Deep Neural Methods for True/Pseudo- Invasion Classification in Colorectal Polyp Whole-Slide Images

Zhiyuan Yang, The University of Western Ontario

Supervisor: Ling, Charles X., The University of Western Ontario
A thesis submitted in partial fulfillment of the requirements for the Master of Science degree in Computer Science
© Zhiyuan Yang 2023

Follow this and additional works at: https://ir.lib.uwo.ca/etd

Recommended Citation
https://ir.lib.uwo.ca/etd/9187

This Dissertation/Thesis is brought to you for free and open access by Scholarship@Western. It has been accepted for inclusion in Electronic Thesis and Dissertation Repository by an authorized administrator of Scholarship@Western. For more information, please contact wlsadmin@uwo.ca.
Abstract

Pathologists often identify colon cancers by inspecting whole-slide images (WSI), which are high-resolution scans of colon tissues extracted through colonoscopy. One specific case of colon cancer, named pseudoinvasion, is hardly differentiable from true invasions even under careful inspections by a panel of expert pathologists. Therefore, Pathologists seek help from artificial intelligence. A type of deep learning model called convolutional neural network (CNN) has been extensively used in image classification. Unfortunately, WSIs are too large and contain a rich amount of detailed information, making the direct use of CNN on WSIs very slow and the training impossible without millions of training samples. However, the number of labelled WSIs for true/pseudo-invasion classification is limited, making the task extremely challenging.

Since it is almost impossible to classify WSIs directly using CNNs, we identify the tissue types on the WSIs and then aggregate the results into a final output. We propose two multi-zoom-level patch-based methods for tissue type recognition, and one method for aggregation. The first method focuses on accuracy by identifying tissue types by patches at three different zoom levels using three CNNs. Then, we apply the weighted averages to combine the classification results. Our second method focuses on efficiency by classifying image patches at a low zoom level and then proceeding only to selected patches at higher zoom levels. Finally, we design a shallow CNN for aggregating the per-patch results from the two proposed tissue-type recognition results into slide-level results for WSIs. Collaborating with pathologists, we collect a private dataset by identifying 150 WSIs and annotating 50 of them. We apply self-supervised learning on a public dataset and transfer the results to our private dataset to increase the performance of our models under limited data.

Our experiments show that our methods can recognize tissues with high accuracy and reasonable efficiency, and aggregate the results into final true/pseudo-invasion classification with promising accuracy under limited data. We developed a web-based tool for the WSI true/pseudo-invasion classification task. The tool can be accessed at [http://ai4path.ca/#/](http://ai4path.ca/#/).

**Keywords:** Deep learning, digital pathology, colorectal carcinoma, true/pseudo-invasion, whole-slide image classification
Summary for Lay Audience

Pathologists often identify colon cancers by inspecting whole-slide images (WSI), which are high-resolution scans of colon tissues extracted through colonoscopy. One specific case of colon cancer, named pseudoinvasion, is hardly differentiable from true invasions even under careful inspections by a panel of expert pathologists. Therefore, Pathologists seek help from artificial intelligence. Deep learning has been extensively used in image classification. Unfortunately, WSIs are too large and contain a rich amount of detailed information, making traditional deep learning methods impossible to use without millions of data. However, the number of labelled WSIs for true/pseudo-invasion classification is limited, making the task extremely challenging.

Since it is almost impossible to classify WSIs directly using CNNs, we identify the tissue types on the WSIs and then aggregate the results into a final output. We propose two multi-zoom-level patch-based methods for tissue type recognition, and one method for aggregation. The first method focuses on accuracy by classifying the entire WSI three times at different zoom levels using CNNs. The second method focuses on efficiency by classifying selected regions at each zoom level. We also collaborate with pathologists to collect a small private dataset for our task, and we apply transfer learning to improve the performance of our models.

Our methods achieved high accuracy with reasonable efficiency under a limited amount of data. We developed a web-based tool for the WSI true/pseudo-invasion classification task. The tool can be accessed at [http://ai4path.ca/#/](http://ai4path.ca/#/).
## Contents

### Abstract

Summary for Lay Audience

List of Figures

List of Tables

List of Appendices

1 Introduction

1.1 Motivation

1.2 Contributions

1.3 Thesis Outline

2 Background and Related Work

2.1 Convolutional Neural Networks

2.1.1 ResNet

2.2 Deep Learning Researches on Digital Pathology

2.2.1 Whole-Slide Image Classification

2.2.2 Cancer Detection

3 Methodology

3.1 Problem Description

3.2 Pipeline Details

3.2.1 The Patch-based WSI Classification Procedure

3.2.2 Multi-zoom-level Patch-based CNN Prediction Weighted Average

3.2.3 Selective multi-zoom-level Patch-based CNN Prediction

3.2.4 Confidence Map CNN for Patch-level Result Aggregation

3.3 Training

3.3.1 Balancing Classes and Weighted Training

3.3.2 Self-supervised Pretraining

3.3.3 Image Colour Normalization

3.3.4 Training Details

4 Experiments

4.1 Datasets
List of Figures

1.1 An example whole-slide image. It has a very high resolution and can be viewed under very high zoom levels. .......................................................... 2
1.2 An overview to our workflow. ............................................................. 6
1.3 An illustration of our online web tool for WSI true/pseudo-invasion classifications. The right box at the top-left corner shows the final confidence. The green box under the red box shows a few changeable parameters for pathologists. The purple box at the bottom-left corner shows the legend of the tissue type classification results on the WSI. The upload-file button is located at the top-right corner of this page. .......................................................... 9
2.1 An illustration of the convolutional layer ............................................. 12
2.2 An example of max-pooling layer ....................................................... 13
2.3 The architecture of LeNet[20] as an example of the standard convolutional neural network .......................................................... 13
2.4 The performance comparison between a 20-layer model and a 56-layer model. Taken from the ResNet paper[10] ................................................. 14
2.5 Residual block ................................................................................... 15
2.6 A general view to the ResNet Architecture ........................................ 15
2.7 ResNet’s performance compared with other SOTA methods. Taken from the ResNet paper[10] .......................................................... 16
2.8 Yan’s WSI classification workflow. [32] .............................................. 19
2.9 An overview of the DenseNet architecture. [12] ................................ 19
2.10 Visualization of Khvostikov’s method. [17] ....................................... 20
2.11 An overview to Shen et al.’s method. [28] ......................................... 21
3.1 An illustration of cells and structures under different zoom levels. Cells similar to hemosiderins are best identified under a high zoom level, and structures similar to the rounded lobular groups can only be identified under a low zoom level. .......................................................... 24
3.2 The per-patch classification procedure .............................................. 25
3.3 Pathologists’ annotations of a type of tissue and its tightest bounding box .......................................................... 26
3.4 Stage 1 multi-zoom-level patch-based CNN prediction weighted average. .......................................................... 27
3.5 Selective multi-zoom-level patch-based CNN pipeline ...................... 29
3.6 A sample per-patch classification result for a WSI. Each coloured dot corresponds to a predicted patch. Dots for backgrounds and patches with low confidence are not shown. .......................................................... 31
3.7 Stage 2 Confidence map CNN and its architecture .......................... 32
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.8</td>
<td>An annotated area of mucus lake tiled into $224 \times 224$ patches at 20x zoom level.</td>
</tr>
<tr>
<td>3.9</td>
<td>An illustration of repetitive results generated from different data augmentation methods. Applying image transpose and vertical flip is the same as rotating the image 90 degrees clockwise.</td>
</tr>
<tr>
<td>3.10</td>
<td>An imprecise annotation. The blue polygon is the annotated area, and the black circles include areas that do not belong to the annotated class.</td>
</tr>
<tr>
<td>3.11</td>
<td>Colour difference between two WSIs. Even though both tissues are extracted from the same body part, the right image has darker tissue than the left image.</td>
</tr>
<tr>
<td>4.1</td>
<td>Example images taken from the NCT-CRC-HE-100K dataset.</td>
</tr>
<tr>
<td>4.2</td>
<td>An example of the annotated area on a WSI in our dataset. Coloured polygons indicate the areas of certain tissue types.</td>
</tr>
<tr>
<td>4.3</td>
<td>Directly classifying downsampled WSIs.</td>
</tr>
<tr>
<td>4.4</td>
<td>Detail of the single zoom-level per-patch prediction model with linear classifier.</td>
</tr>
<tr>
<td>4.5</td>
<td>Normalized confusion matrix for each method.</td>
</tr>
<tr>
<td>4.6</td>
<td>Threshold v.s. False rates.</td>
</tr>
<tr>
<td>4.7</td>
<td>An illustration of different colour normalization methods.</td>
</tr>
<tr>
<td>A.1</td>
<td>The detailed architecture for ResNet with different number of layers.</td>
</tr>
</tbody>
</table>
List of Tables

4.1 The number of training examples in each class on each zoom level .......................... 43
4.2 A confusion matrix for two classes ........................................................................ 44
4.3 WSI patch classification accuracy for each method .................................................. 48
4.4 Final WSI True/Pseudo Invasion Classification Accuracy for each method .............. 50
4.5 Fine-tuning accuracy of models pretrained using different methods ....................... 51
4.6 Impact of colour normalization method in terms of average accuracy and time spent, .................................................................................................................. 54
List of Appendices

Appendix A ......................................................... 64
Chapter 1

Introduction

In this chapter, we discuss the importance of the WSI true/pseudo invasion classification task and the motivation behind our research. We will provide some basic background knowledge about colon cancer, digital pathology and deep learning. We will talk about our contribution toward solving the WSI classification task. An overview of the structure of this thesis will be introduced at the end of this chapter.

