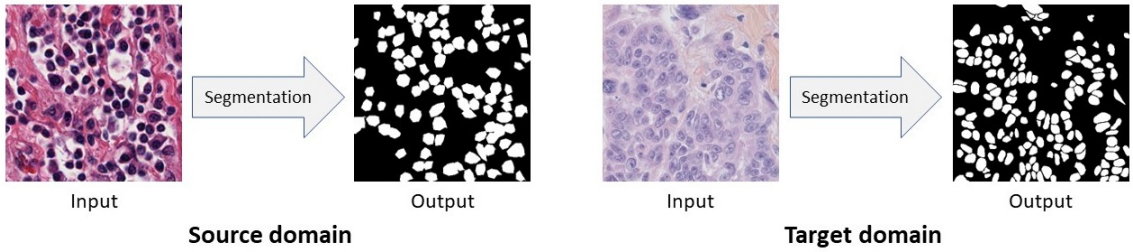

Figure 3.2. Images from different domains look dissimilar while their pixel-level segmentation outputs are similar. In this figure, source domain and target domain images come from Kidney Renal Clear cell carcinoma (KIRC) and Triple Negative Breast Cancer (TNBC), respectively.

In this work, we first propose an Unsupervised Domain Adaptation (UDA) model for nuclei segmentation to close the gap between the annotated source domain and unlabeled target domain. UDA methods are capable to minimize the labeling cost by utilizing cross-domain data and aligning the distribution shift between labeled source domain data and unlabeled target domain data. We empirically and carefully observed that, images from different nuclei datasets, even if collected from different organ or cancer types, exhibit dissimilarity although their corresponding segmentation ground-truth labels are quite similar (see Figure 3.2). In summary, ground-truth labels for nuclei segmentation are domain-invariant. Because of the aforementioned observation, we apply domain adaptation in the output space. Thus, with the help of adversarial learning, we train a robust nuclei segmentation network to generate source-domain look-alike outputs for target images. Adversarial learning attempts to align target-domain predictions with source-domain ground truths via discriminator training. In addition to image-level domain adaptation at the output space, we apply domain-invariant class-conditional feature-level domain adaptation in the feature space. However, simply forcing the target-domain distribution towards the source-domain distribution can destroy the latent structural patterns of the target domain, leading to a drop in the model's accuracy. Consequently, we also use a

reconstruction network to maximize the correlation between target images and target predictions. Again, a reconstruction network alone can not perfectly reconstruct original images (i.e., the reconstructed images lack original texture, style, color distribution, etc.) for which we incorporate adversarial learning into the reconstruction network, which in turn helps us to translate source domain images to the target domain. We additionally train our UDA model with these target-translated source images, and observe a significant performance boost. Finally, we extend our UDA framework to Semi-Supervised Domain Adaptation (SSDA) model considering that we have some annotations available from the target domain.

Conducting extensive experiments on two nuclei segmentation datasets we conclude that, our proposed UDA method, NuSegUDA, outperforms fully-supervised model trained on source domain and evaluated on target domain, and baseline generic and biomedical UDA segmentation models. Experimental result (see Section 4) also shows the impacts of training NuSegUDA with proposed image reconstruction adversarial loss, target-translated source images, and feature-level clustering loss. Furthermore, the accuracy of our SSDA model, NuSegSSDA, is highly competitive to the upper bound of fully-supervised model trained in the target domain.

Therefore, the main contributions of this paper are: **1)** We propose an adversarial learning based Unsupervised Domain Adaptation (UDA) approach, which is applied at both of feature space and output space to solve nuclei segmentation problem for unannotated datasets. **2)** Additionally, we incorporate adversarial learning into a reconstruction network to translate source domain images to the target domain, and train proposed model with these target-translated source images. **3)** Compared to many of the baselines, our proposed method is simple as it does not depend on any data synthesization or data augmentation. **4)** Our proposed UDA framework can be easily extended to Semi-Supervised Domain Adaptation (SSDA) in the scenario

where a small portion of the target domain is labeled. **5)** Extensive and comprehensive experiments on two datasets have demonstrated the superiority of the proposed methods.

3.2 Related Work

In literature, several domain adaptation models have been proposed for generic image segmentation. Isola et al. [38] applied conditional GAN [57] for image-to-image translation problems. CyCADA proposed an Unsupervised Domain Adaptation (UDA) model utilizing both of input space and feature space adaptation [33]. A multi-level adversarial network based domain adaptation approach for semantic segmentation was proposed in AdaptSegNet [79]. Zhang et al. [103] proposed a fully convolutional adaptation network for semantic segmentation. CrDoCo proposed a cross-domain consistency loss based pixel-wise adversarial domain adaptation algorithm [17]. Yang et al. [96] proposed adversarial self-supervision UDA model which maximizes agreement between clean samples and their adversarial examples. Toldo et al. [78] proposed feature-clustering based UDA framework that groups features of the same class into tight and well-separated clusters.

Domain adaptation has also been employed in different biomedical image segmentation tasks. A multi-connected domain discriminator based UDA model for brain lesion segmentation was proposed by Kamnitsas et al. [41]. Dong et al. [22] introduced another UDA framework for cardiothoracic ratio estimation through chest organ segmentation. Huo et al. [36] proposed an end-to-end CycleGAN [108] based whole abdomen MRI to CT image synthesis and CT splegonmegaly segmentation network. Mahmood et al. [56] proposed a nuclei segmentation approach in which a large dataset is generated using synthesization. Gholami et al. [24] proposed a biophysics-based medical image segmentation framework which enriches the training

dataset by generating synthetic tumor-bearing MR images. Hou et al. [34] also synthesized annotated training data for histopathology image segmentation. Haq and Huang [28] utilized adversarial learning at output space along with a reconstruction network for nuclei segmentation. Xia et al. [90] proposed Uncertainty-aware Multi-view Co-Training (UMCT) framework which is capable of utilizing large-scale unlabeled data to improve volumetric medical image segmentation. Raju et al. [65] proposed an user-guided domain adaptation framework for liver segmentation which uses prediction-based adversarial domain adaptation to model the combined distribution of user interactions and mask predictions. EndoUDA proposed another UDA-based segmentation model for gastrointestinal endoscopy imaging which comprises of a shared encoder and a joint loss function for improved unseen target domain generalization [11]. Li et al. [49] proposed another GAN [57] based framework for unsupervised domain adaptation of nuclei segmentation which also utilized self-ensembling and conditional random field [5]. Sharma et al. [71] proposed a mutual information based UDA method for cross-domain nuclei segmentation.

Several previous approaches [79, 22, 28, 78] employed unsupervised domain adaptation technique either in the output space or the feature space. Differently from these approaches, in our work we apply domain adaptation at both of output space and feature space. Additionally, unlike previous works, we utilize a reconstruction network to ensure that the target domain predictions spatially correspond to the target domain images. Also, several recent works [36, 56, 24, 34] applied complicated data synthesization techniques to generate a large training dataset. On the contrary, in our work we simply incorporate adversarial learning so that the source domain images can be translated to the target domain for further training.

3.3 Methodology

In this section, we first describe the problem that we aim to solve. Then, we introduce the details of our proposed Unsupervised Domain Adaptation (UDA) and Semi-Supervised Domain Adaptation (SSDA) framework. Finally, we discuss the implementations of the proposed models.

3.3.1 Problem Definition

In our nuclei segmentation problem, we have nuclei histopathology image patches as input X of size $H \times W \times 3$. The input X comes from either the source domain or the target domain. Depending on the problem (i.e., unsupervised or semi-supervised) and domain (i.e., source or target), we may also have the corresponding pixel-wise ground-truth label Y of size $H \times W \times 1$ which is basically a binary mask. Then, using the segmentation network, we want to predict the segmentation output $\hat{Y}$ of size $H \times W \times 1$.

Formally, in Unsupervised Domain Adaptation (UDA) problem, the source domain consists of N_s annotated images $\{(X_s, Y_s)\}$, and the target domain has N_t unannotated images $\{(X_t)\}$. In the case of Semi-Supervised Domain Adaptation (SSDA) problem, the source domain is the same as it is in UDA problem, and we assume that the target domain has N_t^l images with annotations $\{(X_t^l, Y_t)\}$ and N_t^u unannotated images $\{(X_t^u)\}$. In both of UDA and SSDA problem, the source domain data and target domain data are the related data but they come from different distributions (i.e., different organ or cancer types). For both of unsupervised and semi-supervised domain adaptation, our ultimate goal is to learn nuclei segmentation models that accurately produce the segmentation outputs in the target domain.

3.3.2 Unsupervised Domain Adaptation

We refer our nuclei segmentation Unsupervised Domain Adaptation (UDA) model as NuSegUDA, and the framework is shown in Figure 3.3. NuSegUDA consists of four modules: Segmentation network (S), Reconstruction network (R), Prediction Discriminator (D_P), and Image Discriminator (D_I).

3.3.2.1 Segmentation network

The segmentation network S takes image X as the input and produces the segmentation prediction $\hat{Y}$ of the same size as the input. Here, X can be either the source domain image X_s, or the target domain image X_t. Hence, the source domain prediction $\hat{Y}_s = S(X_s)$, and the target domain prediction $\hat{Y}_t = S(X_t)$. From the perspective of GAN [25] framework, the segmentation network S can be thought as the generator module.

We train S to generate the source domain segmentation predictions $\hat{Y}_s$ to be similar to the source domain ground-truth labels Y_s. Since in Unsupervised Domain Adaptation (UDA) the ground-truth labels are not available for target images, we can not compute any supervised pixel-level loss for target predictions. In practice, we found that combining dice-coefficient loss and entropy minimization loss is more effective than simply using binary cross-entropy loss for nuclei segmentation tasks. Therefore, we define segmentation loss L_{seg} as:

$$L_{dice}(X_s) = 1 - \frac{2.Y_s'.\hat{Y}_s'}{Y_s' + \hat{Y}_s'} \tag{3.1}$$

$$L_{em}(X_s) = -\frac{1}{H \times W} \sum_{h,w} \hat{Y}_s \log(\hat{Y}_s) \tag{3.2}$$

$$L_{seg}(X_s) = L_{dice}(X_s) + L_{em}(X_s) \tag{3.3}$$

where Y_s' and $\hat{Y}_s'$ are the flattened Y_s and $\hat{Y}_s$, respectively.

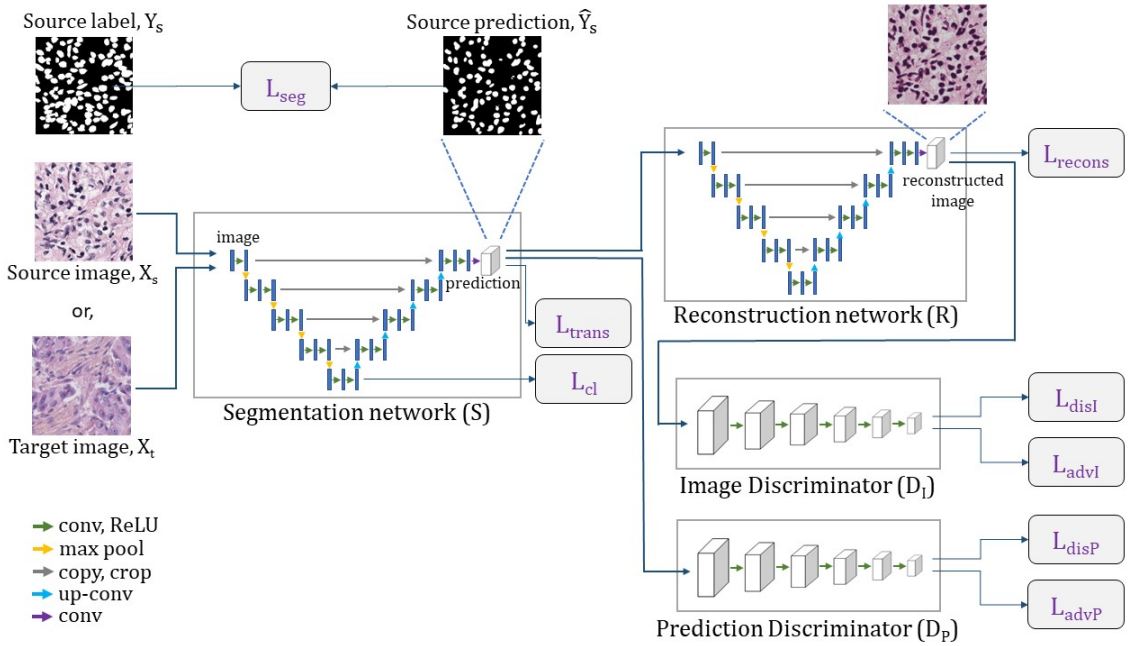

Figure 3.3. Complete architecture of NuSegUDA. Segmentation network generates segmentation outputs, from which reconstruction network reconstructs input images. Prediction discriminator distinguishes between source domain outputs and target domain outputs. Image discriminator distinguishes between original images and reconstructed images.

Here, question may arise that why we are using single segmentation network S in NuSegUDA although we have two different domains. Since we are particularly looking for nuclei from both domain images, it is very unusual to use multiple segmentation networks. Additionally, using two segmentation networks would increase the number of learnable parameters which would slow down the training process in turn. Therefore, single segmentation network helps to prevent the memory issues and training latency in NuSegUDA.

Training the segmentation network S with only the annotated source data teaches S to make accurate predictions for source images. However, this segmentation network may generate incorrect outputs for target images as there are visual

discrepancies between source images and target images (see Figure 3.2). This visual gap between domains causes the domain shift problem. According to our aforementioned observation that nuclei segmentation outputs are domain-invariant, we require S to produce target domain predictions as much as close to the source domain predictions. In other words, we want to make the distribution of target predictions $\hat{Y}_t$ closer to the distribution of source predictions $\hat{Y}_s$. For this reason, we utilize Prediction Discriminator D_P in NuSegUDA, and we define the prediction adversarial loss as:

$$L_{advP}(X_t) = -\frac{1}{H_p \times W_p} \sum_{h_p, w_p} \log\left(D_P(\hat{Y}_t)\right) \tag{3.4}$$

where $\hat{Y}_t = S(X_t)$, and H_p and W_p are height and width of the prediction discriminator output $D_P(\hat{Y}_t)$. The details of the Prediction Discriminator D_P is discussed in 3.3.2.3.

The prediction adversarial loss in Eq. (3.4) helps S to fool the prediction discriminator so that it considers $\hat{Y}_t$ as source domain segmentation outputs. Segmentation loss and the prediction adversarial loss jointly guide S to generate target domain predictions $\hat{Y}_t$ which look similar to source domain ground-truths.

3.3.2.2 Reconstruction network

As we mentioned earlier, the segmentation network S produces domain-invariant predictions for both domains. In other words, we want to generate the target domain predictions in a way so that they become similar to the source domain predictions. However, it is highly probable that the target predictions are not well-correlated with corresponding target input images. In this scenario, the ability of reconstructing the images from the predictions with similar visual appearance as input images will ensure that there is a correlation between the input image and segmentation output.

To ensure that our target domain predictions spatially correspond to the target domain images, reconstruction network R is used in NuSegUDA. In a similar way

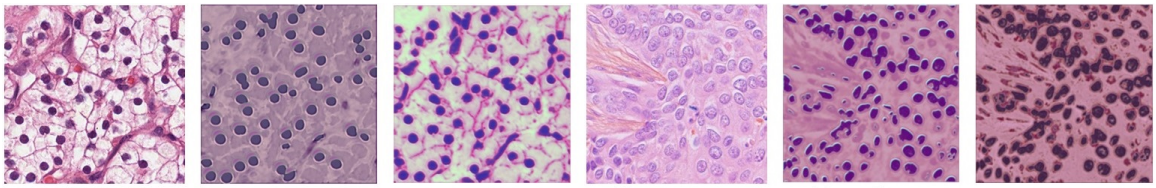

Figure 3.4. Visualization of the target-translated source domain images $X_{s\to t}$ which are also the same as reconstructed source images $\tilde{X}_s$. (a)-(c), and (d)-(f) are chosen from Kidney Renal Clear cell carcinoma (KIRC) domain, and Triple Negative Breast Cancer (TNBC) domain, respectively. In (c) and (f), we see that KIRC domain image is translated (i.e. reconstructed) into TNBC domain styles, and vice versa, respectively. In (b)-(c) and (e)-(f), $X_{s\to t}$ w/o L_{advI} and $X_{s\to t}$ w/ L_{advI} refer to the translated image when NuSegUDA is trained without and with image reconstruction adversarial loss L_{advI}, respectively..

to [89], we consider the segmentation network S and the reconstruction network R as an encoder and a decoder, respectively. R reconstructs target images from the corresponding predictions. Thus, S and R altogether works as an autoencoder.

Using our reconstruction network R, we first reconstruct target input images X_t from $\hat{Y}_t$. Then, we calculate the reconstruction loss as:

$$L_{recons}(X_t) = \frac{1}{H \times W \times C} \sum_{h,w,c} \left(X_t - R(\hat{Y}_t)\right)^2 \tag{3.5}$$

where, $R(\hat{Y}_t))$ is the output of reconstruction network for $\hat{Y}_t$, and C is the number of channels of input image X_t.

Although we use above reconstruction loss to reconstruct the target domain images from its predictions, the reconstructed images may have very different textures and styles (for both of nuclei and background) than the original images (see Figure 3.4). The reason is that the pixel-wise reconstruction loss L_{recons} (in Eq. (3.5)) can not capture the overall pixel distribution of target domain images. To solve this issue, in addition to L_{recons}, we also utilize an Image Discriminator $\mathrm{D_I}$ to distinguish the original images and the reconstructed images. To train R and S to generate

original-alike reconstructed images, we define image reconstruction adversarial loss as:

$$L_{advI}(X_t) = -\frac{1}{H_i \times W_i} \sum_{h_i, w_i} \log\left(D_I(\tilde{X}_t)\right) \tag{3.6}$$

where $\tilde{X}_t = R(\hat{Y}_t))$, and H_i and W_i are height and width of the image discriminator output $D_I(\tilde{X}_t)$. This adversarial loss L_{advI} trains R and S to reconstruct target domain images of similar distributions (in terms of texture, style, color distribution, etc.) to the original images from target domain.

In NuSegUDA, L_{advP} helps the segmentation network S to generate target predictions $\hat{Y}_t$ to be similar to the source predictions $\hat{Y}_s$. And, due to L_{advI}, reconstruction network R learns to reconstruct target images (i.e., $\tilde{X}_t$) which are very similar to the original target images in terms of texture, style, color distribution, etc. In other words, S maps both domain images (i.e., X_s and X_t) to a common prediction subspace $\mathbb{R}^n_p$, and from $\mathbb{R}^n_p$ R reconstructs the images in target domain. Therefore, using S and R we can translate source domain images X_s to the target domain. Thus, target translated source domain images $X_{s\to t} = R(S(X_s))$. Figure 3.4 shows the visualizations of the impacts of image reconstruction adversarial loss L_{advI} on $X_{s\to t}$. Finally, we train the segmentation network S with $\{(X_{s\to t}, Y_s)\}$ using following L_{trans} loss which is a combination of dice-coefficient loss and entropy minimization loss:

$$L_{dice}(X_{s\to t}) = 1 - \frac{2.Y'_s.\tilde{Y}'_s}{Y'_s + \hat{Y}'_s} \tag{3.7}$$

$$L_{em}(X_{s\to t}) = -\frac{1}{H \times W} \sum_{h,w} \tilde{Y}_s \log(\tilde{Y}_s) \tag{3.8}$$

$$L_{trans}(X_{s\to t}) = L_{dice}(X_{s\to t}) + L_{em}(X_{s\to t}) \tag{3.9}$$

where $\tilde{Y}_s = S(X_{s\to t})$. And, Y'_s and $\tilde{Y}'_s$ are the flattened Y_s and $\tilde{Y}_s$, respectively.

3.3.2.3 Discriminators

We utilize two discriminators in NuSegUDA: Prediction Discriminator (D_P), and Image Discriminator (D_I). Prediction Discriminator distinguishes between source domain outputs and target domain outputs, whereas Image Discriminator distinguishes between original images and reconstructed images. We discuss the details of both discriminators in the following.