1.1 Motivation

Colorectal (colon) cancer is the third most common cancer in North America. It usually starts with non-cancerous growths called polyps in the large intestine. Under certain conditions, these polyps can become cancerous and potentially spread to the entire human body. It is crucial to detect cancer before it spreads, as it can lead to a severe decrease in the long-term survival rate. The Canadian cancer statistics 2022 [6] show stunning figures that the 5-year net death rate for both men and women is around 33%. Adults of age forty and above are recommended to take colonoscopies regularly. Colonoscopy is a test that allows doctors to identify potential diseases on the inner lining of human colons by sending a camera inside a colon. A biopsy will be conducted if signs of cancer are found inside the colon. During a biopsy, tissues will be extracted from human colon polyps and sent to pathologists for examination.
With the development of digital pathology, taking high-resolution scans of the extracted tissue under microscopes and storing the image in a database has grown popular. The microscopic scan is called a whole-slide image (WSI), as shown in Figure 1.1. This procedure provides a comfortable environment for pathologists to view patients’ medical images without limits on the availability of the actual tissue. Pathologists will try to identify signs of malignancy on the WSIs of the extracted tissue in different zoom levels with assistance from computer softwares. However, malignancy (true invasion) is usually identified by microscopic submucosal invasion, which is bothered by a problematic mimic called pseudoinvasion. Pseudoinvasion occurs when glands become misplaced in the submucosa. It is very challenging even for a panel of expert pathologists to differentiate pseudoinvasion and true invasion because only a few morphological criteria can be used to help pathologists to make decisions. Also, the key regions can be very small to be spotted by pathologists, and can only be viewed under very high magnifications such as 10x or 20x zoom levels. As a result, hard cases of pseudoinvasion will usually take one or two days for pathologists to come up with a judgment, and extreme cases can take up to a week. Therefore, pathologists seek a tool to help them in conquering this challenge.

Figure 1.1: An example whole-slide image. It has a very high resolution and can be viewed under very high zoom levels.

Deep learning research in the field of computer vision has been growing rapidly in recent
1.1. Motivation

years. Modern deep learning models utilizing convolutional neural networks (CNNs) have shown extraordinary achievements in image classification and object detection tasks. From the early CNN model for image classification, such as AlexNet [19] that achieves a top-5 classification error rate of 15.3%, the currently most popular CNN model called ResNet-50 [10] has grown fast enough to outperform human by having a top-5 error rate of only 5.25%. This leads to an upsurge in applying deep learning to many fields, including digital pathology. One successful deep-learning application in digital pathology is breast cancer detection. For example, Al-Masini et al. [1] has developed a system based on the most well-known object detection model YOLO [27] for detecting mass locations on mammograms. The system has achieved an overall accuracy of 99.7%. For breast lesion classifications, Seokmin Han el al [9] have developed a system by applying GoogleNet [31] on breast ultrasound images. The system has achieved a classification accuracy of approximately 90%. These achievements have proven that deep learning can be used to develop convenient tools to help doctors and pathologists in medical procedures.

Besides the remarkable success of breast cancer as mentioned in the previous paragraph, colorectal cancer remains a challenging field of deep learning. The difference mainly comes from the data types used for constructing deep learning systems. For breast cancers, data are usually ultrasound images, CT scans and magnetic resonance imaging scans. The amount of labelled such images is abundant, and the images usually contain distinctive features for deep learning models to learn and capture.

However, as mentioned previously, WSI is one of the most important data sources for colorectal cancer. WSIs have several characteristics that differentiate them from ordinary medical images. Firstly, the size of WSIs is usually huge as they are high-resolution images taken under microscopes. On average, one WSI can have around three hundred million of pixels, making it as big as tens of thousands of ordinary medical images. This makes WSIs impossible to fit into CNN models as ordinary CNN models only take inputs as small as $224 \times 224$ pixels. One naive solution is to downsample WSIs into the standard input size to CNNs. However,
it is obvious that this approach will fail because downsampling a WSI over around 100 times will lead to a catastrophic loss of information. It is also impossible to build a CNN that takes WSIs as standard inputs as the model can be too large and deep to fit physically into modern computers. Secondly, labelled WSIs are rare and precious. It usually requires hundreds of thousands of images to train a CNN from scratch. With transfer learning methods, this number can be reduced, but the effectiveness of such methods is highly dependent on the domains of the datasets. It is unpractical to collect that many WSIs. Therefore, people take the context of the WSIs into consideration as a complement by annotating the WSIs at different zoom levels. However, annotating a large number of WSIs is also unpractical. Due to their gigantic image sizes, the amount of information to be annotated on each WSI is much richer than ordinary medical images, which leads to a higher demand for professional knowledge. Also, the important features on WSIs can have varying sizes, and sometimes the features need to be annotated across zoom levels. This increased the time taken to annotate one WSI fully. Therefore, the annotation of WSIs is too costly in both time and labour, which leads to a data shortage in training deep learning models for developing assisting tools for pathologists. The data shortage issue is especially serious in the WSI true/pseudo-invasion classification task because the required annotations are too specific and hardly be found in public databases.

There have been some works on applying deep learning models to WSIs. For example, Campanella et al. \cite{3} has built a system for detecting cancer metastasis on WSIs by tiling each WSI into image patches and using a CNN to predict the probability of being positive for each patch. Then, the top-k patches are sent into a recurrent neural network (RNN) to aggregate into a final classification result for the WSI. This system has achieved an area under the curve (AUC) above 0.98 for breast, prostate, and basal cell carcinoma. However, this method does not produce a satisfactory result on the colorectal true/pseudo invasion classification problem due to two reasons. First, the features for breast cancer, prostate cancer and basal cell carcinoma have relatively consistent sizes, so the model constructed for one single zoom level is enough to capture the features. However, features for differentiating true/pseudo invasion need to be
viewed in multiple zoom levels. Second, the training requires a very large dataset. The training datasets for breast cancer, prostate cancer and basal cell carcinoma usually contain around 10,000 WSIs for each cancer. However, the dataset for true/pseudo invasion is very small and contains only around 150 WSIs. Therefore, developing a system with improved performance on the true/pseudo invasion classification task takes time and effort. It requires a better design of the system architecture, more task-specific data to be collected and higher data utilization efficiency.

As a result, the WSI true/pseudo invasion classification task remains unexplored. Pathologists seek a solution to this problem, and if possible, they want an accessible and easy-to-use tool to help them with daily diagnosis.

1.2 Contributions

As we discussed earlier, the colorectal true/pseudo invasion classification task is challenging. It is challenged by the limited amount of annotated WSIs for training and large variance in the sizes of the features to learn. Most commonly, in the existing works for WSI classification tasks, people use two-stage methods. They usually tile the WSIs into collections of patches under one zoom level and use one CNN to classify the image patches. The patches are then aggregated into one final prediction for the corresponding WSI in different ways, such as using a linear classifier or RNN. However, classifications under only one zoom level are only plausible under the assumption that the important features or classes on the WSI can be identified in a small window of fixed size under one zoom level. Such an assumption is not true for the colorectal true/pseudo invasion classification task because some classes are best viewed under very high zoom levels while others can only be seen under very low zoom levels.

In this thesis, we collaborate with pathologists in the Department of Pathology and Laboratory Medicine at Western University. We identify 150 WSIs and annotate 50 of them by labelling regions on the WSI with their tissue types. We take ideas from the WSI examination
Figure 1.2: An overview to our workflow.

procedure of pathologists. Pathologists usually start with low zoom levels to take an overview of the WSI to identify some apparent patterns or regions of interest. Then, they will zoom into each region of interest to identify more evidence of invasion, which includes the presence of certain types of cells, structures of tissues and shapes of groups of cells. Cells are often visible under high zoom levels, while structural information can be captured at medium zoom levels. We imitate such a procedure and propose two multi-zoom-level patch-based methods. As shown in Figure 1.2, we first tile WSIs into patches at low, medium and high zoom levels, and then we train three CNNs to classify the patches at specific zoom levels. These three CNNs are trained and arranged differently in the two approaches. They either independently classify the entire WSI in patches on different zoom levels or only identify selected regions on the WSI based on predictions from other zoom levels. The former focuses on tissue type classification accuracy, while the latter focuses on efficiency. This might seem counterintuitive as the former method runs models in parallel while the latter method runs in sequential order. However, the parallel method classifies every region on a WSI three times under different views, while the sequential method visits each region no more than twice.

The three CNNs we use to classify WSI patches use ResNet [10] as their architecture. Although it is a common practice to apply transfer learning from models pretrained on the Im-
ageNet [7] dataset in the image classification task to a downstream task, we discover that test accuracy from models pretrained on the ImageNet dataset was only a little bit higher than models trained from scratch. Even though the ImageNet dataset contains over one million images spanning one thousand classes, the images contain only natural and environmental objects. The features used to identify cells and tissues on medical images are significantly different from the features for identifying natural images. Therefore, we pretrain the CNNs on the NCT-CRC-HE-100K [16] dataset, which consists of over 100,000 image patches extracted from human colon WSIs at 20x zoom level. This dataset has ten tissue classes closely related to colon cancer, which means the features learnt during pretraining will be more useful to our task. We also compare different pretraining techniques, such as supervised and self-supervised. For the self-supervised pretraining, we use Bootstrap-Your-Own-Latent (BYOL) [8] as the learning approach because BYOL has achieved a superior accuracy than the conventional self-supervised learning methods such as SimCLR [4] and MoCov2 [5] on ResNet-50. After pretraining, the models are further finetuned on our dataset to classify patches dedicated to the true/pseudo-invasion task. More details will be introduced in Chapter 3.

To effectively aggregate the image patch classification results, we discard the classical aggregation methods such as thresholding or applying a linear classifier to the WSI statistics, because these methods fail to capture spatial and morphological information. To best capture the spatial and morphological information, we first arrange the patch classification results based on the position of the corresponding patches to form a label map, which can be considered a small and compact representation of the original WSI. Then, we construct and train a CNN to classify the label map into true invasion or pseudo invasion, producing a final result for the corresponding WSI. However, due to the limited amount of labelled WSI, we cannot generate enough label maps for training a deep model. Thus, we design a shallow CNN with only four convolutional layers. Finally, we adopt standard data augmentation techniques used in computer vision tasks to label maps to create more training samples. Our methods achieve a promising final accuracy on the WSI true/pseudo-invasion classification task on our dataset.
and show a clear improvement compared with single-scale baseline methods.

Besides the multi-zoom level design discussed earlier, our methods have several other highlights. The first is modularity. Our methods follow a two-stage pipeline that first classifies WSIs by patches under multiple zoom levels into different tissue categories and then uses the results of the per-patch predictions as evidence to aggregate into final slide-level classification results. Any per-patch prediction models can be easily substituted without affecting the pipeline integrity, and the aggregation method can be changed based on preference for accuracy or efficiency. The second is understandability. Unlike most deep learning frameworks that act as black boxes, we can visualize the evidence by overlaying the per-patch results directly on the corresponding WSI, as shown in Figure 1.3. This can make sure pathologists understand how the final result arrives. The third is reliability. The experiments on our dataset show that our methods produce high accuracy compared with baseline methods. Pathologists are quite satisfied with the performance and the visualization, and our collaborated work is still in progress for further improvements.

We have built an online web system for pathologists. Our tool is simple and clean compared to the widely used digital pathology assisting tool named QuPath [2] that requires installation and a certain depth of knowledge. The cell detection tool on QuPath has a lot of parameters to tune to get the best detection performance. In contrast, our tool only has two tunable parameters, as shown in Figure 1.3, where one is for tissue type sensitivity, and another is for display transparency. This makes our tool extremely user-friendly. Also, our system is web-based, which means it is easily accessible on any computer without needing installation.

The main contribution of this thesis is as following:

- We developed a two-stage pipeline for the colon WSI true/pseudo-invasion classification task under an extremely limited amount of data. We proposed two multi-zoom-level patch-based methods for colon WSI tissue type classification and one patch-level result aggregation method, achieving high accuracy in tissue recognition and satisfying accuracy in aggregation. See Chapter 3.
1.2. Contributions

- We collected a dataset of 150 WSIs and annotated 50 of them for the task in collaboration with pathologists in the Department of Pathology and Laboratory Medicine at Western University. See Chapter 4 for details.

- We evaluated the performance of our method against the single-zoom-level method and direct usage of CNN for the task. Our method outperformed the baseline by a clear margin in terms of accuracy with a reasonable trade-off in efficiency, as shown in Chapter 4.

- We developed a web tool based on our methods to assist pathologists in their daily work. Our website can be accessed through [http://ai4path.ca/#/](http://ai4path.ca/#/).
1.3 Thesis Outline

We organize this thesis in the following structure: In Chapter 2, we will provide background knowledge to some related work. This includes explanations of convolutional neural networks, the image classification task, and deep learning research on digital pathology. In chapter 3, we will discuss and explain our two-stage multi-zoom level methods for solving the true/pseudo-invasion classification task in detail. We will also cover the techniques used during training. Chapter 4 contains our experiment results, including comparing the baseline method, different pretraining approaches and data preprocessing techniques. Chapter 5 concludes this thesis and discusses the limits of our method and potential future improvements.
Chapter 2

Background and Related Work

In this chapter, we will provide some related background knowledge about the WSI true/pseudo-invasion classification task. We will also review previous works on deep learning applications in digital pathology. Basic knowledge needed to be known about our work will be discussed too.

2.1 Convolutional Neural Networks

Convolutional Neural Network (CNN) is a type of neural network that is widely used for computer vision tasks such as image classification, object detection and image segmentation. In the past, people used feed-forward neural networks with only fully-connected layers to solve tasks involving images. The limitation of such architectures is obvious: all inputs have to be flattened into one-dimensional vectors to get into the model, which leads to a loss of important 2D features such as the spatial, structural and positional features on the input images. The convolutional neural network is a modified version of the feed-forward network. It can take 2D inputs and is able to extract important features from the 2D inputs directly. This is achieved by a particular type of layer called the convolution layer. This layer learns to detect and extract specific types of features, such as edges, corners and shapes on an image, by learning a 2D filter/kernel that looks at a small region of the image and aggregates the region’s information.
When feeding a 2D input to a convolution layer, the 2D filter slides over each pixel position on the input at a specified stride and performs the convolution operation between the filter and the image pixels covered by the filter. Figure 2.1 illustrates the convolution process. As a brief explanation, convolving an image at a given pixel position is equivalent to summing up the pixel-wise multiplication results of the area under the filter with the horizontally and vertically flipped filter. This can also be mathematically shown as the following equation:

\[ y(i, j) = \sum_{u=-k}^{k} \sum_{v=-k}^{k} F(u, v) \cdot I(i - u, j - v) \]

Where \((i, j)\) and \((u, v)\) are the pixel’s coordinates, \(F\) is the filter of size \(k\) and \(I\) is the input image. The convolved images are often called feature maps.

A standard CNN architecture usually consists of a series of convolution blocks followed by many fully connected layers, as shown in Figure 2.3. Each convolution block often contains many convolution layers, followed by a pooling layer. Pooling layers are used to reduce the dimensionality of the feature map as well as provide invariances to small transformations. Similar to convolution layers, a pooling layer slides a pooling filter of a specific size over the input feature map at a specific stride. Most commonly, people use \(2 \times 2\) filters with a stride of 2 to reduce the input size by half. There are many types of pooling filters, such as max pooling, mean pooling and average pooling. The most widely used pooling operation is max pooling, which only keeps the maximum value of the pixel values within the area under the filter, and discards all other values. Figure 2.2 shows an example of a pooling layer with a \(2 \times 2\)
max-pooling filter and stride of 2.

![Figure 2.2: An example of max-pooling layer.](image)

![Figure 2.3: The architecture of LeNet[20] as an example of the standard convolutional neural network](image)

2.1.1 ResNet

After the huge success achieved by the first CNN, researchers have created many more CNN architectures that are deeper than previous ones because they believe increasing model complexity can increase performance. However, stacking more layers reduces the error rate only when the number of layers is small. The performance deteriorates very fast after stacking enough layers, which means the depth and model complexity for traditional CNN architectures are actually limited. The experiment conducted by He et al. [10] compared the performance
between a model with 20 layers and one with 56 layers. As shown in Figure 2.4, the error rate of the 56-layer model is always higher than the 20-layer, which contradicts people’s common belief. He also empirically showed the limit to the number of layers should lie between 16 and 30.

Figure 2.4: The performance comparison between a 20-layer model and a 56-layer model. Taken from the ResNet paper [10]

The phenomenon illustrated in Figure 2.4 is a demonstration of the classic deep learning problem: the exploding/vanishing gradient problem. When training a deep learning model, gradients for each weight with respect to the loss are computed to update the weights. However, the computation involves a series of multiplications following the chain rule for derivative, which takes the gradient to zero if associated gradients are small, or to infinity if associated gradients are large. The former is called the vanishing gradient, and the latter is called the exploding gradient. Although both problems appear on lower layers of deep models quite often, the latter one can be solved by using batch norm layers [14], which explicitly normalizes layer outputs to unit norm. Therefore, the vanishing gradient problem is the actual problem of deep models.

To solve the vanishing gradient problem, the ResNet [10] architecture was proposed and quickly became the most famous and widely used CNN in the world, and it is still state-of-the-art in image classification tasks today. Different from traditional feed-forward deep CNNs that stacks convolution layers and pass input images through the sequence of convolution layers,
ResNet introduced the concept of skip-connection as shown in Figure 2.5, which allows input to skip convolution layers when going through the network. Mathematically, we compute the gradient of a residual block $H$ containing nonlinear layers $F$ given the loss function $L$ as follows:

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial H} \frac{\partial H}{\partial x} = \frac{\partial L}{\partial H} \left( \frac{\partial F}{\partial x} + 1 \right) = \frac{\partial L}{\partial H} \frac{\partial H}{\partial x} + \frac{\partial L}{\partial H}$$

This special design adds an identity function that enables a stronger flow of gradient during back-propagation. It allows lower layers to receive a proper error signal to update their weights. With this design, the vanishing gradient problem is solved. Figure 2.7 shows the performance of ResNet with different numbers of layers. It can be observed that increasing the number of layers always leads to a decrease in the model’s error rate, even when the ResNet model is constructed to contain 152 layers.
2.2 Deep Learning Researches on Digital Pathology

Modern digital pathology technologies have been widely used in hospitals to provide convenience and accessibility. Specially equipment such as microscopic digital scanners can convert an entire glass of tissue specimen into a high-resolution digital slide. This not only enables the involvement of computer software for medical image processing to get better visual qualities but also allows people to develop computer tools for digital pathology analysis.

Cancer is the main focus of digital pathology analysis. This involves not only the detection of cancer but also other tasks, such as cancer grading and survival prediction. However, tradi-
tional methods for solving these tasks largely rely on mathematically extracting hand-crafted features that may differ a lot. Also, the difference in staining, sectioning and fixation can have a huge impact on the extracted features, which limits the generalization of the analysis. In addition, extracting the hand-crafted features often requires domain-specific knowledge, which limits the work to domain experts. Therefore, deep learning methods become an ideal solution to digital pathology analysis tasks because these methods can automatically learn to extract features without prior knowledge.

The study of deep learning methods and their applications in digital pathology has always been hot. Deep learning methods using CNNs have demonstrated great power in computer vision tasks, so they have become the best candidate for analyzing digital slides. Many works have been done on using CNNs for digital slide classification, segmentation and detection tasks, and the interest has continued to grow in recent years.

### 2.2.1 Whole-Slide Image Classification

As previously mentioned, whole-slide images (WSI) are digital scans of entire histopathology slides under microscopes. WSIs are very high in resolution, and they can contain up to trillions of pixels. In a computer, a WSI is stored as multiple copies of itself in a pyramid structure. Each level of the pyramid corresponds to a magnification level, which provides a faster reading speed if people want to view the slide at different zoom levels.

WSI classification tasks are different from ordinary image classification tasks. The biggest difference is the size of the input to deep learning models. Datasets for ordinary image classification tasks usually contain images that are only as large as a few hundred pixels across, while the size of a WSI is usually tens of thousands of pixels across. Thus, most of the current WSI classification methods use machine learning approaches on image patches extracted from WSI at a given zoom level, and then aggregate the patch features to obtain a slide-level result.

Yan et al. [32] proposed a classic pipeline for classifying WSIs, as shown in Figure 2.8. In their proposed workflow, they first tile a WSI into a collection of overlapping image patches
at 20x and 40x zoom levels. Then, they performed patch selection by removing background patches, which are image patches with only white pixels that have RGB values greater than 200. After that, they used AlexNet [19] as a feature extractor to convert the image patches into 4096-dimensional feature vectors. They used P-norm pooling to aggregate patch-level feature vectors into one slide-level feature vector. The P-norm pooling can be mathematically expressed as the following equation:

\[
f_p(v) = \frac{1}{P N} \sum_{i=1}^{N} v_i^p
\]

Where \(N\) is the total number of image patches, \(P\) is a hyperparameter for P-norm pooling, and \(v_i\) is the feature vector for the \(i\)-th patch. To reduce the dimensionality of the aggregated feature vector, they apply feature selection techniques to select the top 100 features. The selection is based on the absolute difference between the feature’s average value on positive examples and negative examples. Finally, they use an SVM to classify the feature vector. Their method won the MICCAI 2014 challenge on the brain tumour classification task by having an accuracy of 97.5%. However, their per-patch feature extraction method only extracts local features at one zoom level, which fails to capture structural and morphological features beyond the size of the patch. Their aggregation method is agnostic to the positional and relational information between patches. Our proposed multi-zoom-level method solves the feature size problem, and our aggregation method learns a CNN to handle the positional information between patches. Therefore, our method can solve harder tasks, such as the true/pseudo-invasion classification task.