Prediction Discriminator As our goal is to generate similar predictions for both of source images and target images, we incorporate prediction discriminator D_P in NuSegUDA. This discriminator takes source domain prediction or target domain prediction as input, and then distinguishes whether the input (i.e., prediction) comes from the source domain or the target domain. To train D_P, we use following cross-entropy loss:

$$L_{disP}(\hat{Y}) = -\frac{1}{H_p \times W_p} \sum_{h_p, w_p} z_p . \log\left(D_P(\hat{Y})\right) + (1 - z_p) . \log\left(1 - D_P(\hat{Y})\right) \tag{3.10}$$

where z_p=0 when D_P takes target domain prediction as its input, and z_p=1 when the input comes from source domain prediction.

Image Discriminator We use image discriminator D_I in NuSegUDA so that the reconstructed image distribution becomes similar to original image distribution. The input of D_I is either the original target image or the reconstructed target image. Then, D_I distinguishes whether the input is original or the reconstructed one. Similar to D_P, we use following cross-entropy loss to train D_I:

$$L_{disI}(X) = -\frac{1}{H_i \times W_i} \sum_{h_i, w_i} z_i . \log\left(D_I(X)\right) + (1 - z_i) . \log\left(1 - D_I(X)\right) \tag{3.11}$$

where z_i=0 when D_I takes reconstructed target image $\tilde{X}_t$ as its input, and z_i=1 when the input comes from original target images X_t.

3.3.2.4 Feature-level adaptation

In addition to image-level domain adaptation at the outputs, we also apply feature-level domain adaptation in NuSegUDA to reduce the domain gap in the feature space. We assume that, our segmentation network S is composed of an encoder S_E and a decoder S_D (i.e., $S = S_E o S_D$). Here, the encoder S_E works as a feature extractor. Due to the discrepancy of input statistics across domains, there is also a shift of feature distribution in the feature space spanned by S_E. Similar to [78], we utilize a clustering loss at the feature-level to serve as a constraint toward a class-conditional feature alignment between domains.

Given source image X_s and target image X_t, we first extract the features $F_s = S_E(X_s)$ and $F_t = S_E(X_t)$. Then, the clustering loss is computed as:

$$L_{cl}(X_s, X_t) = \frac{1}{\mid F_{s,t} \mid} \sum_{f_i \in F_{s,t}, \hat{y}_i \in \hat{Y}_{s,t}} d(f_i, c_{\hat{y}_i}) - \frac{1}{\mid C \mid (\mid C \mid -1)} \sum_{j \in C} \sum_{k \in C, k \neq j} d(c_j, c_k) \tag{3.12}$$

where f_i is the feature vector corresponding to a spatial location of F_s or F_t, $\hat{y}_i$ is the corresponding predicted class, and C is the set of semantic classes which is $\{0, 1\}$ for our nuclei segmentation problem. To compute $\hat{y}_i$, the segmentation prediction $\hat{Y}$ is downsampled to match the spatial dimension of F. We set the function d(.) to L1 norm. In Eq. (3.12), c_j denotes the centroid of semantic class j, which is computed using following formula:

$$c_j = \frac{\sum_{f_i} \sum_{\hat{y}_i} \delta_{j,\hat{y}_i} f_i}{\sum_{\hat{y}_i} \delta_{j,\hat{y}_i}}, j \in \{0, 1\} \tag{3.13}$$

where $\delta_{j,\hat{y}_i}$ is equal to 1 if $\hat{y}_i = j$, and to 0 otherwise.

In Eq. (3.12), the clustering loss is composed of two terms: the first term measures how close the features are from their respective centroids, and the second term measures how far the semantic class centroids are from each other. Therefore, according to the first term, the feature vectors of the same class from same or different domain are tightened around the class feature centroids. And, because of the second term, features from different classes gets a repulsive force applied to feature centroids which moves them apart.

Thus, we minimize the following total loss when training our segmentation network S and reconstruction network R:

$$\begin{aligned} L_{uda}(X_s, X_t) = L_{seg}(X_s) + \lambda_{advP} L_{advP}(X_t) + \lambda_{recons} L_{recons}(X_t) + \\ \lambda_{advI} L_{advI}(X_t) + \lambda_{trans} L_{trans}(X_{s \to t}) + \lambda_{cl} L_{cl}(X_s, X_t) \end{aligned} \tag{3.14}$$

where, λ_{advP}, λ_{recons}, λ_{advI}, λ_{trans} and λ_{cl} are the weights to balance corresponding losses.

3.3.3 Semi-Supervised Domain Adaptation

In Semi-Supervised Domain Adaptation (SSDA) problem, we aims to ensure the best usages of available target domain annotations Y_t when training our segmentation network S. In such scenarios, we extend proposed NuSegUDA framework to NuSegSSDA, a nuclei segmentation SSDA model.

In NuSegSSDA, for unannotated target images X_t^u we follow the same steps as NuSegUDA. However, when we encounter an annotated target data (X_t^l, Y_t) while training, we additionally compute the segmentation loss $L_{seg}(X_t^l)$ in the similar manner to Eq. (3.3). Then, while computing the total loss we incorporate $L_{seg}(X_t^l)$ so

that the segmentation network learns to generate the predictions closer to target ground-truths. Therefore, Eq. (3.14) is now modified as below:

$$L_{ssda}(X_s, X_t^l, X_t^u) = L_{seg}(X_s) + L_{seg}(X_t^l) + \lambda_{advP} L_{advP}(X_t^u) + \lambda_{recons} L_{recons}(X_t^u) + \lambda_{advI} L_{advI}(X_t^u) + \lambda_{trans} L_{trans}(X_{s\to t}) + \lambda_{cl} L_{cl}(X_s, X_t^l, X_t^u) \tag{3.15}$$

3.4 Experiments

3.4.1 Dataset

In our experiments, we use two H&E stained histopathology datasets with ground-truth annotations: 1) KIRC, and 2) TNBC. Both of the datasets that we used are public.

3.4.2 Implementations

In our work, we use U-Net [67] as both of our segmentation network and reconstruction network. We choose U-Net so that our proposed segmentation framework can be directly applied in other biomedical domains. We preferred U-Net over UNet++ [106] because of the less number of parameters. Following DCGAN [64], we designed our prediction discriminator and image discriminator consisting of five convolutional layers. To train NuSegUDA and NuSegSSDA, we followed the training strategy from GAN [25]. Adam optimizer [44] with learning rate 0.0001, 0.001, 0.001 and 0.001 are used in segmentation network, reconstruction network, prediction discriminator, and image discriminator, respectively. We empirically choose 0.001, 0.01, 0.001, 0.001 and 0.002 as λ_{advP}, λ_{recons}, λ_{advI}, λ_{trans} and λ_{cl}, respectively. We implement NuSegUDA and NuSegSSDA using PyTorch [61], and trained on a single GPU. We do not use any data augmentation in our experiments.

	Experiment-1 KIRC → TNBC			**Experiment-2** TNBC → KIRC		
Method	**IoU%**	**Dice score**	**HD**	**IoU%**	**Dice score**	**HD**
U-Net (source-trained)	52.66	0.6875	10.1214	54.82	0.7056	9.2487
DA-ADV	54.93	0.7079	9.6531	55.43	0.7107	9.0142
AdaptSegNet	56.49	0.7198	9.1512	56.87	0.7235	8.3477
CellSegUDA	59.02	0.7394	8.5653	57.09	0.7242	8.1739
OrClEmb	59.23	0.7402	8.5564	57.05	0.7236	8.1923
EndoUDA	59.81	0.7445	8.3317	57.39	0.7277	8.1254
SelfEnsemb	60.02	0.7468	8.2524	57.45	0.7292	8.1121
MaNi	60.09	0.7477	8.2746	57.48	0.7293	8.1493
U-Net (target-trained)	66.57	0.7985	7.7301	62.04	0.7621	7.6281
NuSegUDA (ours)	**60.51**	**0.7525**	**8.0011**	**57.68**	**0.7303**	**8.0881**

Table 3.1. Unsupervised Domain Adaptation (UDA) results for Experiment-1 and Experiment-2. IoU and HD denotes Intersection over Union, and Hausdorff Distance, respectively. Results are from testing on TNBC-test and KIRC-test for experiment-1 and experiment-2, respectively.

3.4.3 Experimental results

3.4.3.1 Unsupervised Domain Adaptation

Experiment-1 (KIRC → TNBC) In our first experiment, we choose KIRC as source domain and TNBC as target domain, denoted by KIRC → TNBC. In our experiment, we choose U-Net [67] as the representative of Convolutional Neural Network (CNN) based approaches. Fully-supervised segmentation model U-Net gives an insight of how it performs when directly applying transfer learning (i.e., training with only KIRC and then test it on TNBC without any modifications). AdaptSegNet [79] and OrClEmb [78] represent generic Unsupervised Domain Adaptation (UDA) models. DA-ADV [22], CellSegUDA [28], EndoUDA [11], SelfEnsemb [49] and MaNi [71] are chosen as the representatives of UDA model for biomedical image segmentation.

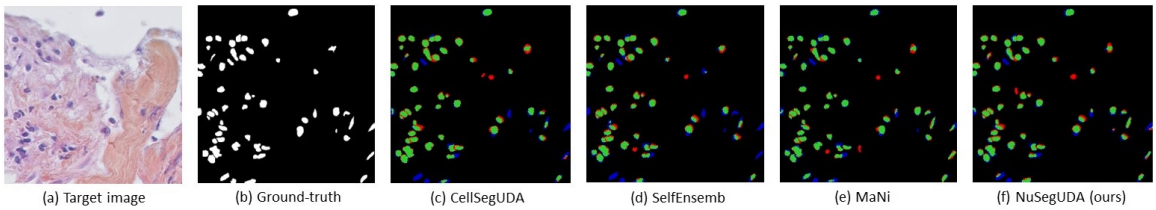

Figure 3.5. Visualizations of Unsupervised Domain Adaptation (UDA) for KIRC→TNBC. In (c)-(f), green pixels, red pixels, and blue pixels indicate the true positives, false positives, and false negatives, respectively. In other words, green and red pixels indicate the predicted nuclei pixels, whereas green and blue pixels indicate the ground-truth nuclei pixels. This average-dense nuclei histopathology image in (a) is chosen so that the reader can easily find out the visual differences without further zooming-in.

From Table 3.1, we see that source-trained U-Net gives the lower-bound of experimental performance (see first row of Table 3.1) which happens because of the visual domain gap between source training images and target test images, also known as domain shift problem. We see that, our proposed UDA model NuSegUDA outperforms all UDA baseline models in terms of IoU%, Dice score, and Hausdorff distance. Specifically, NuSegUDA has 1.28 and 0.42 higher IoU% than best generic UDA baseline OrClEmb, and best biomedical UDA baseline MaNi, respectively. Figure 3.5 shows the visualization results of CellSegUDA, SelfEnsemb, MaNi and NuSegUDA. In Table 3.1, the second to last row (i.e., U-Net (target-trained)) shows the upper-bound of experimental performance (i.e., training U-Net with TNBC-train and testing it on TNBC-test).

Experiment-2 (TNBC → KIRC) We conduct another experiment in the similar way to experiment-1 by selecting TNBC as source and KIRC as target domain. This experiment also reflects the excellence of NuSegUDA compared to other approaches in terms of segmentation accuracies (see last three columns of Table 3.1).

	Experiment-1 KIRC → TNBC			Experiment-2 TNBC → KIRC		
Method	**IoU%**	**Dice**	**HD**	**IoU%**	**Dice**	**HD**
U-Net (source 100% + target 10%)	60.74	0.7534	8.3627	56.89	0.7194	8.5122
CellSegSSDA (source 100% + target 10%)	60.96	0.7557	8.3563	58.81	0.7377	7.9817
NuSegSSDA (source 100% + target 10%) (ours)	**61.12**	**0.7578**	**8.3274**	**58.99**	**0.7401**	**7.9629**
U-Net (source 100% + target 25%)	61.67	0.7607	8.2742	59.32	0.7405	7.9211
CellSegSSDA (source 100% + target 25%)	62.94	0.771	8.0966	59.73	0.7443	**7.8647**
NuSegSSDA (source 100% + target 25%) (ours)	**63.15**	**0.7732**	**8.0487**	**59.79**	**0.7449**	7.8752
U-Net (source 100% + target 50%)	56.73	0.7208	9.1473	59.95	0.7464	7.8461
CellSegSSDA (source 100% + target 50%)	63.59	0.7748	7.9802	60.32	0.7494	7.7958
NuSegSSDA (source 100% + target 50%) (ours)	**63.97**	**0.7802**	**7.9549**	**60.53**	**0.7511**	**7.7754**
U-Net (source 100% + target 75%)	59.06	0.7394	8.6286	61.63	0.7592	7.7026
CellSegSSDA (source 100% + target 75%)	64.96	0.7862	7.8496	61.01	0.7541	7.7275
NuSegSSDA (source 100% + target 75%) (ours)	**65.22**	**0.7901**	**7.7928**	**61.68**	**0.7598**	**7.6872**
U-Net (target 100%)	66.57	0.7985	7.7301	62.04	0.7621	7.6281

Table 3.2. Semi-Supervised Domain Adaptation (SSDA) results for Experiment-1 and Experiment-2. IoU, Dice, and HD denotes Intersection over Union, Dice score, and Hausdorff Distance, respectively. NuSegSSDA refers to our proposed SSDA model. NuSegSSDA (source 100% + target n%) denotes n% annotations available in TNBC-train and KIRC-train for experiment-1 and experiment-2, respectively. Results are from testing on TNBC-test and KIRC-test for experiment-1 and experiment-2, respectively.

3.4.3.2 Semi-Supervised Domain Adaptation

Experiment-1 (KIRC → TNBC) In experiment-1, we assess our Semi-Supervised Domain Adaptation (SSDA) method NuSegSSDA for KIRC → TNBC. Table 3.2 shows the experimental performances of NuSegSSDA. For this experiment, the source dataset KIRC is the same as UDA experiments. However, now we treat TNBC as partially labeled. We train NuSegSSDA considering 10%, 25%, 50% and 75% images from TNBC-train dataset have annotations available. Then, testing on TNBC-test gives us increasing IoUs and Dice scores, and decreasing Hausdorff Distances. This happens because more false negative nuclei can be identified and some false positive

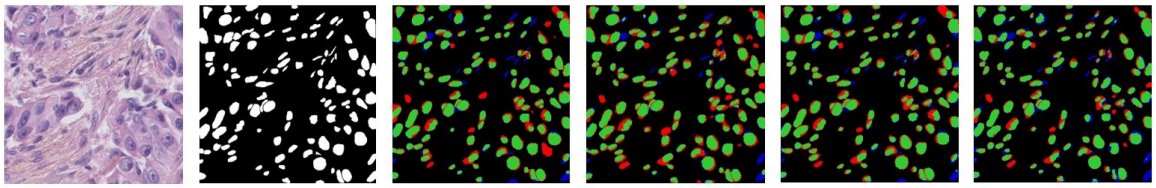

Figure 3.6. Visualizations of Semi-Supervised Domain Adaptation (SSDA) for KIRC→TNBC. In (c)-(f), green pixels, red pixels, and blue pixels indicate the true positives, false positives, and false negatives, respectively..

nuclei can be removed by NuSegSSDA as we train it with more target annotations (see Figure 3.6). We compare NuSegSSDA with fully-supervised model U-Net [67], and baseline biomedical SSDA model CellSegSSDA [28] to demonstrate the superiority of our proposed SSDA model. To train U-Net, we combine full KIRC dataset with the same 10%, 25%, 50% and 75% of TNBC-train we chose to train NuSegSSDA. We observe that, the accuracy of NuSegSSDA approaches to the upper-bound (only lower by 1.35 IoU%) as we train with more annotations from target domain.

Experiment-2 (TNBC → KIRC) In our second experiment, we select TNBC as source and KIRC as target domain. The second experiment also demonstrates the excellence of NuSegSSDA compared to U-Net [67] and CellSegSSDA [28] (see last three columns of Table 3.2). Similar to experiment-1, for the second experiment we again see that the segmentation accuracies of NuSegSSDA increase when more target images are annotated.

3.4.3.3 Ablation Studies

To verify the robustness of proposed UDA framework, we perform extensive ablation studies on the adaptation of NuSegUDA from KIRC to TNBC, and from

				Experiment-1 KIRC → TNBC			**Experiment-2** TNBC → KIRC		
Method	$\boldsymbol{L_{advI}}$	$\boldsymbol{L_{trans}}$	$\boldsymbol{L_{cl}}$	**IoU%**	**Dice**	**HD**	**IoU%**	**Dice**	**HD**
CellSegUDA				59.02	0.7394	8.5653	57.09	0.7242	8.1739
CellSegUDA w/ L_{advI}	✓			59.38	0.7405	8.4316	57.17	0.7252	8.1422
CellSegUDA w/ L_{trans}		✓		58.44	0.7357	8.6123	56.77	0.7209	8.3865
CellSegUDA w/ L_{cl}			✓	59.11	0.7398	8.5734	57.02	0.7237	8.1203
NuSegUDA w/o L_{advI}		✓	✓	58.59	0.7365	8.5914	56.82	0.7212	8.3685
NuSegUDA w/o L_{trans}	✓		✓	59.45	0.7411	8.4021	57.19	0.7253	8.2468
NuSegUDA w/o L_{cl}	✓	✓		60.36	0.7512	8.1963	57.63	0.7298	8.1247
NuSegUDA (ours)	✓	✓	✓	**60.51**	**0.7525**	**8.0011**	**57.68**	**0.7303**	**8.0880**

Table 3.3. Impacts of L_{advI}, L_{trans} and L_{cl} loss on NuSegUDA for Experiment-1 and Experiment-2. IoU, Dice, and HD denotes Intersection over Union, Dice score, and Hausdorff Distance, respectively. NuSegUDA w/o L_{advI}, NuSegUDA w/o L_{trans}, and NuSegUDA w/o L_{cl} refer to our proposed UDA model without image adversarial loss, target-translated source supervised loss, and clustering loss, respectively. Results are from testing on TNBC-test and KIRC-test for experiment-1 and experiment-2, respectively.

TNBC to KIRC. First, we examine the contribution of each loss to the final IoU%, Dice score, and Hausdorff Distance; then, we investigate the effects of different segmentation network backbones on NuSegUDA.

Effectiveness of Losses The contribution of image adversarial loss L_{advI}, target-translated source supervised loss L_{trans}, and clustering loss L_{cl} to our proposed NuSegUDA model is shown in Table 3.3. We see that, simply applying only L_{advI} or L_{cl} to CellSegUDA [28] gives little better performance than CellSegUDA alone. However, when we apply only target-translated source supervised loss L_{trans} to CellSegUDA, the performance is inferior due to the absence of L_{advI} loss. Without applying image-adversarial loss L_{advI}, target-translated source images $X_{s\to t}$ looks very different from the target-domain images in terms of texture, style, color distribution, etc. (see Fig-

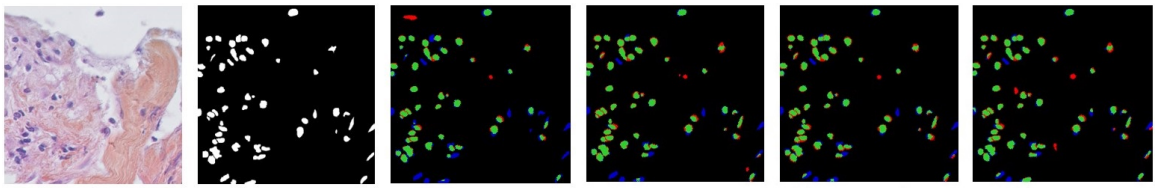

Figure 3.7. Visualizations of the effectiveness of proposed L_{advI}, L_{trans} and L_{cl} loss on NuSegUDA for KIRC→TNBC. In (c)-(f), green pixels, red pixels, and blue pixels indicate the true positives, false positives, and false negatives, respectively..

ure 3.4). As a result, the performance of the model (i.e., CellSegUDA w/ L_{trans}) decreases when trained with these $X_{s\to t}$ images.