2.2.2 Cancer Detection

Cancer detection is a different task from the WSI classification task. In the WSI classification task, the slide-level label often comes from the aggregation of many patch-level results. However, the cancer detection task only focuses on the presence of cancer cells or regions, which
means the local results are often sufficient to tell if an image has cancer.

Jiao et al. [15] proposed a method for detecting colon cancers on WSI using machine learning methods. They extracted hand-crafted image features such as pixel mean and variance. Then, they combine these features with the texture information by using the Gray-Level Co-occurrence Matrix (GLCM) method [18], which is a statistical method based on second-order features. This includes image energy, contrast, correlation and entropy. Finally, they construct an SVM to classify the features into cancer and non-cancer. However, this method still relies on hand-crafted features instead of deep features, which is neither efficient nor general enough.

In contrast, Khvostikov et al. [17] proposed a cancer detection method based on tissue type recognition by deep learning models. They utilized the DenseNet [12] as the image patch classifier for tissue type recognitions, which is a modified version of ResNet [10] with dense skip connections. First, they tile incoming WSIs into collections of non-overlapping image
patches in size of $224 \times 224$ pixels at 20x zoom level. Then, they use the DenseNet model to classify the image patches into different tissue categories, including cancers. They pretrain the DenseNet model on the NCT-CRC-HE-100K [16] dataset, and fine-tune the model on the PATH-DT-MSU dataset.

Figure 2.10: Visualization of Khvostikov’s method. [17].

Figure 2.10 shows an example of the tissue type recognition results on a WSI. Their method not only detects cancer regions but also recognizes other types of tissues, which provides more convenience for views by pathologists. In our work, we take tissue type recognition as an intermediate step and build the final slide-level result on top of the recognition results so that the final result is supported by visible evidence for pathologists to see.

Cancer detection is not only limited to WSIs. Shen et al. [28] proposed a method for predicting breast cancers using mammograms, which are X-ray imaging of the breast. They first train a CNN model for classifying regions of interest (ROI) on a mammogram. Then, they convert the CNN from an ROI detector into a whole image classifier by adding convolution layers to the original CNN and applying further training on image-level labels. Finally, they use the entire CNN to classify the image into cancer or normal directly. Their method achieves high accuracy in classifying breast cancer mammograms. However, we can hardly adapt this
method to our task because their method relies on using a CNN model to classify images as a whole, which is impractical for images with very large sizes.
Chapter 3

Methodology

In this chapter, we explain our approach to solving the WSI true/pseudo-invasion classification task. Firstly, we analyze the problem we aim to solve and make a clearer illustration of the whole slide image classification pipeline we hope to build in Section 3.1. Secondly, we explain the two versions of our multi-zoom-level patch-based method for tissue-type recognition by first explaining how the patch-based classification procedure would work on a single zoom-level, and then expanding the method into multi-zoom-level methods. After that, we describe our aggregation method. Finally, we show the training details of the multi-zoom-level methods in Section 3.3.

3.1 Problem Description

The true/pseudo-invasion classification task lies within the intersection of medical imaging and gigapixel image classification. This task is different from ordinary cell detection tasks because only the presence of cancer cannot identify a case as a true invasion. In this task, we will deal with a special type of medical image called whole-slide image (WSI) to see if a cancerous WSI is a true invasion. WSIs are high-resolution digital scans of microscopic slides of biological tissues. WSIs are usually very large, and the sizes can vary from 10,000 pixels across to 100,000 pixels across. They contain rich information that can be used for our task,
including the presence of certain types of cells, structures of tissues and location of cell groups. However, important features on WSIs can have a very large variance in their sizes, which means they need to be viewed under different zoom levels. As shown in Figure 3.1, cells on a WSI can be as small as 10 or 20 pixels across, while structures can be as large as hundreds or thousands of pixels across. This means traditional image classification methods that take a whole image as input cannot solve our problem, because they need to resize WSIs to a few hundred pixels across so they can fit into a CNN. The resizing process will drastically damage the information on the images because the small features will shrink to a point or even disappear. Annotation of WSIs is very expensive because fully annotating a WSI requires expert knowledge to catch important features and considerable time for precise labelling. To the best of our knowledge, there is currently no public annotated dataset for this specific task. Thus, this task is extremely challenging.

We try to conquer the challenges under limited data. We aim to build a system that can take WSIs as inputs that can capture all important features and output a binary result that represents whether the WSI contains true invasion or not. Therefore, pathologists can use such a system to increase their efficiency and accuracy in diagnosis, or even transfer the task entirely to the system.

### 3.2 Pipeline Details

In this section, we explain our multi-zoom-level pipeline, which is designed to have two stages. The first stage is the WSI tissue type classification stage, and the second stage is the aggregation stage. We will start our discussion by first introducing our stage one methods, which includes a discussion of the patch-based procedure in section 3.2.1, and two proposed multi-zoom-level methods for tissue type classification in sections 3.2.2 and 3.2.3. The first multi-zoom-level method focuses more on the accuracy, and the second method focuses more on efficiency. In section 3.2.4, we explain in detail our stage two aggregation method.
Figure 3.1: An illustration of cells and structures under different zoom levels. Cells similar to hemosiderins are best identified under a high zoom level, and structures similar to the rounded lobular groups can only be identified under a low zoom level.

3.2.1 The Patch-based WSI Classification Procedure

Patch-based methods are the most widely used to solve the WSI size problem. It is to feed a WSI into CNN in small pieces instead of in whole. In this way, the WSIs can successfully go through the CNN without being resized, and thus we can preserve most of the information. This also mimics pathologists’ diagnosis process as they usually start by identifying sub-regions in a WSI, and combine them to get a final result.

We tile the input WSIs into a collection of image patches of $224 \times 224$ pixels at a zoom level. After tiling the WSIs, we preprocess the images by normalizing the image colour. Then, we iteratively feed the image patches into our CNN model for classification. This brings us a collection of classified image patches, which can be used for aggregation. Figure 3.2 shows
the per-patch classification procedure we are using.

![Diagram of the per-patch classification procedure]

Figure 3.2: The per-patch classification procedure

Most commonly, people process the per-patch results and feed them into a linear model after the per-patch classification procedure to aggregate the results for practical applications, which creates an end-to-end pipeline.

3.2.2 Multi-zoom-level Patch-based CNN Prediction Weighted Average

Patch-based classifications on a single zoom level are not accurate enough, and using linear classifiers for aggregation is also inadequate for our task. This combination has two critical flaws. First, image patch classification on a single zoom level brings unbalanced per-class accuracy. This problem can hardly be fixed only by training because some classes are tissue structures too large to be recognized on an image patch that only shows a small part of the whole structure, and some classes are cells too small to be visible on one single zoom level. Second, using a linear classifier on tissue class percentages is insufficient for the true/pseudo-invasion classification task. The diagnosis of the true/pseudo-invasion is challenging because it does not only need to consider the presence of specific cell types, but also the position and structure formed by cells. In this section, we will focus on solving the first flaw. The solution to fix the second flaw will be introduced later in section 3.2.4.

To fix the first flaw, we switch from single zoom level image patch classification to classification in three different zoom levels. The choice of zoom levels comes from the analysis of pathologists’ annotations on WSIs, which are polygons represented by ordered sequences of
vertices. The highest zoom level is chosen to be the lowest level that can make a $224 \times 224$ image patch show the largest classes as a whole. The middle zoom level is chosen to be the middle zoom value between the highest zoom level and the lowest zoom level. The lowest zoom level is the highest level that can make the smallest classes visible on an image patch. The average size of the tightest axis-aligned bounding square to the annotations of a tissue class was used. For each annotation, we want to compute the size of the tightest axis-aligned bounding square. To find the size of the bounding square, we can start by computing the length and width of the tightest axis-aligned bounding box, and take the length of the longer side as the desired length as in Figure 3.3. Once the size of the square for each annotation is computed, we can easily calculate the average size per class, which will be used to choose the zoom levels as described above. As a result of the above procedure, we chose 5x, 10x and 20x as our interest zoom levels.

With the proper zoom levels chosen, we can train three models each for one specific level and construct the first half of the entire pipeline as in Figure 3.4. Each model alone still follows the per-patch classification procedure as in Section 3.2.1. Here we use ResNet-18 as our model for image patch classification. Then, we take an extra step at the end of the per-patch
classification procedure by arranging and storing the per-patch classification results in a grid. Each cell in the grid stores a vector $u \in [0, 1]^n$ of the class confidences for the corresponding patch on the WSI, creating a confidences map. If class label indices are stored in the cells instead of class confidence vectors, a label map will be created. For 5x and 10x zoom levels, the label maps are upsampled to 20x by dividing each grid into 16 or 4 smaller grids to have a matching shape for the final step in stage one: computing a weighted average among all three label maps to get higher accuracy in subarea tissue type classification.

Although averaging confidence values from different models for one class can often lead to a more reliable result, averaging class indices has no practical meaning. Therefore, we use weighted averages only for confidence maps. For label maps, we use weighted majority voting instead. Since a model might accurately predict some classes while frequently giving false results for other classes, we will give each class a weight associated with different models.
Mathematically, the weight $w_{i,j}$ on class $j$ from model $i$ is set as followings:

$$w_{i,j} = \alpha_{i,j} \cdot \beta_{i,j}$$

where $\alpha_{i,j}$ is the accuracy of model $i$ on class $j$, and $\beta_{i,j}$ is a hyperparameter. By default, all $\beta$s are 1. The existence of $\beta_{i,j}$ enables manual weight adjustments based on domain experts’ advice. For example, pathologists view and identify rounded lobular groups often at very high zoom levels, which suggests more weight should be added to this class on the 5x model.

After computing the weighted average of the confidence maps from three different zoom levels, the weighted average needs to be aggregated to obtain a final slide-level classification. The aggregation method will be introduced in Section 3.2.4.

### 3.2.3 Selective multi-zoom-level Patch-based CNN Prediction

The previous method focuses on increasing accuracy. However, the increase in accuracy comes from the increase in the complexity of the entire WSI classification system. The average time taken to classify a WSI increase because the 5x and 10x model brings an average of 5,000 image patches to classify, and the 20x model usually needs to classify over 20,000 image patches. This leads us to the next problem to concern: finding a way to accelerate the system without damaging the final accuracy. Therefore, we have re-designed the previous stage one pipeline so that the total amount of image patches to classify is greatly reduced with a final accuracy close to the previous method. This new stage-one design still consists of models in the same architecture, which is ResNet-18. The per-patch classification procedure shown in section 3.3.1 is used and unchanged in this new design.

Instead of having models in parallel as in Section 3.2.2, we arrange the models in sequential order. In this new pipeline, later models will only classify image patches on regions selected by their previous models instead of the whole WSI. Also, each model will be trained on a specific subset of the classes instead of all, making them more specialized. We have replaced the 5x
model with a 1.25x model for a larger patch area to contain all the large classes, such as rounded lobular group and mucus lake with peripheral dysplastic glands. The 10x model is trained to focus on classifying medium-size classes such as stroma with hemosiderin and angulated gland. The 20x model will be specialized in very small classes, such as LP stroma and luminal necrosis. This design allows the pipeline to mimic pathologists’ diagnosis procedures further.

The whole pipeline is shown in Figure 3.5. The WSI will be first tiled into 224×224 image patches at a 1.25x zoom level with 50 percent overlap. Then, the patches will be sent into the 1.25x model for classification. Image patches will be classified into three groups: large-sized classes, backgrounds and regions of interest (RoI). Among them, patches of large-sized classes will be directly used to construct the confidence map without being further examined.
by models of other zoom levels, as in the previous section. Backgrounds will be discarded. RoI will be tiled into $224 \times 224$ patches under a 10x zoom level without overlap and sent into a 10x model for classification into three categories: medium-sized classes, misc and RoI. Similar to the procedure for 1.25x, medium-sized classes will directly contribute to the confidence map for final classification, misc patches are discarded, and RoI will be tiled into $224 \times 224$ patches under 20x zoom level without overlap. These patches will be classified by a 20x model for small classes and be used to construct the confidence map. Finally, we upsample the predictions from both 1.25x and 10x to 20x so that the predictions are consistent in size for constructing a confidence map. Figure 3.6 shows a sample result for the new per-patch prediction pipeline. Dots gathered in a square shape are upsampled predictions from low zoom levels.

Since the regions for later models to classify are selected based on previous models’ results, misclassifications by previous models can lead to a series of errors, which might accumulate along the pipeline and harm the final output accuracy. Therefore, we set up a threshold to only keep the classification results with high confidence, as the results with low confidence have a high chance of being wrong. By doing so, regions with low classification confidences will be viewed by later models in the pipeline, thus gaining a chance of being correctly classified.

However, WSI regions are no longer guaranteed to be viewed by multiple modes on each zoom level, which might lead to a decrease in tissue type recognition performances and impact the final aggregation accuracy. We will show the impact on accuracy and efficiency in Section 4.5.

### 3.2.4 Confidence Map CNN for Patch-level Result Aggregation

As previously mentioned, a linear classifier does not provide satisfactory test accuracy due to its lack of ability to recognize the position and structure of tissues. The solution we propose is to use a CNN instead. Due to the small size of our dataset, we cannot use deep CNNs such as VGG16[29], ResNet[10] and Inception-V4[31] because the complexity of the model overwhelms the number of training samples. Therefore, we have designed a tiny CNN as shown
3.2. **Pipeline Details**

Figure 3.6: A sample per-patch classification result for a WSI. Each coloured dot corresponds to a predicted patch. Dots for backgrounds and patches with low confidence are not shown.

in Figure 3.7 with an appropriate complexity for our dataset. The CNN has four convolution blocks in sequential order. The first three blocks contain a $3 \times 3$ convolution layer with the number of output channels equal to input channels, a dropout layer\[30\] and a ReLU activation layer\[24\]. Similar to ResNet, skip connections are introduced before the input layer and after the output layer of the first three convolution blocks to allow a stronger gradient flow for fast training.

Unlike ordinary deep CNNs that feed a flattened output from the last convolution layer to fully-connected layers, we choose to feed the output into another special convolution block instead. The last convolution block of our network has a $1 \times 1$ convolution layer with two output channels, a global average pooling (GAP) layer \[21\] and a softmax activation layer as the final output layer of the entire network. The GAP layer was first introduced by Lin et al. to reduce overfitting. In contrast with fully-connect layers with the number of learnable
parameters equal to the product of input dimension and output dimension, the GAP layer has no trainable parameter. Thus, model complexity is reduced, and overfitting is prevented on this layer. The GAP layer aggregates the spatial information by averaging the values of pixels on each input channel, which suits better for 2D inputs than forcing 2D inputs to become 1D vectors to fit into fully-connect layers. Also, the GAP layer is not sensitive to image translation and rotation compared with fully-connected layers because the average value of pixels on a rotated or translated image is always very close to the average of the original image. Most importantly, using a GAP layer instead of fully-connected layers allows input confidence maps to preserve their shapes instead of reshaping to a fixed size. Similar to changing the colour of a cell can change its label into a cancer cell, resizing a confidence map into a fixed input shape to a network brings a high risk of changing its label because the structures of tissues are changed. Therefore, the original input shapes of the confidences maps must be preserved.

Since we are using the GAP layer as the final slide-level true/pseudo-invasion output layer, it is very important that the input to the GAP layer should have only two channels, each containing information related to one type of invasion. We use a $1 \times 1$ convolution layer to reduce
the number of channels of the output from previous convolution blocks. The $1 \times 1$ convolution layer was heavily used on SqueezeNet [13], which has achieved accuracy close to AlexNet [19] on ImageNet [7] with only 2% of AlexNet’s number of parameters. Different from ordinary $3 \times 3$ or $5 \times 5$ convolution kernels that learn to capture local structural features by considering neighbouring pixels, a $1 \times 1$ kernel applies linear weighting along the channels only. This drastically decreases the number of parameters as well as the time taken on the convolution operation. Therefore, the $1 \times 1$ convolution layer is useful in summarizing the input confidence maps and reducing the dimensions.

### 3.3 Training

In this section, we will talk in detail about the methods and techniques used in the training process.

#### 3.3.1 Balancing Classes and Weighted Training

Given our two-stage pipeline mentioned in previous sections, the unbalanced classes problem appears mostly in stage one, as the models are required to identify multiple types of tissues. Since we are training models to classify image patches converted from polygon annotations, the number of training samples for one class is determined by not only the number of annotations but also the size of the annotations. For example, if we tile a WSI at 20x zoom level into patches of $224 \times 224$ pixels, one annotated area of mucus lake can produce nearly thousands of image patches as shown in Figure 3.8. However, for other classes, such as luminal propria stroma and luminal necrosis, one annotated area could only generate dozens of image patches. This creates a huge difference between the number of image patches for training the per-patch classification models. As a result, models trained from unbalanced data can still have very high average accuracy and low loss, even if they completely ignore classes with fewer training samples.
To solve this problem, we apply three techniques: class upsampling, downsampling, and most importantly, the focal loss [22]. We first try to reduce the unbalance by upsampling the classes with fewer training samples by two times and downsampling the classes with more training samples to one-half. When upsampling a class, we apply a random combination of data augmentation techniques such as flipping, rotation and translation to create more training samples. However, sometimes a combination of data augmentation techniques can produce identical samples to other augmentation methods, and we need to check and discard repetitive results. For example, if we flip an image along its diagonal and then flip it vertically, the final output is the same as rotating the image 90 degrees clockwise, as shown in Figure 3.9. To downsample a class, we randomly sample a subset of image patches from each annotation.

With both the upsampling and downsampling, the class imbalance problem is reduced but still not completely solved because the difference between the number of training samples in each class is too big. Therefore, we apply the focal loss [22], which is a modified version of the cross-entropy loss. This loss function improves the cross-entropy loss by assigning a weight to classified training samples based on the classification difficulty and class size. Let $p_t$ be the predicted probability of the true class $t$, the focal loss [22] is defined as the following:

$$Focal(p_t) = \alpha (1 - p_t)\gamma \log(p_t)$$
3.3. Training

Figure 3.9: An illustration of repetitive results generated from different data augmentation methods. Applying image transpose and vertical flip is the same as rotating the image 90 degrees clockwise.

Where $\alpha$ is a weight factor to the class size and $\gamma$ is a hyperparameter to control the effect brought to the loss by the difficulty of making a correct prediction. The weight factor $\alpha$ is set to be the inverse of class frequencies so that classes with more training samples contribute less to the loss function and classes with fewer training samples have a larger weight toward the loss, preventing the model from ignoring the less frequent classes. The $(1 - p_t)^\gamma$ term measures the difficulty of correctly predicting a class. If the predicted probability $p_t$ is close to 1, the model can easily and almost correctly predict the true class. In this case, $(1 - p_t)$ is close to 0, bringing down the loss computed for this training sample. However, if the model makes very poor predictions of the true classes, the output probability for this class will be close to 0, and $(1 - p_t)$ will be close to 1. This leaves the computed loss unchanged. When $\alpha$ is set to 1 and $\gamma$ is set to 0, the focal loss function reduces into the standard cross-entropy loss.

3.3.2 Self-supervised Pretraining

It was shown by studies that transfer learning can significantly accelerate the training process and increase the final accuracy. However, as previously mentioned, transferring the weights learnt from the ImageNet dataset does not bring a huge boost to our training process because we are dealing with images very differently from natural images. Therefore, we choose to pretrain
the model on the NCT-CRC-HE-100K [16] dataset. This dataset contains 100,000 labelled images from 10 classes, which are patches taken from human colon whole-slide images at 20x zoom level. This dataset was originally collected for the medical image classification task, which is highly similar to the first stage goal in our two-stage methods.

![stroma with hemosiderin](image)

Figure 3.10: An imprecise annotation. The blue polygon is the annotated area, and the black circles include areas that do not belong to the annotated class.