Similarly, NuSegUDA w/o L_{advI} gives much worse performance than NuSegUDA which happens because of training NuSegUDA with less-realistic target-translated source domain images. This again validates the effectiveness of L_{advI} on NuSegUDA. Finally, with all the proposed losses enabled, we achieve the best performing model NuSegUDA for both of the experiments which demonstrates the combined impact of newly proposed image adversarial loss, target-translated source supervised loss, and clustering loss on NuSegUDA. Figure 3.7 shows the visualization results of NuSegUDA w/o L_{advI}, NuSegUDA w/o L_{trans}, NuSegUDA w/o L_{cl}, and NuSegUDA.

Impacts of different segmentation networks In NuSegUDA, we use U-Net [67] as the backbone segmentation network. We also assess the model performance by replacing the backbone segmentation network with two more frequently-used Convolutional Neural Network (CNN) based approaches: FCN [53] and UNet++ [106]. As mentioned earlier, CNN based approaches are still the dominant ones for semantic segmentation of nuclei. However, due to the intrinsic locality nature and limited receptive fields of convolution operations, CNN based models may be incapable of

	Experiment-1 KIRC → TNBC			**Experiment-2** TNBC → KIRC		
Segmentation Network	**IoU%**	**Dice score**	**HD**	**IoU%**	**Dice score**	**HD**
FCN	59.23	0.7398	8.4125	55.81	0.7165	8.7365
U-Net	60.51	0.7525	8.0011	57.68	0.7303	8.0880
UNet++	60.57	0.7529	8.0336	57.41	0.7282	8.1575
TransUNet	59.87	0.7476	8.1562	57.02	0.7256	8.1742

Table 3.4. Impacts of different segmentation network backbones in NuSegUDA.

capturing the global context of the input [14, 104, 39]. To this end, we explore the feasibility of Transformers, an alternative to CNNs, as the backbone segmentation network in NuSegUDA. Transformer mainly utilizes self-attention mechanism to extract inherent features [77], and due to this self-attention mechanism, transformers are powerful at modeling the global context of an input [104]. To examine the effectiveness of Vision Transformer based model, we replace U-Net in NuSegUDA with TransUNet [14] which basically combines a hybrid CNN-transformer encoder architecture with a decoder.

Table 3.4 shows the quantitative results of using different segmentation networks in NuSegUDA. We see that, among CNN-based models, UNet++ and U-Net outperform other CNN approaches in Experiment-1, and Experiment-2, respectively. We also see that, Transformer-based model TransUNet does not give any better accuracy than U-Net and UNet++ for both of the experiments. This happens due to our small-sized training datasets, because Vision Transformers (VT) need lot of data for training, usually more than what is necessary to standard CNNs [52].

3.5 Conclusion

Accurate nuclei segmentation is a significant step for cancer diagnosis and further clinical procedures. Collecting a fully annotated nuclei segmentation dataset, or manually labeling an unannotated dataset is expensive, time-consuming and impractical although such annotations are required to train Convolutional Neural Networks in fully-supervised manner. In this work, we propose a novel Unsupervised Domain Adaptation (UDA) framework named NuSegUDA for segmenting nuclei in unannotated datasets by utilizing adversarial learning. In NuSegUDA, we apply domain adaptation at both of feature space and output space. We also incorporate image adversarial loss and target-translated source supervised loss into NuSegUDA, and train the model with target-translated source domain images. Extensive and prominent experimental results validate the effectiveness of each of the newly proposed modules and losses, and the superiority of NuSegUDA over baseline models. Finally, assuming we have a few annotations available, we extend our work to Semi-Supervised Domain Adaptation (SSDA). We expect our proposed UDA and SSDA approaches to be very useful in other biomedical image segmentation tasks.

CHAPTER 4

SELF-SUPERVISED PRE-TRAINING FOR NUCLEI SEGMENTATION

The accurate segmentation of nuclei is crucial for cancer diagnosis and further clinical treatments. For semantic segmentation of nuclei, Vision Transformers (VT) have the potentiality to outperform Convolutional Neural Network (CNN) based models due to their ability to model long-range dependencies (i.e., global context). Usually, VT and CNN models are pre-trained with large-scale natural image dataset (i.e., ImageNet) in fully-supervised manner. However, pre-training nuclei segmentation models with ImageNet is not much helpful because of morphological and textural differences between natural image domain and medical image domain. Also, ImageNet-like large-scale annotated histology dataset rarely exists in medical image domain. In this chapter, we propose a novel region-level Self-Supervised Learning (SSL) approach and corresponding triplet loss for pre-training semantic nuclei segmentation model with unannotated histology images extracted from Whole Slide Images (WSI). Our proposed region-level SSL is based on the observation that, non-background (i.e., nuclei) patches of an input image are difficult to predict from surrounding neighbor patches, and vice versa. We empirically demonstrate the superiority of our proposed SSL incorporated VT model on two public nuclei segmentation datasets.

4.1 Introduction

Nuclei segmentation is considered as a fundamental task of digital histopathology image analysis. For semantic segmentation of nuclei, Convolutional Neural Net-

work (CNN) based approaches give very promising results [54, 66, 107, 28]. However, due to the intrinsic locality nature and limited receptive fields of convolution operations, CNN based models are incapable of capturing the global context of the input [14, 104]. Transformers, an alternative to CNNs, are powerful at modeling the global context of input images [104]. Also, Transformers show superior transferability for downstream tasks, when pre-trained with large-scale dataset. However, Vision Transformers (VT) need lot of data for training, usually more than what is necessary to standard CNNs [52].

Usually, VTs are pre-trained with large-scale annotated natural image dataset like ImageNet [19], and then fine-tuned to downstream tasks [23]. However, histology images are quite different from natural images due to the nuclei and background textures, morphological structures of nuclei, large variations in the shape and appearance of nuclei, clustered and overlapped nuclei, blurred nuclei boundaries, inconsistent staining methods, scanning artifacts, etc. [56, 95]. Due to this domain gap between natural images and medical images, the ImageNet pre-trained models may yield marginal improvement over train-from-scratch models for nuclei segmentation tasks [93]. Unfortunately, in medical image domain, ImageNet-like large-scale annotated histopathology image datasets do not exist, and they are very difficult to produce, because of expensive, time-consuming and tedious labeling process of histology images [95, 12].

In this work, we propose a Transformer-based Self-Supervised Learning (SSL) approach for pre-training so that the segmentation network implicitly acquires a better understanding of the nuclei and background using a large-scale unannotated histology image dataset extracted from Whole Slide Images (WSI). In computer vision, SSL is used to learn useful data representations without using any labels [60, 8, 9]. To achieve this goal, we first divide the image into $k \times k$ patches where k=32. Then,

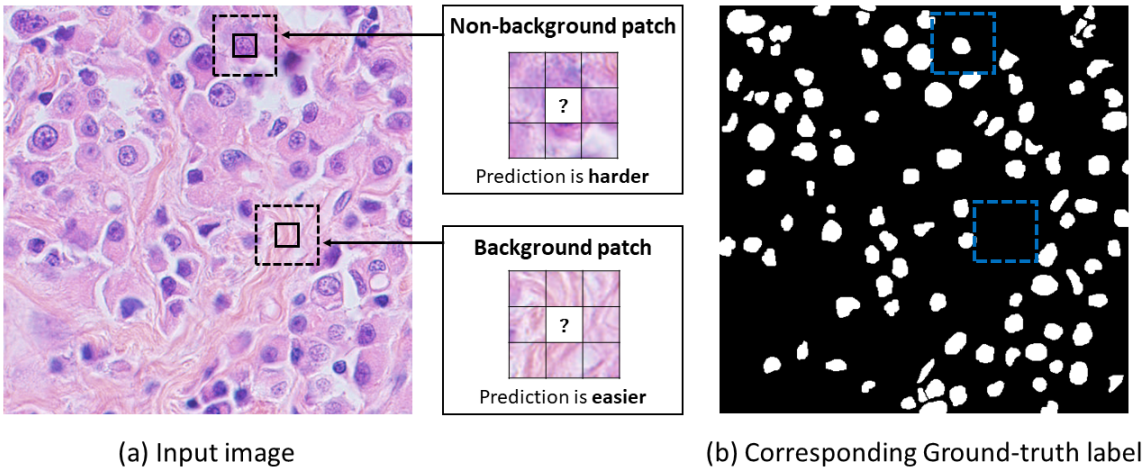

Figure 4.1. We divide the input image into kxk patches. We try to predict patch features using its 8-connected neighboring patches. We see that, predicting a non-background (i.e., nuclei) patch is much harder than predicting a background patch.

we try to predict each of the patch features from its 8-connected neighboring patches. Figure 4.1 shows that predicting non-background patches (i.e., that contain nuclei) is much harder than background patch prediction. Based on this aforementioned observation, we design region-level triplet loss to pre-train the segmentation network. Our pre-trained SSL model learns to separate nuclei features from the background features in the embedding space. Additionally, our SSL approach involves the image-level sub-task of predicting the scale of image, which enables the segmentation network to implicitly acquire further knowledge of nuclei size and shape. Finally, we fine-tune the pre-trained network for nuclei segmentation with a small annotated dataset.

Thus, the main contributions of this paper are: **1)** We propose a novel region-level Self-Supervised Learning (SSL) approach and corresponding triplet loss for pre-training semantic nuclei segmentation model with unannotated histology image dataset. **2)** We incorporate our proposed pre-training technique into a Vision Transformer (VT). To the best of our knowledge, this is the first work focusing on

Transformer-based SSL for semantic segmentation of nuclei. **3)** Extensive experimental results demonstrate the superiority of our proposed SSL incorporated transformer model over baseline methods.

4.2 Related Work

Several Self-Supervised Learning (SSL) approaches have been proposed for nuclei segmentation. An instance-aware SSL model is proposed considering scale-wise triplet learning and count ranking as proxy sub-tasks [93]. Another self-supervised nuclei segmentation approach without requiring annotations is proposed utilizing scale classification as a self-supervision signal to locate nuclei [68].

In literature, Transformer has been employed in various computer vision problems [27, 81, 23, 7, 97, 104, 92, 83, 14]. For natural image segmentation, a pure transformer-based model named SEgmentation TRansformer (SETR) [104] is proposed by treating semantic segmentation problem as a sequence-to-sequence prediction task. For medical image segmentation, TransUNet [14] is proposed to solve multi-organ segmentation task. In recent times, SSL have been applied to Vision Transformers (VT). In Self-supervised vision Transformer (SiT) [2], parts of the input image are corrupted using several local transformation operations, and original image is reconstructed later from the corrupted one. An auxiliary self-supervised localization task also has been proposed which encourages the VTs to learn relative distances between patch embedding pairs [52].

4.3 Methodology

In semantic nuclei segmentation problem, we have nuclei histology image of size $H \times W \times 3$ as input, and we want to predict the segmentation output of size

$H \times W$. We first pre-train our proposed model with unannotated image patches $D_s = \{(X_n)\}$. Then, we fine-tune the model with annotated images $D_t = \{(X_t, Y_t)\}$. In this work, since we propose Transformers-based Self-Supervised learning method for Nuclei segmentation, we name our proposed framework as TransNuSS. Figure 4.2 shows the complete architecture of TransNuSS.

In our work, we adopt TransUNet [14] as the segmentation network. The encoder of the segmentation network consists of a hybrid Convolutional Neural Network (CNN) - Transformer architecture. CNN works as a feature extractor to generate feature map F_x for the input. Then, patch embedding is applied to get embedding $Z_0 \in \mathbb{R}^{n_{patch} \times d}$, where $n_{patch} = \frac{H}{16} \times \frac{W}{16}$ and d is the dimension of embedding space which we set to 768. After that, transformer encoder appears which consists of L layers of Multi-head Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks. The final layer of the transformer encoder produces hidden features $Z_L \in \mathbb{R}^{n_{patch} \times d}$. In encoder, we use ResNet-50 [31] and ViT [23] as CNN and transformer, respectively. The decoder of the segmentation network consists of multiple upsampling steps. At first, hidden features Z_L is reshaped to the shape of $d \times \frac{H}{16} \times \frac{W}{16}$, which we denote as A. Then, a 3×3 convolution is applied to decrease the depth to 512. Finally, multiple upsampling blocks are used to generate the full resolution segmentation mask. We refer the reader to [14] for more details.

4.3.1 Self-Supervised Pre-Training with unannotated dataset

For each image $X_n \in D_s$ of size $H \times W \times 3$, we generate a same-size image X_s by cropping and scaling. To generate X_s, we first randomly select a scaling-factor z_s from a pool $\{1.0, 1.25, 1.5, 1.75, 2.0\}$. We denote this scale-pool as S_p. Now, we randomly crop a patch from X_n, and then scale the cropped patch z_s times so that

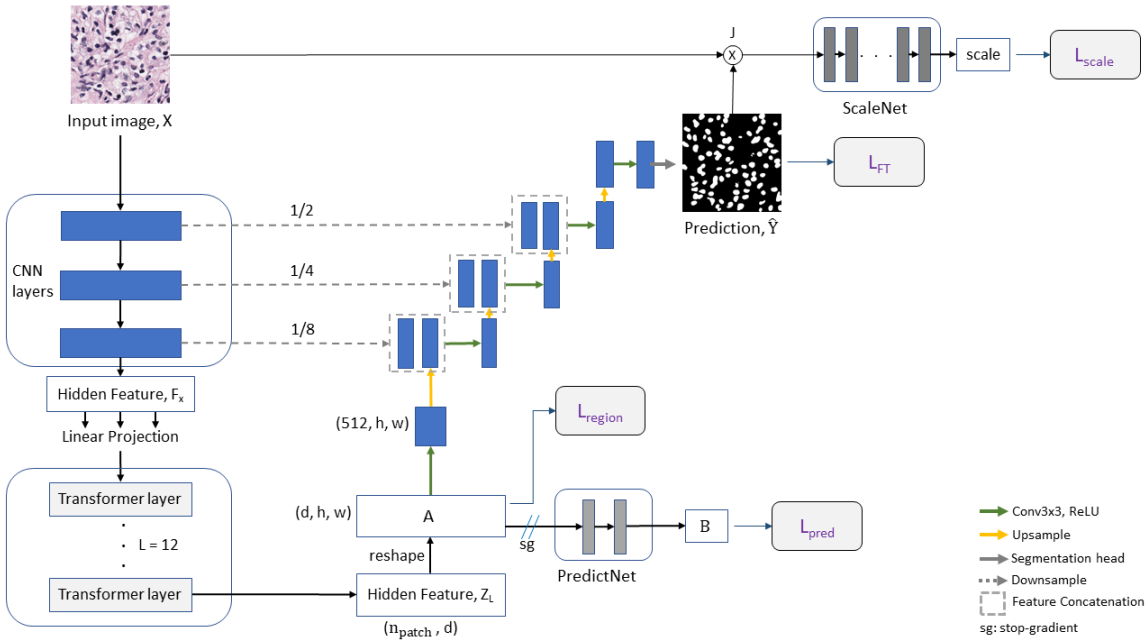

Figure 4.2. Complete architecture of TransNuSS.

the scaled patch becomes of size $H \times W \times 3$. Thus, for self-supervised pre-training, our input consists of $\{(X_n, X_s, z_s)\}$.

4.3.1.1 Region-level triplet loss

For self-supervised pre-training, we consider that we have background and non-background image patches (see Figure 4.1) in an input image. Then, we propose region-level triplet loss to learn the embedding space. We expect that, in the embedding space, feature should have similarity and dissimilarity among same and different types of patches, respectively. We design our triplet loss in a way so that the segmentation network can implicitly learn to separate background and non-background patch features in a given image. To generate triplet samples, we use feature map A which is of size $d \times h \times w$ where $h = \frac{H}{16}$ and $w = \frac{W}{16}$. Therefore, A contains $h \times w$ number (i.e., the number of patches) of d-dimensional feature vectors. As we men-

tioned before, our intuition and observation is that: in this embedding space, we can easily predict a background feature vector from its 8-connected neighboring features vectors, whereas predicting a non-background feature vector is comparatively harder.

We try to predict feature vector $A_{i,j} \in \mathbb{R}^d$ at each spatial location (i, j) of A, where $2 \leq i \leq h-1$ and $2 \leq j \leq w-1$. To compute the corresponding predicted feature vector $B_{i,j} \in \mathbb{R}^d$, we use PredictNet P which consists of two fully-connected layers with $2d$ and d output neurons, respectively. To predict $B_{i,j}$, we first concatenate 8-connected features of $A_{i,j}$ which is denoted as $e_{i,j} \in \mathbb{R}^{8d}$. Thus, $e_{i,j} = (A_{i-1,j-1}, ..., A_{i-1,j+1}, A_{i,j-1}, A_{i,j+1}, A_{i+1,j-1}, ..., A_{i+1,j+1})$. We forward $e_{i,j}$ through PredictNet to predict $B_{i,j}$. Now, we produce a hardness-to-predict matrix $Hard \in \mathbb{R}^{h \times w}$ which computes the prediction difficulty for each of the non-boundary patches. $Hard$ is computed as:

$$Hard_{i,j} = \begin{cases} d_{L1}(A_{i,j}, B_{i,j}), & \text{if } 2 \leq i \leq h-1 \; and \; 2 \leq j \leq w-1 \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

where $B_{i,j} = P(e_{i,j})$, and $d_{L1}(.)$ is the Mean Absolute Error (MAE) between two feature vectors. Now, we normalize $Hard$ matrix, and denote normalized matrix as $Hard'$. To avoid selecting boundary patch features later, we also replace boundary pixel values of $Hard'$ with the median of $Hard'$. For input image pair $\{X_n, X_s\}$, we denote corresponding feature map pair, predicted feature map pair, and normalized hardness matrix pair as $\{An, As\}$, $\{Bn, Bs\}$, and $\{HardN', HardS'\}$, respectively. Now, from $HardN'$ and $HardS'$ we generate following sets of feature vectors:

$$\begin{aligned} FG_n = \{An_{i,j} : HardN'_{i,j} \geq \tau_{fgn}\}; BG_n = \{An_{i,j} : HardN'_{i,j} \leq \tau_{bgn}\} \\ FG_s = \{As_{i,j} : HardS'_{i,j} \geq \tau_{fgs}\}; BG_s = \{As_{i,j} : HardS'_{i,j} \leq \tau_{bgs}\} \end{aligned} \tag{4.2}$$

We empirically set the value of τ_{fgn} and τ_{fgs} to 95th percentile of $HardN'$ and $HardS'$, respectively. And, we set the value of τ_{bgn} and τ_{bgs} to 5th percentile of $HardN'$ and

$HardS'$, respectively. In words, FG_n and BG_n set contain feature vectors from potential foreground (i.e., nuclei) and background patches of X_n, respectively. Similarly, FG_s and BG_s contain probable foreground and background patch features of scaled image X_s, respectively.

We now generate triplet samples $\{(a, p, n)\}$ from feature map pair $\{An, As\}$. We consider a, p and n as as anchor, a positive sample, and a negative sample, respectively, each of which is a d-dimensional feature vector. To generate a triplet sample (a, p, n) from input pair $\{X_n, X_s\}$, we randomly sample feature vector from FG_n, FG_s and $(BG_n \cup BG_s)$, respectively. Thus, a contains the feature vector of a non-background patch from unscaled image X_n, and p contains the feature vector of a non-background patch from probably-scaled image X_s. And, n contains the feature vector of a background patch from either X_n or X_s. We randomly generate m number of triplet samples for an input pair $\{X_n, X_s\}$. We set, m=32 in our experiments. We define our region-level triplet loss as:

$$L_{region}(X_n, X_s) = \frac{1}{m}\sum_{k=1}^{m} max(0, d_{L2}(a_k, p_k) - d_{L2}(a_k, n_k) + c) \tag{4.3}$$

where $d_{L2}(.)$ is the squared L_2 distance between two features, and c is the margin value which is empirically set to 1.0. Triplet loss encourages features from the same class to be located nearby, and pushes apart features from different classes in the embedding space [70, 16]. In other words, being pre-trained with proposed region-level triplet loss, the segmentation network can narrow down the distance between anchor and positive samples in the embedding space, and enlarges the semantic dissimilarity between the anchor and negative samples [93]. Note that, in the embedding space, we separate background and non-background patch features regardless of the corresponding scales of the patches of input image X_n and X_s. This design helps to map multi-scale nuclei features to be located nearby in the feature space. Similarly, multi-scale background

features are also mapped so that they are located far from nuclei features, and remain close to each other in the embedding space. Finally, for accurately predicting the feature vectors, we train PredictNet with following loss function:

$$L_{pred}(X_n, X_s) = \frac{1}{(h-2) \times (w-2)} \sum_{i=2}^{h-1} \sum_{j=2}^{w-1} (d_{L1}(An_{i,j}, Bn_{i,j}) + d_{L1}(As_{i,j}, Bs_{i,j})) \quad (4.4)$$

where $d_{L1}(.)$ is the Mean Absolute Error (MAE) between two features.