Although the most common method for pretraining is supervised learning on a similar task, previous works indicated that self-supervised learning is more beneficial, especially in medical image tasks. Alejandro’s work [26] compared four pretraining methods on four computer vision tasks, including the image classification task. The result shows that self-supervised pretraining almost always brings more utility than other methods, especially when the amount of labelled samples for fine-tuning is limited. Furthermore, Dan’s work [11] shows that self-supervised learning adds more robustness when the labels are noisy due to corruption. In our situation, the fine-tuning dataset contains noise due to imprecise annotation because it is too hard and time-consuming to outline the tissue areas perfectly. For example, in Figure 3.10 the annotation has included a few small areas that do not belong to the class, which might create training image patches with faulty labels. Finally, Dan et al. [25] compared self-supervised learning with normal learning methods, specifically on medical imaging tasks. They evaluated
the robustness and generalizability of both learning methods with corrupted data. They showed that the self-supervised learning method achieves a higher accuracy on classification tasks and higher Dice scores on segmentation tasks. Therefore, we choose self-supervised learning as our pretraining method.

We use Bootstrap-Your-Own-Latent (BYOL)\cite{BYOL} as the self-supervised learning method. This method tries to learn robust representations by iteratively contrasting the model with a slightly different version of itself using different views of the same input. Let $x$ be the input image, $t$ and $t'$ be different augmentations. At each iteration, the augmented images $t(x)$ and $t'(x)$ go respectively into two models with parameters $\theta$ and $\xi$, where $\theta$ is the desired set of parameters and $\xi$ is the exponential moving average of $\theta$ as shown below:

$$
\xi \leftarrow \tau \xi + (1 - \tau) \theta
$$

Where $\tau \in [0, 1]$ is the decay factor for the exponential moving average. Then, BYOL computes the mean-squared error between the predictions made by the two models, as shown in the formula below:

$$
L_{\theta, \xi} = \| h(\theta, t(x)) - h(\xi, t'(x)) \|^2
$$

Where $h(\theta, t(x))$ is the prediction from the model with desired parameters $\theta$ and input augmented by $t$, and $h(\xi, t'(x))$ is the prediction from the model with parameter $\xi$ and input augmented by $t'$. Finally, BYOL updates the parameters $\theta$ by backpropagating the computed loss and starts the next iteration of the same procedure. By augmenting the input differently, the model learns representations robust to the views of the same input, which means it learns to capture the semantic information of the input image.

### 3.3.3 Image Colour Normalization

Image colour is an important feature in image classification tasks. This is even more crucial when dealing with medical images, as slight variations in the image colour can alter the image
class. However, due to many reasons, such as the difference in scanning equipment, lighting conditions and tissue storage methods, whole-slide images tend to have slight variations in the colours. As shown in Figure 3.11, the tissue in the right image is darker than the left one.

![Figure 3.11: Colour difference between two WSIs. Even though both tissues are extracted from the same body part, the right image has darker tissue than the left image.](image)

To make image colours consistent, we first randomly sampled 20 WSIs and randomly collected image patches from these WSIs, excluding background areas. Then, we compute the pixel mean and variance for each image channel from the extracted image patches. We use the computed pixel mean and variance as our standard for all WSIs, and we normalize every WSI patch to have the same mean and variance before they go into our models.

### 3.3.4 Training Details

Our CNNs for tissue type recognition in the multi-zoom-level pipelines are pretrained using BYOL on the NCT-CRC-HE-100K dataset. Then, we finetune our CNNs on the colon tissue image patches extracted from our 50 annotated WSIs on areas annotated by pathologists. The tissue image patches are extracted at corresponding zoom levels to train the specific CNNs. We use 80% of the extracted image patches to train, 10% to validate, and 10% to test our models.

We adjust the hyperparameters to achieve the best accuracy on the validation set. For the
focal loss, we set the difficulty factor $\gamma$ to 2 and class weights $\alpha$ to 1. For the BYOL, we set the moving average decay $\tau$ to 0.99. We use the Adam optimizer for our training. The learning rate we used is 0.001, and the coefficients are set to be 0.9 and 0.999 for the running averages of the gradients and their squares, respectively. The weight decay is set to 0, which means there is no weight decay. When training our CNNs for WSI patch classifications, we use mini-batch training with a batch size of 256 for both self-supervised pretraining and fine-tuning. However, we do not use batch training for the aggregation model because the batch training procedure requires all training samples to have the same dimension so that they can be stacked together to form a batch. We want to keep the native dimension of the confidence/label maps, so we train the aggregation model with a batch size of 1. All models are trained on an RTX3080 GPU.
Chapter 4

Experiments

In this chapter, we will discuss the setup of our experiments, including the explanation of the datasets we used, the evaluation metric, the implementation details, and the baseline methods. Then, we show our results on the datasets with comparisons to the baseline method and analysis. Finally, we perform an ablation study to show the impact and effectiveness of the different techniques we used.

4.1 Datasets

Our proposed methods all consist of two stages, and each stage does different tasks. The first stage predicts the WSI patches into different categories, and the second stage aggregates the per-patch prediction results. Therefore, we need at least two different kinds of datasets. For the first stage, we use the NCT-CRC-HE-100K dataset as the pretraining dataset and the image patches generated from the annotations on our private collection of WSIs as the fine-tuning dataset. For the second stage, we use our collection of WSIs and their slide-level labels to train the model.
4.1. Datasets

4.1.1 NCT-CRC-HE-100K dataset

In this thesis, we use the NCT-CRC-HE-100K dataset [16] as the pretraining dataset to transfer knowledge to our stage-one models. This dataset can be downloaded from their official website here. This dataset was collected by the National Centre of Tumor Disease at Heidelberg, and it was used for studies related to deep learning applications on colon cancers and tissue type classifications. The NCT-CRC-HE-100K dataset consists of 100,000 images from 9 different tissue classes: adipose (ADI), background (BACK), debris (DEB), lymphocytes (LYM), mucus (MUC), smooth muscle (MUS), normal colon mucosa (NORM), cancer-associated stroma (STR), colorectal adenocarcinoma epithelium (TUM). Figure 4.1 shows some images from the NCT-CRC-HE-100K dataset in each class. However, classes in this dataset are not evenly distributed. The largest class is TUM, which contains 14317 images. Small classes such as MUC and NORM only contain around 8500 images.

![Image](image_url)

Figure 4.1: Example images taken from the NCT-CRC-HE-100K dataset.

The images are non-overlapping patches manually taken from 86 hematoxylin & eosin stained whole-slide images. To increase the variability of the normal tissue classes, image patches from healthy regions from gastrectomy slides were added. Each image in the dataset
is a $224 \times 224$ RGB image taken at 20x zoom level, or 0.5 microns per pixel. To reduce the image colour variance, all images are normalized using Macenko’s image colour normalization method \cite{23}. This dataset is used as a pretraining dataset for our models to learn better representations of WSI patches.

4.1.2 Our Private Dataset

Our dataset contains 150 slides identified as true invasion or pseudoinvasion in the department of Pathology and Laboratory Medicine, Western University. The slides are scanned using the Aperio CS scanner. Among the 150 slides, 50 slides are annotated to show the tissue types on them. The annotations are in 9 classes: acellular mucin, angulated gland, desmoplastic stroma, hemorrhage, lp stroma, luminal necrosis, mucus lake with peripheral dysplastic glands, rounded lobular group, stroma with hemosiderin. Each annotation is a polygon with a class name. The polygons are represented by an ordered list of vertices, which are tuples are $(x, y)$ coordinates. Figure 4.2 shows an example of the annotated area on a WSI.

Figure 4.2: An example of the annotated area on a WSI in our dataset. Coloured polygons indicate the areas of certain tissue types.
This dataset is used for both fine-tuning our pretrained models to recognize tissue types, and training the aggregation model in the second stage of our pipeline. When used for fine-tuning, we create a collection of images by extracting $224 \times 224$ image patches from the annotations on each WSI at different zoom levels. Pathologists then view the extracted patches to filter out unrecognizable patches. Table 4.1 shows the number of examples in each class on different zoom levels. However, as previously mentioned, the classes in the fine-tuning datasets are extremely unbalanced. Therefore, we applied a random combination of different image augmentation methods to balance the classes. These methods include random rotation, flip and translation. When used for training the aggregation model, we create a collection of label/confidence maps using the per-patch classification results for each WSI and use the WSI’s true/pseudo invasion result as the label for each label/confidence map.

<table>
<thead>
<tr>
<th>Classes</th>
<th>1.25x</th>
<th>5x</th>
<th>10x</th>
<th>20x</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acellular mucin</td>
<td>25</td>
<td>77</td>
<td>308</td>
<td>1462</td>
</tr>
<tr>
<td>Angulated gland</td>
<td>13</td>
<td>17</td>
<td>69</td>
<td>281</td>
</tr>
<tr>
<td>Desmoplastic stroma</td>
<td>N/A</td>
<td>12</td>
<td>47</td>
<td>205</td>
</tr>
<tr>
<td>Hemorrhage</td>
<td>N/A</td>
<td>2</td>
<td>4</td>
<td>18</td>
</tr>
<tr>
<td>LP stroma</td>
<td>2</td>
<td>11</td>
<td>48</td>
<td>192</td>
</tr>
<tr>
<td>Luminal necrosis</td>
<td>4</td>
<td>20</td>
<td>83</td>
<td>334</td>
</tr>
<tr>
<td>Mucus lake with peripheral dysplastic glands</td>
<td>37</td>
<td>55</td>
<td>225</td>
<td>914</td>
</tr>
<tr>
<td>Rounded lobular group</td>
<td>182</td>
<td>440</td>
<td>1768</td>
<td>7074</td>
</tr>
<tr>
<td>Stroma with hemosiderin</td>
<td>21</td>
<td>41</td>
<td>167</td>
<td>669</td>
</tr>
</tbody>
</table>

Table 4.1: The number of training examples in each class on each zoom level

Since the data collected from patients contain sensitive information, we cannot release them to the public without the consent of every patient. Therefore, we keep this dataset private to our group and only use the WSIs for model training.

### 4.2 Evaluation Metric

As previously discussed, we converted the annotated regions into collections of image patches at the desired zoom levels, which we utilized for training and evaluating our models. To evalu-
ate the models’ performance, we randomly split the collection of image patches, utilizing 80% for training and the remaining 20% for testing. We employed classification accuracy as our evaluation metric, which is defined as the number of correctly classified images divided by the total number of images. The average accuracy of our models was 95.3%.

However, as previously mentioned, the categories in our dataset were unbalanced. Therefore, the average accuracy might not provide enough information. Hence, we analyzed the per-class classification accuracy using a confusion matrix. We normalized the confusion matrix so that each entry in the matrix shows percentages instead of exact numbers, for a more intuitive interpretation. As depicted in Figure 4.5, we observed that our models did not perform well on the fourth class, hemorrhage. This class had a shortage of training samples in our dataset, resulting in a suboptimal accuracy.

<table>
<thead>
<tr>
<th>True Class 1</th>
<th>Predicted Class 1</th>
<th>Predicted Class 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly Predicted Class 1</td>
<td></td>
<td>Incorrectly Predicted 1 as Class 2</td>
</tr>
<tr>
<td>Incorrectly Predicted Class 2 as Class 1</td>
<td>Correctly Predicted Class 2</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.2: A confusion matrix for two classes

4.3 Implementation Details

The first stage of our pipelines involves the use of multiple CNNs to classify WSI patches. We use ResNet-18 architecture for all of the CNNs with an input size of $3 \times 224 \times 224$. We remove the fully connected layers from the original ResNet-18 architecture and replace them with three linear layers. The first linear layer has an input dimension of 512 and an output dimension of 256. The second linear layer has an input dimension of 256 and an output dimension of 128. The third linear layer has an input dimension of 128 and an output dimension equal to the number of classes. We added the ReLU activation function right after the first linear layer and the second linear layer.

For the aggregation model introduced previously in chapter 3.2.2, we use Dropout layers
to prevent overfitting. The dropout probability is set to 0.30. For the pipeline introduced in chapter 3.2.3, we added a threshold to filter out patch predictions with low confidences before they are constructed into label/confidences maps and fed into the aggregation model. We set the threshold to 0.50.

4.4 Baseline Methods

We compare our methods against two baseline methods. In the first baseline, we train a CNN to classify downscaled WSIs directly. The second baseline is a single-zoom-level patch-based prediction pipeline with a linear classifier for patch results aggregation.

4.4.1 Directly Classify Downsized WSIs

As previously mentioned, WSIs are too large to fit into standard CNNs, so we have to either downsize the WSI to a suitable size or tile the WSI into small patches. Although downsizing a WSI will lead to a serious loss of information which decreases the accuracy, classifying a WSI as a whole is significantly more efficient than classifying a collection of image patches. Therefore, the direct classification method might prove useful.

First, we resize the WSIs to $224 \times 224$ pixels. Then, we fine-tune a ResNet-18 \cite{10} model pretrained on the ImageNet \cite{7} dataset to classify the downsized WSIs directly into true invasion and pseudo-invasion. We freeze all the convolution layers and only fine-tune the linear
layers because we do not have enough training examples for training the whole network. We randomly choose 120 WSI images from our dataset to train this model. We use the slide labels as the training labels. The remaining 30 WSIs are used to validate the model.

Different from patch-based methods that can provide intermediate results as evidence, this baseline method skips the tiling and aggregation process by directly outputting slide-level results.

### 4.4.2 Single-zoom-level Patch-based Prediction with SVM

We choose the 20x zoom level for the single-zoom-level method, which is the highest zoom level used in our proposed multi-zoom-level methods. Firstly, we tile the input WSIs into a collection of image patches of $224 \times 224$ pixels. After tiling the WSIs and preprocessing the patches, we iteratively feed the image patches into our CNN model for classification. We use ResNet-50 [10] pretrained on ImageNet [7] as our CNN model. We fine-tune the model on our dataset to classify each image patch into nine different tissue type categories used to identify true/pseudo-invasions.

Per-patch results are processed before aggregation. We process by counting the class frequencies, and building class percentage histograms. The histogram can be seen as a vector $x \in [0, 1]^n$, where $n$ is the number of tissue type classes. Then, we learn a support vector machine (SVM) as the aggregation model. The SVM takes a histogram $x$ as input and outputs a confidence score $y \in [0, 1]$ representing how likely the WSI contains true invasion as shown in the bottom part of Figure 4.4.

### 4.5 Experiment Results

We include two parts in our experimental results. The first part evaluates the tissue type recognition accuracy. The second part evaluates the aggregation accuracy, which is also the final true/pseudo invasion classification accuracy for WSIs.


4.5.2  Patch-level WSI Tissue Type Recognition Results

We train and evaluate the stage-one methods on our own dataset. We convert the annotated areas into collections of image patches on desired zoom levels, and randomly split the collection of image patches. All of the methods in comparison use the same 80% of the collection for training, if applicable (i.e. the zoom-level matches). The remaining 20% are used for testing. The baseline and our models are trained on the same collections for 25 epochs. Table 4.3 shows the classification accuracy for each method. Our multi-zoom-level methods achieve a higher classification accuracy than the single zoom-level method because the areas on a WSI are viewed multiple times and combined to get a better result. The selective multi-zoom-level prediction method has a slightly lower accuracy compared with the multi-zoom-level weighted average method because the areas on a WSI are usually not viewed by all models but only one or two models.

As previously mentioned, the classes in the dataset are extremely unbalanced, so the aver-
age accuracy might not be explanatory enough. Hence, we analyze the per-class classification accuracy using normalized confusion matrices. Figure 4.5 shows the normalized confusion matrix for each method. We notice that the single zoom-level prediction method performs very poorly on the second and the fourth class, which is angulated gland and hemorrhage. These two classes are not only short on the number of training samples but also not suitable to be viewed at a fixed zoom level. Both of our methods have improved the classification accuracy of these two classes.

Efficiency is also one important aspect of evaluating the methods’ practicality in real-life applications. We compare the methods’ average time to classify all the patches from a WSI for tissue type recognition. We randomly sample 10 WSIs from our dataset and calculate the average time taken for each method to run on those WSIs. Table 4.3 also shows the average time taken for each method measured in seconds. We can notice that the multi-zoom-level weighted average method takes around 40% extra time compared with the single zoom-level method because the 10x zoom level and 5x zoom level bring an additional 37.5% to the total number of image patches to classify. However, the selective multi-zoom-level prediction method takes a short average time, which is only half the time taken by the multi-zoom-level weighted average method and even lower than the time taken by the single zoom-level method. The increase in the speed on the selective multi-zoom-level method comes from the fact that the WSI is only fully viewed under 1.25x zoom level and is partially viewed under 10x and 20x zoom levels. On average, a WSI only generates around one hundred image patches at a 1.25x zoom level, and most areas are only viewed once at other zoom levels.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single zoom-level at 20x</td>
<td>92.3%</td>
<td>1083.7s</td>
</tr>
<tr>
<td>Exhaustive Multi-zoom-level weighted average</td>
<td>95.7%</td>
<td>1487.1s</td>
</tr>
<tr>
<td><strong>Selective Multi-zoom-level prediction (Ours)</strong></td>
<td>95.1%</td>
<td>771.5s</td>
</tr>
</tbody>
</table>

Table 4.3: WSI patch classification accuracy for each method
4.5.2 Slide-level WSI True/pseudo-invasion Classification Results

The final goal for this work is to build a system for classifying true/pseudo invasions through WSIs, and the tissue type recognition task is just an intermediate step for us to reach our goal and also provide visible evidence for the final classification result. The final result comes from aggregating the confidence maps, which contain the tissue type recognition results for a WSI. We randomly choose 120 WSIs from our 150 WSIs to train all the aggregation models. The aggregation model is trained on the outputs from the corresponding stage-one pipeline for each method, and the targets are the true/pseudo invasion identified by pathologists. Table [4.4] shows...
the final accuracy for each method. Then we use the remaining 30 WSIs for testing.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directly Classifying Downsized WSIs</td>
<td>59.7%</td>
</tr>
<tr>
<td>Single zoom-level at 20x + SVM</td>
<td>73.1%</td>
</tr>
<tr>
<td>Single zoom-level at 20x + Confidence map CNN</td>
<td>76.4%</td>
</tr>
<tr>
<td>Multi zoom-level weighted average + SVM</td>
<td>77.5%</td>
</tr>
<tr>
<td>Multi zoom-level weighted average + Confidence map CNN (Ours)</td>
<td>84.2%</td>
</tr>
<tr>
<td>Selective multi-zoom-level prediction + SVM</td>
<td>76.9%</td>
</tr>
<tr>
<td>Selective multi-zoom-level prediction + Confidence map CNN (Ours)</td>
<td>83.6%</td>
</tr>
</tbody>
</table>

Table 4.4: Final WSI True/Pseudo Invasion Classification Accuracy for each method

It can be observed that multi-zoom-level methods for tissue type recognition combined with a confidence map CNN for aggregation have achieved noticeably higher accuracy than single zoom-level prediction with a linear classifier for aggregation. We also investigate the performance of multi-zoom-level methods with a linear classifier, and it is clear that a linear classifier is outperformed by the proposed confidence map CNN by a noticeable gap in accuracy. Directly classifying downsized WSIs achieves the lowest accuracy, which proves that WSIs cannot be downsized too much to the degree that important features for models to learn are lost.

### 4.6 Ablation Studies

In this section, we perform ablation experiments on important components to study their importance. First, we test the impact of different pretraining methods. Then, we analyze the patch confidence threshold used in the selective multi-zoom-level prediction methods. Finally, we compare the effectiveness of various image colour normalization methods in increasing the model performance.
4.6.1 Impact of Pretraining Methods

Pretraining is a critical and essential step in training deep learning models to get high accuracy. Many pretraining methods have been developed to learn more meaningful and generalizable knowledge from the data. We choose three pretraining methods: supervised pretraining, self-supervised pretraining (BYOL) and no pretraining. Here, we test the effectiveness of the pretraining methods using one of our CNNs that recognizes tissue types at a 20x zoom level. We pretrain the CNN on NCT-CRC-HE-100K dataset. We then fine-tune the pretrained model on a collection of tissue patches extracted from our WSIs on annotated areas at a 20x zoom level. We use 80% of the collection to train the CNN, and measure the tissue classification accuracy on the remaining 20%. Table 4.5 shows the fine-tuning accuracy of models pretrained using different methods. From the table, we can observe that the model pretrained using BYOL achieves the highest accuracy. The improvement is marginal compared with the supervised pretraining method but noticeable compared with no pretraining. On the other hand, models with pretraining converge much faster during fine-tuning and are more robust to data noise. In essence, pretraining is always beneficial for better performance.