4.3.1.2 Scale loss

According to [68], looking at the size and texture of nuclei should be enough to determine the magnification level (i.e., scale) of input image, and this identification of the scale can generate a preliminary self-supervision signal to locate nuclei. Similar to [68], we compute the attention map J_s for input X_s with $J_s = \hat{Y}_s \odot X_s$, where $\hat{Y}_s$ is segmentation output for X_s, and $\odot$ represents element-wise multiplication. We use a scale classification network ScaleNet C to predict the scale from J_s. For C, we use ResNet-34 [31]. The output of C is a 5-dimensional vector v which gives the scores for each magnification level. Therefore, $v = C(J_s)$. We use negative log-likelihood to train ScaleNet C, and in turn the segmentation network S. Thus, our scale loss is defined as:

$$L_{scale}(X_s, z_s) = -log(p_l) \quad (4.5)$$

where l is the class label for z_s (i.e. index of z_s in S_p), and $p_l = [softmax(v)]_l$. Therefore, the total loss L_{PT} for pre-training TransNuSS is defined as:

$$L_{PT}(X_n, X_s) = L_{region}(X_n, X_s) + L_{pred}(X_n, X_s) + \lambda_{scale} L_{scale}(X_s, z_s) \quad (4.6)$$

where, λ_{scale} is the weight to balance scale loss which is empirically set to 0.5.

4.3.2 Fine-Tuning with annotated dataset

After pre-training, we fine-tune our segmentation network S with a small annotated dataset. For fine-tuning, S takes image X_t as input, and produces the segmentation prediction $\hat{Y_t}$ of the same size as output. We denote the ground-truth label by Y_t. In practice, we found dice-coefficient loss to be more effective than the binary cross-entropy loss for nuclei segmentation tasks. Therefore, we choose dice-coefficient loss as our supervised segmentation loss for fine-tuning:

$$L_{FT}(X_t) = 1 - \frac{2.Y_t'.\hat{Y_t}'}{Y_t' + \hat{Y_t}'}, \tag{4.7}$$

where Y_t' and $\hat{Y_t}'$ are flattened Y_t and $\hat{Y_t}$, respectively.

4.3.3 Implementations

We train TransNuSS with batch size 16, and using four GPUs. To train segmentation network, we use SGD optimizer with learning rate 0.01, momentum 0.9 and weight decay 0.0001. We use SGD optimizer with learning rate 0.001 and 0.0001 to train PredictNet and ScaleNet, respectively. We pre-train our model for 20 epochs, and then fine-tune for 80 epochs.

4.4 Experiments

4.4.1 Dataset

In this paper, we use MoNuSegWSI dataset for pre-training purposes. For fine-tuning the model, we use two datasets: 1) TNBC, and 2) MoNuSeg.

4.4.2 Experimental results

Experiment-1 In our first experiment, we fine-tune our pre-trained TransNuSS model with TNBC dataset. We choose FCN [54], U-Net [66], UNet++ [107] and

ResUNet-50 [21] as the representatives of Convolutional Neural Network (CNN) based approaches. SETR-MLA [104] and TransUnet [14] represent transformer-based semantic segmentation models. TransUnet + L_{drloc} shows the performance when auxiliary self-supervised localization loss [52] is utilized while training TransUNet. AttnSSL [68] and InstSSL [93] are chosen as representatives of Self-Supervised Learning (SSL) models for nuclei segmentation. We choose AttnSSL and InstSSL over the generic SSL models for two reasons: 1) AttnSSL and InstSSL were explicitly devised for nuclei segmentation problem, and 2) these two SSL methods perform significantly well for nuclei segmentation with better performance compared with generic self-supervised methods. We also employ TransUNet in InstSSL (i.e., replacing ResUNet backbone with TransUNet) model which is denoted as InstSSL-ViT in Table 4.1.

From Table 4.1, we see that our proposed TransNuSS model outperforms all other approaches in terms of IoU% and dice score. We also see that, our pre-trained model also achieves superiority over AttnSSL, and InstSSL without fine-tuning. The excellence of TransNuSS is mainly due to our proposed region-level triplet learning, which enables the segmentation network to separate nuclei from the backgrounds in a better manner in feature space. We also see that, MoNuSegWSI-pretrained and then fine-tuned InstSSL, InstSSL-ViT and TransNuSS models outperform ImageNet [19]-pretrained models, which proves the effectiveness of pre-training nuclei segmentation models with Whole Slide Image (WSI) patches. Figure 4.3 shows the visualization results of ResUNet-50, TransUNet, InstSSL and our proposed TransNuSS model. Figure 4.3 shows that, TransNuSS can significantly reduce false positive nuclei generated by other approaches. From Figure 4.3(f), we see that TransNuSS is capable to segment nuclei which were missed out by InstSSL model. Also, our intuition and assumption is that, $HardN'_{i,j}$ and $HardS'_{i,j}$ will have larger values if corresponding patch contains

		Experiment-1 TNBC dataset		**Experiment-2** MoNuSeg dataset	
Method	**Pre-trained on**	**IoU%**	**Dice score**	**IoU%**	**Dice score**
AttnSSL [68]	MoNuSegWSI	45.86	0.6018	59.93	0.7412
InstSSL w/o fine-tuning [93]	MoNuSegWSI	46.91	0.6136	61.05	0.7521
FCN [54]	-	63.01	0.7726	63.81	0.7803
U-Net [66]	-	64.65	0.7824	64.91	0.7982
UNet++ [107]	-	64.35	0.7813	65.38	0.7998
ResUNet-50 [21]	ImageNet	64.96	0.7863	65.79	0.8041
SETR-MLA [104]	ImageNet	64.87	0.7854	65.39	0.8021
TransUNet [14]	ImageNet	65.66	0.7905	66.02	0.8072
TransUNet + L_{drloc} [14, 52]	ImageNet	65.73	0.7894	66.63	0.8101
InstSSL [93]	ImageNet	64.68	0.7831	66.57	0.8112
InstSSL [93]	MoNuSegWSI	65.85	0.7942	67.92	0.8244
InstSSL-ViT [93, 14]	MoNuSegWSI	66.32	0.7991	68.11	0.8256
TransNuSS w/o fine-tuning	MoNuSegWSI	48.11	0.6252	63.43	0.7664
TransNuSS w/o L_{region}	MoNuSegWSI	66.28	0.7951	66.83	0.8147
TransNuSS w/o L_{scale}	MoNuSegWSI	66.72	0.8007	67.66	0.8236
TransNuSS (ours)	MoNuSegWSI	**67.02**	**0.8059**	**68.72**	**0.8307**

Table 4.1. Nuclei segmentation results for Experiment-1 and Experiment-2. IoU denotes intersection over union. Results are from testing on TNBC-test and MoNuSeg-test for experiment-1 and experiment-2, respectively.

nuclei (i.e., is non-background patch). Figure 4.4 shows the visualization of $HardN'$ matrices, which empirically validates our aforementioned intuition.

To understand the impact of each of the losses (i.e., region-level triplet loss, and scale loss), we also pre-train TransNuSS using a single (i.e., either triplet loss or scale loss) loss, and then we fine-tune the pre-trained model. From last three rows of Table 4.1, we see that the proposed TransNuSS outperforms both of TransNuSS w/o L_{region}, and TransNuSS w/o L_{scale}. The overall good performance of TransNuSS comes when both losses are applied together. In summary, both losses complement each other for the excellent performance of TransNuSS.

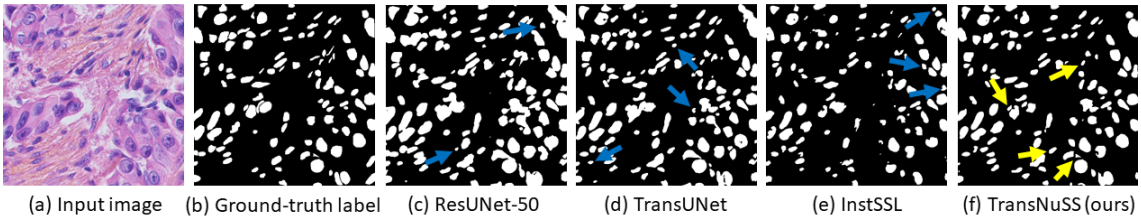

Figure 4.3. Visualization of the nuclei segmentation outputs of ResUNet-50 [21], TransUNet [14], InstSSL [93], and our proposed TransNuSS model. Input image is chosen from TNBC-test dataset. In (c)-(e), blue arrows indicate false positive nuclei that are removed in TransNuSS. In (f), yellow arrows denote missing nuclei from previous models.

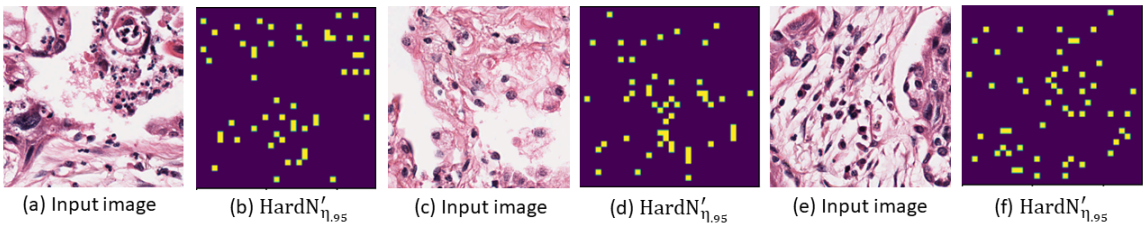

Figure 4.4. Visualizations of $HardN'$ matrices. Yellow color denotes values that are greater than or equal to 95th percentile, and we denote this matrix as $HardN'_{\eta.95}$ (a), (c) and (e) show input images, and (b), (d) and (f) show corresponding $HardN'_{\eta.95}$, respectively. Images are randomly chosen from MoNuSegWSI. In (b), (d) and (f), we zoom-in 32x32 matrices to 512x512 for a better visualization. (b), (d), and (f) show matrices from 1st, 3rd, and 5th epoch, respectively.

Experiment-2 We conduct second experiment with MoNuSeg dataset in the similar way to Experiment-1. This experiment again reflects the excellence of TransNuSS compared to other approaches (see last two columns of Table 4.1).

4.5 Conclusion

Due to a large domain gap between natural images and histology images, ImageNet-pretrained Vision Transformers (VT) does not transfer very well to nuclei segmentation tasks. In this paper, we propose Self-Supervised Learning (SSL)

based region-level triplet learning for pre-training so that VT implicitly learns to separate nuclei from the backgrounds. Prominent experimental results validate the effectiveness of our proposed TransNuSS model.

CHAPTER 5

TranSSCon: CONSISTENT SELF-SUPERVISED PRE-TRAINING FOR NUCLEI SEGMENTATION

For semantic segmentation of nuclei, Convolutional Neural Network (CNN) based approaches show excellent performances. However, the convolution operations of CNN models have limited receptive fields for context modeling, and thus can not model long-range dependencies (i.e., global context). Vision Transformer (VT), on the contrary, has the ability to model the global context at each stage of feature representation learning, and consequently VTs have the potentiality to outperform CNN based models. Usually, VT and CNN models are pre-trained with large-scale natural image dataset (i.e., ImageNet) in fully-supervised manner. However, pre-training nuclei segmentation models with ImageNet is not much helpful because of morphological and textural differences between natural image domain and medical image domain. Also, ImageNet-like large-scale annotated histology dataset rarely exists in medical image domain. In this chapter, we propose region-level, image-level and clustering-based Self-Supervised Learning (SSL) approach for pre-training semantic nuclei segmentation model with unannotated histology images extracted from Whole Slide Images (WSI). Unfortunately, due to the lack of annotations, SSL alone can not guarantee the consistency of the model while pre-training. To reduce disagreements among the predictions, we propose hierarchical, scale and transformation equivariance consistency losses. Thus, we introduce a simple yet effective combination of SSL approaches and consistency losses for pre-training semantic nuclei segmentation model.

5.1 Introduction

Histopathology image analysis is an important step for cancer recognition and diagnosis. Nuclei segmentation is considered as a fundamental task of digital histopathology image analysis [99]. For semantic segmentation of nuclei, Convolutional Neural Network (CNN) based approaches give very promising results [54, 66, 107, 28, 30]. However, CNN based nuclei segmentation methods have several limitations: 1) Due to the intrinsic locality nature and limited receptive fields of convolution operations, CNN based models are incapable of capturing the global context of the input [14, 104]. Thus, these approaches can not model the long-range dependency very well, and this may cause missing out some nuclei to segment (i.e., predicting false negatives). 2) Due to the lack of global context, CNN based methods often mistakenly consider the crowded objects as one connected region, and this may lead to under-segmentation of nuclei. 3) These models show limited transferability for target task (i.e., model trained on one type of organ may not work well on another ones) [14, 104].

Due to the mentioned drawbacks of CNN based models, we explore the feasibility of an alternative approach to solve the semantic nuclei segmentation problem. Transformers, an alternative to CNNs, are powerful at modeling the global context of input images [104]. Thus, if there are any inter-nucleus relationships in the given input image, transformers will be able to explore them and segment those nuclei accordingly. Also, transformers show superior transferability for downstream tasks, when pre-trained with large-scale dataset. So, if we have a large-scale nuclei segmentation pre-training dataset available, transformer based models may transfer better to the target dataset than other approaches even if the target dataset is small enough. Therefore, if properly designed, transformers have the potentiality to outperform CNN based models.

However, Vision Transformers (VT) need lot of data for training, usually more than what is necessary to standard CNNs [52]. Usually, VTs are pre-trained with large-scale annotated natural (i.e., generic) image dataset like ImageNet [19], and then fine-tuned to downstream tasks [23]. But, histology images are quite different from natural images due to the nuclei and background textures, morphological structures of nuclei, large variations in the shape and appearance of nuclei, clustered and overlapped nuclei, blurred nuclei boundaries, inconsistent staining methods, scanning artifacts, etc. [95, 56]. Due to this domain gap between natural images and medical images, the ImageNet pre-trained VT models may yield marginal improvement over train-from-scratch models for nuclei segmentation tasks [93]. As an alternative to ImageNet, we may think of pre-training nuclei segmentation VT models with large-scale annotated histology datasets. However, in medical image domain, ImageNet-like large-scale annotated histopathology image datasets do not exist, and unfortunately they are very difficult to produce, because of expensive, time-consuming and tedious labeling process of histology images [95, 12].

In this work, we propose a Transformer-based Self-Supervised Learning (SSL) approach for pre-training so that the segmentation network implicitly acquires a better understanding of the nuclei and background using a large-scale unannotated histology image dataset extracted from Whole Slide Images (WSI). In computer vision, SSL is used to learn useful data representations without using any labels [8, 60, 9, 109]. To achieve this goal, we first divide the image into $k \times k$ patches where k=32. Then, we try to predict each of the patch features from its 8-connected neighboring patches. Figure 5.1 shows that predicting non-background patches (i.e., that contain nuclei) is much harder than background patch prediction. Based on this aforementioned observation, we design region-level triplet learning and corresponding loss to pre-train the segmentation network. Our pre-trained SSL model learns to separate nuclei fea-

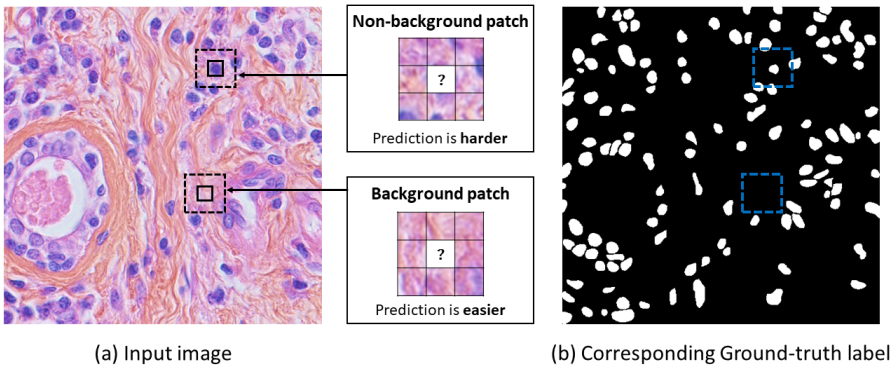

Figure 5.1. We divide the input image into $k \times k$ patches. We try to predict patch features using its 8-connected neighboring patches. We see that, predicting a non-background (i.e., nuclei) patch is much harder than predicting a background patch..

tures from the background features in the embedding space. Additionally, our SSL approach involves the image-level sub-task of predicting the scale of image, which enables the segmentation network to implicitly acquire further knowledge of nuclei size and shape. We also utilize a clustering loss at the feature space which serves as a constraint toward a class-conditional feature alignment.

However, SSL alone can not guarantee that the learned representations and segmentation predictions have the intra-image and inter-image consistency. Therefore, a challenging issue here is the inconsistent and uncertain predictions on the unannotated image dataset. To enforce invariant predictions over different layers of the segmentation network we propose hierarchical consistency enforcement and corresponding consistency loss. Specifically, the hierarchical consistency enforcement module predicts segmentation masks from different layers of the decoder, and minimizes the disagreements among predictions. Nevertheless, self-guided hierarchical consistency loss lacks supervision signal into it, and thus the model may still generate inconsistent predictions. To make the hierarchical predictions further consistent, we utilize supervised learning and propose scale consistency loss. This scale consistency loss

ensures that the predicted scale from different hierarchical predictions are consistent to each other. Hierarchical consistency loss and scale consistency loss reduce intra-image disagreements. To enforce inter-image consistency, we utilize transformation equivariance loss which guides to generate consistent predictions for the same image with probable cropping and scaling. Finally, after pre-training the model with SSL and consistency losses, we fine-tune the pre-trained network for nuclei segmentation with a small annotated dataset.

Thus, the main contributions of this paper are: **1)** We propose region-level, image-level, and clustering-based Self-Supervised Learning (SSL) approaches for pre-training semantic nuclei segmentation model with unannotated histology image dataset. **2)** To enforce intra-image and inter-image consistency while pre-training, we propose hierarchical, scale, and transformation consistency losses. **3)** We introduce a novel combination of SSL techniques and consistency losses for learning from large-scale unannotated histology dataset. **4)** We incorporate our proposed pre-training technique into a Vision Transformer (VT). To the best of our knowledge, this is the first work focusing on Transformer-based consistency-preserving SSL for semantic segmentation of nuclei. **5)** Extensive experimental results demonstrate the superiority of our proposed consistency-guaranteed SSL incorporated transformer model over baseline methods.

5.2 Related Work

Transformer was first designed for sequence-to-sequence prediction tasks. A solely attention mechanism based transformer model was proposed for English constituency parsing tasks [81, 27]. Later, Transformer has been employed in various computer vision problems [7, 23, 92, 14, 83, 97, 104, 63].

As a solution to loose the requirement of manual annotations for neural networks, Self-Supervised Learning (SSL) recently attracts increasing attentions from the community [93]. Several SSL approaches have been recently proposed for nuclei segmentation [93, 68]. In recent times, SSL also has been applied to Vision Transformers (VT) for generic image classification, segmentation, etc. [2, 50, 10, 52, 109]. Recently, for semantic segmentation of nuclei, [29] proposed a region-level SSL approach and corresponding triplet loss for pre-training the model with unannotated histology images extracted from Whole Slide Images (WSI).