<table>
<thead>
<tr>
<th>Pretraining Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>No pretraining</td>
<td>87.4%</td>
</tr>
<tr>
<td>Supervised pretraining</td>
<td>94.9%</td>
</tr>
<tr>
<td>BYOL [8]</td>
<td>95.7%</td>
</tr>
</tbody>
</table>

Table 4.5: Fine-tuning accuracy of models pretrained using different methods

4.6.2 Impact of Patch Confidence Threshold

In our second proposed method, we use a threshold to filter out predictions with low confidence. This is beneficial because predictions with low confidence are more likely to be wrong. However, setting up a correct threshold is crucial because high thresholds can filter out correct predictions, and low thresholds may let wrong predictions pass. Therefore, we need to analyze the model’s false positive (FP) rate and false negative (FN) rate at each threshold. Here, we test
the impact of the patch confidence threshold using our fully established selective multi-zoom-level pipeline and only change the threshold. We change the confidence from 0.05 to 0.95 and measure the corresponding FP and FN rates on our dataset of 150 WSIs at each threshold. Figure 4.6 shows the FP and FN rates corresponding to each threshold value. The figure shows that both false rate lines intersect when the threshold is equal to 0.55. In real-life cancer diagnoses, false positives are preferred over false negatives because patients will go over more physical examinations when diagnosed as cancer-positive. However, a false negative might lead to a delay in treatment and will lead to severe consequences. Therefore, we use 0.5 as our threshold in actual implementation.

4.6.3 Impact of Image Colour Normalization Methods

Colour normalization is always an important step in data processing and is even more crucial when dealing with medical images. Due to the difference in scanning equipment and environmental conditions, WSIs taken in different labs can have biased colours, even when the scanned tissues are from the same patient. Such a bias in image colour might confuse the
model during training because image colour is an important feature in distinguishing between cell types. Therefore, we choose to compare the impact of three image colour normalization methods: standard normalization, Macenko’s normalization and no normalization. The standard normalization method normalizes images by changing the pixel RGB values to a specified mean and variance. Macenko’s colour normalization method is widely used in digital pathology by making histology slides visually consistent, and it is also the colour normalization method used on the NCT-CRC-HE-100K dataset. Figure illustrates the visual effect of different normalization methods.

![Comparison of Different Colour Normalization Methods](image)

**Figure 4.7**: An illustration of different colour normalization methods

We compare the impact of these methods to our final classification results by accuracy and efficiency. In this case, we measure the final results using our selective multi-zoom-level pipeline, trained and tested on the same set of WSIs but colour normalized using different techniques. Table shows the final WSI true/pseudo invasion classification accuracy and the average time taken to classify a WSI. We can see from the table that using image colour normalization brings a significant increase in the final accuracy, but also consumes more time. Macenko’s normalization method achieves the highest accuracy, but it also takes the most time. Considering the trade-off between accuracy and efficiency, the standard normalization method turns out to be the most practical choice.
<table>
<thead>
<tr>
<th>Colour Normalization Method</th>
<th>Accuracy</th>
<th>Time Taken</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Normalization</td>
<td>81.6%</td>
<td>634.7s</td>
</tr>
<tr>
<td>Standard Normalization</td>
<td>84.2%</td>
<td>771.5s</td>
</tr>
<tr>
<td>Macenko’s Normalization [23]</td>
<td>84.8%</td>
<td>907.3s</td>
</tr>
</tbody>
</table>

Table 4.6: Impact of colour normalization method in terms of average accuracy and time spent.
Chapter 5

Conclusion and Future Work

In this chapter, we summarize our work presented in this thesis. Lastly, we discuss the limitations of our work and our future plans.

5.1 Conclusion

The demand for computer-aided assisting tools in the field of digital pathology and cancer diagnosis has been growing for many years. In community hospitals, pathologists often struggle to make an accurate diagnosis of colon cancers because the diagnosis is made based solely on identifying true invasion. The cases of pseudoinvasion bring challenges to pathologists, and they often have to seek external consultation. This not only adds a financial burden but also significantly slows down patients’ treatment process. Many researchers have investigated and developed deep learning systems for cancer diagnosis and whole-slide image classification, but little work has been done on the true/pseudo-invasion classification task as the problem is not well-known, and resources are limited.

In this thesis, we propose two methods to provide end-to-end solutions to the WSI true/pseudo-invasion classification task. Our methods are designed to have two stages: a tissue-type recognition stage and an aggregation stage. In the tissue type recognition stage, both of our methods view and classify a given WSI by image patches at multiple zoom levels using convolutional
neural networks. The multi-zoom-level design makes tissue type recognitions more accurate because different types of tissue have inconsistent sizes that cannot be recognized well in a window of fixed size at one fixed zoom level. We pretrain our CNN models on the NCT-CRC-HE-100K dataset using BYOL to produce more transferable and noise-robust knowledge. The models are fine-tuned on our private dataset to learn to classify task-specific classes.

In the second stage, our methods aggregate the tissue type recognition results into slide-level prediction. However, traditional aggregation methods, such as using a linear classifier, are inadequate to produce accurate results, even with accurate recognition results of tissue types. Thus, we design a small convolutional network for aggregating the patch-level results into slide-level results in the second stage of our methods. The designed network allows the interpretation of spatial and structural patterns about recognized tissue on a WSI to get a more meaningful and accurate aggregation result. However, training convolutional neural networks often require a great number of training examples which we do not have. To solve this issue, we limit the complexity of our aggregation model by reducing its size. We also add skip connections to enable an even stronger gradient flow, and the GAP layer is used at the end of the entire network instead of fully-connected layers to reduce the complexity of the entire network. Such a design allows fast training under limited data while keeping the model’s performance.

For the experimental results, it can be observed that our proposed multi-zoom-level methods outperform the single zoom-level baseline by a clear margin. The multi-zoom-level patch prediction methods significantly increase the tissue type recognition accuracy and provide more reliable results for the final aggregation process. The per-class accuracy is balanced in our methods, and the time efficiency is still competitive with the single zoom-level method. Also, the classified image patch can also be shown directly on a WSI as an overlay, which can act as visual evidence of the final results for pathologists to examine. The CNN-based aggregator in the second stage of our methods also outperforms the traditional linear classifier aggregator, providing more accurate and trustworthy slide-level results.
In the ablation experiments, we show the importance of self-supervised pretraining against supervised pretraining and no pretraining. We observe that BYOL helps with higher fine-tuning accuracy than other supervised pretraining and no pretraining. We also analyze the effect of patch confidence thresholds on our final output’s false positive rates and false negative rates. We pick the most suitable threshold for our demand based on the analysis. Lastly, we show the effectiveness of different image colour normalization methods in data processing by comparing their impacts to the final accuracy and the time taken to classify a WSI. Standard colour normalization was shown to be the most practical and effective choice as it brings high accuracy and adds a reasonable time to the whole classification process.

In conclusion, we propose two strong and practical methods for the WSI true/pseudo-invasion classification task. Our methods first use patch-based prediction at multiple zoom-levels to provide accurate and reliable tissue type recognition results, then aggregate the patch-level results into slide-level results. The two-stage structure provides a chance to visualize intermediate results and also adds modularity for modifications for similar applications. This work gives the possibility of developing cheap and accessible assisting tools for pathologists. We believe this work will inspire the future development of systems that can reduce the workload of pathologists and also the financial burden of the public health system.

5.2 Future Work

Although we believe our current work gives meaningful insights and provides a solid solution to the WSI true/pseudo-invasion classification task, the work still has many aspects that can be further improved, and some problems need to be addressed in the future.

First, the amount of available training samples for this task is limited. We only have 150 slides for this task, and only 55 of them are annotated. The annotation is also with limited precision. With such a small number of available training samples, it is difficult to achieve very high accuracy for clinical use. So far, our methods only provide ways to develop a consultation
system for pathologists to get suggestions. Future works can be done on investigating data-efficient methods, and also on collecting more task-specific data.

Second, our methods are based on a predict-by-patch procedure, which is very slow, and deploying multiple models for multi-zoom-level predictions makes the procedure even slower. Future works can be done on investigating non-patch-based methods or investigating the possibility of paralleling the procedure by multi-processing.
Bibliography


## Appendix A

### The ResNet architecture details

<table>
<thead>
<tr>
<th>layer name</th>
<th>output size</th>
<th>18-layer</th>
<th>34-layer</th>
<th>50-layer</th>
<th>101-layer</th>
<th>152-layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv1</td>
<td>$112 \times 112$</td>
<td></td>
<td></td>
<td>$7 \times 7$, 64, stride 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>conv2.x</td>
<td>$56 \times 56$</td>
<td>$3 \times 3$, 64, stride 2</td>
<td>$3 \times 3$, 64, stride 2</td>
<td>$1 \times 1$, 64</td>
<td>$1 \times 1$, 64</td>
<td>$1 \times 1$, 64</td>
</tr>
<tr>
<td>conv3.x</td>
<td>$28 \times 28$</td>
<td>$3 \times 3$, 128, stride 2</td>
<td>$3 \times 3$, 128, stride 2</td>
<td>$1 \times 1$, 128</td>
<td>$1 \times 1$, 128</td>
<td>$1 \times 1$, 128</td>
</tr>
<tr>
<td>conv4.x</td>
<td>$14 \times 14$</td>
<td>$3 \times 3$, 256, stride 2</td>
<td>$3 \times 3$, 256, stride 2</td>
<td>$1 \times 1$, 256</td>
<td>$1 \times 1$, 256</td>
<td>$1 \times 1$, 256</td>
</tr>
<tr>
<td>conv5.x</td>
<td>$7 \times 7$</td>
<td>$3 \times 3$, 512, stride 2</td>
<td>$3 \times 3$, 512, stride 2</td>
<td>$1 \times 1$, 512</td>
<td>$1 \times 1$, 512</td>
<td>$1 \times 1$, 512</td>
</tr>
<tr>
<td>1×1</td>
<td>average pool, 1000-d fc, softmax</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FLOPs</td>
<td>$1.8 \times 10^9$</td>
<td>$3.6 \times 10^9$</td>
<td>$3.8 \times 10^9$</td>
<td>$7.6 \times 10^9$</td>
<td>$11.3 \times 10^9$</td>
<td></td>
</tr>
</tbody>
</table>

Figure A.1: The detailed architecture for ResNet with different number of layers
Curriculum Vitae

Name: Zhiyuan Yang

Post-Secondary Education and Degrees:
University of Toronto
Toronto, Ontario, Canada
2016 - 2021 B.Sc.

Honours and Awards:
Western Graduate Research Scholarships
2021-2022

Related Work Experience:
Teaching Assistant
The University of Western Ontario
2021 - 2022