In literature, for different weakly-supervised and semi-supervised models, the idea of consistency regularization has been successfully applied [51, 105, 84]. For weakly supervised cardiac segmentation, [102] tackled incomplete shape of scribbles by proposing the shape-consistency loss to regularize cutout equivalence and capture global shape of the heart. Reference [40] proposed a Mean-Teacher based hierarchical consistency enforcement framework with learnable and self-guided mechanisms for semi-supervised histological image segmentation.

5.3 Methodology

In semantic nuclei segmentation problem, we have nuclei histology image of size $H \times W \times 3$ as input, and we want to predict the segmentation output of size $H \times W$. We first pre-train our proposed model with unannotated image patches $D_s = \{(X_n)\}$. Then, we fine-tune the model with annotated images $D_t = \{(X_t, Y_t)\}$. In this work, since we propose Transformers-based Consistency-preserving Self-Supervised learning method for nuclei segmentation, we name our proposed framework as TranSSCon. Figure 5.2 shows the complete architecture of TranSSCon.

In our work, we adopt TransUNet [14] as the segmentation network S. The encoder of the segmentation network consists of a hybrid Convolutional Neural Net-

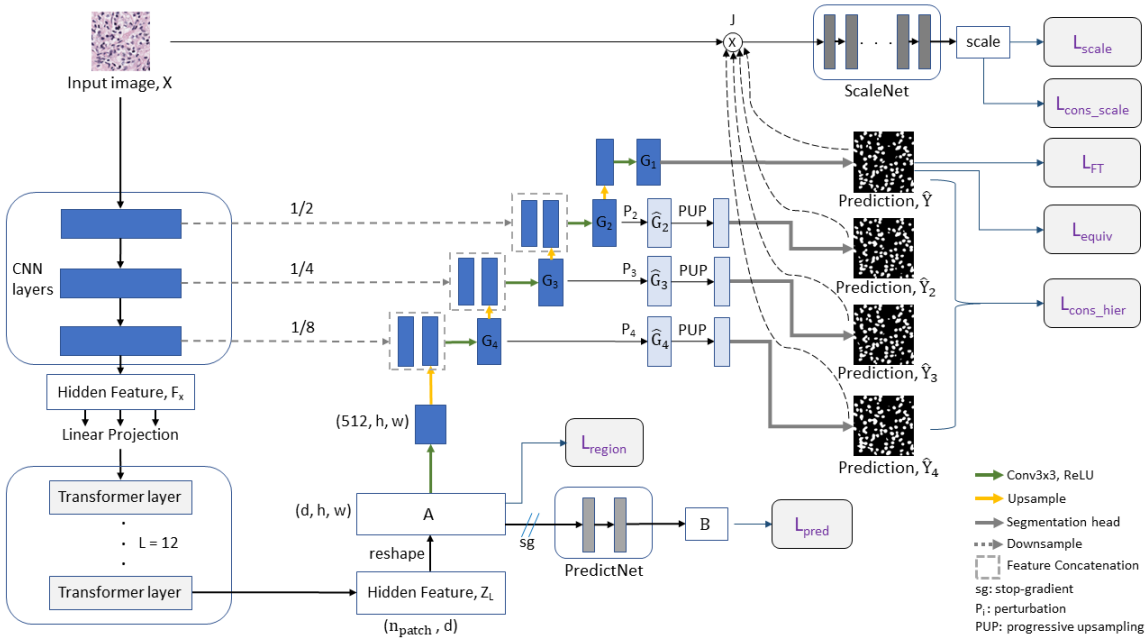

Figure 5.2. Complete architecture of TranSSCon..

work (CNN) - Transformer architecture. In encoder, we use ResNet-50 [31] and ViT [23] as CNN and transformer, respectively. The decoder of the segmentation network consists of multiple upsampling steps. We refer the reader to [14] for more details.

5.3.1 Pre-Training with large-scale unannotated dataset

For each image $X_n \in D_s$ of size $H \times W \times 3$, we generate a same-size image X_s by cropping and scaling. To generate X_s, we first randomly select a scaling-factor z_s from a pool $\{1.0, 1.25, 1.5, 1.75, 2.0\}$. We denote this scale-pool as S_p. Now, we randomly crop a patch from X_n, and then scale the cropped patch z_s times so that the scaled patch becomes of size $H \times W \times 3$. Thus, for pre-training, our input consists of $\{(X_n, X_s, z_s)\}$. The outputs of the segmentation model S are $\hat{Y}_n$ and $\hat{Y}_s$, which denote the segmentation predictions of X_n and X_s, respectively. Therefore, $\hat{Y}_n = S(X_n)$, and $\hat{Y}_s = S(X_s)$.

5.3.1.1 Region-level triplet loss

For self-supervised pre-training, we consider that we have background and non-background image patches (see Figure 5.1) in an input image. Then, we propose region-level triplet loss to learn the embedding space. We expect that, in the embedding space, feature should have similarity and dissimilarity among same and different types of patches, respectively. We design our triplet loss in a way so that the segmentation network can implicitly learn to separate background and non-background patch features in a given image. To generate triplet samples, we use feature map A which is of size $d \times h \times w$ where $h = \frac{H}{16}$ and $w = \frac{W}{16}$. Therefore, A contains $h \times w$ number (i.e., the number of patches) of d-dimensional feature vectors. As we mentioned before, our intuition and observation is that: in this embedding space, we can easily predict a background feature vector from its 8-connected neighboring features vectors, whereas predicting a non-background feature vector is comparatively harder.

We try to predict feature vector $A_{i,j} \in \mathbb{R}^d$ at each spatial location (i, j) of A, where $2 \leq i \leq h-1$ and $2 \leq j \leq w-1$. To compute the corresponding predicted feature vector $B_{i,j} \in \mathbb{R}^d$, we use PredictNet P which consists of two fully-connected layers with $2d$ and d output neurons, respectively. To predict $B_{i,j}$, we first concatenate 8-connected features of $A_{i,j}$ which is denoted as $e_{i,j} \in \mathbb{R}^{8d}$. Thus, $e_{i,j} = (A_{i-1,j-1}, ..., A_{i-1,j+1}, A_{i,j-1}, A_{i,j+1}, A_{i+1,j-1}, ..., A_{i+1,j+1})$. We forward $e_{i,j}$ through PredictNet to predict $B_{i,j}$. Now, we produce a hardness-to-predict matrix $Hard \in \mathbb{R}^{h \times w}$ which computes the prediction difficulty for each of the non-boundary patches. $Hard$ is computed as:

$$Hard_{i,j} = \begin{cases} d_{mae}(A_{i,j}, B_{i,j}), & \text{if } 2 \leq i \leq h-1 \textit{ and } 2 \leq j \leq w-1 \\ 0, & \text{otherwise} \end{cases} \tag{5.1}$$

where $B_{i,j} = P(e_{i,j})$, and $d_{mae}(.)$ is the Mean Absolute Error (MAE) between two feature vectors. Now, we normalize $Hard$ matrix, and denote normalized matrix as $Hard'$. To avoid selecting boundary patch features later, we also replace boundary pixel values of $Hard'$ with the median of $Hard'$. For input image pair $\{X_n, X_s\}$, we denote corresponding feature map pair, predicted feature map pair, and normalized hardness matrix pair as $\{An, As\}$, $\{Bn, Bs\}$, and $\{HardN', HardS'\}$, respectively. Now, from $HardN'$ and $HardS'$ we generate following sets of feature vectors:

$$\begin{aligned} FG_n = \{An_{i,j} : HardN'_{i,j} \geq \tau_{fgn}\}; BG_n = \{An_{i,j} : HardN'_{i,j} \leq \tau_{bgn}\} \\ FG_s = \{As_{i,j} : HardS'_{i,j} \geq \tau_{fgs}\}; BG_s = \{As_{i,j} : HardS'_{i,j} \leq \tau_{bgs}\} \end{aligned} \tag{5.2}$$

We empirically set the value of τ_{fgn} and τ_{fgs} to 95th percentile of $HardN'$ and $HardS'$, respectively. And, we set the value of τ_{bgn} and τ_{bgs} to 5th percentile of $HardN'$ and $HardS'$, respectively. In words, FG_n and BG_n set contain feature vectors from potential foreground (i.e., nuclei) and background patches of X_n, respectively. Similarly, FG_s and BG_s contain probable foreground and background patch features of scaled image X_s, respectively.

We now generate triplet samples $\{(a, p, n)\}$ from feature map pair $\{An, As\}$. We consider a, p and n as as anchor, a positive sample, and a negative sample, respectively, each of which is a d-dimensional feature vector. To generate a triplet sample (a, p, n) from input pair $\{X_n, X_s\}$, we randomly sample feature vector from FG_n, FG_s and $(BG_n \cup BG_s)$, respectively. Thus, a contains the feature vector of a non-background patch from unscaled image X_n, and p contains the feature vector of a non-background patch from probably-scaled image X_s. And, n contains the feature vector of a background patch from either X_n or X_s. We randomly generate m number

of triplet samples for an input pair $\{X_n, X_s\}$. We set, m=32 in our experiments. We define our region-level triplet loss as:

$$L_{region}(X_n, X_s) = \frac{1}{m}\sum_{k=1}^{m} max(0, d_{L2}(a_k, p_k) - d_{L2}(a_k, n_k) + c) \tag{5.3}$$

where $d_{L2}(.)$ is the squared L_2 distance between two features, and c is the margin value which is empirically set to 1.0.

Triplet loss encourages features from the same class to be located nearby, and pushes apart features from different classes in the embedding space [70, 16]. In other words, being pre-trained with proposed region-level triplet loss, the segmentation network can narrow down the distance between anchor and positive samples in the embedding space, and enlarges the semantic dissimilarity between the anchor and negative samples [93]. Note that, in the embedding space, we separate background and non-background patch features regardless of the corresponding scales of the patches of input image X_n and X_s. This design helps to map multi-scale nuclei features to be located nearby in the feature space. Similarly, multi-scale background features are also mapped so that they are located far from nuclei features, and remain close to each other in the embedding space.

Finally, for accurately predicting the feature vectors, we train PredictNet with following loss function:

$$L_{pred}(X_n, X_s) = \frac{1}{(h-2)\times(w-2)}\sum_{i=2}^{h-1}\sum_{j=2}^{w-1}(d_{mae}(An_{i,j}, Bn_{i,j}) + d_{mae}(As_{i,j}, Bs_{i,j})) \tag{5.4}$$

where $d_{mae}(.)$ is the Mean Absolute Error (MAE) between two features.

5.3.1.2 Scale loss

According to [68], looking at the size and texture of nuclei should be enough to determine the magnification level (i.e., scale) of input image, and this identification of the scale can generate a preliminary self-supervision signal to locate nuclei. Similar to [68], we compute the attended image J_s for input X_s with $J_s = \hat{Y}_s \odot X_s$, where $\hat{Y}_s$ is the segmentation output for X_s, and $\odot$ represents element-wise multiplication. We use a scale classification network ScaleNet C to predict the scale from J_s. For C, we use ResNet-34 [31]. The output of C is a 5-dimensional vector v which gives the scores for each magnification level. Therefore, $v = C(J_s)$. We use negative log-likelihood to train ScaleNet C, and in turn the segmentation network S. Thus, our scale loss is defined as:

$$L_{scale}(X_s, z_s) = -log(p_l) \tag{5.5}$$

where l is the class label for z_s (i.e., index of z_s in S_p), and $p_l = [softmax(v)]_l$.

In our segmentation network S, the segmentation head (i.e., classifier) consists of a convolutional operation followed by a sigmoid activation. Thus, the segmentation output $\hat{Y}_s = \sigma(y)$, where y is the output of convolutional operation in segmentation head. The scale loss L_{scale} considers the segmentation output $\hat{Y}_s$ as an attention map that focuses on the nuclei in the input image. In order to force the attention map to focus only on parts of the input image, we need to apply a sparsity regularizer on the segmentation prediction $\hat{Y}_s$ [68, 35]. Similar to [68], we impose the sparsity by picking the 93rd percentile value in y for all images in the batch. In other words, we assume that, on an average 7% of the pixels in an input patch represent nuclei. Thus,

we choose a threshold τ_{sparse} equal to the average of this percentile for all images in the training batch. Formally, τ_{sparse} is defined as:

$$\tau_{sparse} = \frac{1}{b}\sum_{i=1}^{b} y_i^{(\eta)} \tag{5.6}$$

where $y_i^{(\eta)}$ represents the 93rd percentile value in y_i for the i-th image in the training batch, and b is the batch size. Now, we define the sigmoid as $\sigma(y) = \frac{1}{1+exp(-r(y-\tau_{sparse}))}$. This sigmoid function is biased and compressed in order to force sharp transitions in the activated segmentation prediction $\hat{Y}_s$. The compression is determined by r, which we set to 20 in our experiments.

5.3.1.3 Clustering loss

The region-level triplet loss L_{region} guides TranSSCon to separate nuclei features from the background features in the embedding space by extracting and considering the features from two different images X_n and X_s. However, L_{region} does not align the same-class features of a single image, for which we additionally employ a clustering loss in TranSSCon at the feature space.

If ScaleNet C can correctly predict the scale of the attended scaled image J_s, we can consider the corresponding segmentation prediction $\hat{Y}_s$ as the pseudo-label. While pre-training TranSSCon, we apply a clustering loss at the feature-level to serve as a constraint toward a class-conditional feature alignment. Specifically, for the scaled image X_s, we have the corresponding feature map A_s. Then, the clustering loss is computed as:

$$L_{cl}(X_s) = q_s.\left\{\frac{1}{\mid A_s \mid}\sum_{a_i \in A_s, \tilde{y}_i \in \tilde{Y}_{s_ds}} d(a_i, c_{\tilde{y}_i}) - \frac{1}{\mid C \mid (\mid C \mid -1)}\sum_{j \in C}\sum_{k \in C, k \neq j} d(c_j, c_k)\right\} \tag{5.7}$$

where a_i is the feature vector corresponding to a spatial location of A_s, $\tilde{y}_i$ is the corresponding predicted class, and C is the set of semantic classes which is $\{0, 1\}$ for our nuclei segmentation problem. Here, q_s is equal to 1 if the scale of X_s is correctly predicted by ScaleNet C, and to 0 otherwise. We set the function d(.) to L1 norm. To compute $\tilde{y}_i$, the segmentation prediction $\hat{Y}_s$ is downsampled to match the spatial dimension of A_s and then it is converted to predicted class labels (i.e., pseudo-labels). We denote this downsampled pseudo-label by $\tilde{Y}_{s_ds}$. In Eq. (5.7), c_j denotes the centroid of semantic class j, which is computed using following formula:

$$c_j = \frac{\sum_{a_i} \sum_{\tilde{y}_i} \delta_{j,\tilde{y}_i} a_i}{\sum_{\tilde{y}_i} \delta_{j,\tilde{y}_i}}, j \in \{0, 1\} \tag{5.8}$$

where $\delta_{j,\tilde{y}_i}$ is equal to 1 if $\tilde{y}_i = j$, and to 0 otherwise.

5.3.1.4 Consistency loss

Pre-training with region-level triplet loss, scale loss and clustering loss trains TranSSCon to separate nuclei from the background in the embedding space. However, these losses does not consider any intra-image and inter-image prediction consistency in the output space. While pre-training TranSSCon with large scale unannotated dataset, we employ three consistency losses: (a) Hierarchical consistency loss, (b) Scale consistency loss, and (c) Transformation Equivariance loss. We discuss the details of these consistency losses in the following.

a) Hierarchical consistency loss TranSSCon consists of four decoder blocks (see Figure 5.2). To avoid the intra-image prediction disagreement among the unlabeled data, we regularize TranSSCon by a hierarchical consistency. We design Hierarchical Consistency Enforcement (HCE) module to constrain the consistency over the hierarchical outputs of the decoders.

Specifically, the i-th hierarchical representation G_i is first computed from the decoder at i-th block. As shown in Figure 5.2, the blocks are numbered from the deep to shallow ones. Similar to [100] and [40], we apply R stochastic perturbation functions, denoted by P_r with $r \in [1, R]$, to G_i. We randomly introduce the dropout and add noise as our perturbation operations P_r. Thus, the perturbed variant $\hat{G}_i = P_r(G_i)$. Then, instead of one-step upscaling which may introduce additional noisy predictions, we consider a Progressive UPsampling (PUP) strategy [104] that alternates convolutional layer and upsampling operations. While designing PUP layers, we restrict the upsampling to 2x. PUP upsamples the perturbed variant $\hat{G}_i$ to the size $H \times W \times 16$, from which the segmentation head generates the segmentation prediction $\hat{Y}_i$.

The HCE module can provide stronger enforcement among the unlabeled dataset. Besides, the consistency constraint over the perturbed variant representations add more generalization to the model while pre-training. In HCE module, we consider the final prediction (i.e., $\hat{Y}$) as the guidance, and minimize the inconsistency among all other decoders. We define the self-guided hierarchical consistency loss as:

$$L_{cons_hier}(X_n, X_s) = \frac{1}{hr-1}\sum_{i=2}^{hr} d_{mse}(\hat{Y}_{n_i}, \hat{Y}_n) + \frac{1}{hr-1}\sum_{i=2}^{hr} d_{mse}(\hat{Y}_{s_i}, \hat{Y}_s) \tag{5.9}$$

where hr represents the number of hierarchical blocks used in TranSSCon, and $d_{mse}(.)$ is the Mean Squared Error (MSE) between two segmentation predictions. $\hat{Y}_n$ and $\hat{Y}_s$ is the probability prediction from the main decoder for X_n and X_s, respectively. $\hat{Y}_{n_i}$ and $\hat{Y}_{s_i}$ denote the probability prediction from the i-th decoder block for X_n and X_s, respectively. Therefore, $\hat{Y}_n = \hat{Y}_{n_1}$, and $\hat{Y}_s = \hat{Y}_{s_1}$. In our experiments, we set $hr = 4$.

b) Scale consistency loss Although hierarchical consistency loss L_{cons_hier} reduces the disagreement among hierarchical segmentation predictions, we can not employ

any supervision into it. To make the hierarchical predictions more consistent through supervised learning, we add scale consistency loss to TranSSCon.

For $2 \leq i \leq 4$, we first compute the attended image J_{s_i} for input X_s with $J_{s_i} = \hat{Y}_{s_i} \odot X_s$. Here, $\hat{Y}_{s_i}$ is the segmentation prediction of X_s generated from i-th decoder block, and $\odot$ represents element-wise multiplication. Then, we use ScaleNet C to predict the scale from J_{s_i}. Therefore, for $2 \leq i \leq 4$, we get $v_i = C(J_{s_i})$, and then we compute $p_{l_i} = [softmax(v_i)]_l$ where l is the class label for z_s (i.e., index of z_s in S_p). Now, we define the scale consistency loss as:

$$L_{cons_scale}(X_s, z_s) = \frac{1}{hr-1} \sum_{i=2}^{hr} -log(p_{l_i}) \tag{5.10}$$

where hr is the number of hierarchical blocks which we set to 4 in our experiments. We backpropagate the scale consistency loss only through our segmentation network (i.e., skipping the updating of ScaleNet parameters) while pre-training, so that the segmentation network can learn to generate consistent predictions from different hierarchies being guided by a supervision signal.

c) Transformation Equivariance loss While pre-training TranSSCon, we add an equivariance constraint on the segmentation prediction of original image X_n and corresponding scaled image X_s. As previously described, to generate X_s from X_n, we first randomly select a scaling-factor z_s, then randomly crop a patch x_n from X_n, and scale the cropped patch z_s times to make x_n of size $H \times W \times 3$. We assume that, the top-left corner of x_n is located at (r, c) spatial coordinates in X_n, and x_n has an equal height and width of l. We denote this cropping and then z_s-times scaling operations together by the transformation operation $t_{(r,c,l,z_s)}(.)$. Therefore, $X_s = t_{(r,c,l,z_s)}(X_n)$.

While pre-training, we want the segmentation prediction $\hat{Y}_s$ to be equivariant to the prediction obtained from similarly cropping from $\hat{Y}_n$ and then scaling it, where $\hat{Y}_s = S(X_s)$ and $\hat{Y}_n = S(X_n)$ (i.e., the segmentation prediction for X_s and X_n,

respectively). In other words, we want $S(X_s) = S(t_{(r,c,l,z_s)}(X_n)) = t_{(r,c,l,z_s)}(S(X_n))$. In TranSSCon, we define the transformation equivariance loss as:

$$L_{equiv}(X_n, X_s, z_s) = d_{mse}(t_{(r,c,l,z_s)}(S(X_n)), S(X_s)) \tag{5.11}$$

where $d_{mse}(.)$ is the Mean Squared Error (MSE) between two segmentation predictions.

Therefore, the total loss L_{PT} for pre-training TranSSCon is defined as:

$$\begin{aligned} &L_{PT}(X_n, X_s, z_s) = L_{region}(X_n, X_s) + L_{pred}(X_n, X_s) + \lambda_{scale} L_{scale}(X_s, z_s) + \\ &\lambda_{cl} L_{cl}(X_s) + \lambda_{cons_hier} L_{cons_hier}(X_n, X_s) + \lambda_{cons_scale} L_{cons_scale}(X_s, z_s) + \\ &\lambda_{equiv} L_{equiv}(X_n, X_s, z_s) \end{aligned} \tag{5.12}$$

where, λ_{scale}, λ_{cl}, λ_{cons_hier}, λ_{cons_scale} and λ_{equiv} are the weights to balance corresponding losses. We empirically set λ_{scale}, λ_{cl}, λ_{cons_hier}, λ_{cons_scale} and λ_{equiv} to 0.5, 0.1, 0.2, 0.2 and 0.2, respectively.

5.3.2 Fine-Tuning with annotated dataset

After pre-training, we fine-tune our segmentation network S with a small annotated dataset. For fine-tuning, S takes image X_t as input, and produces the segmentation prediction $\hat{Y_t}$ of the same size as output. We denote the ground-truth label by Y_t. In practice, we found dice-coefficient loss to be more effective than the binary cross-entropy loss for nuclei segmentation tasks. Therefore, we choose dice-coefficient loss as our supervised segmentation loss for fine-tuning:

$$L_{FT}(X_t) = 1 - \frac{2.Y_t'.\hat{Y_t}'}{Y_t' + \hat{Y_t}'}, \tag{5.13}$$

where Y_t' and $\hat{Y_t}'$ are flattened Y_t and $\hat{Y_t}$, respectively.

5.3.3 Implementations

To train segmentation network, we use SGD optimizer with learning rate 0.01, momentum 0.9 and weight decay 0.0001. We use SGD optimizer with learning rate 0.001, 0.0001 and 0.0001 to train PredictNet, ScaleNet and PUP, respectively. We pre-train our model for 20 epochs, and then fine-tune for 80 epochs. We implement TranSSCon using PyTorch [61]. We train TranSSCon with batch size 16, and using four GPUs.

5.4 Experiments

5.4.1 Dataset

In our experiments, we use MoNuSegWSI dataset for pre-training purposes. For fine-tuning the model, we use two datasets: 1) TNBC, and 2) MoNuSeg.

5.4.2 Experimental results

5.4.2.1 TranSSCon

Experiment-1 In our first experiment, we fine-tune our pre-trained TranSSCon model with TNBC dataset. We choose ResUNet-50 [21] as the representative of Convolutional Neural Network (CNN) based approaches. TransUNet [14] represents transformer-based semantic segmentation models. TransUNet + L_{drloc} shows the performance when auxiliary self-supervised localization loss [52] is utilized while training TransUNet. AttnSSL [68], InstSSL [93] and TransNuSS [29] are chosen as representatives of Self-Supervised Learning (SSL) models for nuclei segmentation. We choose AttnSSL, InstSSL and TransNuSS over the generic SSL models for two reasons: 1) AttnSSL, InstSSL and TransNuSS were explicitly devised for nuclei segmentation problem, and 2) these three SSL methods perform significantly well for nuclei seg-

		Experiment-1 TNBC dataset		**Experiment-2** MoNuSeg dataset	
Method	**Pre-trained on**	**IoU%**	**Dice**	**IoU%**	**Dice**
AttnSSL	MoNuSegWSI	45.86	0.6018	59.93	0.7412
InstSSL w/o fine-tuning	MoNuSegWSI	46.91	0.6136	61.05	0.7521
TransNuSS w/o fine-tuning	MoNuSegWSI	48.11	0.6252	63.43	0.7664
ResUNet-50	ImageNet	64.96	0.7863	65.79	0.8041
TransUNet	ImageNet	65.66	0.7905	66.02	0.8072
TransUNet + L_{drloc}	ImageNet	65.73	0.7894	66.63	0.8101
InstSSL	ImageNet	64.68	0.7831	66.57	0.8112
InstSSL	MoNuSegWSI	65.85	0.7942	67.92	0.8244
InstSSL-ViT	MoNuSegWSI	66.32	0.7991	68.11	0.8256
TransNuSS	MoNuSegWSI	67.02	0.8059	68.72	0.8307
TranSSCon w/o fine-tuning	MoNuSegWSI	48.75	0.6304	63.71	0.7697
TranSSCon (ours)	MoNuSegWSI	**67.83**	**0.8109**	**69.19**	**0.8337**

Table 5.1. Nuclei segmentation results for Experiment-1 and Experiment-2. IoU and Dice denotes Intersection over Union, and Dice score, respectively. Results are from testing on TNBC-test and MoNuSeg-test for experiment-1 and experiment-2, respectively.

mentation with better performance compared with generic self-supervised methods. We also employ TransUNet in InstSSL (i.e., replacing ResUNet backbone with TransUNet) model which is denoted by InstSSL-ViT in Table 5.1. In our experiments, we choose Intersection-over-Union (IoU) and Dice score as the evaluation matrices.

From Table 5.1, we see that our proposed TranSSCon model outperforms all other approaches in terms of IoU% and dice score. Our pre-trained model (see the second last row in Table 5.1) also achieves superiority over AttnSSL, and InstSSL and TransNuSS without fine-tuning. The excellence of TranSSCon is mainly due to our proposed combination of the consistency losses (i.e., hierarchical, scale, and transformation equivariance) with region-level and image-level SSL strategies, which enables the segmentation network to separate nuclei from the backgrounds in a better

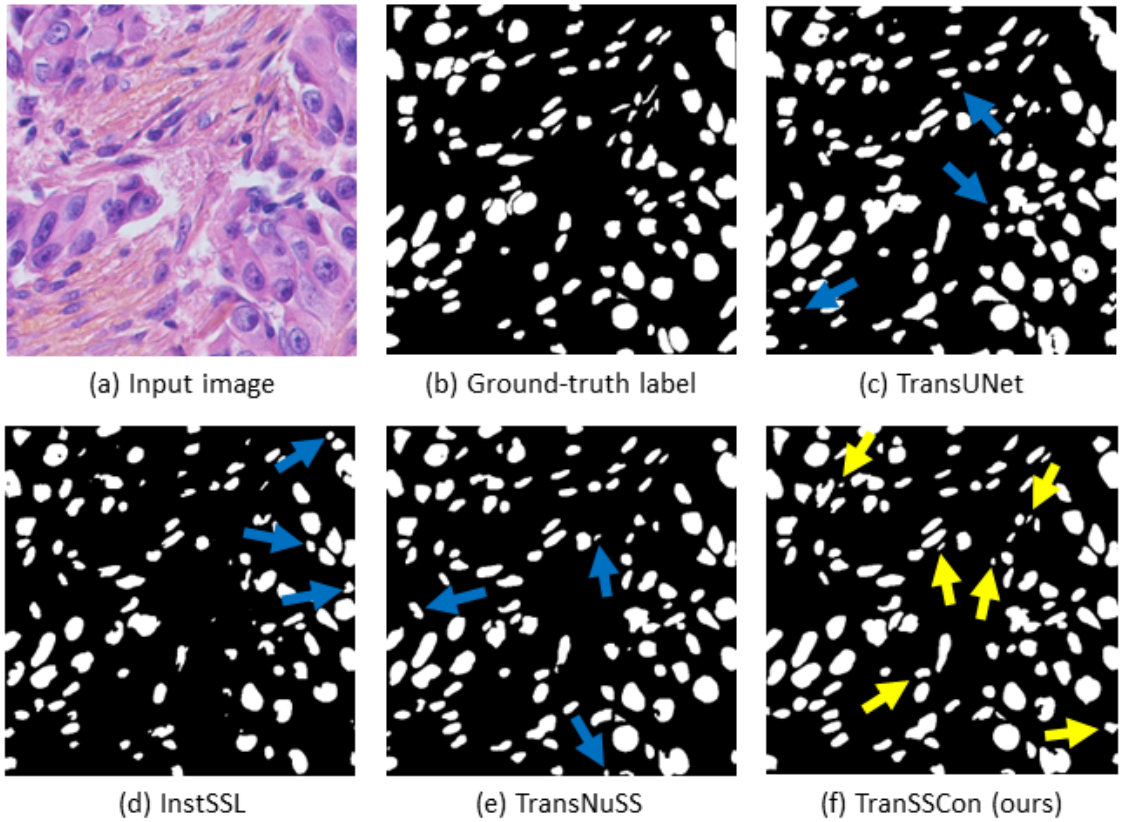

Figure 5.3. Visualization of the nuclei segmentation outputs of TransUNet [14], InstSSL [93], TransNuSS [29], and our proposed TranSSCon model. Input image is chosen from TNBC-test dataset. In (c)-(e), blue arrows indicate false positive nuclei that are removed in TranSSCon. In (f), yellow arrows denote missing nuclei from previous models..

manner in feature space while reducing inter-image and intra-image disagreements. From Table 5.1, we also see that MoNuSegWSI-pretrained and then fine-tuned InstSSL, InstSSL-ViT, TransNuSS and TranSSCon models outperform ImageNet[19]-pretrained models, which proves the effectiveness of pre-training nuclei segmentation models with Whole Slide Image (WSI) patches.

Figure 5.3 shows the visualization results of TransUNet, InstSSL, TranNuSS, and our proposed TranSSCon model. Figure 5.3 shows that TranSSCon can significantly reduce false positive nuclei generated by other approaches. From Figure 5.3(f),

we see that TranSSCon is capable to segment nuclei which were missed out by other models.

Experiment-2 We conduct second experiment with MoNuSeg dataset in the similar way to Experiment-1. This experiment again reflects the excellence of TranSSCon compared to other approaches (see last two columns of Table 5.1).

5.5 Conclusion

Accurate nuclei segmentation is a significant step for cancer diagnosis and further clinical procedures. For semantic segmentation of nuclei, Vision Transformers (VT) have the potentiality to outperform Convolutional Neural Network (CNN) based models due to their ability to model long-range dependencies. But, VTs need lot of annotated data for training, which is highly unavailable in biomedical domain. Moreover, due to a large domain gap between natural images and histology images, ImageNet-pretrained VT does not transfer very well to nuclei segmentation tasks. In this paper, we first propose Self-Supervised Learning (SSL) based region-level triplet learning, image-level scale loss, and clustering loss for pre-training so that VT implicitly learns to separate nuclei from the backgrounds. We then propose hierarchical consistency loss, scale consistency loss, and transformation equivariance loss to reduce the disagreements at both of output space and and feature space while pre-training. Consistency losses helps the model to preserve consistency among different layers, and different views of the same image. We combine the proposed consistency losses with SSL strategies for pre-training nuclei segmentation model with large-scale unannotated histology dataset. Prominent experimental results validate the effectiveness of our proposed TranSSCon model.

CHAPTER 6

DiffNuSS: DIFFUSION MODEL BASED SELF-SUPERVISED PRE-TRAINING FOR NUCLEI SEGMENTATION

The convolution operations of CNN models have limited receptive fields for context modeling, and thus can not model long-range dependencies (i.e., global context). Vision Transformer (VT), on the contrary, has the ability to model the global context at each stage of feature representation learning, and consequently VTs have the potentiality to outperform CNN based models. Usually, VT and CNN models are pre-trained with large-scale natural image dataset (i.e., ImageNet) in fully-supervised manner. However, pre-training nuclei segmentation models with ImageNet is not much helpful because of morphological and textural differences between natural image domain and medical image domain. Also, ImageNet-like large-scale annotated histology dataset rarely exists in medical image domain. In this chapter, we propose Denoising Diffusion Probabilistic Model (DDPM) based Self-Supervised Learning (SSL) approach for pre-training semantic nuclei segmentation model with unannotated histology images extracted from Whole Slide Images (WSI). We feed-forward the DDPM outputs (i.e., estimated noise) to a generation module for predicting the segmentation mask. Since DDPM are capable of extracting powerful and discriminative features via generative pre-training for dense prediction tasks, we combine SSL with DDPM. To pre-train the model for generating realistic segmentation masks and acquiring knowledge of nuclei, we employ a discriminator and scale loss, respectively. Thus, we introduce a simple yet effective combination of DDPM, generation

module, discriminator, and scale loss for label-efficient pre-training of semantic nuclei segmentation model.

6.1 Introduction

Histopathology image analysis is an important step for cancer recognition and diagnosis. Nuclei segmentation is considered as a fundamental task of digital histopathology image analysis [99]. For semantic segmentation of nuclei, Convolutional Neural Network (CNN) based approaches give very promising results [54, 66, 107, 28, 30]. However, CNN based nuclei segmentation methods have several limitations: 1) Due to the intrinsic locality nature and limited receptive fields of convolution operations, CNN based models are incapable of capturing the global context of the input [14, 104]. Thus, these approaches can not model the long-range dependency very well, and this may cause missing out some nuclei to segment (i.e., predicting false negatives). 2) Due to the lack of global context, CNN based methods often mistakenly consider the crowded objects as one connected region, and this may lead to under-segmentation of nuclei. 3) These models show limited transferability for target task (i.e., model trained on one type of organ may not work well on another ones) [14, 104].

Due to the mentioned drawbacks of CNN based models, we explore the feasibility of an alternative approach to solve the semantic nuclei segmentation problem. Transformers, an alternative to CNNs, are powerful at modeling the global context of input images [104]. Thus, if there are any inter-nucleus relationships in the given input image, transformers will be able to explore them and segment those nuclei accordingly. Also, transformers show superior transferability for downstream tasks, when pre-trained with large-scale dataset. So, if we have a large-scale nuclei segmentation pre-training dataset available, transformer based models may transfer better to the target dataset than other approaches even if the target dataset is small enough.

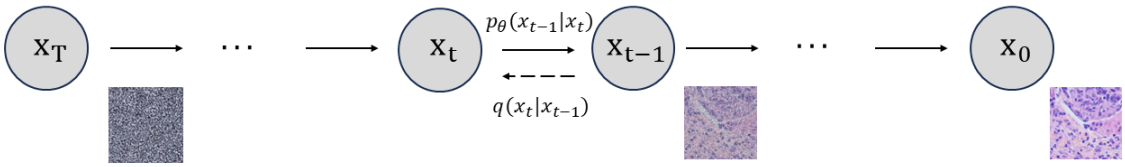

Figure 6.1. Denoising Diffusion Probabilistic Model (DDPM).

Therefore, if properly designed, transformers have the potentiality to outperform CNN based models.

However, Vision Transformers (VT) need lot of data for training, usually more than what is necessary to standard CNNs [52]. Usually, VTs are pre-trained with large-scale annotated natural (i.e., generic) image dataset like ImageNet [19], and then fine-tuned to downstream tasks [23]. In literature, for semantic segmentation problem, pre-training is often used to improve the label-efficiency of segmentation models [6]. However, histology images are quite different from natural images due to the nuclei and background textures, morphological structures of nuclei, large variations in the shape and appearance of nuclei, clustered and overlapped nuclei, blurred nuclei boundaries, inconsistent staining methods, scanning artifacts, etc. [95, 56]. Due to this domain gap between natural images and medical images, the ImageNet pre-trained VT models may yield marginal improvement over train-from-scratch models for nuclei segmentation tasks [93]. As an alternative to ImageNet, we may think of pre-training nuclei segmentation VT models with large-scale annotated histology datasets. However, in medical image domain, ImageNet-like large-scale annotated histopathology image datasets do not exist, and unfortunately they are very difficult to produce, because of expensive, time-consuming and tedious labeling process of histology images [95, 12].

In this work, we propose a Transformer-based Self-Supervised Learning (SSL) approach for pre-training so that the segmentation network implicitly acquires a better understanding of the nuclei and background using a large-scale unannotated histology image dataset extracted from Whole Slide Images (WSI). In computer vision, SSL is used to learn useful data representations without using any labels [8, 60, 9, 109]. To advance the efficacy further, here we combine SSL with Denoising Diffusion Probabilistic Model (DDPM). The essential idea of DDPM is to systematically and slowly destroy structure in a data distribution through an iterative forward diffusion process, and then learn a reverse diffusion process that restores structure in data [74, 32, 42]. DDPM is basically a generative Markov chain which converts a simple known distribution (e.g., a Gaussian) into a target (data) distribution using a diffusion process [74, 32]. Figure 6.1 shows the forward and reverse diffusion process of DDPM. Generally, in DDPM (or, Diffusion Models, for brevity), noise is added to clean data and model is trained to separate the noisy data back into clean data and noise components. This strategy requires the model to learn the data distribution. Denoising objectives are well-suited for training dense prediction models (e.g., semantic segmentation, etc.) because they can be defined on a per-pixel level [6]. Additionally, denoising models are powerful deep generative models that can obtain better sample quality than state-of-the-arts GANs [20, 26, 3]. Diffusion models are also able to acquire discriminative representations for classification via generative pre-training, and this generative pre-training can enhance the label utilization of semantic segmentation models [91, 98, 4]. The latent representation learned by diffusion models is found to be useful in discriminative tasks (e.g., image segmentation, classification and anomaly detection) [18]. Moreover, Gaussian-based denoising is compatible with convolutional networks and Vision Transformers.

In this paper, at first we utilize denoising models to extract powerful discriminative feature from the unannotated histology images while pre-training. Then, we pre-train a generation module with the those meaningful features. Since the diffusion models estimate the noise added to the perturbed input data, and the adversarial learning model generated images for given the noisy vectors, we can reasonably connect the diffusion model with the adversarial learning. We use a discriminator to guide the segmentation prediction networks so the predicted mask becomes very similar to the ground-truth annotations. Additionally, our SSL approach involves the image-level sub-task of predicting the scale of image, which enables the segmentation network to implicitly acquire further knowledge of nuclei size and shape.

Thus, the main contributions of this paper are: **1)** We propose a novel Diffusion model based Self-Supervised Learning (SSL) approach for pre-training semantic nuclei segmentation model with large-scale unannotated histology image dataset. To the best of our knowledge, this is the first work focusing on utilizing diffusion models for pre-training nuclei segmentation models without using any annotations. **2)** We introduce a novel combination of Denoising models, Generation module, and and scale loss for label-efficient SSL. **3)** We incorporate Vision Transformer (VT) into our proposed pre-training technique. **4)** Extensive experimental results demonstrate the superiority of our proposed DDPM incorporated SSL approach over baseline methods.

6.2 Related Work

Transformer was first designed for sequence-to-sequence prediction tasks. A solely attention mechanism based transformer model was proposed for English constituency parsing tasks [81, 27]. Later, Transformer has been employed in various computer vision problems [7, 23, 92, 14, 83, 97, 104, 63].

As a solution to loose the requirement of manual annotations for neural networks, Self-Supervised Learning (SSL) recently attracts increasing attentions from the community [93]. Several SSL approaches have been recently proposed for nuclei segmentation [93, 68]. In recent times, SSL also has been applied to Vision Transformers (VT) for image classification, segmentation, etc. [2, 50, 10, 52, 109, 29].

In literature, diffusion models have been used for a variety of computer vision problems [15, 3, 1, 4, 6, 82, 48]. Diffusion models also have been successfully utilized in medical imaging problems [85, 75, 69, 62, 87, 88, 59, 72, 86, 43, 76]. MedSegDiff [87] proposed diffusion model based medical image segmentation method utilizing dynamic conditional encoding and feature frequency parser. MedSegDiff-V2 [88] enhances MedSegDiff by incorporating the transformer mechanism into the original UNet backbone. DiffMix [59] proposes a data augmentation technique using a conditioned diffusion model for imbalanced pathology nuclei datasets. Diffusion Adversarial Representation Learning (DARL) [43] combines diffusion models with adversarial learning, and applies it to self-supervised vessel segmentation without ground-truth labels.

6.3 Methodology

In semantic nuclei segmentation problem, we have nuclei histology image of size $H \times W \times 3$ as input, and we want to predict the segmentation output of size $H \times W$. We first pre-train our proposed model with unannotated image patches $D_u = \{(x^u)\}$. Then, we fine-tune the model with annotated images $D_l = \{(x^l, y^l)\}$. In this work, since we propose Diffusion model based Self-Supervised learning method for Nuclei segmentation, we name our proposed framework as DiffNuSS. Figure 6.2 shows the complete architecture of DiffNuSS.

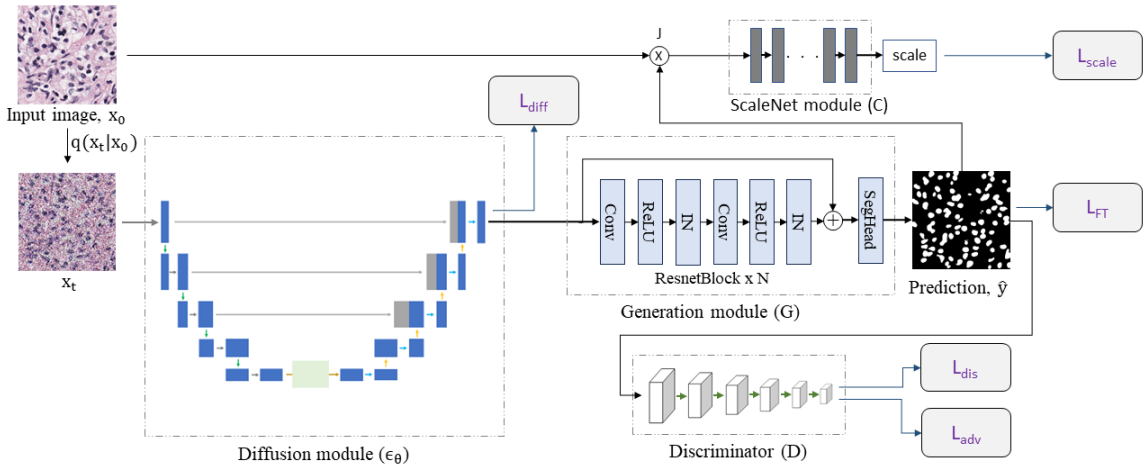

Figure 6.2. Complete architecture of DiffNuSS..

6.3.1 Pre-Training with large-scale unannotated dataset

For each image $x^u \in D_u$ of size $H \times W \times 3$, we generate a same-size image x^s by cropping and scaling. To generate x^s, we first randomly select a scaling-factor z^s from a pool $\{1.0, 1.25, 1.5, 1.75, 2.0\}$. We denote this scale-pool as S_p. Now, we randomly crop a patch from x^u, and then scale the cropped patch z^s times so that the scaled patch becomes of size $H \times W \times 3$. Thus, for pre-training, our input consists of $\{(x^s, z^s)\}$.

As shown in Figure 6.2, DiffNuSS is comprised of four modules: a diffusion model ϵ_θ to estimate the latent features, a generation module G to predict the segmentation masks, a discriminator D to distinguish real and fake images of the segmentation masks, and scale loss. We discuss the details of each modules in the following.

6.3.1.1 Diffusion model

In literature, Denoising Diffusion Probabilistic Models (or, Diffusion Models for brevity) [74, 32] transform noise $x_T \sim N(0, I)$ to the sample x_0 by gradually denoising x_T to less noisy samples x_t. Formally, the forward diffusion process is is defined as:

$$q(x_t \mid x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta I), \tag{6.1}$$

for some fixed variance schedule $\beta_1, ..., \beta_t$. We can obtain a noisy sample x_t from the data x_0 by:

$$q(x_t \mid x_0) := \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t} x_0, (1-\bar{\alpha}_t) I) \tag{6.2}$$

We can simplify Eq. (6.2) as:

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \epsilon \sim N(0, 1) \tag{6.3}$$

where $\alpha_t := 1 - \beta_t, \bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$.

Pre-trained diffusion models approximates a reverse process:

$$p_\theta(x_{t-1} \mid x_t) := \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sum_\theta (x_t, t). \tag{6.4}$$

The denoising model is typically parameterized by different variants of UNet [66] architecture. In DiffNuSS, we adopt TransUNet [14] as the diffusion model ϵ_θ.

In DiffNuSS, the diffusion model can be considered as a noise predictor $\epsilon_\theta(x_t, t)$ which predicts the noise component at the step t. While pretraining, we are given input image x^s. From the perspective of diffusion models, we consider that $x_0^s = x^s$. Now, using the forward diffusion process from Eq. (6.3), we sample the noisy image x_t^s from x_0^s. Here, t is uniformly sampled time step in $[0, T]$. We set $T = 2000$ in our experiments.

Since the diffusion model ϵ_θ learns the distribution of images to estimate meaningful latent features of the inputs, we use following standard loss for training diffusion model in DiffNuSS:

$$L_{diff}(x^s) := \mathbb{E}_{t,x_0,\epsilon}[\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0^s + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2] \tag{6.5}$$

where $\epsilon \sim N(0, I)$.

6.3.1.2 Generation module

The generation module takes the latent feature $\epsilon_\theta(x_t^s, t)$ from the noisy image x_t^s as its input, and predicts the segmentation mask. Therefore, the output of the generation module, $\hat{y}^s = G(\epsilon_\theta(x_t^s, t))$. In DiffNuSS, the generation module G consists of N residual blocks (ResnetBlock). ResnetBlock consists of convolutional layer, ReLU activation, and instance normalization (also known as contrast normalization) [80] layer. After ResnetBlocks, there is a Segmentation Head in generation module, which consists of a convolutional layer and outputs the segmentation prediction.

6.3.1.3 Discriminator

According to [28, 30], ground-truth labels for nuclei segmentation are domain-invariant. While pretraining DiffNuSS, to generate realistic nuclei segmentation masks without using any ground-truth labels, DiffNuSS is trained by adversarial learning using a discriminator D. The discriminator D tries to distinguish the estimated segmentation masks from the real masks (i.e., the ground-truth masks from the fine-tuning dataset). We define the adversarial loss as:

$$L_{adv}(x^s) = -\frac{1}{H_p \times W_p} \sum_{h_p, w_p} \log\left(D(M_\tau(\hat{y}^s))\right) \tag{6.6}$$

where $\hat{y}^s = G(\epsilon_\theta(x_t^s, t))$, M_τ is binarization operator with a fixed threshold τ, and H_p and W_p are height and width of the discriminator output. This adversarial loss helps the noise predictor and generator module to fool the discriminator so that it considers $M_\tau(\hat{y^s})$ as real (i.e., very similar to the ground-truths) segmentation masks. In other words, the adversarial loss L_{adv} guides DiffNuSS to generate unannotated images predictions $\hat{y}^s$ which looks similar to the annotated images ground-truths.

The discriminator D takes unannoated image binarized prediction or annotated image ground-truths as input, and then distinguishes whether the input (i.e., prediction) looks like fake or real. To train D, we use following cross-entropy loss:

$$L_{dis}(\hat{y}) = -\frac{1}{H_p \times W_p} \sum_{h_p, w_p} z.\log\left(D(\hat{y})\right) + (1-z).\log\left(1 - D(M_\tau(\hat{y}))\right) \tag{6.7}$$

where z=0 when D takes unannoated image prediction as its input, and z=1 when the input comes from annotated images ground-truths.

6.3.1.4 Scale loss

According to [68], looking at the size and texture of nuclei should be enough to determine the magnification level (i.e., scale) of input image, and this identification of the scale can generate a preliminary self-supervision signal to locate nuclei. Similar to [68], we compute the attended image j^s for input x^s with $j^s = \hat{y^s} \odot x^s$, where $\hat{y^s}$ is the segmentation output for x^s, and $\odot$ represents element-wise multiplication. We use a scale classification network ScaleNet C to predict the scale from j^s. For C, we use ResNet-34 [31]. The output of C is a 5-dimensional vector v which gives the scores for each magnification level. Therefore, $v = C(j^s)$. We use negative log-likelihood to train ScaleNet C, and in turn the noise estimator and the generation module. Thus, our scale loss is defined as:

$$L_{scale}(x^s, z^s) = -log(p_l) \tag{6.8}$$

where l is the class label for z^s (i.e., index of z^s in S_p), and $p_l = [softmax(v)]_l$.

In our generation network G, the segmentation head (i.e., classifier) consists of a convolutional operation followed by a sigmoid activation. Thus, the segmentation output $\hat{y^s} = \sigma(y)$, where y is the output of convolutional operation in segmentation head. The scale loss L_{scale} considers the segmentation output $\hat{y^s}$ as an attention map that focuses on the nuclei in the input image. In order to force the attention map to focus only on parts of the input image, we need to apply a sparsity regularizer on the segmentation prediction $\hat{y^s}$ [68, 35]. Similar to [68], we impose the sparsity by picking the 93rd percentile value in y for all images in the batch. In other words, we assume that, on an average 7% of the pixels in an input patch represent nuclei. Thus, we choose a threshold τ_{sparse} equal to the average of this percentile for all images in the training batch. Formally, τ_{sparse} is defined as:

$$\tau_{sparse} = \frac{1}{b}\sum_{i=1}^{b} y_i^{(\eta)} \tag{6.9}$$

where $y_i^{(\eta)}$ represents the 93rd percentile value in y_i for the i-th image in the training batch, and b is the batch size. Now, we define the sigmoid as $\sigma(y) = \frac{1}{1+exp(-r(y-\tau_{sparse}))}$. This sigmoid function is biased and compressed in order to force sharp transitions in the activated segmentation prediction $\hat{y^s}$. The compression is determined by r, which we set to 20 in our experiments.

Therefore, the total loss L_{PT} for pre-training DiffNuSS is defined as:

$$L_{PT}(x^s, z^s) = L_{diff}(x^s) + \lambda_{adv} L_{adv}(x^s) + \lambda_{scale} L_{scale}(x^s, z^s) \tag{6.10}$$

where, λ_{adv} and λ_{scale} are the weights to balance corresponding losses. We empirically set λ_{adv} and λ_{scale} to 0.001, and 0.5, respectively.

6.3.2 Fine-Tuning with annotated dataset

After pre-training, we fine-tune DiffNuSS with a small annotated dataset. For fine-tuning, DiffNuSS takes image x^l as input, and produces the segmentation prediction $\hat{y^l}$ of the same size as output. We denote the ground-truth label by y^l. In practice, we found dice-coefficient loss to be more effective than the binary cross-entropy loss for nuclei segmentation tasks. Therefore, we choose dice-coefficient loss as our supervised segmentation loss for fine-tuning:

$$L_{FT}(x_l) = 1 - \frac{2.y'.\hat{y'}}{y' + \hat{y'}}, \tag{6.11}$$

where y' and $\hat{y'}$ are flattened y^l and $\hat{y^l}$, respectively.

6.3.3 Implementations

We use SGD optimizer with learning rate 0.0001, 0.0001, and 0.001 to train generation module, ScaleNet, and discriminator, respectively. Following DCGAN [64], we designed our prediction discriminator and image discriminator consisting of five convolutional layers. We pre-train our model for 20 epochs, and then fine-tune for 80 epochs. We implement DiffNuSS using PyTorch [61]. We train DiffNuSS with batch size 16, and using four GPUs.

6.4 Experiments

6.4.1 Dataset

In this paper, we use MoNuSegWSI dataset for pre-training purposes. For fine-tuning the model, we use two datasets: 1) TNBC, and 2) MoNuSeg.

6.4.2 Experimental results

Experiment-1 In our first experiment, we fine-tune our pre-trained DiffNuSS model with TNBC dataset. We choose ResUNet-50 [21] as the representative of Convolutional Neural Network (CNN) based approaches. TransUNet [14] represents transformer-based semantic segmentation models. TransUNet + L_{drloc} shows the performance when auxiliary self-supervised localization loss [52] is utilized while training TransUNet. Decoder Denoising Pretraining (DDeP) [6] and Label-Efficient Semantic Segmentation (LESS) [4] models represent generic semantic segmentation models pretrained using diffusion techniques. We also choose MedSegDiff-V2 [88] as a representative of diffusion model based medical image segmentation framework. AttnSSL [68], InstSSL [93] and TransNuSS [29] are chosen as representatives of Self-Supervised Learning (SSL) models for nuclei segmentation. We choose AttnSSL, InstSSL and TransNuSS over the generic SSL models for two reasons: 1) AttnSSL, InstSSL and TransNuSS were explicitly devised for nuclei segmentation problem, and 2) these three SSL methods perform significantly well for nuclei segmentation with better performance compared with generic self-supervised methods. We also employ TransUNet in InstSSL (i.e., replacing ResUNet backbone with TransUNet) model which is denoted by InstSSL-ViT in Table 6.1. In our experiments, we choose Intersection-over-Union (IoU) and Dice score as the evaluation matrices.

From Table 6.1, we see that our proposed DiffNuSS model outperforms all other approaches in terms of IoU% and dice score. Our pre-trained model (see the second last row in Table 6.1) also achieves superiority over AttnSSL, and InstSSL and TransNuSS without fine-tuning. The excellence of DiffNuSS is mainly due to our proposed combination of diffusion models, generation module and scale loss, which enables the diffusion models extract features in a way so the whole network learns to separate nuclei from the backgrounds in a better manner. From Table 6.1, we also see

		Experiment-1 TNBC dataset		Experiment-2 MoNuSeg dataset	
Method	**Pre-trained on**	**IoU%**	**Dice**	**IoU%**	**Dice**
AttnSSL	MoNuSegWSI	45.86	0.6018	59.93	0.7412
InstSSL w/o fine-tuning	MoNuSegWSI	46.91	0.6136	61.05	0.7521
TransNuSS w/o fine-tuning	MoNuSegWSI	48.11	0.6252	63.43	0.7664
ResUNet-50	ImageNet	64.96	0.7863	65.79	0.8041
LESS	MoNuSegWSI	57.56	0.7271	61.47	0.7597
DDeP	MoNuSegWSI	65.12	0.7881	61.86	0.7624
MedSegDiff-V2	ImageNet	65.45	0.7893	61.14	0.7527
TransUNet	ImageNet	65.66	0.7905	66.02	0.8072
TransUNet + L_{drloc}	ImageNet	65.73	0.7894	66.63	0.8101
InstSSL	ImageNet	64.68	0.7831	66.57	0.8112
InstSSL	MoNuSegWSI	65.85	0.7942	67.92	0.8244
InstSSL-ViT	MoNuSegWSI	66.32	0.7991	68.11	0.8256
TransNuSS	MoNuSegWSI	67.02	0.8059	68.72	0.8307
DiffNuSS w/o fine-tuning	MoNuSegWSI	48.43 ± 0.09	0.6289 ± 0.0011	63.52 ± 0.12	0.7691 ± 0.0015
DiffNuSS (ours)	MoNuSegWSI	**67.74 ± 0.11**	**0.8105 ± 0.0008**	**69.03 ± 0.14**	**0.8325 ± 0.0012**

Table 6.1. Nuclei segmentation results for Experiment-1 and Experiment-2. IoU and Dice denotes Intersection over Union, and Dice score, respectively. Results are from testing on TNBC-test and MoNuSeg-test for experiment-1 and experiment-2, respectively. For DiffNuss, we report the Mean and Stdev. obtained over five runs.

that MoNuSegWSI-pretrained and then fine-tuned InstSSL, InstSSL-ViT, TransNuSS and DiffNuSS ImageNet[19]-pretrained models, which proves the effectiveness of pre-training nuclei segmentation models with Whole Slide Image (WSI) patches.

Figure 6.3 shows the visualization results of TransUNet, InstSSL, LESS, DDeP, TranNuSS, and our proposed DiffNuSS model. Figure 6.3 shows that DiffNuSS can significantly reduce false positive nuclei generated by other approaches. From Figure 6.3(h), we see that DiffNuSS is capable to segment nuclei which were missed out by other models.

Experiment-2 We conduct second experiment with MoNuSeg dataset in the similar way to Experiment-1. This experiment again reflects the excellence of DiffNuSS compared to other approaches (see last two columns of Table 6.1).

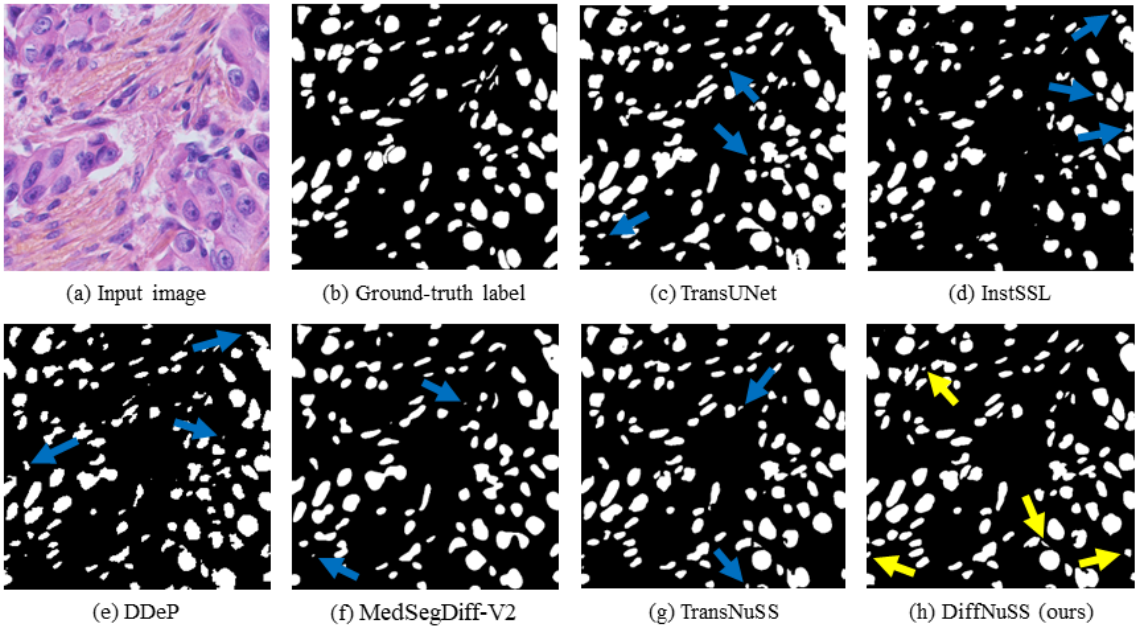

Figure 6.3. Visualization of the nuclei segmentation outputs of TransUNet [14], InstSSL [93], DDeP [6], MedSegDiff-V2 [88], TransNuSS [29], and our proposed DiffNuSS model. Input image is chosen from TNBC-test dataset. In (c)-(g), blue arrows indicate false positive nuclei that are removed in DiffNuSS. In (h), yellow arrows denote missing nuclei from previous models..

6.5 Conclusion

Accurate nuclei segmentation is a significant step for cancer diagnosis and further clinical procedures. For semantic segmentation of nuclei, Vision Transformers (VT) have the potentiality to outperform Convolutional Neural Network (CNN) based models due to their ability to model long-range dependencies. But, VTs need lot of annotated data for training, which is highly unavailable in biomedical domain. Moreover, due to a large domain gap between natural images and histology images, ImageNet-pretrained VT does not transfer very well to nuclei segmentation tasks. In this paper, we propose Denoising Diffusion Probabilistic Model (DDPM) based Self-Supervised Learning (SSL) approach for pre-training so that VT learns to ex-

tract powerful features and utilize them for dense prediction task. In addition to DDPM, we utilize a generation module to predict the segmentation mask, and additionally use a discriminator and scale loss. The discriminator pre-trains the model to generate masks similar to the ground truths, while scale loss provides a preliminary self-supervision signal to the model to locate nuclei. Prominent experimental results validate the effectiveness of our proposed DiffNuSS model.

CHAPTER 7

CONCLUSIONS

The accurate segmentation of nuclei is crucial for cancer diagnosis and further clinical treatments. To successfully train fully-supervised Convolutional Neural Network (CNN) or Vision Transformer (VT) models, we need at least a few amount of annotated data. Unfortunately, such well-annotated histopathology datasets, even if very small-sized, are highly rare. Therefore, due to high unavailability of annotated nuclei segmentation dataset and tedious labeling process, we require to discover a way for training nuclei segmentation models with unlabeled datasets.

In this thesis, I present my work towards solving this critical problem by utilizing Adversarial Learning, Self-Supervised Learning (SSL), and Diffusion Models. Thus, my approaches can be summarized into three directions: 1) employing adversarial learning based unsupervised and semi-supervised domain adaptation techniques to solve nuclei segmentation problem for unannotated datasets; 2) proposing SSL based approaches for pre-training VT models with unannotated image dataset; 3) introducing Denoising Diffusion Probabilistic Model (DDPM) based approach for pre-training nuclei segmentation model.

In the first approach, I apply Unsupervised Domain Adaptation (UDA) and Semi-Supervised Domain Adaptation (SSDA) with the help of another labeled dataset that may come from another organs or sources. Later, I extend the model by utilizing an adversarial learning incorporated reconstruction network to translate the source-domain images to the target domain for further training. Then, in my second approach, I introduce a novel region-level SSL based framework for pre-training

semantic nuclei segmentation model with a large-scale unannotated histopathology image dataset extracted from Whole Slide Images (WSI). Additionally, I propose hierarchical, scale, and transformation equivariance loss to reduce the disagreements among predictions. Finally, in the third approach, I utilize DDPM for extracting discriminative and powerful features. Then, I combine a generation module, a discriminator, and scale loss with DDPM for effective label-efficient SSL based pre-training. Extensive and comprehensive experiments demonstrate the superiority of the proposed methods over the baseline models.

In conclusion, I expect the techniques described in this thesis to be very useful in other biomedical image segmentation tasks. On the future directions, I would like to devise approaches for utilizing the limited annotations to the fullest. In future, my next move would be to generate pseudo ground-truth masks for the unannotated images (e.g., target domain images, pre-training dataset, etc.) for further training. Also, for pre-training models, it might be very interesting to combine the proposed DDPM-based model with the consistency losses.

Bibliography

[1] Tomer Amit, Eliya Nachmani, Tal Shaharbany, and Lior Wolf. Segdiff: Image segmentation with diffusion probabilistic models. *arXiv preprint arXiv:2112.00390*, 2021.

[2] Sara Atito, Muhammad Awais, and Josef Kittler. Sit: Self-supervised vision transformer. *arXiv preprint arXiv:2104.03602*, 2021.

[3] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22669–22679, 2023.

[4] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021.

[5] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE transactions on pattern analysis and machine intelligence*, 26(9):1124–1137, 2004.

[6] Emmanuel Asiedu Brempong, Simon Kornblith, Ting Chen, Niki Parmar, Matthias Minderer, and Mohammad Norouzi. Denoising pretraining for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4175–4186, 2022.

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[8] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021.

[11] Numan Celik, Sharib Ali, Soumya Gupta, Barbara Braden, and Jens Rittscher. Endouda: a modality independent segmentation approach for endoscopy imaging. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 303–312. Springer, 2021.

[12] Cheng Chen, Qi Dou, Hao Chen, Jing Qin, and Pheng-Ann Heng. Synergistic image and feature adaptation: Towards cross-modality domain adaptation for medical image segmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 865–872, 2019.

[13] Cheng Chen, Qi Dou, Hao Chen, Jing Qin, and Pheng-Ann Heng. Synergistic image and feature adaptation: Towards cross-modality domain adaptation for medical image segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 865–872, 2019.

[14] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*, 2021.

[15] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. *arXiv preprint arXiv:2211.09788*, 2022.

[16] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 403–412, 2017.

[17] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1791–1800, 2019.

[18] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[19] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[20] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[21] Foivos I Diakogiannis, François Waldner, Peter Caccetta, and Chen Wu. Resunet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 162:94–114, 2020.

[22] Nanqing Dong, Michael Kampffmeyer, Xiaodan Liang, Zeya Wang, Wei Dai, and Eric Xing. Unsupervised domain adaptation for automatic estimation of cardiothoracic ratio. In *International conference on medical image computing and computer-assisted intervention*, pages 544–552. Springer, 2018.

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

[24] Amir Gholami, Shashank Subramanian, Varun Shenoy, Naveen Himthani, Xiangyu Yue, Sicheng Zhao, Peter Jin, George Biros, and Kurt Keutzer. A novel domain adaptation framework for medical image segmentation. In *International MICCAI Brainlesion Workshop*, pages 289–298. Springer, 2019.

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[26] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[27] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on visual transformer. *arXiv preprint arXiv:2012.12556*, 2020.

[28] Mohammad Minhazul Haq and Junzhou Huang. Adversarial domain adaptation for cell segmentation. In *Medical Imaging with Deep Learning*, pages 277–287. PMLR, 2020.

[29] Mohammad Minhazul Haq and Junzhou Huang. Self-supervised pre-training for nuclei segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 303–313. Springer, 2022.

[30] Mohammad Minhazul Haq, Hehuan Ma, and Junzhou Huang. Nusegda: Domain adaptation for nuclei segmentation. *Frontiers in Big Data*, 6, 2023.

[31] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[32] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[33] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017.

[34] Le Hou, Ayush Agarwal, Dimitris Samaras, Tahsin M Kurc, Rajarsi R Gupta, and Joel H Saltz. Robust histopathology image analysis: to label or to synthe-

size? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8533–8542, 2019.

[35] Le Hou, Vu Nguyen, Ariel B Kanevsky, Dimitris Samaras, Tahsin M Kurc, Tianhao Zhao, Rajarsi R Gupta, Yi Gao, Wenjin Chen, David Foran, et al. Sparse autoencoder for unsupervised nucleus detection and representation in histopathology images. *Pattern recognition*, 86:188–200, 2019.

[36] Yuankai Huo, Zhoubing Xu, Shunxing Bao, Albert Assad, Richard G Abramson, and Bennett A Landman. Adversarial synthesis learning enables segmentation without target modality ground truth. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 1217–1220. IEEE, 2018.

[37] Humayun Irshad, Laleh Montaser-Kouhsari, Gail Waltz, Octavian Bucur, JA Nowak, Fei Dong, Nicholas W Knoblauch, and Andrew H Beck. Crowdsourcing image annotation for nucleus detection and segmentation in computational pathology: evaluating experts, automated methods, and the crowd. In *Pacific symposium on biocomputing Co-chairs*, pages 294–305. World Scientific, 2014.

[38] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[39] Haozhe Jia, Haoteng Tang, Guixiang Ma, Weidong Cai, Heng Huang, Liang Zhan, and Yong Xia. Psgr: Pixel-wise sparse graph reasoning for covid-19 pneumonia segmentation in ct images. *arXiv preprint arXiv:2108.03809*, 2021.

[40] Qiangguo Jin, Hui Cui, Changming Sun, Jiangbin Zheng, Leyi Wei, Zhenyu Fang, Zhaopeng Meng, and Ran Su. Semi-supervised histological image segmen-

tation via hierarchical consistency enforcement. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part II*, pages 3–13. Springer, 2022.

[41] Konstantinos Kamnitsas, Christian Baumgartner, Christian Ledig, Virginia Newcombe, Joanna Simpson, Andrew Kane, David Menon, Aditya Nori, Antonio Criminisi, Daniel Rueckert, et al. Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. In *International conference on information processing in medical imaging*, pages 597–609. Springer, 2017.

[42] Amirhossein Kazerouni, Ehsan Khodapanah Aghdam, Moein Heidari, Reza Azad, Mohsen Fayyaz, Ilker Hacihaliloglu, and Dorit Merhof. Diffusion models for medical image analysis: A comprehensive survey. *arXiv preprint arXiv:2211.07804*, 2022.

[43] Boah Kim, Yujin Oh, and Jong Chul Ye. Diffusion adversarial representation learning for self-supervised vessel segmentation. *arXiv preprint arXiv:2209.14566*, 2022.

[44] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[45] Neeraj Kumar, Ruchika Verma, Sanuj Sharma, Surabhi Bhargava, Abhishek Vahadane, and Amit Sethi. A dataset and a technique for generalized nuclear segmentation for computational pathology. *IEEE transactions on medical imaging*, 36(7):1550–1560, 2017.

[46] Neeraj Kumar, Ruchika Verma, Sanuj Sharma, Surabhi Bhargava, Abhishek Vahadane, and Amit Sethi. A dataset and a technique for generalized nu-

clear segmentation for computational pathology. *IEEE transactions on medical imaging*, 36(7):1550–1560, 2017.

[47] Neeraj Kumar, Ruchika Verma, Deepak Anand, Yanning Zhou, Omer Fahri Onder, Efstratios Tsougenis, Hao Chen, Pheng-Ann Heng, Jiahui Li, Zhiqiang Hu, et al. A multi-organ nucleus segmentation challenge. *IEEE transactions on medical imaging*, 39(5):1380–1391, 2019.

[48] Zeqiang Lai, Yuchen Duan, Jifeng Dai, Ziheng Li, Ying Fu, Hongsheng Li, Yu Qiao, and Wenhai Wang. Denoising diffusion semantic segmentation with mask prior modeling. *arXiv preprint arXiv:2306.01721*, 2023.

[49] Chaoqun Li, Yitian Zhou, Tangqi Shi, Yenan Wu, Meng Yang, and Zhongyu Li. Unsupervised domain adaptation for the histopathological cell segmentation through self-ensembling. In *MICCAI Workshop on Computational Pathology*, pages 151–158. PMLR, 2021.

[50] Chunyuan Li, Jianwei Yang, Pengchuan Zhang, Mei Gao, Bin Xiao, Xiyang Dai, Lu Yuan, and Jianfeng Gao. Efficient self-supervised vision transformers for representation learning. *arXiv preprint arXiv:2106.09785*, 2021.

[51] Jia Li, Yu Rong, Hong Cheng, Helen Meng, Wenbing Huang, and Junzhou Huang. Semi-supervised graph classification: A hierarchical graph perspective. In *The World Wide Web Conference*, pages 972–982, 2019.

[52] Yahui Liu, Enver Sangineto, Wei Bi, Nicu Sebe, Bruno Lepri, and Marco Nadai. Efficient training of visual transformers with small datasets. *Advances in Neural Information Processing Systems*, 34, 2021.

[53] 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*, pages 3431–3440, 2015.

[54] 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*, pages 3431–3440, 2015.

[55] Faisal Mahmood, Daniel Borders, Richard Chen, Gregory N McKay, Kevan J Salimian, Alexander Baras, and Nicholas J Durr. Deep adversarial training for multi-organ nuclei segmentation in histopathology images. *IEEE transactions on medical imaging*, 2019.

[56] Faisal Mahmood, Daniel Borders, Richard J Chen, Gregory N McKay, Kevan J Salimian, Alexander Baras, and Nicholas J Durr. Deep adversarial training for multi-organ nuclei segmentation in histopathology images. *IEEE transactions on medical imaging*, 39(11):3257–3267, 2019.

[57] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[58] Peter Naylor, Marick Laé, Fabien Reyal, and Thomas Walter. Segmentation of nuclei in histopathology images by deep regression of the distance map. *IEEE transactions on medical imaging*, 38(2):448–459, 2018.

[59] Hyun-Jic Oh and Won-Ki Jeong. Diffmix: Diffusion model-based data synthesis for nuclei segmentation and classification in imbalanced pathology image datasets. *arXiv preprint arXiv:2306.14132*, 2023.

[60] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[61] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[62] Walter HL Pinaya, Mark S Graham, Robert Gray, Pedro F Da Costa, Petru-Daniel Tudosiu, Paul Wright, Yee H Mah, Andrew D MacKinnon, James T Teo, Rolf Jager, et al. Fast unsupervised brain anomaly detection and segmentation with diffusion models. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 705–714. Springer, 2022.

[63] Tim Prangemeier, Christoph Reich, and Heinz Koeppl. Attention-based transformers for instance segmentation of cells in microstructures. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 700–707. IEEE, 2020.

[64] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[65] Ashwin Raju, Zhanghexuan Ji, Chi Tung Cheng, Jinzheng Cai, Junzhou Huang, Jing Xiao, Le Lu, ChienHung Liao, and Adam P Harrison. User-guided domain adaptation for rapid annotation from user interactions: a study on pathological liver segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 457–467. Springer, 2020.

[66] 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*, pages 234–241. Springer, 2015.

[67] 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*, pages 234–241. Springer, 2015.

[68] Mihir Sahasrabudhe, Stergios Christodoulidis, Roberto Salgado, Stefan Michiels, Sherene Loi, Fabrice André, Nikos Paragios, and Maria Vakalopoulou. Self-supervised nuclei segmentation in histopathological images using attention. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 393–402. Springer, 2020.

[69] Pedro Sanchez, Antanas Kascenas, Xiao Liu, Alison Q O'Neil, and Sotirios A Tsaftaris. What is healthy? generative counterfactual diffusion for lesion localization. In *MICCAI Workshop on Deep Generative Models*, pages 34–44. Springer, 2022.

[70] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[71] Yash Sharma, Sana Syed, and Donald E Brown. Mani: Maximizing mutual information for nuclei cross-domain unsupervised segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 345–355. Springer, 2022.

[72] Aman Shrivastava and P Thomas Fletcher. Nasdm: Nuclei-aware semantic histopathology image generation using diffusion models. *arXiv preprint arXiv:2303.11477*, 2023.

[73] Korsuk Sirinukunwattana, Shan E Ahmed Raza, Yee-Wah Tsang, David RJ Snead, Ian A Cree, and Nasir M Rajpoot. Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE transactions on medical imaging*, 35(5):1196–1206, 2016.

[74] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[75] David Stojanovski, Uxio Hermida, Pablo Lamata, Arian Beqiri, and Alberto Gomez. Echo from noise: synthetic ultrasound image generation using diffusion models for real image segmentation. *arXiv preprint arXiv:2305.05424*, 2023.

[76] Fenghe Tang, Jianrui Ding, Lingtao Wang, Min Xian, and Chunping Ning. Multi-level global context cross consistency model for semi-supervised ultrasound image segmentation with diffusion model. *arXiv preprint arXiv:2305.09447*, 2023.

[77] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *arXiv preprint arXiv:2009.06732*, 2020.

[78] Marco Toldo, Umberto Michieli, and Pietro Zanuttigh. Unsupervised domain adaptation in semantic segmentation via orthogonal and clustered embeddings. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1358–1368, 2021.

[79] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7472–7481, 2018.

[80] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6924–6932, 2017.

[81] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[82] Hefeng Wang, Jiale Cao, Rao Muhammad Anwer, Jin Xie, Fahad Shahbaz Khan, and Yanwei Pang. Dformer: Diffusion-guided transformer for universal image segmentation. *arXiv preprint arXiv:2306.03437*, 2023.

[83] Xiyue Wang, Sen Yang, Jun Zhang, Minghui Wang, Jing Zhang, Junzhou Huang, Wei Yang, and Xiao Han. Transpath: Transformer-based self-supervised learning for histopathological image classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 186–195. Springer, 2021.

[84] Zicheng Wang, Zhen Zhao, Xiaoxia Xing, Dong Xu, Xiangyu Kong, and Luping Zhou. Conflict-based cross-view consistency for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19585–19595, 2023.

[85] Julia Wolleb, Florentin Bieder, Robin Sandkühler, and Philippe C Cattin. Diffusion models for medical anomaly detection. In *International Conference on Medical image computing and computer-assisted intervention*, pages 35–45. Springer, 2022.

[86] Julia Wolleb, Robin Sandkühler, Florentin Bieder, Philippe Valmaggia, and Philippe C Cattin. Diffusion models for implicit image segmentation ensembles. In *International Conference on Medical Imaging with Deep Learning*, pages 1336–1348. PMLR, 2022.

[87] Junde Wu, Huihui Fang, Yu Zhang, Yehui Yang, and Yanwu Xu. Medsegdiff: Medical image segmentation with diffusion probabilistic model. *arXiv preprint arXiv:2211.00611*, 2022.

[88] Junde Wu, Rao Fu, Huihui Fang, Yu Zhang, and Yanwu Xu. Medsegdiff-v2: Diffusion based medical image segmentation with transformer. *arXiv preprint arXiv:2301.11798*, 2023.

[89] Xide Xia and Brian Kulis. W-net: A deep model for fully unsupervised image segmentation. *arXiv preprint arXiv:1711.08506*, 2017.

[90] Yingda Xia, Dong Yang, Zhiding Yu, Fengze Liu, Jinzheng Cai, Lequan Yu, Zhuotun Zhu, Daguang Xu, Alan Yuille, and Holger Roth. Uncertainty-aware multi-view co-training for semi-supervised medical image segmentation and domain adaptation. *Medical Image Analysis*, 65:101766, 2020.

[91] Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. *arXiv preprint arXiv:2303.09769*, 2023.

[92] Enze Xie, Wenjia Wang, Wenhai Wang, Mingyu Ding, Chunhua Shen, and Ping Luo. Segmenting transparent objects in the wild. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 696–711. Springer, 2020.

[93] Xinpeng Xie, Jiawei Chen, Yuexiang Li, Linlin Shen, Kai Ma, and Yefeng Zheng. Instance-aware self-supervised learning for nuclei segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 341–350. Springer, 2020.

[94] Yan Xu, Zhipeng Jia, Liang-Bo Wang, Yuqing Ai, Fang Zhang, Maode Lai, I Eric, and Chao Chang. Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features. *BMC bioinformatics*, 18(1):281, 2017.

[95] Yan Xu, Zhipeng Jia, Liang-Bo Wang, Yuqing Ai, Fang Zhang, Maode Lai, I Eric, and Chao Chang. Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features. *BMC bioinformatics*, 18(1):1–17, 2017.

[96] Jinyu Yang, Chunyuan Li, Weizhi An, Hehuan Ma, Yuzhi Guo, Yu Rong, Peilin Zhao, and Junzhou Huang. Exploring robustness of unsupervised domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9194–9203, 2021.

[97] Jinyu Yang, Jiali Duan, Son Tran, Yi Xu, Sampath Chanda, Liqun Chen, Belinda Zeng, Trishul Chilimbi, and Junzhou Huang. Vision-language pre-training with triple contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15671–15680, 2022.

[98] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Yingxia Shao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *arXiv preprint arXiv:2209.00796*, 2022.

[99] Siqi Yang, Jun Zhang, Junzhou Huang, Brian C Lovell, and Xiao Han. Minimizing labeling cost for nuclei instance segmentation and classification with cross-domain images and weak labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 697–705, 2021.

[100] Lequan Yu, Shujun Wang, Xiaomeng Li, Chi-Wing Fu, and Pheng-Ann Heng. 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*, pages 605–613. Springer, 2019.

[101] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[102] Ke Zhang and Xiahai Zhuang. Shapepu: A new pu learning framework regularized by global consistency for scribble supervised cardiac segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part VIII*, pages 162–172. Springer, 2022.

[103] Yiheng Zhang, Zhaofan Qiu, Ting Yao, Dong Liu, and Tao Mei. Fully convolutional adaptation networks for semantic segmentation. In *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition, pages 6810–6818, 2018.

[104] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6881–6890, 2021.

[105] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. Pixel contrastive-consistent semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7273–7282, 2021.

[106] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11. Springer, 2018.

[107] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 3–11. Springer, 2018.

[108] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[109] Adrian Ziegler and Yuki M Asano. Self-supervised learning of object parts for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14502–14511, 2022.

BIOGRAPHICAL STATEMENT

Mohammad Minhazul Haq was born in Dhaka, Bangladesh. He completed his B.S. degree in Computer Science and Engineering from Bangladesh University of Engineering and Technology (BUET) in 2014. After graduating from BUET, he worked as a Software Engineer at Therap Services LLC. for two years. In Fall 2016, Minhazul came to USA for pursuing his Ph.D. in Computer Science at The University of Texas at Arlington. His main research interests are Machine Learning, Deep Learning, Computer Vision, and Medical Image Analysis.